Springer Texts in Statistics

Peter K. Dunn · Gordon K. Smyth

Generalized Linear Models With Examples in R

Springer

Springer Texts in Statistics

Series Editors

R. DeVeaux
S.E. Fienberg
I. Olkin

More information about this series at http://www.springer.com/series/417

Peter K. Dunn • Gordon K. Smyth

Generalized Linear Models With Examples in R

Springer

Peter K. Dunn
Faculty of Science, Health, Education and Engineering
School of Health of Sport Science
University of the Sunshine Coast
QLD, Australia

Gordon K. Smyth
Bioinformatics Division
Walter and Eliza Hall Institute of Medical Research
Parkville, VIC, Australia

ISSN 1431-875X ISSN 2197-4136 (electronic)
Springer Texts in Statistics
ISBN 978-1-4419-0117-0 ISBN 978-1-4419-0118-7 (eBook)
https://doi.org/10.1007/978-1-4419-0118-7

Library of Congress Control Number: 2018954737

This Springer imprint is published by the registered company Springer Science+Business Media, LLC part of Springer Nature.
The registered company address is: 233 Spring Street, New York, NY 10013, U.S.A.

To my wife Alison; our children Jessica, Emily, Jemima, Samuel, Josiah and Elijah; and my parents: Thank you for your love and support and for giving so much so I could get this far. PKD

To those who taught me about glms 40 years ago and to all the students who, in the years since, have patiently listened to me on the subject, given feedback and generally made it rewarding to be a teacher. GKS

Preface

A sophisticated analysis is wasted if the results cannot be communicated effectively to the client.
Reese [4, p. 201]

Our purpose in writing this book is to combine a good applied introduction to generalized linear models (GLMs) with a thorough explanation of the theory that is understandable from an elementary point of view.

We assume students to have basic knowledge of statistics and calculus. A working familiarity with probability, probability distributions and hypothesis testing is assumed, but a self-contained introduction to all other topics is given in the book including linear regression. The early chapters of the book give an introduction to linear regression and analysis of variance suitable for a second course in statistics. Students with more advanced backgrounds, including matrix algebra, will benefit from optional sections that give a detailed introduction to the theory and algorithms. The book can therefore be read at multiple levels. It can be read by students with only a first course in statistics, but at the same time, it contains advanced material suitable for graduate students and professionals.

The book should be appropriate for graduate students in statistics at either the masters or PhD levels. It should be also be appropriate for advanced undergraduate students taking majors in statistics in Britain or Australia. Students in psychology, biometrics and related disciplines will also benefit. In general, it is appropriate for anyone wanting a practical working knowledge of GLMs with a sound theoretical background.

R is a powerful and freely available environment for statistical computing and graphics that has become widely adopted around the world. This book includes a self-contained introduction to R (Appendix A), and use of R is integrated into the text throughout the book. This includes comprehensive R code examples and complete code for most data analyses and case studies. Detailed use of relevant R functions is described in each chapter.

A practical working knowledge of good applied statistical practice is developed through the use of real data sets and numerous case studies. This book makes almost exclusive use of real data. These data sets are collected in the R package **GLMsData** [1] (see Appendix A for instructions for obtaining

this R package), which has been prepared especially for use with this book and which contains 97 data sets. Each example in the text is cross-referenced with the relevant data set so that readers can load the relevant data to follow the analysis in their own R session. Complete reproducible R code is provided with the text for most examples.

The development of the theoretical background sometimes requires more advanced mathematical techniques, including the use of matrix algebra. However, knowledge of these techniques is not required to read this book. We have ensured that readers without this knowledge can still follow the theoretical development, by flagging the corresponding sections with a star * in the margin. Readers unfamiliar with these techniques may skip these sections and problems without loss of continuity. However, those with the necessary knowledge can gain more insight by reading the optional starred sections.

A set of problems is given at the end of each chapter and at the end of the book. The balance between theory and practice is evident in the list of problems, which vary in difficulty and purpose. These problems cover many areas of application and test understanding, theory, application, interpretation and the ability to read publications that use GLMs.

This book begins with an introduction to multiple linear regression. In a book about GLMs, at least three reasons exist for beginning with a short discussion of multiple linear regression:

- Linear regression is *familiar*. Starting with regression consolidates this material and establishes common notation, terminology and knowledge for all readers. Notation and new terms are best introduced in a familiar context.
- Linear regression is *foundational*. Many concepts and ideas from linear regression are used as approximations in GLMs. A firm foundation in linear regression ensures a better understanding of GLMs.
- Linear regression is *motivational*. GLMs often *improve* linear regression. Studying linear regression reveals its weaknesses and shows how GLMs can often overcome most of these, motivating the need for GLMs.

Connections between linear regression and GLMs are emphasized throughout this book.

This book contains a number of important but advanced topics and tools that have not typically been included in introductions to GLMs before. These include Tweedie family distributions with power variance functions, saddlepoint approximations, likelihood score tests, modified profile likelihood and randomized quantile residuals, as well as regression splines and orthogonal polynomials. Particular features are the use of saddlepoint approximations to clarify the asymptotical distribution of residual deviances from GLMs and an explanation of the relationship between score tests and Pearson statistics. Practical and specific guidelines are developed for the use of asymptotic approximations.

Throughout this book, R functions are shown in `typewriter font` followed by parentheses; for example, `glm()`. Operators, data frames and variables in R are shown in `typewriter font`; for example, `Smoke`. R packages are shown in **bold and sans serif font**; for example, **GLMsData**.

We thank those who have contributed to the writing of this book and especially students who have contributed to earlier versions of this text. We particularly thank Janette Benson, Alison Howes and Martine Maron for the permission to use data.

This book was prepared using LaTeX and R version 3.4.3 [3], integrated using Sweave [2].

Sippy Downs, QLD, Australia — Peter K. Dunn
Parkville, VIC, Australia — Gordon K. Smyth
December 2017

References

[1] Dunn, P.K., Smyth, G.K.: GLMsData: Generalized linear model data sets (2017). URL https://CRAN.R-project.org/package=GLMsData. R package version 1.0.0

[2] Leisch, F.: Dynamic generation of statistical reports using literate data analysis. In: W. Härdle, B. Rönz (eds.) Compstat 2002—Proceedings in Computational Statistics, pp. 575–580. Physika Verlag, Heidelberg, Germany (2002)

[3] R Development Core Team: R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing, Vienna, Austria (2017). URL https://www.R-project.org

[4] Reese, R.A.: Data analysis: The need for models? The Statistician **35**(2), 199–206 (1986). Special Issue: Statistical Modelling

Contents

Chapter 1
Statistical Models

. . . all models are approximations. Essentially, all models are wrong, but some are useful. However, the approximate nature of the model must always be borne in mind.
Box and Draper [2, p. 424]

1.1 Introduction and Overview

This chapter introduces the concept of a statistical model. One particular type of statistical model—the generalized linear model—is the focus of this book, and so we begin with an introduction to statistical models in general. This allows us to introduce the necessary language, notation, and other important issues. We first discuss conventions for describing data mathematically (Sect. 1.2). We then highlight the importance of plotting data (Sect. 1.3), and explain how to numerically code non-numerical variables (Sect. 1.4) so that they can be used in mathematical models. We then introduce the two components of a statistical model used for understanding data (Sect. 1.5): the systematic and random components. The class of regression models is then introduced (Sect. 1.6), which includes all models in this book. Model interpretation is then considered (Sect. 1.7), followed by comparing physical models and statistical models (Sect. 1.8) to highlight the similarities and differences. The purpose of a statistical model is then given (Sect. 1.9), followed by a description of the two criteria for evaluating statistical models: accuracy and parsimony (Sect. 1.10). The importance of understanding the limitations of statistical models is then addressed (Sect. 1.11), including the differences between observational and experimental data. The generalizability of models is then discussed (Sect. 1.12). Finally, we make some introductory comments about using R for statistical modelling (Sect. 1.13).

1.2 Conventions for Describing Data

The concepts in this chapter are best introduced using an example.

Example 1.1. A study of 654 youths in East Boston [10, 18, 20] explored the relationships between lung capacity (measured by forced expiratory volume,

P. K. Dunn, G. K. Smyth, *Generalized Linear Models with Examples in R*,
Springer Texts in Statistics, https://doi.org/10.1007/978-1-4419-0118-7_1

or FEV, in litres) and smoking status, age, height and gender (Table 1.1). The data are available in R as the data frame `lungcap` (short for 'lung capacity'), part of the **GLMsData** package [4]. For information about this package, see Appendix B; for more information about R, see Appendix A. Assuming the **GLMsData** package is installed in R (see Sect. A.2.4), load the **GLMsData** package and the `lungcap` data frame as follows:

```
> library(GLMsData)     # Load the  GLMsData  package
> data(lungcap)         # Make the data set  lungcap  available for use
> head(lungcap)         # Show the first few lines of data
  Age   FEV Ht Gender Smoke
1   3 1.072 46      F     0
2   4 0.839 48      F     0
3   4 1.102 48      F     0
4   4 1.389 48      F     0
5   4 1.577 49      F     0
6   4 1.418 49      F     0
```

(The `#` character and all subsequent text is ignored by R.) The data frame `lungcap` consist of five variables: `Age`, `FEV`, `Ht`, `Gender` and `Smoke`. Some of these variables are numerical variables (such as `Age`), and some are non-numerical variables (such as `Gender`). Any one of these can be accessed individually using `$` as follows:

```
> head(lungcap$Age)     # Show first six values of  Age
[1] 3 4 4 4 4 4
> tail(lungcap$Gender)  # Show last six values of  Gender
[1] M M M M M M
Levels: F M
```

Table 1.1 The forced expiratory volume (FEV) of youths, sampled from East Boston during the middle to late 1970s. FEV is in L; age is in completed years; height is in inches. The complete data set consists of 654 observations in total (Example 1.1)

Non-smokers						Smokers					
Females			Males			Females			Males		
FEV	Age	Height	FEV	Age	Height	FEV	Age	Height	FEV	Age	Height
1.072	3	46.0	1.404	3	51.5	2.975	10	63.0	1.953	9	58.0
0.839	4	48.0	0.796	4	47.0	3.038	10	65.0	3.498	10	68.0
1.102	4	48.0	1.004	4	48.0	2.387	10	66.0	1.694	11	60.0
1.389	4	48.0	1.789	4	52.0	3.413	10	66.0	3.339	11	68.5
1.577	4	49.0	1.472	5	50.0	3.120	11	61.0	4.637	11	72.0
1.418	4	49.0	2.115	5	50.0	3.169	11	62.5	2.304	12	66.5
1.569	4	50.0	1.359	5	50.5	3.102	11	64.0	3.343	12	68.0
1.196	5	46.5	1.776	5	51.0	3.069	11	65.0	3.751	12	72.0
1.400	5	49.0	1.452	5	51.0	2.953	11	67.0	4.756	13	68.0
1.282	5	49.0	1.930	5	51.0	3.104	11	67.5	4.789	13	69.0
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮

The length of any one variable is found using `length()`:

```
> length(lungcap$Age)
[1] 654
```

The dimension of the data set is:

```
> dim(lungcap)
[1] 654   5
```

That is, there are 654 cases and 5 variables. □

For these data, the sample size, usually denoted as n, is $n = 654$. Each youth's information is recorded in one row of the R data frame. FEV is called the *response variable* (or the *dependent variable*) since FEV is assumed to change in response to (or depends on) the values of the other variables. The response variable is usually denoted by y. In Example 1.1, y refers to 'FEV (in litres)'. When necessary, y_i refers to the ith value of the response. For example, $y_1 = 1.072$ in Table 1.1. Occasionally it is convenient to refer to all the observations y_i together instead of one at a time.

The other variables—age, height, gender and smoking status—can be called candidate variables, carriers, exogenous variables, independent variables, input variables, predictors, or regressors. We call these variables *explanatory variables* in this book. Explanatory variables are traditionally denoted by x. In Example 1.1, let x_1 refer to age (in completed years), and x_2 refer to height (in inches). When necessary, the value of, say, x_2 for Observation i is denoted x_{2i}; for example, $x_{2,1} = 46$.

Distinguishing between quantitative and qualitative explanatory variables is essential. Explanatory variables that are qualitative, like gender, are called *factors*. Gender is a factor with two *levels*: `F` (female) and `M` (male). Explanatory variables that are quantitative, like height and age, are called *covariates*.

Often, the key question of interest in an analysis concerns the relationship between the response variable and one or more explanatory variables, though other explanatory variables are present and may also influence the response. Adjusting for the effects of other correlated variables is often necessary, so as to understand the effect of the variable of key interest. These other variables are sometimes called *extraneous variables*. For example, we may be interested in the relationship between FEV (as the response variable) and smoking status (as the explanatory variable), but acknowledge that age, height and gender may also influence FEV. Age, height and gender are extraneous variables.

Example 1.2. Viewing the *structure* of a data frame can be informative:

```
> str(lungcap)          # Show the *structure* of the data frame
'data.frame':        654 obs. of  5 variables:
 $ Age   : int  3 4 4 4 4 4 4 5 5 5 ...
 $ FEV   : num  1.072 0.839 1.102 1.389 1.577 ...
 $ Ht    : num  46 48 48 48 49 49 50 46.5 49 49 ...
 $ Gender: Factor w/ 2 levels "F","M": 1 1 1 1 1 1 1 1 1 1 ...
 $ Smoke : int  0 0 0 0 0 0 0 0 0 0 ...
```

The size of the data frame is given, plus information about each variable: `Age` and `Smoke` consists of integers, `FEV` and `Ht` are numerical, while `Gender` is a factor with two *levels*. Each variable can be summarized numerically using `summary()`:

```
> summary(lungcap)     # Summarize the data
      Age              FEV              Ht        Gender
 Min.   : 3.000   Min.   :0.791   Min.   :46.00   F:318
 1st Qu.: 8.000   1st Qu.:1.981   1st Qu.:57.00   M:336
 Median :10.000   Median :2.547   Median :61.50
 Mean   : 9.931   Mean   :2.637   Mean   :61.14
 3rd Qu.:12.000   3rd Qu.:3.119   3rd Qu.:65.50
 Max.   :19.000   Max.   :5.793   Max.   :74.00
     Smoke
 Min.   :0.00000
 1st Qu.:0.00000
 Median :0.00000
 Mean   :0.09939
 3rd Qu.:0.00000
 Max.   :1.00000
```

Notice that quantitative variables are summarized differently to qualitative variables. `FEV`, `Age` and `Ht` (all quantitative) are summarized with the minimum and maximum values, the first and third quartiles, and the mean and median. `Gender` (qualitative) is summarised by giving the number of males and females in the data. The variable `Smoke` is qualitative, and numbers are used to designate the levels of the variable. In this case, R has no way of determining if the variable is a factor or not, and assumes the variable is quantitative by default since it consists of numbers. To explicitly tell R that `Smoke` is qualitative, use `factor()`:

```
> lungcap$Smoke <- factor(lungcap$Smoke,
                  levels=c(0, 1),                  # The values of  Smoke
                  labels=c("Non-smoker","Smoker")) # The labels
> summary(lungcap$Smoke)     # Now, summarize the redefined variable Smoke
Non-smoker     Smoker
       589         65
```

(The information about the data set, accessed using `?lungcap`, explains that `0` represents non-smokers and `1` represents smokers.) We notice that non-smokers outnumber smokers. □

1.3 Plotting Data

Understanding the lung capacity data is difficult because there is so much data. How can the impact of age, height, gender and smoking status on FEV be understood? Plots (Fig. 1.1) may reveal many, but probably not all, important features of the data:

```
> plot( FEV ~ Age, data=lungcap,
       xlab="Age (in years)",      # The x-axis label
       ylab="FEV (in L)",          # The y-axis label
       main="FEV vs age",          # The main title
       xlim=c(0, 20),              # Explicitly set x-axis limits
       ylim=c(0, 6),               # Explicitly set y-axis limits
       las=1)                      # Makes axis labels horizontal
```

This R code uses the `plot()` command to produce plots of the data. (For more information on plotting in R, see Sect. A.3.10.) The formula `FEV ~ Age` is read as '`FEV` is modelled by `Age`'. The input `data=lungcap` indicates that `lungcap` is the data frame in which to find the variables `FEV` and `Age`. Continue by plotting `FEV` against the remaining variables:

```
> plot( FEV ~ Ht, data=lungcap, main="FEV vs height",
       xlab="Height (in inches)", ylab="FEV (in L)",
       las=1, ylim=c(0, 6) )
> plot( FEV ~ Gender, data=lungcap,
       main="FEV vs gender", ylab="FEV (in L)",
       las=1, ylim=c(0, 6))
> plot( FEV ~ Smoke,  data=lungcap, main="FEV vs Smoking status",
       ylab="FEV (in L)", xlab="Smoking status",
       las=1, ylim=c(0, 6))
```

(Recall that `Smoke` was declared a factor in Example 1.2.) Notice that R uses different types of displays for plotting FEV against covariates (top panels) than against factors (bottom panels). Boxplots are used (by default) for plotting FEV against factors: the solid horizontal centre line in each box represents the median (not the mean), and the limits of the central box represent the upper and lower quartiles of the data (approximately 75% of the observations are less than the upper quartile, and approximately 25% of the observations are less than the lower quartile). The lines from the central box extend to the largest and smallest values, except for outliers which are indicated by individual points (such as a large FEV for a few smokers). In R, outliers are defined, by default, as observations more than 1.5 times the interquartile range (the difference between the upper and lower quartiles) more extreme than the upper or lower limits of the central box.

The plots (Fig. 1.1) show a moderate relationship (reasonably large variation) between FEV and age, that is possibly linear (at least until about 15 years of age). However, a stronger relationship (less variation) is apparent between FEV and height, but this relationship does not appear to be linear.

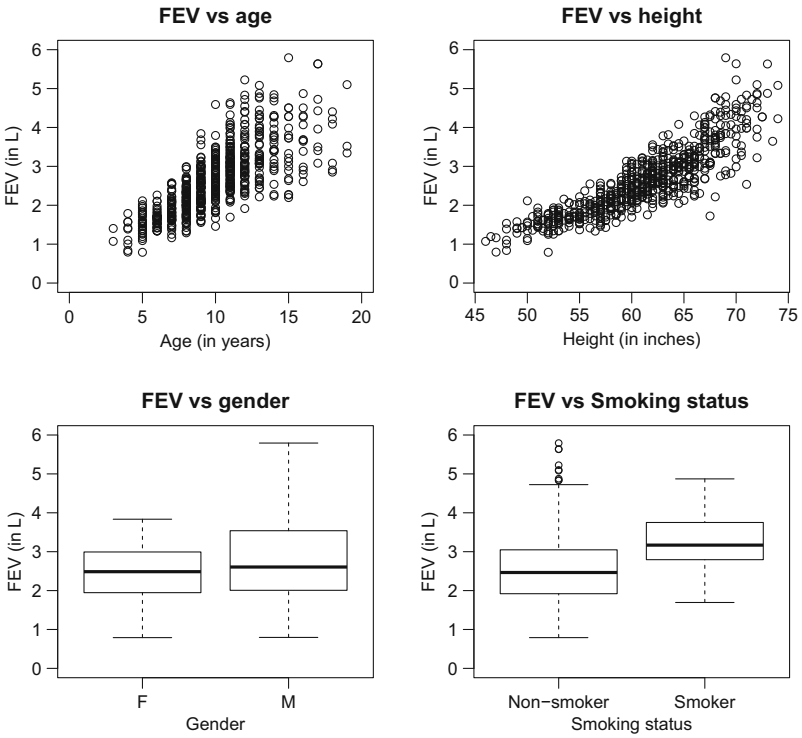

Fig. 1.1 Forced expiratory volume (FEV) plotted against age (top left), height (top right), gender (bottom left) and smoking status (bottom right) for the data in Table 1.1 (Sect. 1.3)

The variation in FEV appears to increase for larger values of FEV also. In general, it also appears that males have a slightly larger FEV, and show greater variation in FEV, than females. Smokers appear to have a larger FEV than non-smokers.

While many of these statements are expected, the final statement is surprising, and may suggest that more than one variable should be examined at once. The plots in Fig. 1.1 only explore the relationships between FEV and each explanatory variable individually, so we continue by exploring relationships involving more than two variables at a time.

One way to do this is to plot the data separately for smokers and non-smokers (Fig. 1.2), using similar scales on the axes to enable comparisons:

```
> plot( FEV ~ Age,
    data=subset(lungcap, Smoke=="Smoker"),  # Only select smokers
    main="FEV vs age\nfor smokers",         # \n means `new line'
    ylab="FEV (in L)", xlab="Age (in years)",
    ylim=c(0, 6), xlim=c(0, 20), las=1)
```

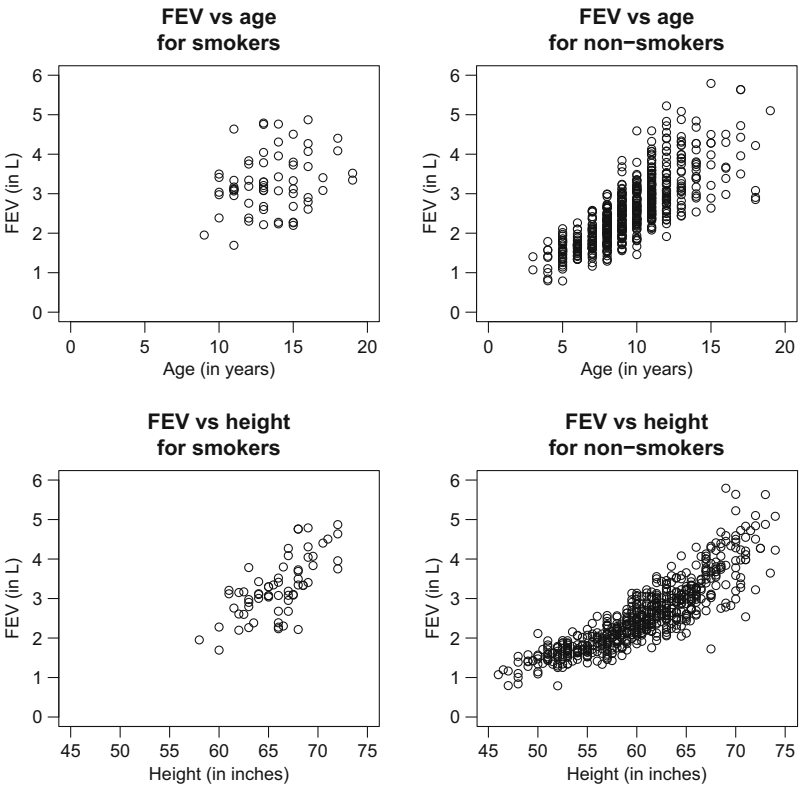

Fig. 1.2 Plots of the lung capacity data: the forced expiratory volume (FEV) plotted against age, for smokers (top left panel) and non-smokers (top right panel); and the forced expiratory volume (FEV) plotted against height, for smokers (bottom left panel) and non-smokers (bottom right panel) (Sect. 1.3)

```
> plot( FEV ~ Age,
    data=subset(lungcap, Smoke=="Non-smoker"),  # Only select non-smokers
    main="FEV vs age\nfor non-smokers",
    ylab="FEV (in L)", xlab="Age (in years)",
    ylim=c(0, 6), xlim=c(0, 20), las=1)
> plot( FEV ~ Ht, data=subset(lungcap, Smoke=="Smoker"),
    main="FEV vs height\nfor smokers",
    ylab="FEV (in L)", xlab="Height (in inches)",
    xlim=c(45, 75), ylim=c(0, 6), las=1)
> plot( FEV ~ Ht, data=subset(lungcap, Smoke=="Non-smoker"),
    main="FEV vs height\nfor non-smokers",
    ylab="FEV (in L)",  xlab="Height (in inches)",
    xlim=c(45, 75), ylim=c(0, 6), las=1)
```

Note that == is used to make logical comparisons. The plots show that smokers tend to be older (and hence taller) than non-smokers and hence are likely to have a larger FEV.

Another option is to distinguish between smokers and non-smokers when plotting the FEV against Age. For these data, there are so many observations that distinguishing between smokers and non-smokers is difficult, so we first adjust Age so that the values for smokers and non-smokers are slightly separated:

```
> AgeAdjust <- lungcap$Age + ifelse(lungcap$Smoke=="Smoker", 0, 0.5)
```

The code ifelse(lungcap$Smoke=="Smoker", 0, 0.5) adds zero to the value of Age for youth labelled with Smoker, and adds 0.5 to youth labelled otherwise (that is, non-smokers). Then we plot FEV against this variable: (Fig. 1.3, top left panel):

```
> plot( FEV ~ AgeAdjust, data=lungcap,
    pch = ifelse(Smoke=="Smoker", 3, 20),
    xlab="Age (in years)", ylab="FEV (in L)", main="FEV vs age", las=1)
```

The input pch indicates the plotting character to use when plotting; then, ifelse(Smoke=="Smoker", 3, 20) means to plot with plotting character 3 (a 'plus' sign) if Smoke takes the value "Smoker", and otherwise to plot with plotting character 20 (a filled circle). See ?points for an explanation of the numerical codes used to define different plotting symbols. Recall that in Example 1.2, Smoke was declared as a factor with two levels that were labelled Smoker and Non-smoker. The legend() command produces the legend:

```
> legend("topleft", pch=c(20, 3), legend=c("Non-smokers","Smokers") )
```

The first input specifies the location (such as "center" or "bottomright"). The second input gives the plotting notation to be explained (such as the points, using pch, or the line types, using lty). The legend input provides the explanatory text. Use ?legend for more information.

A boxplot can also be used to show relationships (Fig. 1.3, top right panel):

```
> boxplot(lungcap$FEV ~ lungcap$Smoke + lungcap$Gender,
    ylab="FEV (in L)", main="FEV, by gender\n and smoking status",
    las=2,    # Keeps labels perpendicular to the axes
    names=c("F:\nNon", "F:\nSmoker", "M:\nNon", "M:\nSmoker"))
```

Another way to show the relationship between three variables is to use an *interaction plot*, which shows the relationship between the levels of two factors and (by default) the mean response of a quantitative variable. The appropriate R function is interaction.plot() (Fig. 1.3, bottom panels):

```
> interaction.plot( lungcap$Smoke, lungcap$Gender, lungcap$FEV,
                    xlab="Smoking status", ylab="FEV (in L)",
                    main="Mean FEV, by gender\n and smoking status",
                    trace.label="Gender", las=1)
> interaction.plot( lungcap$Smoke, lungcap$Gender, lungcap$Age,
                    xlab="Smoking status", ylab="Age (in years)",
                    main="Mean age, by gender\n and smoking status",
                    trace.label="Gender", las=1)
```

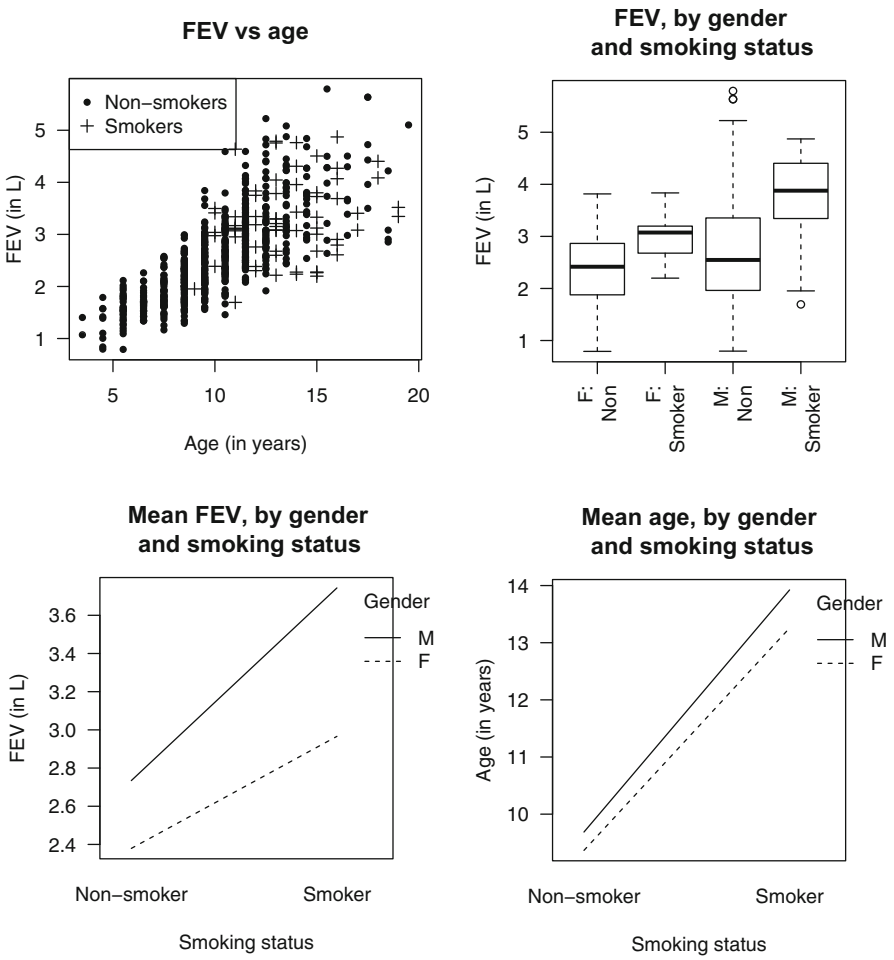

Fig. 1.3 Plots of the lung capacity data: the forced expiratory volume (FEV) plotted against age, using different plotting symbols for non-smokers and smokers (top left panel); a boxplot of FEV against gender and smoking status (top right panel); an interaction plot of the mean FEV against smoking status according to gender (bottom left panel); and an interaction plot of the mean age against smoking status according to gender (bottom right panel) (Sect. 1.3)

This plot shows that, in general, smokers have a larger FEV than non-smokers, for both males and females. The plot also shows that the mean age of smokers is higher for both males and females.

To make any further progress quantifying the relationship between the variables, mathematics is necessary to create a *statistical model*.

1.4 Coding for Factors

Factors represent categories (such as smokers or non-smokers, or males and females), and so must be coded numerically to be used in mathematical models. This is achieved by using *dummy variables*.

The variable `Gender` in the `lungcap` data frame is loaded as a factor by default, as the data are non-numerical:

```
> head(lungcap$Gender)
[1] F F F F F F
Levels: F M
```

To show the coding used by R for the variable `Gender` in the `lungcap` data set, use `contrasts()`:

```
> contrasts(lungcap$Gender)
  M
F 0
M 1
```

(The function name is because, under certain conditions, the codings are called contrasts.) The output shows the two levels of `Gender` on the left, and the name of the dummy variable across the top. When the dummy variable `M` is equal to one, the dummy variable refers males. Notice `F` is not listed across the top of the output as a dummy variable, since it is the *reference level*. By default in R, the reference level is the first level alphabetically or numerically. In other words, the dummy variable, say x_3, is:

$$x_3 = \begin{cases} 0 & \text{if } \texttt{Gender} \text{ is } \texttt{F} \text{ (females)} \\ 1 & \text{if } \texttt{Gender} \text{ is } \texttt{M} \text{ (males).} \end{cases} \tag{1.1}$$

Since these numerical codes are arbitrarily assigned, other levels may be set as the reference level in R using `relevel()`:

```
> contrasts( relevel( lungcap$Gender, "M") )  # Now, M is the ref. level
  F
M 0
F 1
```

As seen earlier in Example 1.2, the R function `factor()` is used to explicitly declare a variable as a factor when necessary (for example, if the data use numbers to designate the factor levels):

```
> lungcap$Smoke <- factor(lungcap$Smoke,
                          levels=c(0, 1),
                          labels=c("Non-smoker","Smoker"))
> contrasts(lungcap$Smoke)
           Smoker
Non-smoker      0
Smoker          1
```

This command assigns the values of `0` and `1` to the labels `Non-smoker` and `Smoker` respectively:

$$x_4 = \begin{cases} 0 & \text{if } \texttt{Smoke} \text{ is } \texttt{0} \text{ (non-smoker)} \\ 1 & \text{if } \texttt{Smoke} \text{ is } \texttt{1} \text{ (smokers).} \end{cases} \tag{1.2}$$

For a factor with k levels, $k-1$ dummy variables are needed. For example, if smoking status had three levels (for example, 'Never smoked', 'Former smoker', 'Current smoker'), then two dummy variables are needed:

$$x_5 = \begin{cases} 1 & \text{for former smokers} \\ 0 & \text{otherwise;} \end{cases} \qquad x_6 = \begin{cases} 1 & \text{for current smokers} \\ 0 & \text{otherwise.} \end{cases} \tag{1.3}$$

Then $x_5 = x_6 = 0$ uniquely refers to people who have never smoked.

The coding discussed here is called *treatment coding*. Many types of coding exist to numerically code factors. Treatment coding is commonly used (and is used in this book, and in R by default) since it usually leads to a direct interpretation. Other codings are also possible, with different interpretations useful in different contexts. In any analysis, the definition of the dummy variables being used should be made clear.

1.5 Statistical Models Describe Both Random and Systematic Features of Data

Consider again the lung capacity data from Example 1.1 (p. 1). At any given combination of height, age, gender and smoking status, many different values of FEV could be recorded, and so produce a *distribution* of recorded FEV values. A model for this distribution of values is called the *random component* of the statistical model. At this given combination of height, age, gender and smoking status, the distribution of FEV values has a mean FEV. The mathematical relationship between the mean FEV and given values of height, age, gender and smoking status is called the *systematic component* of the model. A statistical model consists of a random component and a systematic component to explain these two features of real data. In this context, the *role* of a statistical model is to mathematically represent both the systematic and random components of data.

Many systematic components for the lung capacity data are possible. One simple systematic component is

$$\mu_i = \beta_0 + \beta_1 x_{1i} + \beta_2 x_{2i} + \beta_3 x_{3i} + \beta_4 x_{4i} \tag{1.4}$$

for Observation i, where μ_i is the *expected value* of y_i, so that $\mu_i = \mathrm{E}[y_i]$ for $i = 1, 2, \ldots, n$. The β_j (for $j = 0, 1, 2, 3$ and 4) are unknown *regression parameters*. The explanatory variables are age x_1, height x_2, the dummy

variable x_3 defined in (1.1) for gender, and the dummy variable x_4 defined in (1.2) for smoking status. This is likely to be a poor systematic component, as the plots (Fig. 1.1) show that the relationship between FEV and height is non-linear, for example. Other systematic components are also possible.

The randomness about this systematic component may take many forms. For example, using $\text{var}[y_i] = \sigma^2$ assumes that the variance of the responses y_i is constant about μ_i, but makes no assumptions about the distribution of the responses. A popular assumption is to assume the responses have a normal distribution about the mean μ_i with constant variance σ^2, written $y_i \sim N(\mu_i, \sigma^2)$, where '$\sim$' means 'is distributed as'. Both assumptions are likely to be poor for the lung capacity data, as the plots (Fig. 1.1) show that the variation in the observed FEV increases for larger values of FEV. Other assumptions are also possible, such as assuming the responses come from other probability distributions beside the normal distribution.

1.6 Regression Models

The systematic component (1.4) for the lung capacity data is one possible representation for explaining how the mean FEV changes as height, age, gender and smoking status vary. Many other representation are also possible. Very generally, a *regression model* assumes that the mean response μ_i for Observation i depends on the p explanatory variables x_{1i} to x_{pi} via some general function f through a number of regression parameters β_j (for $j = 0, 1, \ldots q$). Mathematically,

$$\text{E}[y_i] = \mu_i = f(x_{1i}, \ldots, x_{pi}; \beta_0, \beta_1, \ldots, \beta_q).$$

Commonly, the parameters β_j are assumed to combine the effects of the explanatory variables linearly, so that the systematic component often takes the more specific form

$$\mu_i = f(\beta_0 + \beta_1 x_{1i} + \cdots + \beta_p x_{pi}). \tag{1.5}$$

Regression models with this form (1.5) are *regression models linear in the parameters*. All the models discussed in this book are regression models linear in the parameters. The component $\beta_0 + \beta_1 x_{1i} + \cdots + \beta_p x_{pi}$ is called the *linear predictor*.

Two special types of regression models linear in the parameters are discussed in detail in this book:

- Linear regression models: The systematic component of a linear regression model assumes the form

$$\text{E}[y_i] = \mu_i = \beta_0 + \beta_1 x_{1i} + \cdots + \beta_p x_{pi}, \tag{1.6}$$

while the randomness is assumed to have constant variance σ^2 about μ_i. Linear regression models are formally defined and discussed in Chaps. 2 and 3.

- Generalized linear models: The systematic component of a generalized linear model assumes the form

$$\begin{aligned} \mu_i &= g^{-1}(\beta_0 + \beta_1 x_{1i} + \cdots + \beta_p x_{pi}) \\ \text{or alternatively: } g(\mu_i) &= \beta_0 + \beta_1 x_{1i} + \cdots + \beta_p x_{pi} \end{aligned}$$

where $g()$ (called a *link function*) is a monotonic, differentiable function (such as a logarithm function). The randomness is explained by assuming y has a distribution from a specific family of probability distributions (which includes common distributions such as the normal, Poisson and binomial as special cases). Generalized linear models are discussed from Chap. 5 onwards. An example of a generalized linear model appears in Example 1.5. Linear regression models are a special case of generalized linear models.

The following notational conventions apply to regression models linear in the parameters:

- The number of explanatory variables is p: x_1, x_2, $\ldots x_p$.
- The number of regression parameters is denoted p'. If a constant term β_0 is in the systematic component (as is almost always the case) then $p' = p+1$, and the regression parameters are β_0, β_1, $\ldots \beta_p$. If a constant term β_0 is *not* in the systematic component then $p' = p$, and the regression parameters are β_1, β_2, $\ldots \beta_p$.

Example 1.3. For the `lungcap` data (Example 1.1, p. 1), a possible systematic component is given in (1.4) for some numerical values of β_0, β_1, β_2, β_3 and β_4, for $i = 1, 2, \ldots, 654$. This systematic relationship implies a *linear* relationship between μ and the covariates `Age` x_1 (which may be reasonable from Fig. 1.1, top left panel), and `Height` x_2, (which is probably *not* reasonable from Fig. 1.1, top right panel). The model has $p = 4$ explanatory variables, and $p' = 5$ unknown regression parameters.

One model for the random component, suggested in Sect. 1.5, was that the variation of the observations about this systematic component was assumed to be approximately constant, so that $\text{var}[y_i] = \sigma^2$. Combining the two components, a possible linear regression model for modelling the FEV is

$$\begin{cases} \text{var}[y_i] = \sigma^2 & \text{(random component)} \\ \mu_i = \beta_0 + \beta_1 x_{1i} + \beta_2 x_{2i} + \beta_3 x_{3i} + \beta_4 x_{4i} & \text{(systematic component).} \end{cases} \tag{1.7}$$

Often the subscripts i are dropped for simplicity when there is no ambiguity. The values of the parameters β_0, β_1, β_2, β_3, β_4 (for the systematic component) and σ^2 (for the random component) are unknown, and must be estimated.

This is the model implied in Sect. 1.5, where it was noted that both the systematic and random components in (1.7) are likely to be inappropriate for these data (Fig. 1.1). □

Example 1.4. Some other possible systematic components involving FEV (y), age (x_1), height (x_2), gender (x_3) and smoking status (x_4) include:

$$\mu = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_4 x_4 \tag{1.8}$$

$$\mu = \beta_0 + \beta_2 x_2 + \beta_3 x_2^2 + \beta_4 x_4 \tag{1.9}$$

$$\mu = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \beta_4 x_4 \tag{1.10}$$

$$\mu = \beta_0 + \beta_1 \log x_1 + \beta_2 x_2 + \beta_4 x_4 \tag{1.11}$$

$$\mu = \beta_0 + \beta_2 x_2 + \beta_3 x_1 x_2 + \beta_4 x_4 \tag{1.12}$$

$$1/\mu = \beta_1 x_1 + \beta_2 x_2 + \beta_4 x_4 \tag{1.13}$$

$$\log \mu = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_4 x_4 \tag{1.14}$$

$$\mu = \beta_0 + \exp(\beta_1 x_1) - \exp(\beta_2 x_2) + \beta_4 x_4^2 \tag{1.15}$$

All these systematic components apart from (1.15) are linear in the parameters and could be used as the systematic component of a generalized linear model. Only (1.8)–(1.12) could be used to specify a linear regression model. □

Example 1.5. The noisy miner is a small but aggressive native Australian bird. A study [11] of the habitats of the noisy miner recorded (Table 1.2; data set: `nminer`) the abundance of noisy miners (that is, the number observed; `Minerab`) in two hectare transects located in buloke woodland patches with varying numbers of eucalypt trees (`Eucs`). To plot the data (Fig. 1.4), a small amount of randomness is first added in the vertical direction to avoid over plotting, using `jitter()`:

```
> data(nminer)      # Load the data
> names(nminer)     # Show the variables
[1] "Miners"  "Eucs"    "Area"    "Grazed"  "Shrubs"  "Bulokes" "Timber"
[8] "Minerab"
> plot( jitter(Minerab) ~ Eucs, data=nminer, las=1, ylim=c(0, 20),
   xlab="Number of eucalypts per 2 ha", ylab="Number of noisy miners" )
```

See `?nminer` for more information about the data and the other variables.

The random component certainly does not have constant variance, as the observations are more spread out for a larger numbers of eucalypts. Because the responses are counts, a *Poisson distribution* with mean μ_i for Observation i may be suitable for modelling the data. We write $y_i \sim \text{Pois}(\mu_i)$, where $\mu_i > 0$.

The relationship between μ and the number of eucalypts also seems nonlinear. A possible model for the systematic component is $\text{E}[y_i] = \mu_i = \exp(\beta_0 + \beta_1 x_i)$, where x_i is the number of eucalypt trees at location i. This

Table 1.2 The number of eucalypt trees and the number of noisy miners observed in two hectare transects in buloke woodland patches within the Wimmera Plains of western Victoria, Australia (Example 1.5)

Number of eucalypts	Number of noisy miners	Number of eucalypts	Number of noisy miners	Number of eucalypts	Number of noisy miners
2	0	32	19	0	0
10	0	2	0	0	0
16	3	16	2	0	0
20	2	7	0	3	0
19	8	10	3	8	0
18	1	15	1	8	0
12	8	30	7	15	0
16	5	4	1	21	3
3	0	4	0	24	4
12	4	19	7	15	6
		11	0		

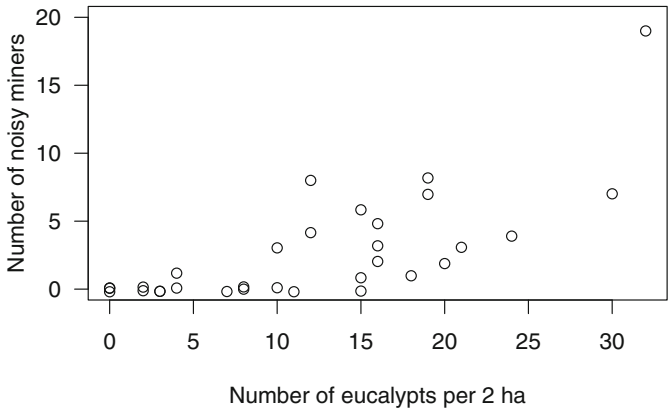

Fig. 1.4 The number of noisy miners (observed in two hectare transects in buloke woodland patches within the Wimmera Plains of western Victoria, Australia) plotted against the number of eucalypt trees. A small amount of randomness is added to the number of miners in the vertical direction to avoid over-plotted observations (Example 1.5)

functional form ensures $\mu_i > 0$, as required for the Poisson distribution, and may also be appropriate for modelling the non-linearity.

Combining the two components, one possible model for the data, dropping the subscripts i, is:

$$\begin{cases} y \sim \text{Pois}(\mu) & \text{(random component)} \\ \mu = \exp(\beta_0 + \beta_1 x) & \text{(systematic component)} \end{cases} \tag{1.16}$$

where $\mu = \text{E}[y]$. This is an example of a *Poisson generalized linear model* (Chap. 10).

We also note that one location (with 19 noisy miners) has more than twice the number of noisy miners observed than the location with the next largest number of noisy miners (with eight noisy miners). □

1.7 Interpreting Regression Models

Models are most useful when they have sensible interpretations. Compare these two systematic components:

$$\mu = \beta_0 + \beta_1 x \tag{1.17}$$

$$\log \mu = \beta_0 + \beta_1 x. \tag{1.18}$$

The first model (1.17) assumes a linear relationship between μ and x, and hence that an increase of one in the value of x is associated with an increase of β_1 in the value of μ. The second model (1.18) assumes a linear relationship between $\log \mu$ and x, and hence that an increase of one in the value of x will increase the value of $\log \mu$ by β_1. This implies that when the value of x increases by one, μ increases (approximately) by a *factor* of $\exp(\beta_1)$. To see this, write the second systematic component (1.18) as

$$\mu_x = \exp(\beta_0 + \beta_1 x) = \exp(\beta_0)\exp(\beta_1)^x.$$

Hence if the value of x increases by 1, to $x + 1$, we have

$$\mu_{x+1} = \exp(\beta_0)\exp(\beta_1)^{x+1} = \mu_x \exp(\beta_1).$$

A researcher should consider which is more sensible for the application. Furthermore, models that are based on underlying theory or sensible approximations to the problem (Sect. 1.10) produce models with better and more meaningful interpretations. Note that the systematic component (1.17) is suitable for a linear regression model, and that both systematic components are suitable for a generalized linear model.

Example 1.6. For the `lungcap` data, consider a model relating FEV y to height x. Model (1.17) would imply that an increase in height of one inch is associated with an increase in FEV of β_1 L. In contrast, Model (1.18) would imply that an increase in height of one inch is associated with an increase in FEV by a factor of $\exp(\beta_1)$ L. □

A further consideration when interpreting models is when models contain more than one explanatory variable. In these situations, the regression parameters should be interpreted with care, since the explanatory variables may not be independent. For example, for the lung capacity data, the age and height of youth are related (Fig. 1.5): older youth are taller, on average:

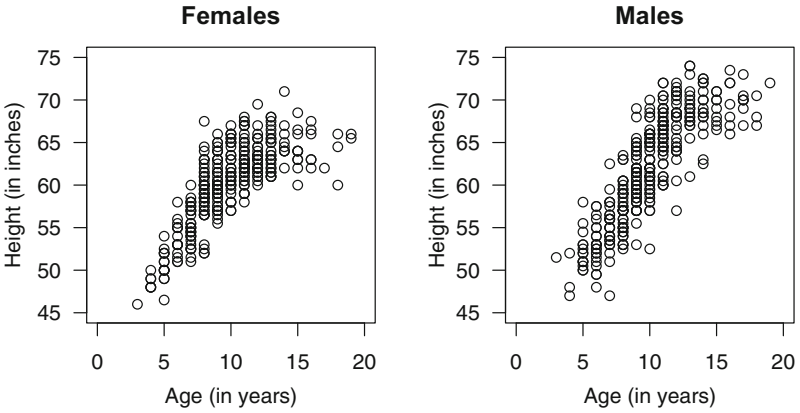

Fig. 1.5 A strong relationship exists between the height and the age of the youth in the lung capacity data: females (left panel) and males (right panel)

```
> plot( Ht ~ Age, data=subset(lungcap, Gender=="F"), las=1,
    ylim=c(45, 75), xlim=c(0, 20), # Use similar scales for comparisons
    main="Females", xlab="Age (in years)", ylab="Height (in inches)" )
> plot( Ht ~ Age, data = subset(lungcap, Gender=="M"), las=1,
    ylim=c(45, 75), xlim=c(0, 20), # Use similar scales for comparisons
    main="Males", xlab="Age (in years)", ylab="Height (in inches)" )
```

In a model containing both age and height, it is not possible to interpret both regression parameters independently, as expecting age to change while height stays constant is unreasonable in youth. Note that height tends to increase with age initially, then tends to stay similar as the youth stop (or slow) their growing.

Further comments on model interpretation for specific models are given as appropriate, such as in Sect. 2.7.

1.8 All Models Are Wrong, but Some Are Useful

Previous sections introduced regression models as a way to understand data. However, when writing about statistical models, Box and Draper [2, p. 424] declared "all models are wrong". What do they mean? Were they correct? One way to understand this is to contrast statistical models with some physical models in common use. For example, biologists use *models* of the human skeleton to teach anatomy, which capture enough important information about the real situation for the necessary purpose. Models are not an exact representation of reality: the skeleton is probably made of plastic, not bones; no-one may have a skeleton with the exact dimensions of the model skeleton. However, models *are* useful approximations for representing the necessary detail for the purpose at hand.

Similar principles apply to *statistical models*: they are mathematical approximations to reality that represent the important features of *data* for the task at hand. The complete quote from Box and Draper clarifies [2, p. 424], "... Essentially, all models are wrong, but some are useful. However, the approximate nature of the model must always be borne in mind".

Despite the many similarities between physical and statistical models, two important differences exist:

- A model skeleton shows the structure of an average or typical skeleton, which is equivalent to the systematic component of a statistical model. But no-one has a skeleton exactly like the model: some bones will be longer, skinnier, or a different shape. However, the model skeleton makes no attempt to indicate the *variation* that is present in skeletons in the population. The model skeleton ignores the variation from person to person (the random component). In contrast, the statistical model represents both the systematic trend and the randomness of the data. The random component is modelled explicitly by making precise statements about the random variation (Sect. 1.5).
- Most physical models are based on what is known to be true. Biologists *know* what a typical real skeleton looks like. Consequently, knowing whether a physical model is adequate is generally easy, since the model represents the important, known features of the true situation. However, statistical models are often developed where the *true* model is unknown, or is only artificially assumed to exist. In these cases, the model must be developed from the available data.

1.9 The Purpose of a Statistical Model Affects How It Is Developed: Prediction vs Interpretation

The *role* of a statistical model is to accurately represent the important systematic and random features of the data. But what is the *purpose* of developing statistical models? For regression models, there are two major motivations:

- Prediction: To produce accurate predictions from new or future data.
- Understanding and interpretation: To understand how variables relate to each other.

For example, consider the lung capacity study. The purpose of this study may be to determine whether there is a (potentially causal) relationship between smoking and FEV. Here we want to understand whether smoking has an effect on FEV, and in what direction. For this purpose, the size and significance of coefficients in the model are of interest. If smoking decreases lung function, this would have implications for health policy.

A different health application is to establish the normal weight range for children of a given age and gender. Here the purpose is to be able to judge whether a particular child is out of the normal range, in which case some intervention by health carers might be appropriate. In this case, a prediction curve relating weight to age is desired, but the particular terms in the model would not be of interest. The lung capacity data is in fact an extract from a larger study [19] in which the pulmonary function of the same children was measured at multiple time points (a *longitudinal study*), with the aim of establishing the normal range for FEV at each age.

Being aware of the major purpose of a study may affect how a regression model is fitted and developed. If the major purpose is interpretation, then it is important that all terms are reliably estimated and have good support from the data. If the major purpose is prediction, then any predictor that improves the precision of prediction may be included in the model, even if the causal relationship between the predictor and the response is obscure or if the regression coefficient is relatively uncertain. This means that sometimes one might include more terms in a regression model when the purpose is prediction than when the purpose is interpretation and understanding.

1.10 Accuracy vs Parsimony

For any set of data, there are typically numerous systematic components that could be chosen and various random components may also be possible. How do we choose a statistical model from all the possible options?

Sometimes, statistical models are based on underlying theory, or from an understanding of the physical features of the situation, and are built with this knowledge in mind. In these situations, the statistical model may be critiqued by how well the model explains the known features of the theoretical situation.

Sometimes, approximations to the problem can guide the choice of model. For example, for the lung capacity data, consider lungs roughly as cylinders, whose heights are proportional to the height of the child, and assume the FEV is proportional to lung volume. Then volume $\propto (\text{radius})^2 x_2$ may be a suitable model. This approach implies FEV is proportional to x_2, as in Models (1.8)–(1.11) (p. 14).

Sometimes, statistical models are based on data, often without guiding theory, and no known 'true' state exists with which to compare. After all, statistical models are artificial, mathematical constructs. The model is a representation of an unknown, but assumed, underlying true state. How can we know if the statistical model is adequate?

In general, an adequate statistical model balances two criteria:

- Accuracy: The model should accurately describe both the systematic and random components.
- Parsimony: The model should be as simple as possible.

According to the *principle of parsimony* (or *Occam's Razor*), the simplest accurate model is the preferred model. In other words, prefer the simplest accurate model not contradicting the data. A model too simple or too complex does not model the data well. Complex models may fit the given data well but usually do not generalize well to other data sets (this is called *over-fitting*).

Example 1.7. Figure 1.6 (top left panel) shows the systematic component of a linear model (represented by the solid line) fitted to some data. This model does not represent the systematic trend of the data. The variation around this linear model is large and not random: observations are consistently smaller than the fitted model, then consistently larger, then smaller.

The systematic component of the fitted cubic model (Fig. 1.6, top centre panel) represents the systematic trend of the data, and suggests a small amount of random variation about this trend.

The fitted 10th order polynomial (Fig. 1.6, top right panel) suggests a small amount of randomness, as the polynomial passes close to every observation. However, the systematic polynomial component incorrectly represents both the systematic *and* random components in the data. Because the systematic component also represents the randomness, predictions based on this model are suspect (predictions near $x = -1$ are highly dubious, for example).

The principle of parsimony suggests the cubic model is preferred. This model is simple, accurate, and does not contradict the data. Researchers focused only on producing a model passing close to each observation (and hence selecting the 10th order polynomial) have a poor model. This is called *over-fitting*.

The data were actually generated from the model

$$\begin{cases} y \sim N(\mu, 0.35) \\ \mu = x^3 - 3x + 5. \end{cases}$$

The notation $y \sim N(\mu, 0.35)$ means the responses come from a normal distribution with mean μ and variance $\sigma^2 = 0.35$.

Suppose new data were observed from this same true model (for example, from a new experiment or from a new sample), and linear, cubic and 10th order polynomial models were refitted to this new data (Fig. 1.6, bottom panels). The new fitted linear model (Fig. 1.6, bottom left panel) still does not fit the data well. The new fitted 10th order polynomial (Fig. 1.6, bottom right panel) is very different compared to the one fitted to the first data set, even though the data for both were generated from the same model. In contrast, the new fitted cubic model (Fig. 1.6, bottom centre panel) is very similar for both data sets, suggesting the cubic model represents the systematic and random components well. □

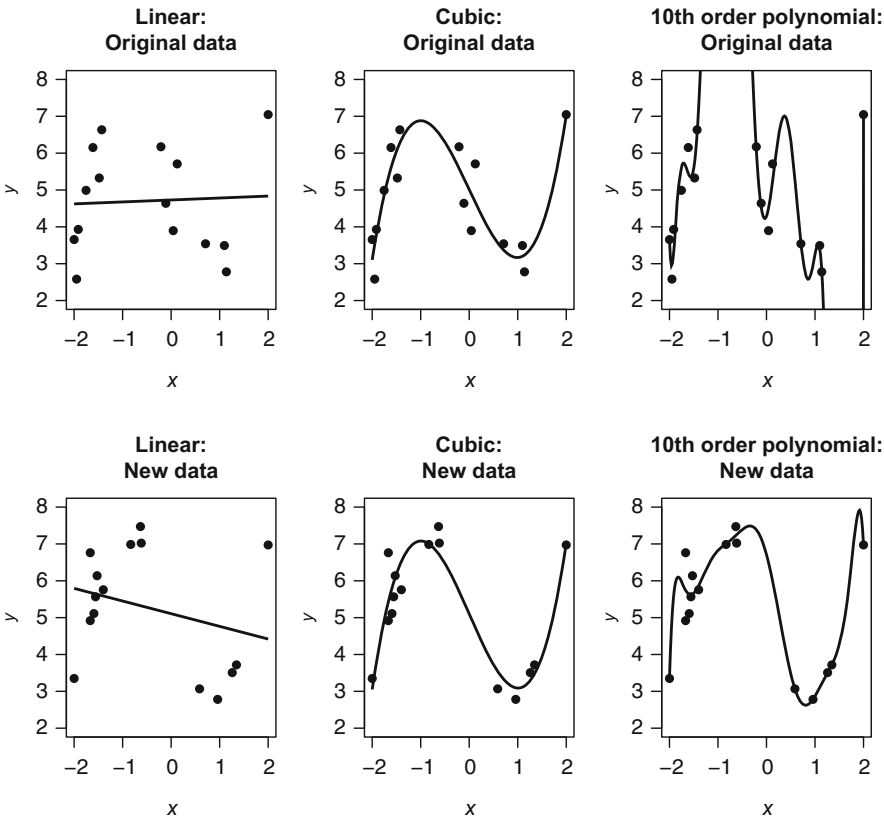

Fig. 1.6 Three different systematic components for an artificial data set. Left panels: the data modelled using a linear model; centre panels: using a cubic model; right panels: using a 10th order polynomial. The lines represent the systematic component of the fitted model. The top panels show the models fitted to some data; the bottom panels shows the models fitted to data randomly generated from the same model used to generate the data in the top panels. A good model would be similar for both sets of data (Example 1.7)

1.11 Experiments vs Observational Studies: Causality vs Association

All models must be used and understood within limitations imposed by how the data were collected. The method of data collection influences the conclusions that can be drawn from the analysis. An important aspect of this concerns whether researchers intervene to apply treatments to subjects or simply observe pre-existing processes.

In an *observational study*, researchers may use elaborate equipment to collect physical measures or may ask subjects to respond to carefully designed questionnaires, but do not influence the processes being observed.

Observational studies generally only permit conclusions about *associations* between variables, not a cause-and-effect. While the relationship may in fact be causal, the use of observational data by itself it not usually sufficient to confirm this conclusion. In contrast, researchers conducting a *designed experiment* do intervene to control the values of the explanatory variables that appear in the data. The distinguishing feature of an experiment versus an observational study is that the researchers conducting the study are able to determine which experimental condition is applied to each subject. A well-designed randomized experiment allows inference to be made about *cause-and-effect* relationships between the explanatory and response variables.

Statistical models treat experimental and observational studies in the same way, and the statistical conclusions are superficially similar, but scientific conclusions from experiments are usually much stronger. In an observational study, the best that can be done is to measure all other extraneous variables that are likely to affect the response, so that the analysis can adjust for as many uncontrolled effects as possible. In this way, good quality data and careful statistical analysis can go a long way towards correcting for many influences that cannot be controlled in the study design.

Example 1.8. The lung capacity data (Example 1.1) is a typical observational study. The purpose of the study is to explore the effects of smoking on lung capacity, as measured by FEV (explored later in Problem 11.15). Whether or not each participant is a smoker is out of the control of the study designers, and there are many physical characteristics, such as age and height, that have direct effects on lung capacity, and some quite probably have larger effects than the effect of interest (that of smoking). Hence it was necessary to record information on the height, age and gender of participants (which become extraneous variables) so that the influence of these variables can be taken into account. The aim of the analysis therefore is to try to measure the association between smoking and lung capacity after adjusting for age, height and gender. It is always possible that there are other important variables that influence FEV that have not been measured, so any association discovered between FEV and smoking should not be assumed to be cause-and-effect. □

1.12 Data Collection and Generalizability

Another feature of data collection that affects conclusions is the population from which the subjects or cases are drawn. In general, conclusions from fitting and analysing a statistical model only apply to the population from which the cases are drawn. So, for example, if subjects are drawn from women aged over 60 in Japan, then conclusions do not necessarily apply to men, to women in Japan aged under 60, or to women aged over 60 elsewhere.

Similarly, the conclusions from a regression model cannot necessarily be applied (extrapolated) outside the range of the data used to build the model.

Example 1.9. The lung capacity data (Example 1.1) is from a sample of youths from the middle to late 1970s in Boston. Using the results to infer information about other times and locations may or may not be appropriate. The study designers might hope that Boston is representative of much of the United States in terms of smoking among youth, but generalizing the results to other countries with different lifestyles or to the present day may be doubtful.

The youths in the FEV study are aged from 3 to 19. As no data exists outside this age range, no statistical model can be verified to apply outside this age range. In the same way, no statistical model applies for youth under 46 inches tall or over 74 inches tall. FEV cannot be expected to increase linearly for all ages and heights. □

1.13 Using R for Statistical Modelling

A computer is indispensable in any serious statistical work for performing the necessary computations (such as estimating the values of β_j), for producing graphics, and for evaluating the final model.

Although the theory and applications of GLMs discussed throughout this book apply generally, the implementation is possible in various statistical computer packages. This book discusses how to perform these analyses using R (all computations in this book are performed in R version 3.4.3). A short introduction to using R is given in Appendix A (p. 503).

This section summarizes and collates some of the relevant R commands introduced in this chapter. For more information on some command `foo`, type `?foo` at the R command prompt.

- `library()`: Loads extra R functionality that is contained in an R package. For example, use `library(GLMsData)` to make the data frames associated with this book available in R. See Appendix B (p. 525) for information about obtaining and installing this package.
- `data()`: Loads data frames.
- `names(x)`: Lists the names of the variables in the data frame `x`.
- `summary(object)`: Produces a summary of the variable `object`, or of the data frame `object`.
- `factor(x)`: Declares `x` as a factor. The first input is the variable to be declared as a factor. Two further inputs are optional. The second (optional) input `levels` is the list of the levels of the factor; by default the levels of the factor are sorted by numerical or alphabetical order. The third (optional) input `labels` gives the labels to assign to the levels of the factor in the order given by `levels` (or the order assumed by default).

- `relevel(x, ref)`: Changes the reference level for factor `x`. The first input is the factor, and the second input `ref` is the level of the factor to use as the reference level.
- `plot()`: Plots data. See Appendix A.3.10 (p. 516) for more information.
- `legend()`: Adds a legend to a plot.

1.14 Summary

Chapter 1 introduces the idea of a statistical model. In this context, y refers to the response variable, n to the number of observations, and $x_1, x_2, \ldots, x_p$ to the p explanatory variables. Quantitative explanatory variables are called covariates; qualitative explanatory variables are called factors (Sect. 1.2). Factors must be *coded* numerically for use in statistical models (Sect. 1.4) using dummy variables. Treatment codings are commonly used, and are used by default in R. $k-1$ dummy variables are required for a factor with k levels.

Plots are useful for an initial examination of data (Sect. 1.3), but statistical models are necessary for better understanding. Statistical models explain the two components of data: The *systematic component* models how the mean response changes as the explanatory variables change; the *random component* models the variation of the data about the mean (Sect. 1.5). In this way, statistical models represent both the systematic and random components of data (Sect. 1.8), and can be used for prediction, and for understanding relationships between variables (Sect. 1.9). Two criteria exist for an adequate model: simplicity and accuracy. The simplest model that accurately describes the systematic component and the randomness is preferred (Sect. 1.10).

Regression models 'linear in the parameters' have a systematic component of the form $\mathrm{E}[y_i] = \mu_i = f(\beta_0 + \beta_1 x_{1i} + \cdots + \beta_p x_{pi})$ (Sect. 1.6). In these models, the number of regression parameters is denoted p'. If a constant term β_0 is in the systematic component, as is almost always the case, then $p' = p + 1$; otherwise $p' = p$ (Sect. 1.6).

Statistical models should be able to be sensibly interpreted (Sect. 1.7). However, fitted models should be interpreted and understood within the limitations of the data and of the model (Sect. 1.11). For example: in observational studies, data are simply observed, and no cause-and-effects conclusions can be drawn. In experimental studies, data are produced when the researcher has some control over the values of at least some of the explanatory variables to use; cause-and-effect conclusions may be drawn (Sect. 1.11). In general, conclusions from fitting and analysing a statistical model only apply to the population represented by the sample (Sect. 1.12).

Computers are invaluable in statistical modelling, especially for estimating parameters and graphing (Sect. 1.13).

Problems

Selected solutions begin on p. 529.

1.1. The plots in Fig. 1.7 (data set: `paper`) show the strength of Kraft paper [7, 8] for different percentages of hardwood concentrations. Which systematic component, if any, appears most suitable for modelling the data? Explain.

1.2. The plots in Fig. 1.8 (data set: `heatcap`) show the heat capacity of solid hydrogen bromide y measured as a function of temperature x [6, 16]. Which systematic component, if any, appears best for modelling the data? Explain.

1.3. Consider the data plotted in Fig. 1.9. In the panels, quadratic, cubic and quartic systematic components are shown with the data. Which systematic component appears best for modelling the data? Explain.

The data are actually randomly generated using the systematic component $\mu = 1 + 10\exp(-x/2)$ (with added randomness), which is not a polynomial at all. Explain what this demonstrates about fitting systematic components.

1.4. Consider the data plotted in Fig. 1.10 (data set: `toxo`). The data show the proportion of the population y testing positive to toxoplasmosis against the annual rainfall x for 34 cities in El Salvador [5]. Analysis suggests a cubic model fits the data reasonably well (though substantial variation still exists). What important features of the data are evident from the plot? Which of the plotted systematic components appears better? Explain.

1.5. For the following systematic components used in a regression model, determine if they are appropriate for regression models linear in the parameters, linear regression models, and/or generalized linear models. In all cases, β_j refers to model parameters, μ is the expected value of the response variable, while x, x_1 and x_2 refer to explanatory variables.

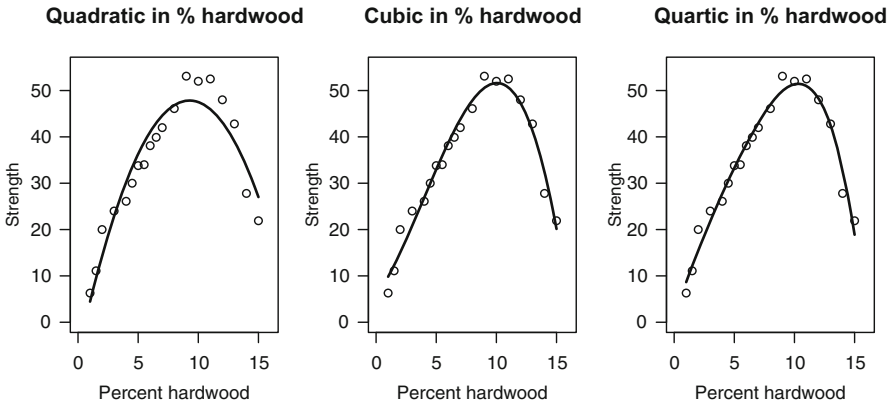

Fig. 1.7 Three different systematic components for the Kraft paper data set: fitted quadratic, cubic and quartic systematic components are shown (Problem 1.1)

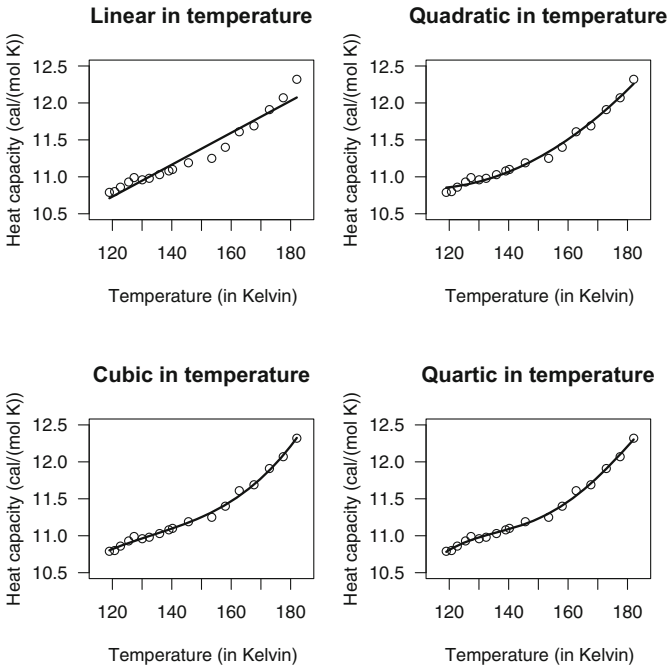

Fig. 1.8 Plots of the heat capacity data: fitted linear, quadratic, cubic and quartic systematic components are shown (Problem 1.2)

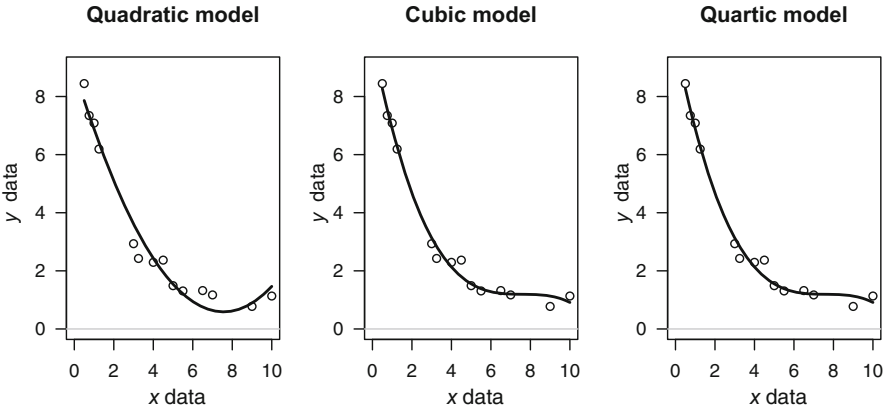

Fig. 1.9 Three different systematic components for a data set: fitted quadratic, cubic and quartic systematic components are shown (Problem 1.3)

1. $\mu = \beta_0 + \beta_1 x_1 + \beta_2 \log x_2$.
2. $\mu = \beta_0 + \exp(\beta_1 + \beta_2 x)$.
3. $\mu = \exp(\beta_0 + \beta_1 x)$ for $\mu > 0$.
4. $\mu = 1/(\beta_0 + \beta_1 x_1 + \beta_2 x_1 x_2)$ for $\mu > 0$.

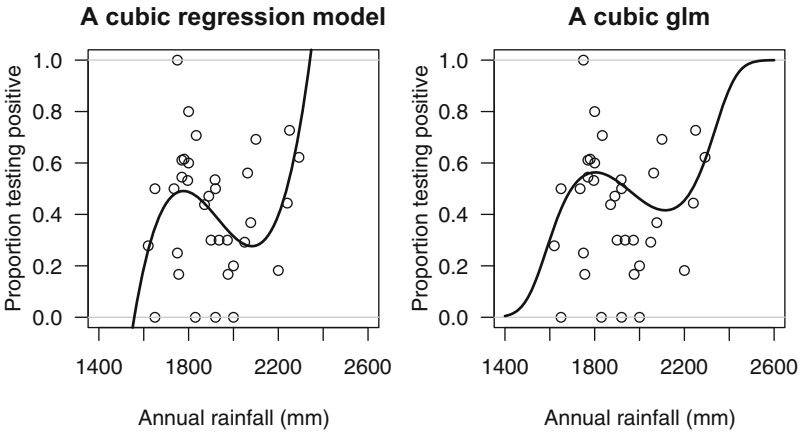

Fig. 1.10 The toxoplasmosis data, and two fitted cubic systematic components (Problem 1.4)

1.6. Load the data frame `turbines` from the package **GLMsData**. Briefly, the data give the proportion of turbines developing fissures after a given number of hours of run-time [13, 14].

1. Use `names()` to determine the names of the variables in the data frame.
2. Determine which variables are quantitative and which are qualitative.
3. For any qualitative variables, define appropriate dummy variables using treatment coding.
4. Use R to summarize each variable.
5. Use R to create a plot of the proportion of failures (turbines with fissures) against run-time.
6. Determine the important features of the data evident from the plot.
7. Would a linear regression model seem appropriate for modelling the data? Explain.
8. Read the help for the data frame (use `?turbines` after loading the **GLMsData** package in R), and determine whether the data come from an observational or experimental study, then discuss the implications.

1.7. Load the data frame `humanfat`. Briefly, the data record the percentage body fat y, age, gender and body mass index (BMI) of 18 adults [12]. The relationship between y and BMI is of primary interest.

1. Use `names()` to determine the names of the variables in the data.
2. Determine which variables are quantitative and which are qualitative. Identify which variables are extraneous variables.
3. For any qualitative variables, define appropriate dummy variables using treatment coding.
4. Use R to summarize each variable.

5. Plot the response against each explanatory variable, and discuss any important features of the data.
6. Would a linear regression model seem appropriate for modelling the data? Explain.
7. Read the help for the data frame (use `?humanfat` after loading the **GLMsData** package in R), and determine whether the data come from an experiment or observational study. Explain the implications.
8. After reading the help, determine the population to which the results can be expected to generalize.
9. Suppose a linear regression model was fitted to the data with systematic component $\mu = \beta_0 + \beta_1 x_1$, where x_1 is BMI. Interpret the systematic component of this model.
10. Suppose a generalized linear model was fitted to the data with systematic component $\log \mu = \beta_0 + \beta_1 x_1 + \beta_2 x_2$, where x_1 is BMI, and x_2 is 0 for females and 1 for males. Interpret the systematic component of this model.
11. For both models given above, determine the values of p and p'.

1.8. Load the data frame `hcrabs`. Briefly, the data give the number of male satellite crabs y attached to female horseshoe crabs of various weights (in g), widths (in cm), colours and spine conditions [1, 3].

1. Determine which variables are quantitative and which are qualitative.
2. For any qualitative variables, define appropriate dummy variables using treatment coding.
3. Use R to summarize each variable.
4. Produce appropriate plots to help understand the data.
5. Find the correlation between weight and width, and comment on the implications.
6. Read the help for the data frame (use `?hcrabs` after loading package **GLMsData** in R), and determine whether the data come from an experiment or observational study. Explain the implications.
7. After reading the help, determine the population to which the results can be expected to generalize.
8. Suppose a linear regression model was fitted to the data with systematic component $\mu = \beta_0 + \beta_1 x_1$, where x_1 is the weight of the crab. Interpret the systematic component of this model. Comment on the suitability of the model.
9. Suppose a generalized linear model was fitted to the data with systematic component $\log \mu = \beta_0 + \beta_1 x_1$, where x_1 is the weight of the crab. Interpret the systematic component of this model. Comment on the suitability of the model.
10. For the model given above, determine the values of p and p'.

1.9. Children were asked to build towers as high as they could out of cubical and cylindrical blocks [9, 17]. The number of blocks used and the time taken were recorded.

1. Load the data frame `blocks` from the package **GLMsData**, and produce a summary of the variables.
2. Produce plots to examine the relationship between the *time* taken to build towers, and the block type, trial number, and age.
3. In words, summarize the relationship between the four variables.
4. Produce plots to examine the relationship between the *number* of blocks used to build towers, and the block type, trial number, and age.
5. Summarize the relationship between the four variables in words.

1.10. In a study of foetal size [15], the mandible length (in mm) and gestational age for 167 foetuses were measured from the 15th week of gestation onwards. Load the data frame `mandible` from the package **GLMsData**, then use R to create a plot of the data.

1. Determine the important features of the data evident from the plot.
2. Is a linear relationship appropriate? Explain.
3. Is a model assuming constant variation appropriate? Explain.

References

[1] Agresti, A.: An Introduction to Categorical Data Analysis, second edn. Wiley-Interscience (2007)
[2] Box, G.E.P., Draper, N.R.: Empirical Model-Building and Response Surfaces. Wiley, New York (1987)
[3] Brockmann, H.J.: Satellite male groups in horseshoe crabs, *limulus polyphemus*. Ethology **102**, 1–21 (1996)
[4] Dunn, P.K., Smyth, G.K.: GLMsData: Generalized linear model data sets (2017). URL https://CRAN.R-project.org/package=GLMsData. R package version 1.0.0
[5] Efron, B.: Double exponential families and their use in generalized linear regression. Journal of the American Statistical Association **81**(395), 709–721 (1986)
[6] Giauque, W.F., Wiebe, R.: The heat capacity of hydrogen bromide from 15°K. to its boiling point and its heat of vaporization. The entropy from spectroscopic data. Journal of the American Chemical Society **51**(5), 1441–1449 (1929)
[7] Hand, D.J., Daly, F., Lunn, A.D., McConway, K.Y., Ostrowski, E.: A Handbook of Small Data Sets. Chapman and Hall, London (1996)
[8] Joglekar, G., Scheunemyer, J.H., LaRiccia, V.: Lack-of-fit testing when replicates are not available. The American Statistician **43**, 135–143 (1989)
[9] Johnson, B., Courtney, D.M.: Tower building. Child Development **2**(2), 161–162 (1931)

[10] Kahn, M.: An exhalent problem for teaching statistics. Journal of Statistical Education **13**(2) (2005)

[11] Maron, M.: Threshold effect of eucalypt density on an aggressive avian competitor. Biological Conservation **136**, 100–107 (2007)

[12] Mazess, R.B., Peppler, W.W., Gibbons, M.: Total body composition by dualphoton (^{153}Gd) absorptiometry. American Journal of Clinical Nutrition **40**, 834–839 (1984)

[13] Myers, R.H., Montgomery, D.C., Vining, G.G.: Generalized Linear Models with Applications in Engineering and the Sciences. Wiley, Chichester (2002)

[14] Nelson, W.: Applied Life Data Analysis. Wiley Series in Probability and Statistics. John Wiley Sons, New York (1982)

[15] Royston, P., Altman, D.G.: Regression using fractional polynomials of continuous covariates: Parsimonious parametric modelling. Journal of the Royal Statistical Society, Series C **43**(3), 429–467 (1994)

[16] Shacham, M., Brauner, N.: Minimizing the effects of collinearity in polynomial regression. Industrial and Engineering Chemical Research **36**, 4405–4412 (1997)

[17] Singer, J.D., Willett, J.B.: Improving the teaching of applied statistics: Putting the data back into data analysis. The American Statistician **44**(3), 223–230 (1990)

[18] Smyth, G.K.: Australasian data and story library (OzDASL) (2011). URL http://www.statsci.org/data

[19] Tager, I.B., Weiss, S.T., Muñoz, A., Rosner, B., Speizer, F.E.: Longitudinal study of the effects of maternal smoking on pulmonary function in children. New England Journal of Medicine **309**(12), 699–703 (1983)

[20] Tager, I.B., Weiss, S.T., Rosner, B., Speizer, F.E.: Effect of parental cigarette smoking on the pulmonary function of children. American Journal of Epidemiology **110**(1), 15–26 (1979)

Chapter 2
Linear Regression Models

Almost all of statistics is linear regression, and most of what is left over is non-linear regression.
Robert Jennrich, in the discussion of Green [4, p. 182]

2.1 Introduction and Overview

The most common of all regression models is the linear regression model, introduced in this chapter. This chapter also introduces the notation and language used in this book so a common foundation is laid for all readers for the upcoming study of generalized linear models: linear regression models are a special case of generalized linear models. We first define linear regression models and introduce the relevant notation and assumptions (Sect. 2.2). We then describe least-squares estimation for simple linear regression models (Sect. 2.3) and multiple regression models (Sects. 2.4 and 2.5). The use of the R functions to fit linear regression models is explained in Sect. 2.6, followed by a discussion of the interpretation of linear regression models (Sect. 2.7). Inference procedures are developed for the regression coefficients (Sect. 2.8), followed by analysis of variance methods (Sect. 2.9). We then discuss methods for comparing nested models (Sect. 2.10), and for comparing non-nested models (Sect. 2.11). Tools to assist in model selection are then described (Sect. 2.12).

2.2 Linear Regression Models Defined

In this chapter, we consider linear regression models for modelling data with a response variable y and p explanatory variables $x_1, x_2, \ldots, x_p$. A linear regression model consists of the usual two components of a regression model (random and systematic components), with specific forms.

The random component assumes that the responses y_i have constant variances σ^2, or that the variances are proportional to known, positive weights w_i; that is, $\text{var}[y_i] = \sigma^2/w_i$ for $i = 1, 2, \ldots n$. The w_i are called *prior weights*,

P. K. Dunn, G. K. Smyth, *Generalized Linear Models with Examples in R*,
Springer Texts in Statistics, https://doi.org/10.1007/978-1-4419-0118-7_2

which provide the possibility of giving more weight to some observations than to others. The systematic component assumes that the expected value of the response $\mathrm{E}[y_i] = \mu_i$ is linearly related to the explanatory variables x_j such that $\mu_i = \beta_0 + \sum_{j=1}^{p} \beta_j x_{ji}$.

Combining these components, a linear regression model has the general form

$$\begin{cases} \mathrm{var}[y_i] = \sigma^2 / w_i \\ \mu_i = \beta_0 + \sum_{j=1}^{p} \beta_j x_{ji} \end{cases} \tag{2.1}$$

where $\mathrm{E}[y_i] = \mu_i$, and the prior weights w_i are known. The *regression parameters* $\beta_0, \beta_1, \ldots, \beta_p$, as well as the error variance σ^2, are unknown and must be estimated from the data. Recall, the number of regression parameters for Model (2.1) is $p' = p + 1$. β_0 is often called the *intercept*, since it is the value of y when all the explanatory variables are zero. The parameters $\beta_1, \ldots \beta_p$ are sometimes called the *slopes* for the corresponding explanatory variables.

A linear regression model with systematic component $\mu = \beta_0 + \beta_1 x_1$ (that is, $p = 1$ and $p' = 2$) is called a *simple linear regression model* or a *simple regression model*. A linear regression model with all prior weights w_i set to one is called an *ordinary linear regression model*, to be distinguished from a *weighted linear regression model* when the prior weights are not all one. A linear regression model with $p > 1$ is often called a *multiple linear regression model* or *multiple regression model*. Figure 2.1 shows how the systematic and random components combine to specify the model in the case of simple linear regression with all prior weights set to one.

The assumptions necessary for establishing Model (2.1) are:

- Suitability: The same regression model is appropriate for all the observations.
- Linearity: The true relationship between μ and each quantitative explanatory variable is linear.
- Constant variance: The unknown part of the variance of the responses, σ^2, is *constant*.
- Independence: The responses y are *independent* of each other.

Example 2.1. The mean birthweight y (in kg) and gestational ages x (in weeks) of 1513 infants born to Caucasian mothers at St George's hospital, London, between August 1982 and March 1984 [2] were recorded from volunteers (Table 2.1; data set: `gestation`).

```
> library(GLMsData); data(gestation); str(gestation)
'data.frame':        21 obs. of  4 variables:
 $ Age   : int  22 23 25 27 28 29 30 31 32 33 ...
 $ Births: int  1 1 1 1 6 1 3 6 7 7 ...
 $ Weight: num  0.52 0.7 1 1.17 1.2 ...
 $ SD    : num  NA NA NA NA 0.121 NA 0.589 0.319 0.438 0.313 ...
> summary(gestation) # Show the first few lines of the data
```

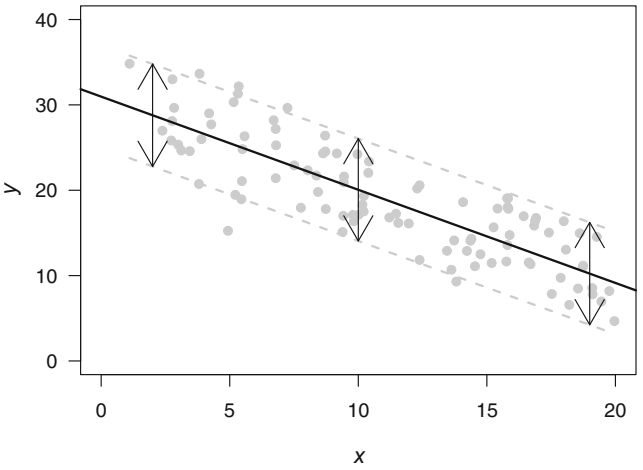

Fig. 2.1 A simple linear regression model, with all prior weights set to 1. The points show the observations, and the solid dark line shows the values of μ from the linear relationship (the systematic component). The arrows and dotted lines indicate that the variation (random component) is approximately constant for all values of x (Sect. 2.2)

Table 2.1 Mean birthweights and gestational ages of babies born to Caucasian mothers at St George's hospital, London, between August 1982 and March 1984 who were willing to participate in the research (Example 2.1)

Gestational age (weeks) x_i	Number of births m_i	Birthweight means (kg) y_i	Gestational age (weeks) x_i	Number of births m_i	Birthweight means (kg) y_i
22	1	0.520	35	29	2.796
23	1	0.700	36	43	2.804
25	1	1.000	37	114	3.108
27	1	1.170	38	222	3.204
28	6	1.198	39	353	3.353
29	1	1.480	40	401	3.478
30	3	1.617	41	247	3.587
31	6	1.693	42	53	3.612
32	7	1.720	43	9	3.390
33	7	2.340	44	1	3.740
34	7	2.516			

```
      Age            Births           Weight           SD
 Min.   :22.00   Min.   :  1.00   Min.   :0.520   Min.   :0.1210
 1st Qu.:29.00   1st Qu.:  1.00   1st Qu.:1.480   1st Qu.:0.3575
 Median :34.00   Median :  7.00   Median :2.516   Median :0.4270
 Mean   :33.76   Mean   : 72.05   Mean   :2.335   Mean   :0.4057
 3rd Qu.:39.00   3rd Qu.: 53.00   3rd Qu.:3.353   3rd Qu.:0.4440
 Max.   :44.00   Max.   :401.00   Max.   :3.740   Max.   :0.5890
                                                  NA's   :6
```

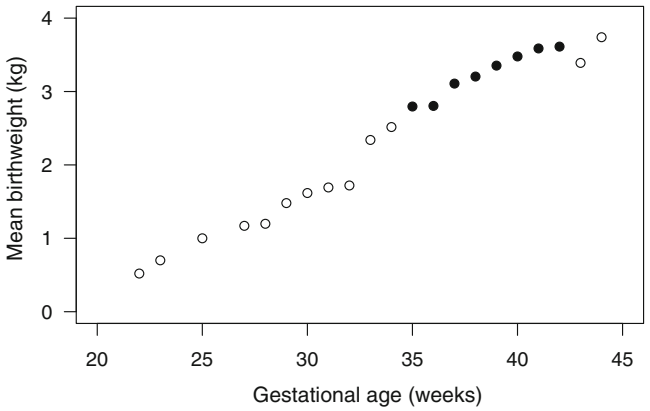

Fig. 2.2 A plot of mean birthweights against gestational ages from Table 2.1. The hollow dots are used for the means based on fewer than 20 observations, and filled dots for other observations (Example 2.1)

The mean birthweight (`Weight`) and standard deviation of birthweights (`SD`) of all the babies at given gestational ages are recorded. Notice the appearance of `NA` in the data; `NA` means 'not available'. Here the `NA`s appear because standard deviations cannot be computed for gestational ages where only one birth was recorded.

The relationship between the expected mean birthweight of babies $\mu = \mathrm{E}[y]$ and gestational age x is approximately linear over the given gestational age range (Fig. 2.2):

```
> plot( Weight ~ Age, data=gestation, las=1, pch=ifelse( Births<20, 1, 19),
    xlab="Gestational age (weeks)", ylab="Mean birthweight (kg)",
    xlim=c(20, 45), ylim=c(0, 4))
```

The construct `pch=ifelse(Births<20, 1, 19)` means that if the number of births m is fewer than 20, then plot using `pch=1` (an empty circle), and otherwise use `pch=19` (a filled circle).

Note that, for example, there are $m = 3$ babies born at $x = 30$ weeks gestation. This means that three observations have been combined to make this entry in the data, so this information should be weighted accordingly. There are $n = 21$ rows in the data frame (and 21 gestational ages given), but a total of $\sum_{i=1}^{n} m_i = 1513$ births are represented.

The responses y_i here represent *sample mean* birthweights. If birthweights of *individual* babies at gestational age x_i have variance σ^2, then expect the sample means y_i to have variance σ^2/m_i, where m_i is the sample size of group i. A sensible random component is $\mathrm{var}[y_i] = \sigma^2/m_i$, so that the known prior weights are $w_i = m_i$. A possible model for the data is

$$\begin{cases} \mathrm{var}[y_i] = \sigma^2/m_i \\ \mu_i = \beta_0 + \beta_1 x_i. \end{cases} \tag{2.2}$$

Model (2.2) is a weighted linear regression model. Mean birthweights based on larger numbers of observations contain more information than mean birthweights based on smaller numbers of observations. Using prior weights enables the observations to be suitably weighted to reflect this. □

2.3 Simple Linear Regression

2.3.1 Least-Squares Estimation

Many of the principles of linear regression can be seen in the case of simple linear regression, when there is only an intercept and a single covariate in the model; that is,

$$\begin{cases} \text{var}[y_i] = \sigma^2/w_i \\ \mu_i = \beta_0 + \beta_1 x_i, \end{cases}$$

where $\text{E}[y_i] = \mu_i$.

For regression models to be used in practice, estimates of the intercept β_0 and slope β_1 are needed, as well as the variance σ^2. For any given intercept and slope, the deviations between the observed data y_i and the model μ_i are given by

$$e_i = y_i - \mu_i = y_i - \beta_0 - \beta_1 x_i. \tag{2.3}$$

It makes sense to choose the fitted line (that is, the estimates of β_0 and β_1) in such a way as to make the deviations as small as possible. To summarize the deviations, we can square them (to avoid negative quantities) then sum them, to get

$$S(\beta_0, \beta_1) = \sum_{i=1}^{n} w_i e_i^2 = \sum_{i=1}^{n} w_i (y_i - \mu_i)^2 = \sum_{i=1}^{n} w_i (y_i - \beta_0 - \beta_1 x_i)^2.$$

The non-negative weights w_i may be used to weight observations according to their precision (for example, mean birthweights based on larger sample sizes are estimated with greater precision, so can be allocated larger weights). S summarizes how far the fitted line is from the observations y_i. Smaller values of S mean the line is closer to the y_i, in general. The least-squares principle is to estimate β_0 and β_1 by those values that minimize S.

Example 2.2. Consider the `gestation` data from Example 2.1. We can try some values for β_0 and β_1, and compute the corresponding value of S.

```
> y   <- gestation$Weight
> x   <- gestation$Age
> wts <- gestation$Births
> beta0.A <- -0.9; beta1.A <- 0.1   # Try these values for beta0 and beta1
> mu.A <- beta0.A + beta1.A * x
> SA <- sum( wts*(y - mu.A)^2 ); SA
[1] 186.1106
```

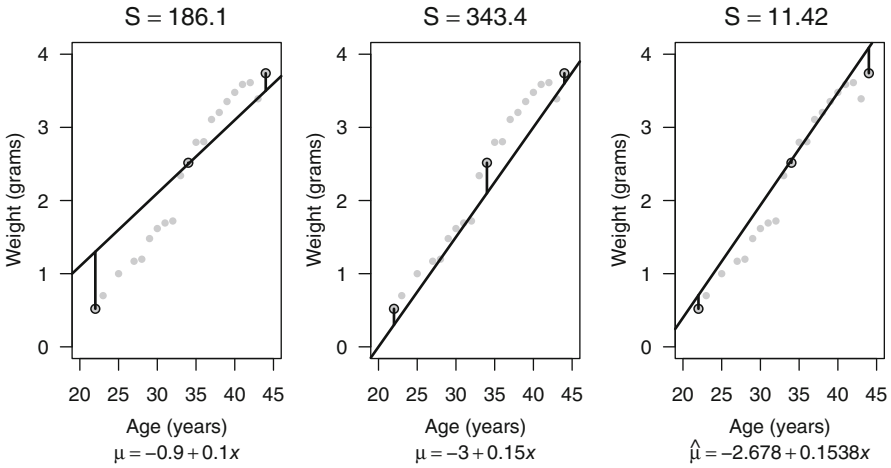

Fig. 2.3 Three possible systematic components relating weight and age. For three observations, the deviations from the postulated equation are shown by thin vertical lines (Example 2.2)

This shows that the values $\beta_0 = -0.9$ and $\beta_1 = 0.1$ produce $S = 186.1$ (Fig. 2.3, left panel). Suppose we try different values for β_0 and β_1:

```
> beta0.B <- -3; beta1.B <- 0.150
> mu.B <- beta0.B + beta1.B * x
> SB <- sum( wts*(y - mu.B)^2 ); SB
[1] 343.4433
```

Using $\beta_0 = -3$ and $\beta_1 = 0.15$ produces $S = 343.4$ (centre panel), so the values of β_0 and β_1 used in the left panel are preferred over those used in the centre panel.

The smallest possible value for S is achieved using the least-squares estimates $\hat{\beta}_0$ and $\hat{\beta}_1$ (right panel). □

2.3.2 Coefficient Estimates

The least-squares estimators of β_0 and β_1 can be found by using calculus to minimize the sum of squares $S(\beta_0, \beta_1)$. The derivatives of S with respect to β_0 and β_1 are

$$\frac{\partial S(\beta_0, \beta_1)}{\partial \beta_0} = 2 \sum_{i=1}^{n} w_i (y_i - \mu_i); \tag{2.4}$$

$$\frac{\partial S(\beta_0, \beta_1)}{\partial \beta_1} = 2 \sum_{i=1}^{n} w_i x_i (y_i - \mu_i). \tag{2.5}$$

Solving $\partial S/\partial\beta_0 = \partial S/\partial\beta_1 = 0$ (Problem 2.2) gives the following solutions for β_0 and β_1:

$$\hat{\beta}_0 = \bar{y}_w - \hat{\beta}_1 \bar{x}_w; \tag{2.6}$$

$$\hat{\beta}_1 = \frac{\text{SS}_{xy}}{\text{SS}_x} = \frac{\sum_{i=1}^n w_i(x_i - \bar{x}_w)y_i}{\sum_{i=1}^n w_i(x_i - \bar{x}_w)^2}, \tag{2.7}$$

where $\bar{x}_w$ and $\bar{y}_w$ are the weighted means

$$\bar{x}_w = \frac{\sum_{i=1}^n w_i x_i}{\sum_{i=1}^n w_i} \quad \text{and} \quad \bar{y}_w = \frac{\sum_{i=1}^n w_i y_i}{\sum_{i=1}^n w_i}.$$

Here $\hat{\beta}_0$ and $\hat{\beta}_1$ are the *least-squares estimators* of β_0 and β_1 respectively. They can be shown to be unbiased estimators of β_0 and β_1 respectively (Problem 2.5). The *fitted values* are estimated by $\hat{\mu}_i = \hat{\beta}_0 + \hat{\beta}_1 x_i$, for $i = 1, \ldots, n$.

The minimized value of $S(\beta_0, \beta_1)$, evaluated at the least-squares estimates $\beta_0 = \hat{\beta}_0$ and $\beta_1 = \hat{\beta}_1$, is called the *residual sum-of-squares* (RSS):

$$\text{RSS} = \sum_{i=1}^n w_i(y_i - \hat{\mu}_i)^2 = \sum_{i=1}^n w_i(y_i - \hat{\beta}_0 - \hat{\beta}_1 x_i)^2, \tag{2.8}$$

because $r_i = y_i - \hat{\mu}_i$ are called the *raw residuals*. (Contrast this with the deviations given in (2.3).)

Example 2.3. For the `gestation` data model (2.2), the least-squares parameter estimates can be computed using (2.6) and (2.7):

```
> xbar <- weighted.mean(x, w=wts)     # The weighted mean of x (Age)
> SSx <- sum( wts*(x-xbar)^2 )
> ybar <- weighted.mean(y, w=wts)     # The weighted mean of y (Weight)
> SSxy <- sum( wts*(x-xbar)*y )
> beta1 <- SSxy / SSx; beta0 <- ybar - beta1*xbar
> mu <- beta0 + beta1*x
> RSS <- sum( wts*(y - mu )^2 )
> c( beta0=beta0, beta1=beta1, RSS=RSS )
     beta0      beta1        RSS
-2.6783891  0.1537594 11.4198322
```

This is not how the model would be fitted in R in practice, but we proceed this way to demonstrate the formulae above. The usual way to fit the model (see Sect. 2.6) would be to use `lm()`:

```
> lm(Weight ~ Age, weights=Births, data=gestation)
Call:
lm(formula = Weight ~ Age, data = gestation, weights = Births)

Coefficients:
(Intercept)          Age
    -2.6784       0.1538
```

Either way, the systematic component of the model is estimated as

$$\hat{\mu} = -2.678 + 0.1538x \tag{2.9}$$

with RSS = 11.42. □

2.3.3 Estimating the Variance σ^2

By definition, $\sigma^2/w_i = \text{var}[y_i] = \text{E}[(y_i - \mu_i)^2]$, so it is reasonable to try to estimate σ^2 by the average of the squared deviations $w_i(y_i - \hat{\mu}_i)^2 = \text{RSS}$. This leads to the superficially attractive proposal of estimating σ^2 by

$$\hat{\sigma}^2 = \frac{\text{RSS}}{n}.$$

If the μ_i were known and not estimated (by $\hat{\mu}_i$), this would be an ideal estimator. Unfortunately the process of estimating $\hat{\mu}_i$ is based on minimizing RSS, making RSS smaller than it would be by random variation and introducing a negative bias into $\hat{\sigma}^2$. In other words, $\hat{\sigma}^2$ is a *biased* estimate of σ^2. The correct way to adjust for the fact that the regression parameters have been estimated is to divide by $n - 2$ instead of n. This leads to

$$s^2 = \frac{\text{RSS}}{n-2}. \tag{2.10}$$

This is an unbiased estimator of σ^2, and is the estimator almost always used in practice.

The divisor $n - 2$ here is known as the *residual degrees of freedom*. The residual degrees of freedom are equal to the number of observations minus the number of coefficients estimated in the systematic component of the linear regression model. One can usefully think of the process of estimating each coefficient as "using up" the equivalent of one observation. For simple linear regression, there are two coefficients needing to be estimated, so that the equivalent of only $n - 2$ independent observations remain to estimate the variance. The terminology *degrees of freedom* arises from the following observation. If the first $n - 2$ values of $r_i = y_i - \hat{\beta}_0 - \hat{\beta}_1 x_i$ were known, then the remaining two values could be inferred from $\hat{\beta}_0$ and $\hat{\beta}_1$. In other words, there are only $n - 2$ degrees of freedom available to the residuals r_i given the coefficient estimates.

Example 2.4. In Example 2.3 using the `gestation` data, compute:

```
> df <- length(y) - 2
> s2 <- RSS / df
> c( df = df, s=sqrt(s2), s2=s2 )
        df          s         s2
19.0000000  0.7752701  0.6010438
```

The estimate of σ^2 is $s^2 = 0.6010$. This information is automatically computed by R when the `lm()` function is used (see Sect. 2.6). □

2.3.4 Standard Errors of the Coefficients

The variances of the parameter estimates given in Sect. 2.3.2 (p. 36) are

$$\text{var}[\hat{\beta}_0] = \sigma^2 \left(\frac{1}{\sum w_i} + \frac{\bar{x}_w^2}{\text{SS}_x} \right) \quad \text{and} \quad \text{var}[\hat{\beta}_1] = \frac{\sigma^2}{\text{SS}_x},$$

where $\bar{x}_w$ is the weighted mean. An estimate of $\text{var}[\hat{\beta}_j]$, written $\widehat{\text{var}}[\hat{\beta}_j]$, is found by substituting s^2 for the unknown true variance σ^2.

The term *standard error* is commonly used in statistics to denote the standard deviation of an estimated quantity. The standard errors of the coefficients are the square roots of $\widehat{\text{var}}[\hat{\beta}_j]$:

$$\text{se}(\hat{\beta}_0) = s \left(\frac{1}{\sum w_i} + \frac{\bar{x}_w^2}{\text{SS}_x} \right)^{1/2} \quad \text{and} \quad \text{se}(\hat{\beta}_1) = \frac{s}{\sqrt{\text{SS}_x}}.$$

Example 2.5. For the `gestation` data model, the standard errors of the coefficients are:

```
> var.b0 <- s2 * ( 1/sum(wts) + xbar^2 / SSx )
> var.b1 <- s2 / SSx
> sqrt( c( beta0=var.b0, beta1=var.b1) )  # The std errors
      beta0       beta1
0.371172341 0.009493212
```

This information is automatically computed by R when the `lm()` function is used (see Sect. 2.6). □

2.3.5 Standard Errors of Fitted Values

For a given value of the explanatory variable, say x_g, the best estimate of the mean response is the fitted value $\hat{\mu}_g = \hat{\beta}_0 + \hat{\beta}_1 x_g$. Since $\hat{\mu}_g$ is a function of the estimated parameters $\hat{\beta}_0$ and $\hat{\beta}_1$, the estimate of μ_g also contains uncertainty. The variance of $\hat{\mu}_g$ is

$$\text{var}[\hat{\mu}_g] = \sigma^2 \left\{ \frac{1}{\sum w_i} + \frac{(x_g - \bar{x})^2}{\text{SS}_x} \right\}.$$

An estimate of $\text{var}[\hat{\mu}_g]$, written $\widehat{\text{var}}[\hat{\mu}_g]$, is found by substituting s^2 for the unknown true variance σ^2. The standard error of $\hat{\mu}_g$, written $\text{se}(\hat{\mu}_g)$, is the square root of the variance.

Example 2.6. For the `gestation` data model, suppose we wish to use the model to estimate the mean birthweight for a gestation length of 30 weeks:

```
> x.g <- 30
> mu.g <- beta0 + x.g * beta1
> var.mu.g <- s2 * ( 1/sum(wts) + (x.g-xbar)^2 / SSx )
> se.mu.g <- sqrt(var.mu.g)
> c( mu=mu.g, se=sqrt(var.mu.g))
      mu       se
1.934392 0.088124
```

The mean birthweight is estimated as $\hat{\mu}_g = 1.934\,\text{kg}$, with a standard error of $\text{se}(\hat{\mu}_g) = 0.08812\,\text{kg}$. □

2.4 Estimation for Multiple Regression

2.4.1 Coefficient Estimates

Now we return to the general situation, when there are p explanatory variables, and p' regression coefficients β_j to be estimated, for $j = 0, 1, \ldots, p$, including the intercept. The regression model is given by Eq. (2.1).

As for simple linear regression, we define the sum of squared deviations between the observations y_i and the model means by

$$S = \sum_{i=1}^{n} w_i (y_i - \mu_i)^2.$$

For any given set of coefficients β_j, S measures how close the model means μ_i are to the observed responses y_i. Smaller values of S indicate that the μ_i are closer to the y_i, in general. The *least-squares estimators* of β_j are defined to be those values of β_j that minimize S, and are denoted $\hat{\beta}_0, \ldots, \hat{\beta}_p$.

Using calculus, the minimum value of S occurs when

$$\frac{\partial S}{\partial \beta_j} = 0 \quad \text{for } j = 0, 1, \ldots, p. \tag{2.11}$$

The least-squares estimators are found by solving the set of $p+1$ simultaneous equations (2.11). The solutions to these equations are best computed using matrix algorithms, but the least-squares estimators can be well understood and interpreted by writing them as:

$$\hat{\beta}_j = \frac{\sum_{i=1}^{n} w_i x_{ij}^* y_i}{\sum_{i=1}^{n} w_i (x_{ij}^*)^2}, \tag{2.12}$$

for $j = 0, \ldots, p$, where x_{ij}^* give the values for jth explanatory variable x_j after being adjusted for the all other explanatory variables $x_0, \ldots, x_p$ apart from x_j. The adjusted explanatory variable x_j^* is that part of x_j that cannot be explained by regression on the other explanatory variables.

The *fitted values* are

$$\hat{\mu}_i = \hat{\beta}_0 + \sum_{j=1}^{p} \hat{\beta}_j x_{ji}, \tag{2.13}$$

and the *residuals* are the deviations of the responses from the fitted values:

$$r_i = y_i - \hat{\mu}_i.$$

The values of the adjusted explanatory variable x_j^* are the residuals from the linear regression of x_j on the explanatory variables other than x_j. Although not immediately obvious, the formulae for the least-squares estimators (2.12) are of the same form as that for the slope in simple linear regression (2.7). In simple linear regression, the covariate x needs to be adjusted only for the intercept term, so $x_i^* = (x_i - \bar{x})$. Substituting this into (2.12) gives (2.7).

Note that σ^2 doesn't appear in the least-squares equations. This means we do not need to know the value of σ^2 in order to estimate the coefficients β_j.

Example 2.7. For the lung capacity data (`lungcap`), Fig. 2.4 shows that the relationship between FEV and height is not linear, so a linear model is not appropriate. However, plotting the *logarithm* of FEV against height does show an approximate linear relationship (the function `scatter.smooth()` adds a smooth curve to the plotted points):

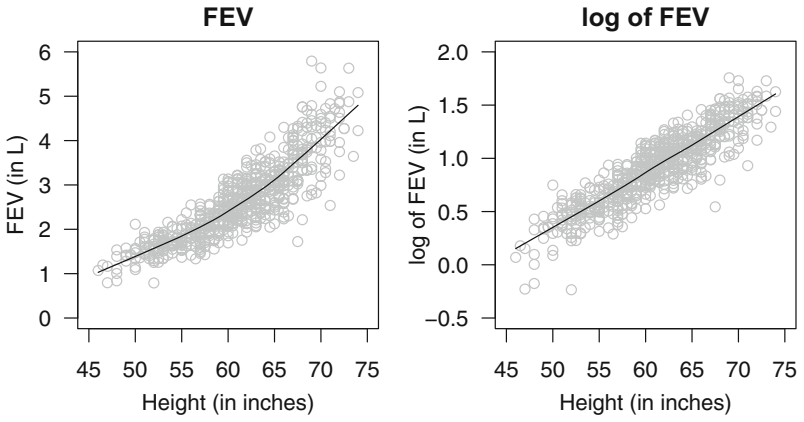

Fig. 2.4 FEV plotted against height (left panel), and the logarithm of FEV plotted against height (right panel) for the `lungcap` data (Example 2.7)

```
> scatter.smooth( lungcap$Ht, lungcap$FEV, las=1, col="grey",
    ylim=c(0, 6), xlim=c(45, 75), # Use similar scales for comparisons
    main="FEV", xlab="Height (in inches)", ylab="FEV (in L)" )
> scatter.smooth( lungcap$Ht, log(lungcap$FEV), las=1, col="grey",
    ylim=c(-0.5, 2), xlim=c(45, 75), # Use similar scales for comparisons
    main="log of FEV", xlab="Height (in inches)", ylab="log of FEV (in L)")
```

For the `lungcap` data then, fitting a linear model for $y = \log(\text{FEV})$ may be appropriate. On this basis, a possible linear regression model to fit to the data would be

$$\begin{cases} \text{var}[y_i] = \sigma^2 \\ \mu_i = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \beta_4 x_4, \end{cases} \tag{2.14}$$

where $\mu = \text{E}[y]$ for $y = \log(\text{FEV})$, x_1 is height, x_2 is age, x_3 is the dummy variable (1.1) for gender (0 for females; 1 for males), and x_4 is the dummy variable (1.2) for smoking (0 for non-smokers; 1 for smokers). Here, $p' = 5$ and $n = 654$. □

2.4.2 Estimating the Variance σ^2

The value of S evaluated at the least-squares estimates of β_j is called the *residual sum-of-squares* (RSS):

$$\text{RSS} = \sum_{i=1}^{n} w_i (y_i - \hat{\mu}_i)^2. \tag{2.15}$$

The residual degrees of freedom associated with RSS is equal to the number of observations minus the number of regression coefficients that were estimated in evaluating RSS, in this case $n - p'$. As for simple linear regression, an unbiased estimator of σ^2 is obtained by dividing RSS by the corresponding degrees of freedom:

$$s^2 = \frac{\sum_{i=1}^{n} w_i (y_i - \hat{\mu}_i)^2}{n - p'} = \frac{\text{RSS}}{n - p'}.$$

2.4.3 Standard Errors

Write $\mathcal{I}_j^* = \sum_{i=1}^{n} w_i \left(x_{ij}^*\right)^2$ for the sum of squares of the jth explanatory variable adjusted for the other variables. This quantity $\mathcal{I}_j^*$ is a measure of how well the regression model is leveraged to estimate the jth coefficient. It

tends to be larger when x_j is independent of the other explanatory variables and smaller when x_j is correlated with one or more of the other variables. The variance of the jth coefficient is

$$\text{var}[\hat{\beta}_j] = \sigma^2 / \mathcal{I}_j^*.$$

An estimate of $\text{var}[\hat{\beta}_j]$, written $\widehat{\text{var}}[\hat{\beta}_j]$, is found by substituting s^2 for the unknown true variance σ^2. Then, the standard error becomes

$$\text{se}(\hat{\beta}_j) = s / \sqrt{\mathcal{I}_j^*}.$$

* 2.5 Matrix Formulation of Linear Regression Models

* *2.5.1 Matrix Notation*

Using matrix algebra to describe data is convenient, and useful for simplifying the mathematics. Denote the $n \times 1$ vector of responses as $\mathbf{y}$, and the $n \times p'$ matrix of explanatory variables, called the *model matrix*, as $\mathrm{X} = [\mathbf{x}_0, \mathbf{x}_1, \ldots, \mathbf{x}_p]$, where $\mathbf{x}_j$ is the $n \times 1$ vector of values for x_j. We write $\mathbf{x}_0$ for the vector of ones (the constant term) for convenience. The linear regression model in matrix form is

$$\begin{cases} \text{var}[\mathbf{y}] = \mathrm{W}^{-1}\sigma^2 \\ \boldsymbol{\mu} = \mathrm{X}\boldsymbol{\beta}, \end{cases} \tag{2.16}$$

where $\mathrm{E}[\mathbf{y}] = \boldsymbol{\mu}$, and W^{-1} is a known, positive-definite symmetric matrix of size $n \times n$. A special case occurs when the diagonal elements (i, i) of W^{-1} are $1/w_i$ and the off-diagonal elements are zero, equivalent to (2.1). Most commonly, observations are weighted identically, so that $\mathrm{W}^{-1} = \mathrm{I}_n$, where I_n is an $n \times n$ identity matrix.

Example 2.8. For the `gestation` data in Example 2.1 (p. 32), $n = 21$ and so $\mathbf{y}$ is a 21×1 vector, and X is a 21×2 model matrix (that is, $p' = 2$). The vector $\mathbf{y}$, matrix X, and covariance matrix W^{-1} are

$$\mathbf{y} = \begin{bmatrix} 0.520 \\ 0.700 \\ 1.000 \\ \vdots \\ 3.612 \\ 3.390 \\ 3.740 \end{bmatrix}; \quad \mathrm{X} = \begin{bmatrix} 1 & 22 \\ 1 & 23 \\ 1 & 25 \\ \vdots & \vdots \\ 1 & 42 \\ 1 & 43 \\ 1 & 44 \end{bmatrix}; \quad \mathrm{W}^{-1} = \begin{bmatrix} 1/1 & 0 & 0 & \ldots & 0 & 0 & 0 \\ 0 & 1/1 & 0 & \ldots & 0 & 0 & 0 \\ 0 & 0 & 1/1 & \ldots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \ldots & 1/53 & 0 & 0 \\ 0 & 0 & 0 & \ldots & 0 & 1/9 & 0 \\ 0 & 0 & 0 & \ldots & 0 & 0 & 1/1 \end{bmatrix}.$$

The columns of X are the vector of ones and the gestational ages. □

Example 2.9. To write the model proposed for the `lungcap` data in Example 2.7, first recall that $p' = 5$ and $n = 654$. Then, the 654×1 vector $\mathbf{y} = \log(\text{FEV})$, the 654×5 model matrix X, and the 5×1 vector $\boldsymbol{\beta}$ are

$$\mathbf{y} = \begin{bmatrix} 0.0695 \\ -0.176 \\ 0.0971 \\ \vdots \\ 1.48 \end{bmatrix}; \quad \mathrm{X} = \begin{bmatrix} 1 & 3 & 46 & 0 & 0 \\ 1 & 4 & 48 & 0 & 0 \\ 1 & 4 & 48 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & 18 & 70.5 & 1 & 1 \end{bmatrix}; \quad \boldsymbol{\beta} = \begin{bmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \\ \beta_3 \\ \beta_4 \end{bmatrix},$$

where the columns of X are the constant term (always one), `Age`, `Ht`, the dummy variable for `Gender`, and the dummy variable for `Smoke`. The weight matrix W is the 654×654 identity matrix I_{654}. Model (2.14) written in matrix notation is then

$$\begin{cases} \mathrm{var}[\mathbf{y}] = \mathrm{I}_{654}\sigma^2 \\ \boldsymbol{\mu} = \mathrm{X}\boldsymbol{\beta}, \end{cases}$$

where $\mathrm{E}[\mathbf{y}] = \mathrm{E}[\log(\text{FEV})] = \boldsymbol{\mu}$. □

* 2.5.2 Coefficient Estimates

The simultaneous solutions to the least-squares equations (2.11) are most conveniently found using matrix algebra. Using matrix notation, write the weighted sum-of-squared deviations (Sect. 2.4.1) as

$$S = (\mathbf{y} - \boldsymbol{\mu})^T \mathrm{W}(\mathbf{y} - \boldsymbol{\mu}), \tag{2.17}$$

where $\boldsymbol{\mu} = \mathrm{X}\boldsymbol{\beta}$. Differentiating S with respect to $\boldsymbol{\beta}$ and setting to zero shows that the minimum value of S (the RSS) occurs when

$$\mathrm{X}^T\mathrm{W}\mathrm{X}\boldsymbol{\beta} = \mathrm{X}^T\mathrm{W}\mathbf{y} \tag{2.18}$$

(Problem 2.4). The matrix $\mathrm{X}^T\mathrm{W}\mathrm{X}$ must be invertible for this equation to have a unique solution, and so X must be of full column-rank. The solution can be written as

$$\hat{\boldsymbol{\beta}} = (\mathrm{X}^T\mathrm{W}\mathrm{X})^{-1}\mathrm{X}^T\mathrm{W}\mathbf{y}. \tag{2.19}$$

Using matrix algebra, it is straightforward to show that $\hat{\boldsymbol{\beta}}$ is an unbiased estimator of $\boldsymbol{\beta}$ (Problem 2.6). Then the fitted values are $\hat{\boldsymbol{\mu}} = \mathrm{X}\hat{\boldsymbol{\beta}}$.

Although not immediately obvious, the matrix formula for $\hat{\boldsymbol{\beta}}$ (2.19) has essentially the same form as the non-matrix expressions (2.7) and (2.12). In each case, the formula for $\hat{\beta}$ consists of a sum of cross-products of x and y (here $\mathrm{X}^T\mathrm{W}\mathbf{y}$) divided by a sum of squares of x values (here $\mathrm{X}^T\mathrm{W}\mathrm{X}$). The

expressions (2.12) and (2.19) are equivalent, although the matrix version is more appropriate for computation.

Numerically efficient algorithms do not implement Eq. (2.19) by inverting $\mathrm{X}^T\mathrm{WX}$ explicitly. A more efficient approach is to obtain $\hat{\boldsymbol{\beta}}$ directly as the solution to the linear system of Eqs. (2.18). The default numerical algorithms used by the built-in regression functions in R are even more sophisticated, and avoid computing $\mathrm{X}^T\mathrm{WX}$ altogether. This is done via the QR-decomposition of X, such that $\mathrm{XW}^{1/2} = \mathrm{QR}$ where Q satisfies $\mathrm{Q}^T\mathrm{Q} = \mathrm{I}$ and R is an upper-triangular matrix. Details of these computations are beyond the scope of this book. Rather, it will be sufficient to know that R implements efficient and stable numerical algorithms for computing $\hat{\boldsymbol{\beta}}$ and other regression output.

Example 2.10. Consider fitting the linear regression model (2.14) to the lung capacity data. Observations are not weighted and hence $\mathrm{W}^{-1} = \mathrm{I}_n$, so use R as follows:

```
> data(lungcap)
> lungcap$Smoke <- factor(lungcap$Smoke, levels=c(0, 1),
                      labels=c("Non-smoker","Smoker"))
> Xmat <- model.matrix( ~ Age + Ht + factor(Gender) + factor(Smoke),
                       data=lungcap)
```

Here, `model.matrix()` is used to combine the variables as columns of a matrix, after declaring `Smoke` as a factor.

```
> head(Xmat)
  (Intercept) Age Ht factor(Gender)M factor(Smoke)Smoker
1           1   3 46               0                    0
2           1   4 48               0                    0
3           1   4 48               0                    0
4           1   4 48               0                    0
5           1   4 49               0                    0
6           1   4 49               0                    0
> XtX <- t(Xmat) %*% Xmat  # t() is transpose; %*% is matrix multiply
> y <- log(lungcap$FEV)
> inv.XtX <- solve( XtX )  # solve  returns the matrix inverse
> XtY <- t(Xmat) %*% y
> beta <- inv.XtX %*% XtY; drop(beta)
       (Intercept)                 Age                  Ht
       -1.94399818          0.02338721          0.04279579
   factor(Gender)M factor(Smoke)Smoker
        0.02931936         -0.04606754
```

(`drop()` drops any unnecessary dimensions. In this case it reduces a single-column matrix to a vector.) The fitted model has the systematic component

$$\hat{\mu} = -1.944 + 0.02339\texttt{Age} + 0.04280\texttt{Ht} + 0.02932\texttt{Gender} - 0.04607\texttt{Smoke},$$

where `Gender` is 0 for females and 1 for males, and `Smoke` is 0 for non-smokers and 1 for smokers. Slightly more efficient code would have been to compute $\hat{\boldsymbol{\beta}}$ by solving a linear system of equations:

```
> beta <- solve(XtX, XtY); beta
                            [,1]
(Intercept)          -1.94399818
Age                   0.02338721
Ht                    0.04279579
factor(Gender)M       0.02931936
factor(Smoke)Smoker -0.04606754
```

giving the same result. An even more efficient approach would have been to use the QR-decomposition:

```
> QR <- qr(Xmat)
> beta <- qr.coef(QR, y); beta
        (Intercept)                 Age                  Ht
        -1.94399818          0.02338721          0.04279579
    factor(Gender)M factor(Smoke)Smoker
         0.02931936         -0.04606754
```

again giving the same result. □

* 2.5.3 Estimating the Variance σ^2

After computing $\hat{\boldsymbol{\beta}}$, the fitted values are obtained as $\hat{\boldsymbol{\mu}} = \mathrm{X}\hat{\boldsymbol{\beta}}$. The variance σ^2 is estimated from the RSS as usual:

$$s^2 = \frac{(\mathbf{y} - \hat{\boldsymbol{\mu}})^T \mathrm{W} (\mathbf{y} - \hat{\boldsymbol{\mu}})}{n - p'} = \frac{\mathrm{RSS}}{n - p'}.$$

Example 2.11. In Example 2.10, for the model relating log(FEV) to age, height, gender and smoking status for the `lungcap` data, compute:

```
> mu <- Xmat %*% beta
> RSS <- sum( (y - mu)^2 ); RSS
[1] 13.73356
> s2 <- RSS / ( length(lungcap$FEV) - length(beta) )
> c(s=sqrt(s2), s2=s2)
         s         s2
0.14546857 0.02116111
```

The estimate of σ^2 is $s^2 = 0.02116$. Of course, these calculations are performed automatically by `lm()`. □

* 2.5.4 Estimating the Variance of $\hat{\boldsymbol{\beta}}$

Using (2.19), the covariance matrix for $\hat{\boldsymbol{\beta}}$ is (Problem 2.7)

$$\text{var}[\hat{\boldsymbol{\beta}}] = \sigma^2(\mathrm{X}^T\mathrm{W}\mathrm{X})^{-1}. \tag{2.20}$$

The diagonal elements of $\text{var}[\hat{\boldsymbol{\beta}}]$ are the values of $\text{var}[\hat{\beta}_j]$. An estimate of this covariance matrix is found by using s^2 as an estimate of σ^2:

$$\widehat{\text{var}}[\hat{\boldsymbol{\beta}}] = s^2(\mathrm{X}^T\mathrm{W}\mathrm{X})^{-1}. \tag{2.21}$$

The diagonal elements of $\widehat{\text{var}}[\hat{\boldsymbol{\beta}}]$ are the values of $\widehat{\text{var}}[\hat{\beta}_j]$, from which the estimated standard errors of the individual parameters are computed: $\text{se}(\hat{\beta}_j) = \sqrt{\widehat{\text{var}}[\hat{\beta}_j]}$.

Example 2.12. For the model relating FEV to age, height, gender and smoking status, as used in Examples 2.10 and 2.11 (data set: `lungcap`):

```
> var.matrix <- s2 * inv.XtX
> var.betaj <- diag( var.matrix )   # diag()  grabs the diagonal elements
> sqrt( var.betaj )
        (Intercept)                 Age                  Ht
        0.078638583         0.003348451         0.001678968
    factor(Gender)M factor(Smoke)Smoker
        0.011718565         0.020910198
```

Hence, $\text{se}(\hat{\beta}_0) = 0.07864$ and $\text{se}(\hat{\beta}_1) = 0.003348$, for example. Of course, these calculations are performed automatically by `lm()`. □

* 2.5.5 Estimating the Variance of Fitted Values

For known values of the explanatory variables, given in the row vector $\mathbf{x}_g$ of length p' say, the best estimate of the mean response is the fitted value $\hat{\mu}_g = \mathbf{x}_g\hat{\boldsymbol{\beta}}$. Since $\hat{\mu}_g$ is a function of the estimated parameters $\hat{\boldsymbol{\beta}}$, the estimate of μ_g also contains uncertainty. The variance of $\hat{\mu}_g$ is

$$\text{var}[\hat{\mu}_g] = \text{var}[\mathbf{x}_g\hat{\boldsymbol{\beta}}] = \mathbf{x}_g(\mathrm{X}^T\mathrm{W}\mathrm{X})^{-1}\mathbf{x}_g^T\sigma^2.$$

An estimate of $\text{var}[\hat{\mu}_g]$, written $\widehat{\text{var}}[\hat{\mu}_g]$, is found by substituting s^2 for the unknown true variance σ^2. The standard error is then

$$\text{se}(\hat{\mu}_g) = s\sqrt{(\mathrm{X}^T\mathrm{W}\mathrm{X})^{-1}\mathbf{x}_g^T}.$$

Example 2.13. For the `lungcap` data, Example 1.6 suggested a linear relationship between log(FEV) and height. Suppose we wish to estimate the mean of

log(FEV) for females (that is, $x_3 = 0$) that smoke (that is, $x_4 = 1$), aged 18 who are 66 inches tall using the model in (2.14):

```
> xg.vec <- matrix( c(1, 18, 66, 0, 1), nrow=1)
>    ### The first "1" is the constant term
> mu.g <- xg.vec %*% beta
> var.mu.g <- sqrt( xg.vec %*% (solve(t(Xmat)%*%Xmat)) %*% t(xg.vec) * s2)
> c( mu.g, var.mu.g )
[1] 1.25542621 0.02350644
```

The estimate of log(FEV) is $\hat{\mu}_g = 1.255\,\text{L}$, with a standard error of $\text{se}(\hat{\mu}_g) = \sqrt{0.02351} = 0.1533\,\text{L}$. □

2.6 Fitting Linear Regression Models Using R

Performing explicit computations in R to estimate unknown model parameters, as demonstrated in Sects. 2.3 and 2.5, is tedious and unnecessary. In R, linear regression models are conveniently fitted to data using the function `lm()`. Basic use of the `lm()` function requires specifying the response and explanatory variables.

Example 2.14. Fitting the regression model (2.2) for the birthweight data frame `gestation` (Example 2.1, p. 32) requires the prior weights (the number of birth, `Births`) to be explicitly supplied in addition to the response and explanatory variable:

```
> gest.wtd <- lm( Weight ~ Age, data=gestation,
                  weights=Births)   # The prior weights
> summary(gest.wtd)
Call:
lm(formula = Weight ~ Age, data = gestation, weights = Births)

Weighted Residuals:
     Min       1Q   Median       3Q      Max
-1.62979 -0.60893 -0.30063 -0.08845  1.03880

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -2.678389   0.371172  -7.216 7.49e-07 ***
Age          0.153759   0.009493  16.197 1.42e-12 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.7753 on 19 degrees of freedom
Multiple R-squared:  0.9325,        Adjusted R-squared:  0.9289
F-statistic: 262.3 on 1 and 19 DF,  p-value: 1.416e-12
```

The first argument to the `lm()` function is a *model formula*: `Weight ~ Age`. The symbol `~` is read as 'is modelled by'. The response variable (in this case

`Weight`) is placed on the left of the `~`, and the explanatory variables are placed on the right of the `~` and are joined by `+` signs if there are more than one. The second argument `data=gestation` indicates the data frame in which the variables are located. The argument `weights` specifies the prior weights w_i, and can be omitted if all the prior weights are equal to one.

We can also fit the regression *without* using prior weights for comparison:

```
> gest.ord <- lm( Weight ~ Age, data=gestation); coef(gest.ord)
(Intercept)         Age
  -3.049879    0.159483
```

Using the prior weights (Fig. 2.5, solid line), the regression line is closer to the observations weighted more heavily (which contain more information) than the ordinary regression line (dashed line):

```
> plot( Weight ~ Age, data=gestation, type="n",
    las=1, xlim=c(20, 45), ylim=c(0, 4),
    xlab="Gestational age (weeks)", ylab="Mean birthweight (in kg)" )
> points( Weight[Births< 20] ~ Age[Births< 20], pch=1,  data=gestation )
> points( Weight[Births>=20] ~ Age[Births>=20], pch=19, data=gestation )
> abline( coef(gest.ord), lty=2, lwd=2)
> abline( coef(gest.wtd), lty=1, lwd=2)
> legend("topleft", lwd=c(2, 2), bty="n",
    lty=c(2, 1, NA, NA), pch=c(NA, NA, 1, 19),   # NA shows nothing
    legend=c("Ordinary regression", "Weighted regression",
             "Based on 20 or fewer obs.","Based on more than 20 obs."))
```

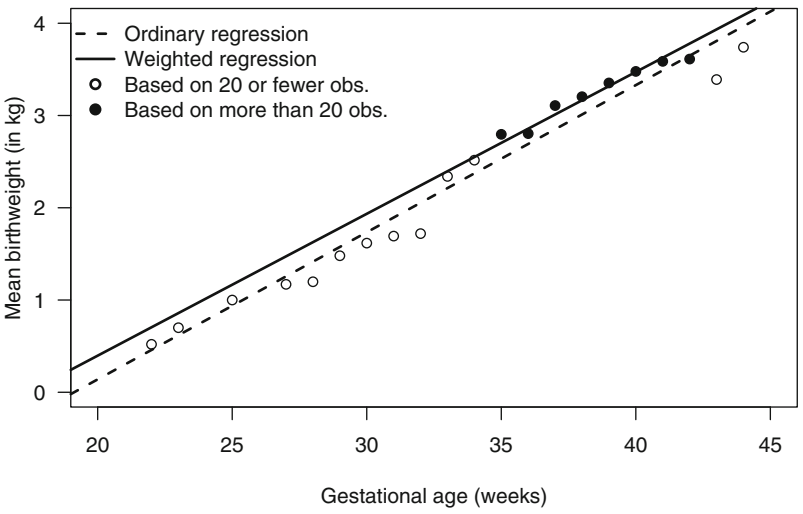

Fig. 2.5 A plot of birthweights against gestational age from Table 2.1. The filled dots are used for the means based on more than 20 observations, and hollow dots for other observations. The solid line is the ordinary regression line, while the dashed line is weighted regression line (Example 2.1)

The systematic components are drawn using `abline()`, which needs the intercept and the slope to draw the straight lines (which are both returned using `coef()`). □

Example 2.15. Consider fitting the Model (2.14) to the lung capacity data (`lungcap`), using age, height, gender and smoking status as explanatory variables, and log(FEV) as the response:

```
> # Recall, Smoke  has been declared previously as a factor
> lm( log(FEV) ~ Age + Ht + Gender + Smoke, data=lungcap )
Call:
lm(formula = log(FEV) ~ Age + Ht + Gender + Smoke, data = lungcap)

Coefficients:
(Intercept)          Age           Ht      GenderM  SmokeSmoker
   -1.94400      0.02339      0.04280      0.02932     -0.04607
```

The output of the `lm()` command as shown above is brief, and shows that the estimated systematic component is

$$\hat{\mu} = -1.944 + 0.02339x_1 + 0.04280x_2 + 0.02932x_3 - 0.04607x_4 \qquad (2.22)$$

where $\mu = \mathrm{E}[\log \mathrm{FEV}]$, for `Age` x_1 and `Ht` x_2. `Gender` is a factor, but does not need to be explicitly declared as a factor (using `factor()`) since the variable `Gender` is non-numerical (Sect. 1.4). The default coding used in R sets $x_3 = 0$ for females `F` and $x_3 = 1$ for males `M`, as in (1.1) (p. 10). The `M` following the name of the variable `Gender` in the R output indicates that `Gender` is 1 for males (see Sect. 1.4). `Smoke` is a factor, but must be explicitly declared as a factor (using `factor()`).

The constant term in the model is included implicitly by R, since it is almost always necessary. To explicitly *exclude* the constant in the model (which is unusual), use one of these forms:

```
> lm( log(FEV) ~ 0 + Age + Ht + Gender + Smoke, data=lungcap ) # No const.
> lm( log(FEV) ~ Age + Ht + Gender + Smoke - 1, data=lungcap ) # No const.
```

R returns more information about the fitted model by directing the output of `lm()` to an output *object*:

```
> LC.m1 <- lm( log(FEV) ~ Age + Ht + Gender + Smoke, data=lungcap )
```

The output object `LC.m1` contains a great deal of information about the fitted model:

```
> names( LC.m1 )  # The names of the components of LC.m1
 [1] "coefficients"  "residuals"     "effects"       "rank"
 [5] "fitted.values" "assign"        "qr"            "df.residual"
 [9] "contrasts"     "xlevels"       "call"          "terms"
[13] "model"
```

```
1  > data(lungcap)
2  > lungcap$Smoke <- factor(lungcap$Smoke, levels=c(0, 1),
3                            labels=c("Non-smoker","Smoker"))
4  > LC.m1 <- lm( log(FEV) ~ Age + Ht + Gender + Smoke, data=lungcap )
5  > summary(LC.m1)
6
7  Call:
8  lm(formula = log(FEV) ~ Age + Ht + Gender + Smoke, data = lungcap)
9
10 Residuals:
11      Min       1Q   Median       3Q      Max
12 -0.63278 -0.08657  0.01146  0.09540  0.40701
13
14 Coefficients:
15              Estimate Std. Error t value Pr(>|t|)
16 (Intercept) -1.943998   0.078639 -24.721  < 2e-16 ***
17 Age          0.023387   0.003348   6.984  7.1e-12 ***
18 Ht           0.042796   0.001679  25.489  < 2e-16 ***
19 GenderM      0.029319   0.011719   2.502   0.0126 *
20 SmokeSmoker -0.046068   0.020910  -2.203   0.0279 *
21 ---
22 Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
23
24 Residual standard error: 0.1455 on 649 degrees of freedom
25 Multiple R-squared:  0.8106,        Adjusted R-squared:  0.8095
26 F-statistic: 694.6 on 4 and 649 DF,  p-value: < 2.2e-16
```

Fig. 2.6 The output of the `summary()` command after using `lm()` for the `lungcap` data

Each of these components can be accessed directly using constructs like, for example, `LC.m1$coefficients`. However, most of the useful information is accessed using R functions, such as `coef(LC.m1)`, as demonstrated below. These functions are discussed throughout this chapter, and are summarized in Sect. 2.14. A summary of the information contained in the `LC.m1` object is displayed using the `summary()` command (Fig. 2.6). Most of this output is explained in later sections, which refer back to the output in Fig. 2.6.

For now, observe that the parameter estimates are shown in the table in the middle of the output (starting from line 14), in the column labelled `Estimate`. The estimated standard errors appear in the column labelled `Std. Error`. The parameter estimates are explicitly obtained using:

```
> coef( LC.m1 )
(Intercept)         Age          Ht     GenderM SmokeSmoker
-1.94399818  0.02338721  0.04279579  0.02931936 -0.04606754
```

The estimate of σ is:

```
> summary( LC.m1 )$sigma
[1] 0.1454686
```

This information (as well as the residual degrees of freedom) appears in line 24 of the output shown in Fig. 2.6. □

2.7 Interpreting the Regression Coefficients

After fitting a model, interpretation of the model is strongly encouraged to determine if the model makes physical sense, and to understand the story the model is telling (Sect. 1.7).

The systematic component of linear regression model fitted to the gestation data (Example 2.14) is

$$\hat{\mu} = -2.678 + 0.1538x,$$

where $\mu = \mathrm{E}[y]$, where y is the mean birthweight (in kg), and x is the gestational age in weeks. This model indicates that the mean birthweight increases by approximately 0.1538 kg for each extra week of gestation, on average, over the range of the data. The random component implies that the variation of the weights around μ is approximately constant with $s^2 = 0.6010$.

The interpretation for the systematic component model fitted to the lung capacity data (Example 2.15) is different, because the response variable is log(FEV). This means that the systematic component is

$$\begin{aligned}\mu &= \mathrm{E}[\log(\mathrm{FEV})] \\ &= -1.944 + 0.02339x_1 + 0.04280x_2 + 0.02932x_3 - 0.04607x_4 \end{aligned} \tag{2.23}$$

for `Age` x_1, `Ht` x_2, the dummy variable for `Gender` x_3 and the dummy variable for `Smoke` x_4. The regression coefficients can only be interpreted for their impact on $\mu = \mathrm{E}[\log(\mathrm{FEV})]$ and not on $\mathrm{E}[\mathrm{FEV}]$ directly. However, since $\mathrm{E}[\log y] \approx \log \mathrm{E}[y] = \log \mu$ (Problem 2.11), then (2.23) can be written as

$$\begin{aligned}\log \mu &= \log \mathrm{E}[\mathrm{FEV}] \\ &\approx -1.944 + 0.02339x_1 + 0.04280x_2 + 0.02932x_3 - 0.04607x_4. \end{aligned} \tag{2.24}$$

Now the parameter estimates can be used to approximately interpret the effects of the explanatory variables on $\mu = \mathrm{E}[\mathrm{FEV}]$ directly. For example, an increase in height x_2 of one inch is associated with an increase in the mean FEV by a *factor* of $\exp(0.04280) = 1.044$, assuming all other variables are kept constant.

Parameter estimates for qualitative explanatory variables indicate how much the value of μ changes *compared* to the reference level (after adjusting for the effect of other variables), provided treatment coding is used (Sect. 1.4). For the systematic component in (2.24), the value of μ will change by a factor of $\exp(-0.04607) = 0.9550$ for smokers (`Smoke=1`) compared to non-smokers (`Smoke=0`). In other words, FEV is likely to be a factor of 0.9550 lower for smokers, assuming all other variables are kept constant.

The random component of the model (Example 2.15) indicates the variation of log(FEV) around $\mu = \mathrm{E}[\log(\mathrm{FEV})]$ is approximately constant, with $s^2 = 0.02116$.

Interpreting the effects of correlated covariates is subtle. For example, in the lung capacity data, height and age are positively correlated (Sect. 1.7). Height generally increases with age for youth, so the effect on FEV of increasing age for fixed height is not the same as the overall increase in FEV as age increases. The overall increase in FEV would reflect the combined effects of height and age as both increase. The coefficient in the linear model reflects only the net effect of a covariate, eliminating any concomitant changes in the other covariates that might normally be present if all the covariate varied in an uncontrolled fashion.

Also, note that the data are observational, so no cause-and-effect conclusion is implied (Sect. 1.7).

2.8 Inference for Linear Regression Models: *t*-Tests

2.8.1 Normal Linear Regression Models

Up to now, no specific statistical distribution has been assumed for the responses in the regression. The responses have simply been assumed to be independent and to have constant variance. However, to undertake formal statistical inference we need to be more specific. The usual assumption of linear regression is that the responses are normally distributed, either with constant variance or with variances that are proportional to the known weights. This can be stated as:

$$\begin{cases} y_i \sim N(\mu_i, \sigma^2/w_i) \\ \mu_i = \beta_0 + \sum_{j=1}^{p} \beta_j x_{ji}. \end{cases} \tag{2.25}$$

Model (2.25) is called a *normal linear regression model*. Under the assumptions of this model, hypothesis tests and confidence intervals can be developed. In practice, the assumption of normality is not as crucial is it might appear, as most of the tests we will develop remain valid for large n even when the responses are not normally distributed. The main significance of the normality therefore is to develop tests and confidence intervals that are valid for small sample sizes.

2.8.2 The Distribution of $\hat{\beta}_j$

Expressions for computing estimates of $\text{var}[\hat{\beta}_j]$ were given in Sects. 2.3.4 and 2.5.4. When a normal linear regression model (2.25) is adopted, the entire distributions of the regression parameters are known, not just the variance. Using Model (2.25), the $\hat{\beta}_j$ are random variables which follow normal distri-

butions, since $\hat{\beta}_j$ is a linear combination of the y_i (Sect. 2.5.2). Specifically, for normal linear regression models,

$$\hat{\beta}_j \sim N(\beta_j, \text{var}[\hat{\beta}_j]). \tag{2.26}$$

This means that $\hat{\beta}_j$ has a normal distribution with mean β_j and variance $\text{var}[\hat{\beta}_j]$. Note that $\text{var}[\hat{\beta}_j]$ is a product of σ (approximately inversely proportional to $\sqrt{n}$) and the known values of the explanatory variable and weights. From (2.26),

$$Z = \frac{\hat{\beta}_j - \beta_j}{\text{se}(\hat{\beta}_j)},$$

where $\text{se}(\hat{\beta}_j) = \sqrt{\text{var}[\hat{\beta}_j]}$, and Z has a standard normal distribution when σ^2 is known. When σ^2 is unknown, estimate σ^2 by s^2 and hence estimate $\text{var}[\hat{\beta}_j]$ by $\widehat{\text{var}}[\hat{\beta}_j]$. Then

$$T = \frac{\hat{\beta}_j - \beta_j}{\text{se}(\hat{\beta}_j)}$$

has a Student's t distribution with $n - p'$ degrees of freedom, where $\text{se}(\hat{\beta}_j) = \sqrt{\widehat{\text{var}}[\hat{\beta}_j]}$. Note that Student's t-distribution converges to the standard normal as the degrees of freedom increase.

2.8.3 Hypothesis Tests for β_j

Consider testing the null hypothesis H_0: $\beta_j = \beta_j^0$ against a one-sided alternative (H_A: $\beta_j > \beta_j^0$ or H_A: $\beta_j < \beta_j^0$) or a two-sided alternative (H_A: $\beta_j \neq \beta_j^0$), where β_j^0 is some hypothesized value of β_j (usually zero). The statistic

$$T = \frac{\hat{\beta}_j - \beta_j^0}{\text{se}(\hat{\beta}_j)} \tag{2.27}$$

is used to test this hypothesis. When H_0 is true, T has a t-distribution with $n - p'$ degrees of freedom when σ^2 is unknown, so we determine significance by referring to this distribution.

Each individual t-test determines whether evidence exists that the parameter is statistically significantly different from β_j^0 *in the presence of the other variables currently in the model.*

Example 2.16. After fitting Model (2.22) to the lung capacity data in R (data set: `lungcap`), the output of the `summary()` command in Fig. 2.6 (p. 51) reports information about the parameter estimates in the table in the centre of the output (starting from line 14):

- the `Estimate` column contains the parameter estimates $\hat{\beta}_j$;
- the `Std. Error` column contains the corresponding standard errors $\text{se}(\hat{\beta}_j)$;
- the `t value` column contains the corresponding t-statistic (2.27) for testing H_0: $\beta_j = 0$;
- the `Pr(>|t|)` column contains the corresponding two-tailed P-values for the hypothesis tests. (The one-tailed P-value is the two-tailed P-value divided by two.)

Line 22 in Fig. 2.6 (p. 51) regarding `Signif. codes` needs explanation. The `***` indicates a two-tailed P-value between 0 and 0.001; `**` indicates a two-tailed P-value between 0.001 and 0.01; `*` indicates a two-tailed P-value between 0.01 and 0.05; `.` indicates a two-tailed P-value between 0.05 and 0.10.

This information can be accessed directly using `coef(summary())`:

```
> round(coef( summary( LC.m1 ) ), 5)
             Estimate Std. Error   t value Pr(>|t|)
(Intercept) -1.94400    0.07864 -24.72067  0.00000
Age          0.02339    0.00335   6.98449  0.00000
Ht           0.04280    0.00168  25.48933  0.00000
GenderM      0.02932    0.01172   2.50196  0.01260
SmokeSmoker -0.04607    0.02091  -2.20311  0.02794
```

For example, consider a hypothesis test for β_4 (the coefficient for `Smoke`). To test H_0: $\beta_4 = 0$ against the alternative H_A: $\beta_4 \neq 0$ (in the presence of age, height and gender), the output shows that the t-score is $t = -2.203$, and the corresponding two-tailed P-value is 0.02794. Thus, some evidence exists that smoking status is statistically significant when age, height and gender are in the model. If gender was omitted from the model and the relevant null hypothesis retested, the test has a different meaning: this second test determines if age is significant in the model adjusted only for height (but not gender). Consequently, we should expect the test statistic and P-values to be different, and so the conclusion may differ also. □

2.8.4 Confidence Intervals for β_j

While hypothesis tests are useful for detecting statistical significance, often the *size* of the effect is of greater interest. This can be estimated by computing confidence intervals. The estimates $\hat{\beta}_j$ and the corresponding standard errors $\text{se}(\hat{\beta}_j)$ can be used to form $100(1-\alpha)\%$ confidence intervals for each estimate using

$$\hat{\beta}_j \pm t^*_{\alpha/2,n-p'}\text{se}(\hat{\beta}_j),$$

where $t^*_{\alpha/2,n-p'}$ is the value such that an area $\alpha/2$ is in each tail of the t-distribution with $n - p'$ degrees of freedom. Rather than explicitly using the

formula, confidence intervals are found in R using the `confint()` command. By default, 95% confidence intervals are produced; other levels are produced by using, for example, `level=0.90` in the call to `confint()`.

Example 2.17. For the lung capacity data (data set: `lungcap`), find the 95% confidence interval for all five regression coefficients in model `LC.m1` using `confint()`:

```
> confint( LC.m1 )
                   2.5 %       97.5 %
(Intercept) -2.098414941 -1.789581413
Age          0.016812109  0.029962319
Ht           0.039498923  0.046092655
GenderM      0.006308481  0.052330236
SmokeSmoker -0.087127344 -0.005007728
```

For example, the 95% confidence interval for β_4 is from -0.08713 to -0.005008. □

2.8.5 Confidence Intervals for μ

The fitted values $\hat{\mu}$ are used to estimate the mean value for given values of the explanatory variables. Using the expressions for computing $\text{var}[\hat{\mu}_g]$ (Sect. 2.3.5; Sect. 2.5.5), the $100(1-\alpha)\%$ confidence interval for the fitted value is

$$\hat{\mu}_g \pm t^*_{\alpha/2,n-p'}\text{se}(\hat{\mu}),$$

where $\text{se}(\hat{\mu}_g) = \sqrt{\text{var}[\hat{\mu}_g]}$, and where $t^*_{\alpha/2,n-p'}$ is the value such that an area $\alpha/2$ is in each tail of the t-distribution with $n-p'$ degrees of freedom. Rather than explicitly using the formulae, R returns the standard errors when making predictions using `predict()` with the input `se.fit=TRUE`, from which the confidence intervals can be formed.

Example 2.18. For the lung capacity data (data set: `lungcap`), suppose we wish to estimate $\mu = \text{E}[\log(\text{FEV})]$ for female smokers aged 18 who are 66 inches tall. Using R, we first create a new data frame containing the values of the explanatory variables for which we need to make the prediction:

```
> new.df <- data.frame(Age=18, Ht=66, Gender="F", Smoke="Smoker")
```

Then, use `predict()` to compute the estimates of μ:

```
> out <- predict( LC.m1, newdata=new.df, se.fit=TRUE)
> names(out)
[1] "fit"            "se.fit"         "df"             "residual.scale"
> out$se.fit
[1] 0.02350644
```

```
> tstar <- qt(df=LC.m1$df, p=0.975 ) # For a 95% CI
> ci.lo <- out$fit - tstar*out$se.fit
> ci.hi <- out$fit + tstar*out$se.fit
> CIinfo <- cbind( Lower=ci.lo, Estimate=out$fit, Upper=ci.hi)
> CIinfo
     Lower Estimate    Upper
1 1.209268 1.255426 1.301584
```

The prediction is $\hat{\mu} = 1.255$, and the 95% confidence interval is from 1.209 to 1.302. Based on the discussion in Sect. 2.7, an approximate confidence interval for E[FEV] is

```
> exp(CIinfo)
     Lower Estimate    Upper
1 3.351032 3.509334 3.675114
```

This idea can be extended to compute the confidence intervals for 18 year-old female smokers for varying heights:

```
> newHt <- seq(min(lungcap$Ht), max(lungcap$Ht), by=2)
> newlogFEV <- predict( LC.m1, se.fit=TRUE,
    newdata=data.frame(Age=18, Ht=newHt, Gender="F", Smoke="Smoker"))
> ci.lo <- exp( newlogFEV$fit - tstar*newlogFEV$se.fit )
> ci.hi <- exp( newlogFEV$fit + tstar*newlogFEV$se.fit )
```

Notice that the intervals do not have the same width over the whole range of the data:

```
> cbind( Ht=newHt, FEVhat=exp(newlogFEV$fit), SE=newlogFEV$se.fit,
        Lower=ci.lo, Upper=ci.hi, CI.Width=ci.hi - ci.lo)
   Ht   FEVhat         SE    Lower    Upper  CI.Width
1  46 1.491095 0.04886534 1.354669 1.641259 0.2865900
2  48 1.624341 0.04585644 1.484469 1.777392 0.2929226
3  50 1.769494 0.04289937 1.626540 1.925011 0.2984711
4  52 1.927618 0.04000563 1.781987 2.085151 0.3031639
5  54 2.099873 0.03719000 1.951990 2.258959 0.3069685
6  56 2.287520 0.03447163 2.137804 2.447722 0.3099183
7  58 2.491936 0.03187542 2.340743 2.652894 0.3121513
8  60 2.714619 0.02943370 2.562170 2.876138 0.3139672
9  62 2.957201 0.02718813 2.803464 3.119368 0.3159041
10 64 3.221460 0.02519123 3.065984 3.384820 0.3188364
11 66 3.509334 0.02350644 3.351032 3.675114 0.3240817
12 68 3.822932 0.02220493 3.659826 3.993308 0.3334820
13 70 4.164555 0.02135689 3.993518 4.342917 0.3493998
14 72 4.536705 0.02101728 4.353286 4.727852 0.3745665
15 74 4.942111 0.02121053 4.740502 5.152294 0.4117924
```

□

2.9 Analysis of Variance for Regression Models

A linear regression model, having been fitted to the data by least squares, yields a fitted value

$$\hat{\mu}_i = \hat{\beta}_0 + \sum_{j=1}^{p} x_{ij}\hat{\beta}_j$$

for each observation y_i. Each observation therefore can be separated into a component predicted by the model, and the remainder or residual that is left over, as

$$y_i = \hat{\mu}_i + (y_i - \hat{\mu}_i).$$

In other words, DATA = FIT + RESIDUAL.

The simplest possible regression model is that with $p = 0$ and no covariates x_{ij}. In that case $\hat{\mu} = \hat{\beta}_0 = \bar{y}_w$, where $\bar{y}_w = \sum_{i=1}^n w_i y_i / \sum_{i=1}^n w_i$ is the weighted mean of the observations. In order to evaluate the contribution of the covariates x_{ij}, it is more useful to consider the corresponding decomposition of the mean-corrected data,

$$y_i - \bar{y}_w = (\hat{\mu}_i - \bar{y}_w) + (y_i - \hat{\mu}_i).$$

Squaring each of these terms and summing them over i leads to the key identity

$$\text{SST} = \text{SSREG} + \text{RSS}$$

where $\text{SST} = \sum_{i=1}^n w_i(y_i - \bar{y}_w)^2$ is the *total sum of squares*, $\text{SSREG} = \sum_{i=1}^n w_i(\hat{\mu}_i - \bar{y}_w)^2$ is the *regression sum of squares*, and $\text{RSS} = \sum_{i=1}^n w_i(y_i - \hat{\mu}_i)^2$ is the *residual sum of squares*. The cross-product terms $(\hat{\mu}_i - \bar{y}_w)(y_i - \hat{\mu}_i)$ sum to zero, and so don't appear in this identity. The identity embodies the principle that variation in the response variable comes from two sources: firstly a systematic component that can be attributed to changes in the explanatory variables (SSREG), and secondly a random component that cannot be predicted (RSS). This identity is the basis of what is called *analysis of variance*, because it analyses the sources from which variance in the data arises.

It is of key interest to know whether the explanatory variables are useful predictors of the responses. This question can be answered statistically by testing whether the regression sum of squares SSREG is larger than would be expected due to random variation; in other words, whether SSREG is large relative to RSS after taking the number of explanatory variables into account. The null hypothesis is the assertion that $\beta_j = 0$ for all $j = 1, \ldots, p$. To develop such a test, first note that RSS/σ^2 has a chi-square distribution with $n - p'$ degrees of freedom, for a *normal* linear regression model. Likewise, under the null hypothesis, it can be shown that SSREG/σ^2 has a chi-square

Table 2.2 The general form of an analysis of variance table for a linear regression model (Sect. 2.9)

Source of variation	Sums of squares	df	Mean square	F
Systematic component	SSReg	$p' - 1$	$\text{MSReg} = \frac{\text{SSReg}}{p' - 1}$	$F = \frac{\text{MSReg}}{\text{MSE}}$
Random component	RSS	$n - p'$	$\text{MSE} = \frac{\text{RSS}}{n - p'} = s^2$	
Total variation	SST	$n - 1$		

distribution with $p' - 1$ degrees of freedom for a *normal* linear regression model. This means that the ratio

$$F = \frac{\text{SSReg}/(p' - 1)}{\text{RSS}/(n - p')} = \frac{\text{MSReg}}{\text{MSE}} \tag{2.28}$$

follows an F-distribution with $(p' - 1, n - p')$ degrees of freedom. The MSE, the *mean-square error*, is equal to s^2, the unbiased estimator of σ^2 that we have previously seen. MSReg is the mean-square for the regression.

A large value for F means that the proportion of the variation that can be explained by the systematic component is large relative to s^2; a small value for F means that the proportion of the variation that can be explained by the systematic component is small relative to s^2.

The computations are conveniently arranged in an analysis of variance (ANOVA) table (Table 2.2).

The R `summary()` command does not show the details of the ANOVA table (Fig. 2.6, p. 51), but the *results* are reported in the final line of output (line 26): the F-statistic is labelled `F-statistic`, followed by the corresponding degrees of freedom (labelled `DF`), and the P-value for the test (labelled `p-value`). The F-statistic and the corresponding degrees of freedom are returned using `summary(LC.m1)$fstatistic`. There is also an `anova()` function that is demonstrated in the next section.

The proportion of the total variation explained by the regression is the *coefficient of determination*,

$$R^2 = \frac{\text{SSReg}}{\text{SST}} = 1 - \frac{\text{RSS}}{\text{SST}}. \tag{2.29}$$

Clearly, by the definition, R^2 is bounded between zero and one. R^2 is sometimes also called *multiple* R^2, because it is equal to the squared Pearson correlation coefficient between the y_i and the fitted values $\hat{\mu}_i$, using the weights w_i. R reports the value of R^2 in the model `summary()`, as shown in Fig. 2.6 (p. 51), where R^2 is labelled `Multiple R-squared` on line 25.

Adding a new explanatory variable to the regression model cannot increase RSS and hence R^2 tends to increase with the size p of the model even if the explanatory variables have no real explanatory power. For this reason, some statisticians like to adjust R^2 for the number of explanatory variables in the model. The adjusted R^2, denoted $\bar{R}^2$, is defined by

$$\bar{R}^2 = 1 - \frac{\text{RSS}/(n-p')}{\text{SST}/(n-1)} = 1 - (1-R^2)\frac{n-1}{n-p'}.$$

It can be seen that $1-\bar{R}^2$ is the ratio of the residual to the total in the mean square column of the ANOVA table, whereas $1-R^2$ is the corresponding ratio for the sums of squares column. However $\bar{R}^2$ is not the ratio of MSReg to $\text{SST}/(n-1)$, because the entries is the mean square column do not sum. Unlike R^2, $\bar{R}^2$ may be negative. This occurs whenever MSReg $<$ MSE, which can be taken to indicate a very poor model. In the model `summary()` (Fig. 2.6, p. 51), R reports $\bar{R}^2$, called `Adjusted R-squared`. F and R^2 are closely related quantities (Problem 2.8), but it is F that is used to formally test whether the regression is statistically significant.

Example 2.19. For the lung capacity data (data set: `lungcap`), and Model (2.22) with age, height, gender and smoking status as explanatory variables, compute RSS and SST (recalling that `y = log(FEV)`):

```
> mu <- fitted( LC.m1 ); RSS <- sum( (y - mu)^2 )
> SST <- sum( (y - mean(y) )^2 )
> c(RSS=RSS, SST=SST, SSReg = SST-RSS)
     RSS      SST    SSReg
13.73356 72.52591 58.79236
> R2 <- 1 - RSS/SST          # Compute R2 explicitly
> c( "Output R2" = summary(LC.m1)$r.squared, "Computed R2" = R2,
    "adj R2"    = summary(LC.m1)$adj.r.squared)
  Output R2 Computed R2      adj R2
  0.8106393   0.8106393   0.8094722
```

The analysis of variance table (Table 2.3) compiles the necessary information. Compare these results to the output of `summary(LC.m1)` in Fig. 2.6 (p. 51). The summary of the F-test, which includes the numerator and denominator degree of freedom, is

```
> summary(LC.m1)$fstatistic
   value    numdf    dendf
694.5804   4.0000 649.0000
```

□

Table 2.3 The ANOVA table for Model (2.22) fitted to the lung capacity data, partitioning the total sum-of-squares into components due to the systematic and random components (Example 2.19)

Source	SS	df	MS	F
Systematic component	58.79	4	14.70	694.6
Random component	13.73	649	0.02116	
Total variation	72.53	653		

2.10 Comparing Nested Models

2.10.1 Analysis of Variance to Compare Two Nested Models

Rather than evaluating a single model, a researcher may wish to compare two models. First consider comparing two nested linear regression models. Model A is *nested* in Model B if Model A can be obtained from Model B by setting some parameter(s) in Model B to zero or, more generally, if Model A is a special case of Model B. For example, for the lung capacity data a researcher may wish to compare two models with the systematic components

$$\begin{aligned} \text{Model A:} \quad & \mu_A = \beta_0 + \beta_1 x_1 \qquad\qquad\qquad + \beta_4 x_4; \\ \text{Model B:} \quad & \mu_B = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \beta_4 x_4. \end{aligned}$$

Model A is nested in Model B, since Model A is a special case of Model B obtained by setting $\beta_2 = \beta_3 = 0$.

In comparing these models, we wish to know whether the more complex Model B is necessary, or whether the simpler Model A will suffice. Formally, the null hypothesis is that the two models are equivalent, so that we test H_0: $\beta_2 = \beta_3 = 0$ against the alternative that β_2 and β_3 are not both zero.

Consider using the `lungcap` data frame, and fitting the two models:

```
> LC.A <- lm( log(FEV) ~ Age               + Smoke, data=lungcap )
> LC.B <- lm( log(FEV) ~ Age + Ht + Gender + Smoke, data=lungcap )
```

Now compute the respective RSS:

```
> RSS.A <- sum( resid(LC.A)^2 )   # resid()  computes residuals
> RSS.B <- sum( resid(LC.B)^2 )
> c( ModelA=RSS.A, ModelB=RSS.B)
  ModelA   ModelB
28.91982 13.73356
```

The difference between the values of RSS is called the *sum-of-squares* (or SS):

```
> SS <- RSS.A - RSS.B; SS
[1] 15.18626
> DF <- df.residual(LC.A) - df.residual(LC.B); DF
[1] 2
```

The SS measures the reduction in the RSS gained by using the more complex Model B. This reduction in RSS is associated with an increase of two degrees of freedom. Is this reduction statistically significant?

The formal test requires comparing the SS divided by the change in the degrees of freedom, to the RSS for Model B divided by the degrees of freedom for Model B:

```
> df.B <- df.residual(LC.B); df.B
[1] 649
> Fstat <- (SS/DF) / ( RSS.B/df.B ); Fstat
[1] 358.8249
```

A P-value is found by comparing to an F-distribution with $(2, 649)$ degrees of freedom:

```
> pf(Fstat, df1=DF, df2=df.B, lower.tail=FALSE)
[1] 1.128849e-105
```

The P-value is almost zero, providing strong evidence that Model B is significantly different from Model A. In R, the results are displayed using `anova()`:

```
> anova( LC.A, LC.B )
Analysis of Variance Table

Model 1: log(FEV) ~ Age + Smoke
Model 2: log(FEV) ~ Age + Ht + Gender + Smoke
  Res.Df    RSS Df Sum of Sq      F    Pr(>F)
1    651 28.920
2    649 13.734  2    15.186 358.82 < 2.2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

More generally, consider fitting two nested models, say Model A and Model B, with systematic components

$$\begin{aligned} &\text{Model A:} \quad \hat{\mu}_A = \beta_0 + \beta_1 x_1 + \cdots + \beta_{p_A} x_{p_A} \\ &\text{Model B:} \quad \hat{\mu}_B = \beta_0 + \beta_1 x_1 + \cdots + \beta_{p_A} x_{p_A} + \cdots + \beta_{p_B} x_{p_B}. \end{aligned}$$

Model A is nested in Model B, because Model A is obtained by setting $\beta_{p_A+1}, \ldots, \beta_{p_B} = 0$ in Model B. The difference between the RSS computed for each model is the SS due to the difference between the models, based on $p'_B - p'_A$ degrees of freedom. Assuming H_0: $\beta_{p_A+1} = \cdots = \beta_{p_B} = 0$ is

true, the models are identical and SS is equivalent to residual variation. The test-statistic is

$$F = \frac{(\text{RSS}_A - \text{RSS}_B)/(p'_B - p'_A)}{s^2} = \frac{\text{SS}_B/(p'_B - p'_A)}{\text{RSS}_B/(n - p'_B)}. \tag{2.30}$$

A P-value is deduced by referring to an F-distribution with $(p'_B - p'_A, n - p'_B)$ degrees of freedom.

2.10.2 Sequential Analysis of Variance

The analysis of variance table just described is useful for comparing any two nested models. Commonly, a *sequence* of nested models is compared. For each pair of nested models in the sequence, the change in the RSS (the SS) and the corresponding change in the degrees of freedom are recorded and organised in a table.

As an example, consider model `LC.B` fitted to the `lungcap` data (Sect. 2.10.1, p. 61), which explores the relationship between `FEV` and `Smoke`, with the extraneous variables `Age`, `Ht` and `Gender`. A sequence of nested models could be compared:

```
> LC.0 <- lm( log(FEV) ~ 1, data=lungcap)   # No explanatory variables
> LC.1 <- update(LC.0, . ~ . + Age)         # Age
> LC.2 <- update(LC.1, . ~ . + Ht)          # Age and Height
> LC.3 <- update(LC.2, . ~ . + Gender)      # Age, Height and Gender
> LC.4 <- update(LC.3, . ~ . + Smoke)       # Then, add smoking status
```

Notice the use of `update()` to update models. To update model `LC.0` to form model `LC.1`, specify which components of `LC.0` should be changed. The first input is the model to be changed, and the second is the component of the model specification to change. Here we wish to change the formula given in `LC.0`. The left-hand side of the formula remains the same (as specified by `.`) but the original right-hand side (indicated by `.`) has `Age` added. Of course, `LC.1` could be also specified directly.

The RSS can be computed for each model:

```
> RSS.0 <- sum( resid(LC.0)^2 )
> RSS.1 <- sum( resid(LC.1)^2 )
> RSS.2 <- sum( resid(LC.2)^2 )
> RSS.3 <- sum( resid(LC.3)^2 )
> RSS.4 <- sum( resid(LC.4)^2 )
> RSS.list <- c( Model4=RSS.4, Model3=RSS.3, Model2=RSS.2,
                 Model1=RSS.1, Model0=RSS.0)
> RSS.list
  Model4   Model3   Model2   Model1   Model0
13.73356 13.83627 13.98958 29.31586 72.52591
```

Notice that the RSS reduces as the models become more complex. The change in the RSS, the SS, can also be computed:

```
> SS.list <- diff(RSS.list);   SS.list
    Model3     Model2     Model1     Model0
 0.1027098  0.1533136 15.3262790 43.2100549
```

The changes in the degrees of freedom between these nested models are all one in this example. As before, we compare these changes in RSS to an estimate of $\sigma^2 = \text{MSE}$, using the F-statistic (2.30):

```
> s2 <- summary(LC.4)$sigma^2   # One way to get MSE
> F.list <- (SS.list / 1) / s2;   F.list
     Model3      Model2      Model1      Model0
   4.853708    7.245064  724.266452 2041.956379
> P.list <- pf( F.list, 1, df.residual(LC.4), lower.tail=FALSE)
> round(P.list, 6)
  Model3   Model2   Model1   Model0
0.027937 0.007293 0.000000 0.000000
```

These computations are all performed in R by using `anova()`, and providing as input the final model in the set of nested models:

```
> anova(LC.4)
Analysis of Variance Table

Response: log(FEV)
           Df Sum Sq Mean Sq   F value    Pr(>F)
Age         1 43.210  43.210 2041.9564 < 2.2e-16 ***
Ht          1 15.326  15.326  724.2665 < 2.2e-16 ***
Gender      1  0.153   0.153    7.2451  0.007293 **
Smoke       1  0.103   0.103    4.8537  0.027937 *
Residuals 649 13.734   0.021
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The F-values and P-values are the same as those found in the calculations above.

This discussion shows that a series of *sequential* tests is performed. The last formally tests if `Smoke` is significant in the model, given that `Age`, `Ht` and `Gender` are already in the model. In other words, the F-test for `Smoke` *adjusts* for `Age`, `Ht` and `Gender`. In general, the F-tests in sequential ANOVA tables are always adjusted for all previous terms in the model.

Because the F-tests are adjusted for other terms in the model, numerous F-tests are possible to test for the effect of `Smoke`, depending on the order in which the corresponding nested models are compared. For example, tests based on `Smoke` include:

- Test for `Smoke` without adjusting for any other explanatory variables;
- Test for `Smoke` after first adjusting for `Age`;
- Test for `Smoke` after first adjusting for `Ht`;
- Test for `Smoke` after first adjusting for both `Age` and `Gender`.

These tests consider different hypotheses regarding `Smoke` so may produce different results. In contrast, t-tests (Sect. 2.8.3) present the same information after all explanatory variables are in the model whatever order the variables are added, as t-tests are adjusted for all other variables in the final model.

Because the t-tests of Sect. 2.8.3 always adjust for *all* other terms in the model, the results from the t- and F-tests are generally different. However the final F-test in a sequential ANOVA table, if it is on 1 degree of freedom, is equivalent to the corresponding two-sided t-test. For example, the P-value for `Smoke` in the above ANOVA table ($P = 0.0279$) is the same as the P-value for `Smoke` given in Sect. 2.8.3, and the F-statistic for `Smoke` is the square of the t-statistic for `Smoke`. In general, the square of a t-statistic on ν degrees of freedom yields an F-statistic on $(1, \nu)$ degrees of freedom, so any two-sided t-test can be expressed as an F-test.

The ANOVA table shows the results of F-tests for the variables in the presented order. The models higher in the table are special cases of the models lower in the table (that is, models higher in the table are nested within models lower in the table). The order in which the explanatory variables are fitted is important, except in very special cases (usually in an experiment explicitly designed to ensure the order of fitting is not important).

More generally, testing a series of sequential models is equivalent to separating the systematic component into contributions from each explanatory variable (Table 2.4).

Example 2.20. Model `LC.4` (in Sect. 2.10.2) fits the explanatory variables `Age`, `Ht`, `Gender` and `Smoke` in that order (data set: `lungcap`). Consider fitting the explanatory variables in reverse order:

```
> LC.4.rev <- lm(log(FEV) ~ Smoke + Gender + Ht + Age, data=lungcap)
> anova(LC.4.rev)
```

Table 2.4 The general form of an analysis of variance table for a normal linear regression model, separating the systematic component into the contributions for each explanatory variable (Sect. 2.10.2)

	Source of variation	SS	df	Mean square	F
⎧	Due to x_1	$\text{SS}(x_1)$	df_1	MS_1	$\frac{\text{MS}_1}{\text{MSE}}$
⎪	Due to x_2 (adjusted for x_1)	$\text{SS}(x_2 \mid x_1)$	df_2	MS_2	$\frac{\text{MS}_2}{\text{MSE}}$
⎪	Due to x_3 (adjusted for x_1 and x_2)	$\text{SS}(x_3 \mid x_1, x_2)$	df_3	MS_3	$\frac{\text{MS}_3}{\text{MSE}}$
⎪	⋮	⋮	⋮	⋮	⋮
⎩	Due to x_p (adjusted for $x_1, \ldots, x_{p-1}$)	$\text{SS}(x_p \mid x_1, \ldots, x_{p-1})$	df_p	MS_p	$\frac{\text{MS}_p}{\text{MSE}}$
	Due to randomness	RSS	$n - p'$	MSE	
	Total variation	SST	$n - 1$		

```
Analysis of Variance Table

Response: log(FEV)
           Df Sum Sq Mean Sq  F value    Pr(>F)
Smoke       1  4.334   4.334  204.790 < 2.2e-16 ***
Gender      1  2.582   2.582  122.004 < 2.2e-16 ***
Ht          1 50.845  50.845 2402.745 < 2.2e-16 ***
Age         1  1.032   1.032   48.783 7.096e-12 ***
Residuals 649 13.734   0.021
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The level of significance of `Smoke` depends on whether this variable is added first (model `LC.4`) or last, after adjusting for `Age`, `Ht` and `Gender`. Sometimes, a variable may be significant when added first, but not at all significant when added after other variables. Thus the effect of a variable may depend on whether or not the model is adjusted for other variables. □

2.10.3 Parallel and Independent Regressions

Section 2.10.1 discussed the general case of testing any two nested models. We now discuss a particular set of nested models that are commonly compared, using the lung capacity data `lungcap`. For simplicity, we consider the case of one covariate (height x_2) and one factor (smoking status x_4) to fix ideas.

A naive (and obviously untrue) model is that $\mu = \mathrm{E}[\log(\mathrm{FEV})]$ does not depend on smoking status or height (Fig. 2.7, p. 68, top left panel). The fitted systematic component is

$$\hat{\mu} = 0.9154, \tag{2.31}$$

with $\mathrm{RSS} = 72.53$ on 653 degrees of freedom. Note that this model simply estimates the mean value of $y = \log(\mathrm{FEV})$:

```
> mean(log(lungcap$FEV))
[1] 0.915437
```

To consider if the influence of height x_2 on $\mu = \mathrm{E}[\log(\mathrm{FEV})]$ is significant, the fitted model is (Fig. 2.7, top right panel)

$$\hat{\mu} = -2.271 + 0.05212x_2, \tag{2.32}$$

with $\mathrm{RSS} = 14.82$ on 652 degrees of freedom. This regression model does not differentiate between smokers and non-smokers. Is the relationship different for smokers and non-smokers?

To consider this, add smoking status x_4 as an explanatory variable (Fig. 2.7, bottom left panel):

$$\hat{\mu} = -2.277 + 0.05222x_2 - 0.006830x_4, \tag{2.33}$$

with RSS = 14.82 on 651 degrees of freedom, and where $x_4 = 0$ refers to non-smokers and $x_4 = 1$ to smokers. Using (2.33), the two separate systematic components are

$$\hat{\mu} = \begin{cases} -2.277 + 0.05222x_2 \text{ for non-smokers (set } x_4 = 0) \\ -2.284 + 0.05222x_2 \text{ for smokers (set } x_4 = 1) \end{cases}$$

with different intercepts. Model (2.33) produces two parallel regression lines; only the intercepts differ but are so similar than the two lines can hardy be distinguished on the plot (Fig. 2.7, bottom left panel). This model assumes two separate systematic components, but a common random component and so a common estimate of σ^2.

Notice that the regression equation intercepts for smokers and non-smokers are the same if the coefficient for x_4 is zero. Hence, to formally test if the intercepts are different, a test of the corresponding β is conducted. In R:

```
> printCoefmat(coef(summary(lm( log(FEV) ~ Ht + Smoke, data=lungcap))))
              Estimate Std. Error  t value Pr(>|t|)
(Intercept) -2.2767801  0.0656677 -34.6712   <2e-16 ***
Ht           0.0522196  0.0010785  48.4174   <2e-16 ***
SmokeSmoker -0.0068303  0.0205450  -0.3325   0.7397
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The evidence suggests that different intercepts are not needed when the slopes of the lines are common. This is not unexpected given Fig. 2.7.

Perhaps the relationships between $\mu = \text{E}[\log(\text{FEV})]$ and height have different intercepts and slopes for smokers and non-smokers also (Fig 2.7, bottom right panel). Different slopes can be modelled using the *interaction* between height and smoking status as an explanatory variable:

$$\hat{\mu} = -2.281 + 0.05230x_2 - 0.002294x_4 + \overbrace{0.002294x_2.x_4}^{\text{interaction}}, \tag{2.34}$$

with RSS = 14.82 on 650 degrees of freedom. Model (2.34) produces two separate systematic components; both the intercepts and slopes differ (Fig. 2.7, bottom right panel):

$$\hat{\mu} = \begin{cases} -2.281 + 0.05230x_2 \text{ for non-smokers (set } x_4 = 0) \\ -2.137 + 0.05000x_2 \text{ for smokers (set } x_4 = 1). \end{cases}$$

This is not equivalent to fitting two separate linear regression models, since the same estimate of σ^2 is shared by both systematic components.

Notice that the regression equation slopes for smokers and non-smokers are the same if the coefficient for the interaction between x_2 and x_4 is zero. Hence, to formally test if the slopes are different, a test of the corresponding

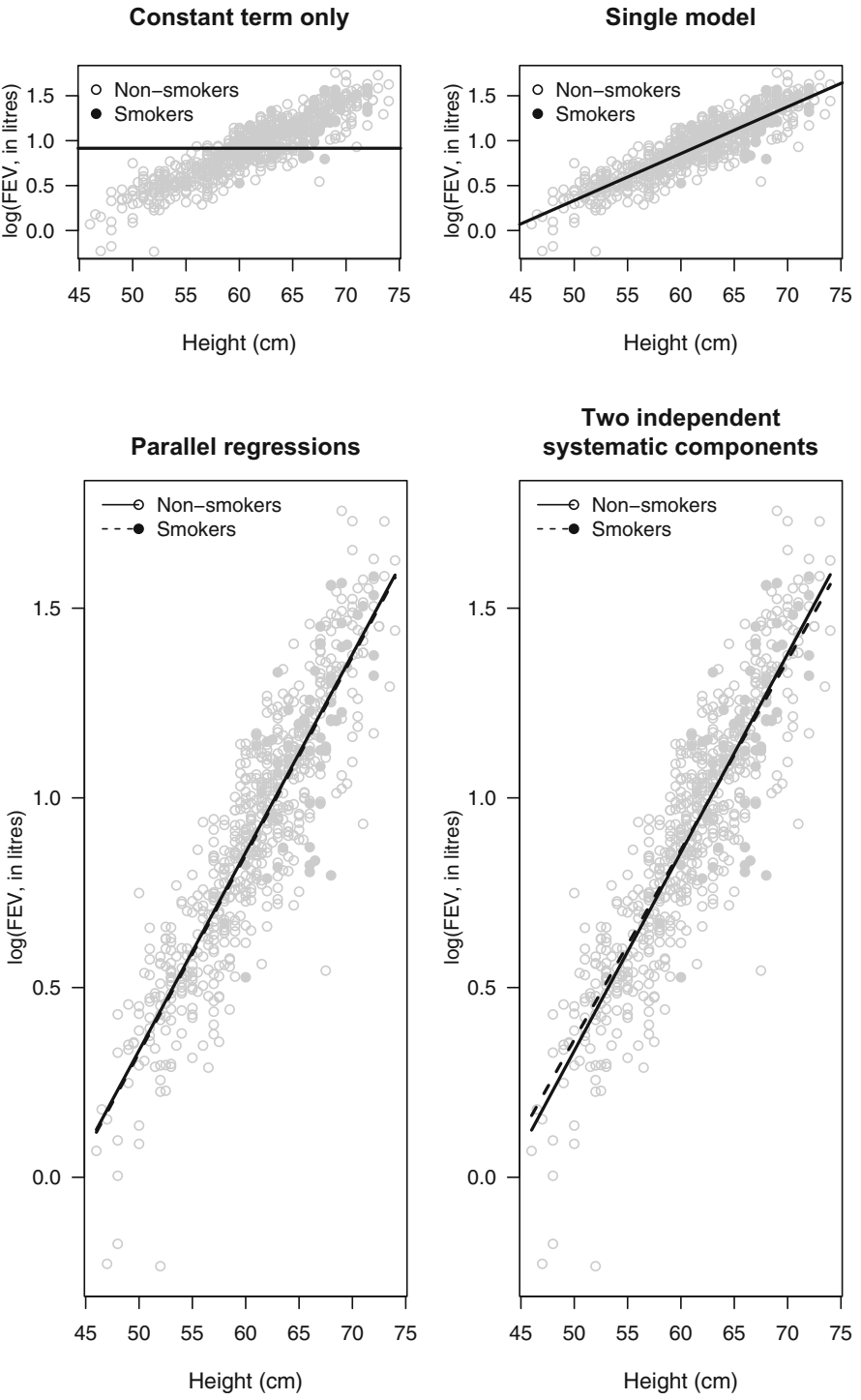

Fig. 2.7 The logarithm of FEV plotted against height. Top left: log(FEV) is constant; top right: log(FEV) depends on height only; bottom left: parallel regression lines; bottom right: two independent lines (Sect. 2.10.3)

Table 2.5 Summarizing Models (2.31)–(2.34) fitted to the lung capacity data (Sect. 2.10.3)

Source of variation	SS	df	MS	F
x_2	57.70	1	57.70	2 531
$x_4 \mid x_2$	0.002516	1	0.002516	0.1104
$x_1.x_4 \mid x_4, x_2$	0.003318	1	0.003318	0.1455
Due to randomness	14.82	650	0.02280	
Total variation	72.53	653		

β is conducted. R indicates the interaction between two explanatory variables by joining the interacting variables with : (a colon).

```
> LC.model <- lm( log(FEV) ~ Ht + Smoke + Ht:Smoke, data=lungcap)
```

A model including all main effects plus the interaction can also be specified using * (an asterisk). The above model, then, could be specified equivalently as:

```
> LC.model <- lm( log(FEV) ~ Ht * Smoke, data=lungcap)
```

There is no evidence to suggest that different intercepts and slopes are needed for smokers and non-smokers:

```
> printCoefmat(coef(summary(LC.model)))
                Estimate Std. Error  t value Pr(>|t|)
(Intercept)    -2.2814140  0.0668241 -34.1406   <2e-16 ***
Ht              0.0522961  0.0010977  47.6420   <2e-16 ***
SmokeSmoker     0.1440396  0.3960102   0.3637   0.7162
Ht:SmokeSmoker -0.0022937  0.0060125  -0.3815   0.7030
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Models (2.31)–(2.34) represent four ways to use linear regression models to model the relationship between $\mu = \mathrm{E}[\log(\mathrm{FEV})]$, height and smoking status. Notice that the models are nested, so the methods in Sect. 2.10.1 (p. 61) are appropriate for comparing the models statistically (Sect. 2.10.3). In the order in which the models are presented in Table 2.5, models higher in the table are *nested* within models lower in the table.

The value of RSS reduces as the models become more complex. R produces similar output using the `anova()` command, using the final model as the input:

```
> anova(LC.model)
Analysis of Variance Table

Response: log(FEV)
           Df Sum Sq Mean Sq   F value Pr(>F)
Ht          1 57.702  57.702 2531.1488 <2e-16 ***
Smoke       1  0.003   0.003    0.1104 0.7398
```

```
Ht:Smoke    1  0.003   0.003    0.1455 0.7030
Residuals 650 14.818   0.023
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The table indicates that the model including only `Ht` is hard to improve upon; neither `Smoke` nor the interaction are statistically significant.

This analysis shows that height is important in the model, but the impact of smoking status is less assured. Of course, in this example, we have not even considered age and gender, or even if the model above follows the necessary assumptions. In any case, the analysis suggests that height has a larger effect on $\mu = \mathrm{E}[\log(\text{FEV})]$ than smoking status in youth.

2.10.4 The Marginality Principle

For the model fitted above, suppose that the interaction between height and smoking status was necessary in the model. Then, height and smoking status main-effects should be included in the model whether they are statistically significant or not. Interactions indicate variations of the main-effect terms, which makes no sense if the main effects are not present. This idea is called the *marginality principle*. This principle states that:

- If higher-order powers of a covariate appear in a model, then the lower order power should also be in the model. For example, if x^2 is in a model then x should be also. (If x^2 remains in the model but x is removed, then the model is artificially constrained to fitting a quadratic model that has zero slope when $x = 0$, something which is not usually required.)
- If the interaction between two or more factors appears in the model, then the individual factors and lower-order interactions should appear also.
- If the interaction between factors and covariates appears in the linear model, then the individual factors and covariates should appear also.

2.11 Choosing Between Non-nested Models: AIC and BIC

The hypothesis tests discussed in Sect. 2.10 only apply when the models being compared are *nested*. However, sometimes researchers wish to compare non-nested models, so those testing methods do not apply. This section introduces quantities for comparing models that are not necessarily nested.

First, recall that the two criteria for selecting a statistical model are accuracy and parsimony (Sect. 1.10). The RSS simply measures the accuracy: adding a new explanatory variable to the model never makes the RSS larger,

and almost always makes it smaller. Adding many explanatory variables produces smaller values of the RSS, but also produces a more complicated model.

Akaike's An Information Criterion (AIC) balances these two criteria, by measuring the accuracy using the RSS but penalizing the complexity of the model as measured by the number of estimated parameters. For a normal linear regression model,

$$\text{AIC} = n\log(\text{RSS}/n) + 2p' \tag{2.35}$$

when σ^2 is unknown. Using this definition, smaller values of the AIC (closer to $-\infty$) represent better models. A formal, more general, definition for the AIC appears in Sect. 4.12. The term $2p'$ is called the *penalty*, since it penalizes more complex linear regression models (models with larger values of p') by a factor of $k = 2$. Note that the value of the AIC is not meaningful by itself; it is useful for comparing models.

Other quantities similar to the AIC are also defined, with different forms for the penalty. One example is the Bayesian Information Criterion (BIC), also called Schwarz's criterion [10]:

$$\text{BIC} = n\log(\text{RSS}/n) + p'\log n, \tag{2.36}$$

when σ^2 is unknown. The BIC is inclined to select lower dimensional (more parsimonious) models than is AIC, as the penalty for extra parameters is more severe ($k = \log n > 2$) unless the number of observations is very small.

The AIC and BIC focus on the two different purposes of a statistical model (Sect. 1.9). The AIC focuses more on creating a model for making good predictions. Extra explanatory variables may be included in the model if they are more likely to help than not, even though the evidence for their importance might not be convincing. The BIC requires stronger evidence for including explanatory variables, so produces simpler models having simpler interpretations. AIC is directed purely at prediction, while BIC is a compromise between interpretation and prediction. Neither AIC nor BIC are formal testing methods, so no test statistics or P-values can be produced.

Both the AIC and the BIC are found in R using the `extractAIC()` command. The AIC is returned by default, and the BIC returned by specifying the penalty `k=log(nobs(fit))` where `fit` is the fitted model, and `nobs()` extracts the number of observations used to fit the model.

Example 2.21. Consider the lung capacity data again (Example 1.1; data set: `lungcap`). Suppose the researcher requires smoking status x_4 in the model, and one of age x_1 or height x_2. The two possible systematic components to consider are

$$\begin{aligned} \text{Model A:} \quad & \mu_A = \beta_0 + \beta_1 x_1 \qquad\quad + \beta_4 x_4; \\ \text{Model B:} \quad & \mu_B = \beta_0 \qquad + \beta_2 x_2 + \beta_4 x_4. \end{aligned}$$

The models are not nested, so the methods of Sect. 2.10 are not appropriate. The AIC is extracted using R as follows:

```
> LC.A <- lm( log(FEV) ~ Age + Smoke, data=lungcap )
> extractAIC(LC.A)
[1]     3.000 -2033.551
> LC.B <- lm( log(FEV) ~ Ht  + Smoke, data=lungcap )
> extractAIC(LC.B)
[1]     3.000 -2470.728
```

The first value reported is the equivalent degrees of freedom; for linear regression models, the equivalent degrees of freedom is the number of estimated regression parameters in the model. The AIC is the second value reported; thus the AIC is lower (closer to $-\infty$) for the second model which uses `Ht`. To extract the BIC, the same function `extractAIC()` is used, but the penalty is adjusted:

```
> k <- log( length(lungcap$FEV) )
> extractAIC(LC.A, k = k)
[1]     3.000 -2020.102
> extractAIC(LC.B, k = k)
[1]     3.000 -2457.278
```

The BIC is lower (closer to $-\infty$) for the second model. The AIC and the BIC both suggest the combination of `Ht` and `Smoke` is more useful as a set of explanatory variables than the combination of `Age` and `Smoke`. This is not surprising, since `Ht` directly measures a physical trait. □

2.12 Tools to Assist in Model Selection

2.12.1 Adding and Dropping Variables

In situations where many explanatory variables are candidates for inclusion in the model, selecting the optimal set is tedious and difficult, especially because the order in which the variables are added is usually important. Exploring the possible models is more convenient using the R functions `add1()` and `drop1()`. These functions explore the impact of adding one variable (`add1()`) and dropping one variable (`drop1()`) from the current model, one at a time. The function `step()` repeatedly uses `add1()` and `drop1()` to suggest a model, basing the decisions on the values of the AIC (by default) or the BIC.

Example 2.22. Consider the lung capacity data (data set: `lungcap`), and the four explanatory variables `Age`, `Ht`, `Gender` and `Smoke`. The command `drop1()` is used by providing a model, and each term is removed one at a time:

```
> drop1( lm( log(FEV) ~ Age + Ht + Gender + Smoke, data=lungcap), test="F")
Single term deletions

Model:
log(FEV) ~ Age + Ht + Gender + Smoke
       Df Sum of Sq    RSS     AIC  F value    Pr(>F)
<none>              13.734 -2516.6
Age     1    1.0323 14.766 -2471.2  48.7831 7.096e-12 ***
Ht      1   13.7485 27.482 -2064.9 649.7062 < 2.2e-16 ***
Gender  1    0.1325 13.866 -2512.3   6.2598   0.01260 *
Smoke   1    0.1027 13.836 -2513.7   4.8537   0.02794 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The output shows the value of the AIC for the original model, and also when `Age`, `Ht`, `Gender` and `Smoke` are removed from model one at a time. The AIC is the smallest (closest to $-\infty$) when none of the explanatory variables are removed (indicated by the row labelled `<none>`), suggesting no changes are needed to the model. The F-test results for omitting terms are displayed using `test="F"`, otherwise `drop1()` reports only the AIC.

In a similar fashion, using `add1()` *adds* explanatory variables one at a time. Using `add1()` requires two inputs: the simplest and the most complex systematic components to be considered. For the lung capacity data, we are particularly interested in the relationship between FEV and smoking status, and so we ensure that the minimum model contains smoking status.

```
> LC.full   <- lm( log(FEV) ~ Age + Ht + Gender + Smoke, data=lungcap)
> add1( lm( log(FEV) ~ Smoke, data=lungcap),  LC.full, test="F" )
Single term additions

Model:
log(FEV) ~ Smoke
       Df Sum of Sq    RSS     AIC  F value    Pr(>F)
<none>              68.192 -1474.5
Age     1    39.273 28.920 -2033.5  884.045 < 2.2e-16 ***
Ht      1    53.371 14.821 -2470.7 2344.240 < 2.2e-16 ***
Gender  1     2.582 65.611 -1497.8   25.616 5.426e-07 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The output shows that any one of the explanatory variables can be added to the simple model `log(FEV) ~ Smoke` and improve the model (the AIC becomes closer to $-\infty$). Since the AIC is smallest when `Ht` is added, we would add `Ht` to the systematic component, and then use `add1()` again. □

2.12.2 Automated Methods for Model Selection

If many explanatory variables are candidates for inclusion in a statistical model, many statistical models are possible. For example, with ten possible

explanatory variables, $2^{10} = 1024$ models are possible, ignoring possible interactions. While comparing every possible model is an option, theory or practical knowledge are usually used to reduce the number of model comparisons needed. Nevertheless, many comparison may still be made, and so the task may be automated using computer software based on specific rules. The three most common automated procedures for selecting models are forward regression, backward elimination and stepwise regression.

Forward regression starts with essential explanatory variables in the model (often just the constant β_0), and each explanatory variable not in the current model is added one at a time. If adding any variables improves the current model, the variable making the greatest improvement is added, and the process is repeated with the remaining variables not in the model. At each step, the AIC closest to $-\infty$ is adopted. (The BIC can be used by setting the appropriate penalty.) The process is repeated with all explanatory variables not in the model until the model cannot be improved by adding more explanatory variables.

Backward elimination is similar but *removes* explanatory variables at each step. The process starts with all explanatory variables in the model, and at each step removes each explanatory variables in the current model one at a time. If removing any variables improves the current model, the variable making the greatest improvement is removed, and the process is repeated with the remaining variables in the model. At each step, the model with the AIC closest to $-\infty$ is adopted. The process is repeated with all explanatory variables in the model until the model cannot be improved by removing more explanatory variables.

At each step of stepwise regression, explanatory variables not in the model are *added* one at a time, and explanatory variables in the current model are *removed* one at a time. If adding or removing any variable improves the current model, the variable making the greatest improvement is added or removed as necessary, and the process is repeated. At each step the model with the AIC closest to $-\infty$ is adopted. Interactions are only considered between lower-order terms already in the current model, according to the marginality principle (Sect. 2.10.4). For example, R only considers adding the interaction `Ht:Gender` if both `Ht` and `Gender` are in the current model.

These procedures are implemented in the R function `step()`, which (by default) uses the AIC to select models. `step()` can perform forward regression (using the input argument `direction="forward"`), backward elimination (`direction="backward"`) or stepwise regression (`direction="both"`). The output is often voluminous if many steps are needed to find the final model and a large number of explanatory variables are being considered.

The `step()` function has three commonly-used inputs. The input `object` and the input `scope` together indicate the range of models for R to consider, and their use depends on which type of approach is used (as indicated by `direction`); see Example 2.23 for a demonstration.

Example 2.23. Consider again the lung capacity data `lungcap`. First, consider forward regression. The first argument in `step()` is the minimal acceptable model. From Example 2.22, no variables can be removed from the model

```
> min.model <- lm(log(FEV) ~ Age + Ht + Gender + Smoke, data=lungcap)
```

to improve the model, so we begin with this as the minimal model. We now use `step()` to suggest a model for the `lungcap` data, considering models as complex as:

```
> max.model <- lm( log(FEV) ~ (Smoke + Age + Ht + Gender)^2, data=lungcap)
```

which specifies all two-way interactions between the variables.

The use of `step()` requires the minimum model and maximum model that is to be considered to be specified. The output is voluminous, so is not shown.

```
> auto.forward <- step( min.model, direction="forward",
                       scope=list(lower=min.model, upper=max.model) )
```

The use of `step()` for backward elimination is similar:

```
> auto.backward <- step( max.model, direction="backward",
                        scope=list(lower=min.model, upper=max.model) )
```

The use of `step()` for stepwise regression (which uses `add1()` and `drop1()` repeatedly) is also similar.

```
> auto.both <- step( min.model, direction="both",
                    scope=list(lower=min.model, upper=max.model) )
```

In this case, the three approaches produce the same models:

```
> signif( coef(auto.forward), 3 )
(Intercept)         Age          Ht     GenderM SmokeSmoker
    -1.9400      0.0234      0.0428      0.0293     -0.0461
> signif( coef(auto.backward), 3 )
(Intercept) SmokeSmoker         Age          Ht     GenderM
    -1.9400     -0.0461      0.0234      0.0428      0.0293
> signif( coef(auto.both), 3 )
(Intercept)         Age          Ht     GenderM SmokeSmoker
    -1.9400      0.0234      0.0428      0.0293     -0.0461
```

Again, we note that we have not considered if the model is appropriate.

The three methods do not always produce the same suggested model. To explain, consider some explanatory variable `x1`. The variable `x1` might never enter the model using the forward and stepwise regression procedures, so interactions with `x1` are never even considered (using the marginality principle). However in backward elimination, an interaction involving `x1` might not be able to be removed from the model, so `x1` must remain in the model (using the marginality principle). □

2.12.3 Objections to Using Stepwise Procedures

Automated stepwise procedures may be convenient (and appear in most statistical packages), but numerous objections exist [6, §4.3]. The objections are philosophical in nature (stepwise methods do not rely on any theory or understanding of the data; stepwise methods test hypothesis that are never asked, or even of interest), or relate to multiple testing issues (standard errors of the regression parameter estimates in the final model are too low; P-values are too small; confidence intervals are too narrow; R^2 values are too high; the distribution of the ANOVA test statistic does not have an F-distribution; regression parameter estimates are too large in absolute value; models selected using automated procedures often do not fit well to new data sets). Many authors strongly recommend against using automated procedures. Comparing *all* possible sub-models presents the same objections. Other methods may be used to assist in model selection [3, 13].

2.13 Case Study

A study [15, 16] compiled data from 90 countries (29 industrialized; 61 non-industrialized) on the average annual sugar consumption and the estimated mean number of decayed, missing and filled teeth (DMFT) at age 12 years (Table 2.6; data set: `dental`). A plot of the data (Fig. 2.8, left panel) suggests a relationship between DMFT and sugar consumption. Also, whether or not the country is industrialized or not seems important (Fig. 2.8, right panel):

```
> data(dental); summary(dental)
      Country      Indus        Sugar            DMFT
 Albania  : 1   Ind   :29   Min.   : 0.97   Min.   :0.300
 Algeria  : 1   NonInd:61   1st Qu.:14.53   1st Qu.:1.600
 Angolia  : 1               Median :33.79   Median :2.300
 Argentina: 1               Mean   :30.14   Mean   :2.656
 Australia: 1               3rd Qu.:44.32   3rd Qu.:3.350
 Austria  : 1               Max.   :63.02   Max.   :8.100
 (Other)  :84
> plot( DMFT ~ Sugar, las=1, data=dental, pch=ifelse( Indus=="Ind", 19, 1),
    xlab="Mean annual sugar consumption\n(kg/person/year)",
    ylab="Mean DMFT at age 12")
> legend("topleft", pch=c(19, 1), legend=c("Indus.","Non-indus."))
> boxplot(DMFT ~ Indus, data=dental, las=1,
     ylab="Mean DMFT at age 12", xlab="Type of country")
```

Consider fitting the linear regression model, including interactions:

```
> lm.dental <- lm( DMFT ~ Sugar * Indus, data=dental)
> anova(lm.dental)
Analysis of Variance Table
```

Table 2.6 The estimated mean number of decayed, missing and filled teeth (DMFT) at age 12 years, and the mean annual sugar consumption (in kg/person/year, computed over the five years prior to the survey) for 90 countries. The first five observations for both categories are shown (Sect. 2.13)

Industrialized			Non-industrialized		
Country	Mean annual sugar consumption	DMFT	Country	Mean annual sugar consumption	DMFT
Albania	22.16	3.4	Algeria	36.60	2.3
Australia	49.96	2.0	Angolia	12.00	1.7
Austria	47.32	4.4	Argentina	34.56	3.4
Belgium	40.86	3.1	Bahamas	34.40	1.6
Canada	42.12	4.3	Bahrain	34.86	1.3
⋮	⋮	⋮	⋮	⋮	⋮

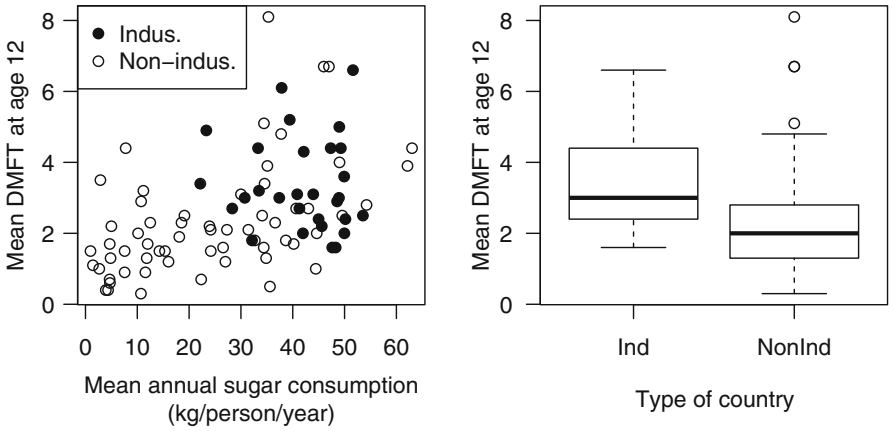

Fig. 2.8 Left panel: a plot of the mean number of decayed, missing and filled teeth (DMFT) at age 12 against the mean annual sugar consumption in 90 countries; right panel: a boxplot showing a difference in the distributions between the mean DMFT for industrialized and non-industrialized countries (Sect. 2.13)

```
Response: DMFT
            Df  Sum Sq Mean Sq F value    Pr(>F)
Sugar        1  49.836  49.836 26.3196 1.768e-06 ***
Indus        1   1.812   1.812  0.9572   0.33065
Sugar:Indus  1   6.674   6.674  3.5248   0.06385 .
Residuals   86 162.840   1.893
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

From this ANOVA table, the effect of sugar consumption is significant without adjusting for any other variables. The effect of `Indus` is not significant after adjusting for `Sugar`. The interaction between sugar consumption and whether

the country is industrialized is marginally significant after adjusting for sugar consumption and the industrialization. Consider the fitted model:

```
> coef( summary( lm.dental ) )
                     Estimate Std. Error    t value    Pr(>|t|)
(Intercept)        3.90857067 1.28649859  3.0381461 0.003151855
Sugar             -0.01306504 0.03014315 -0.4334332 0.665785323
IndusNonInd       -2.74389029 1.32480815 -2.0711605 0.041341018
Sugar:IndusNonInd  0.06004128 0.03198042  1.8774386 0.063847913
```

This output indicates that the mean sugar consumption is *not* significant after adjusting for the other variables. Furthermore, the coefficient for the sugar consumption is negative (though not statistically significant), suggesting greater sugar consumption is associated with *lower* mean numbers of DMFT. Recall this interpretation is for `Indus=="Ind"` (that is, for industrialized countries, when `Indus=0`). For non-industrialized countries, the coefficient for sugar consumption is

```
> sum( coef(lm.dental)[ c(2, 4) ] )
[1] 0.04697624
```

For non-industrialized countries, the coefficient for the sugar consumption is positive. Plotting the two lines (using `abline()`) is informative (Fig. 2.9):

```
> dental.cf <- coef( lm.dental )
> abline(a=dental.cf[1], b=dental.cf[2], lwd=2, lty=1)
> abline(a=sum( dental.cf[c(1, 3)]), b=sum(dental.cf[c(2, 4)]),
        lwd=2, lty=2)
```

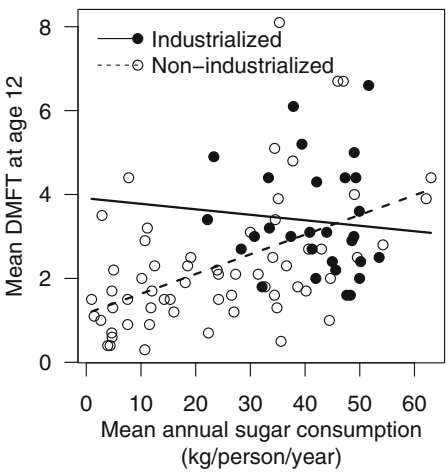

Fig. 2.9 A plot of the mean number of decayed, missing and filled teeth (DMFT) at age 12 and the mean annual sugar consumption in 90 countries showing the fitted model (Sect. 2.13)

Both the intercept and slope for `NonInd` are computed as the sum of the appropriate two coefficients.

Both the ANOVA F-test and the t-test show the interaction is of marginal importance. In fact, the two tests are equivalent (for example, compare the corresponding P-values). We decide to retain the interaction, so `Sugar` and `Indus` must remain in the model by the marginality principle (Sect. 2.10.3).

How can the model be interpreted? For non-industrialized countries, increasing average sugar consumption is related to increasing average number of DMFT at age 12 in children. An increase in mean annual sugar consumption of one kg/person/year is associated with a mean increase of $-0.01307 + 0.06004 = 0.04698$ DMFT in children at age 12. For industrialized countries, the average number of DMFT at age 12 appears to be unrelated to sugar consumption. Since industrialized countries in general have superior personal dental hygiene, dental facilities, and fluoridation of water, the effect of sugar consumption on DMFT may be reduced. However, note that the data for the industrialized countries span a much narrower range of sugar consumptions than those for non-industrialized countries:

```
> range( dental$Sugar[dental$Indus=="Ind"] )      # Industrialized
[1] 22.16 53.54
> range( dental$Sugar[dental$Indus=="NonInd"] )   # Non-industrialized
[1]  0.97 63.02
```

Note that the mean number of DMFT is recorded for children at age 12 (that is, for individuals), but the sugar consumption is an average for the whole population. This means that any connection between the sugar consumption and number of DMFT for *individuals* cannot be made. For example, individuals who do *not* consume sugar may be those individuals with the larger numbers of DMFT. Assuming that the relationships observed for a population also applies to individuals within the population is called the *ecological fallacy*. Also, since the data are observational, no cause-and-effect can be inferred. Even though the regression model has been successfully fitted, closer inspection suggests the model can be improved (Sect. 3.15.1).

2.14 Using R for Fitting Linear Regression Models

An introduction to using R is given in Appendix A (p. 503). For fitting linear regression models, the function `lm()` is used, as has been demonstrated numerous times in this chapter (Sects. 2.6 and 2.10.3 are especially relevant). Common inputs to `lm()` are:

- `formula`: The first input is the model formula, taking the form `y ~ x1 + x2 + x3 + x1:x2` as an example.

- `data`: The data frame containing the variables may be given as an input using the `data` argument (in the form `data=lungcap`).
- `weights`: The prior weights are supplied using the `weights` input argument. The default is to set all prior weights to one.
- `subset`: Sometimes a model needs to be fitted to a subset of the data, when the `subset` input is used. For example, to fit a linear regression model for only the females in the lung capacity data, use, for example
  `lm(log(FEV) ~ Age, data=lungcap, subset=(Gender=="F"))`
  since `Gender=="F"` selects females. Alternatively, the `subset()` function can be used to create a data frame that is a subset of the original data frame; for example:
  `lm(log(FEV) ~ Age, data=subset(lungcap, Gender=="F"))`

Other inputs are also defined; see `?lm` for more information. The explanatory variables in the formula are re-ordered so that all main effects are fitted before any interactions. Furthermore, all two-variables interactions are fitted, then all three-variable interactions, and so on. Use `terms()` to fit explanatory variables in a given order.

The function `update()` updates a model. Rather than specifying the model completely, only the *changes* from a current model are given (see Sect. 2.10.1, p. 61). Typical use: `update(old, changes)`, where `old` is the old model, and `changes` indicates the *changes* to the `old` model. Typically `changes` specifies a different formula from the `old` model. The `changes` formula may contain dots `.` on either side of the `~`, which are replaced by the expression in the `old` formula on the corresponding side of the formula.

Usually, the output from a fitted model is sent to an output object: `fit <- lm( log(FEV) ~ Age + Ht + Gender + Smoke, data=lungcap)`, for example. The output object `fit` contains substantial information; see `names(fit)`. The most useful information is extracted from `fit` using extractor functions, which include:

- `coef(fit)` (or `coefficients(fit)`) extracts the parameter estimates $\hat{\beta}_j$;
- `df.residual(fit)` extracts the residual degrees of freedom;
- `fitted(fit)` (or `fitted.values(fit)`) extracts the fitted values $\hat{\mu}$.

Other useful R functions used with linear regression models include:

`summary(fit)`: The `summary()` of the model prints the following: the parameter estimates with the corresponding standard errors, t-statistics and two-tailed P-values for testing H_0: $\beta_j = 0$; the estimate of s; the value of R^2; the value of $\bar{R}^2$; the results of the overall ANOVA test for the regression. See Fig. 2.6 (p. 51).
The output of `summary()` (for example, `out <- summary(fit)`) contains substantial information (see `names(out)`). For example, `out$r.squared` displays the value of R^2 and `out$sigma` displays the value of s. `coef(out)`

displays the parameter estimates and standard errors, plus the t-values and two-tailed P-values for testing H_0: $\beta_j = 0$. See `?summary.lm` for further information.

`anova()`: The `anova()` function can be used in two ways:

1. `anova(fit)`: When a single model `fit` is given as input, an ANOVA table is produced that sequentially tests the significance of each explanatory variable as it is added to the model (Sect. 2.10.2).
2. `anova(fit1, fit2, ...)`: Compares any set of fitted nested models `fit1`, `fit2` and so on by providing all models to `anova()`. The models are then tested against one another in the specified order, where models earlier in the list of models are nested in later models (Sect. 2.10.1).

`confint(fit)`: Returns the 95% confidence interval for all the regression coefficients β_j in the systematic component. For different confidence levels, use `confint(fit, level=0.99)`, for example, which creates 99% confidence intervals.

`drop1()` and `add1()`: Drops or adds explanatory variables one at a time from the given model using the AIC by default, while obeying the marginality principle. F-test results are displayed by using `test="F"`. To use `add1()`, the second input shows the maximum scope of the models to be considered

`step()`: Uses automated methods for suggesting a linear regression model based on the AIC by default. Common usage is `step(object, scope, direction)`, where `direction` is one of `"forward"` for forward regression, `"backward"` for backward elimination, or `"both"` for stepwise regression. `object` is an initial linear regression model, and `scope` defines extent of the models to be considered. Section 2.12.2 (p. 73) demonstrates the use of `step()` for the three types of automated methods. Decisions can be based on the BIC by using the input `k=log(nobs(fit))`, where `fit` is the fitted model.

`extractAIC(fit)`: Returns the number of estimated regression parameters as the first output value, and the AIC for the given model as the second output value. To compute the BIC instead of the AIC, use `extractAIC(fit, k=log(nobs(fit)))`, where `fit` is the fitted model.

`abline()`: Draws a straight line on the current plot. In the form `abline(a=2, b=-3)`, the straight line with intercept 2 and slope -3 is drawn. For a simple linear regression `model`, the slope and intercept are returned using `coef(fit)`, so that `abline(coef(fit))` draws the systematic component of the fitted simple linear regression model. The form `abline(h=1)` draws a horizontal line at $y = 1$, and the form `abline(v=-1)` draws a vertical line at $x = -1$.

2.15 Summary

Chapter 2 focuses on linear regression models. These models have the form (Sect. 2.2):

$$\begin{cases} \text{var}[y_i] = \sigma^2/w_i \\ \mu_i = \beta_0 + \sum_{j=1}^{p} \beta_j x_{ji}, \end{cases}$$

where $\text{E}[y_i] = \mu_i$, the w_i are known positive prior weights, σ^2 is the unknown variance, and $\beta_0, \ldots, \beta_p$ are the unknown regression parameters. There are p explanatory variables, and p' parameters β_j to be estimated.

Special names are given in special cases (Sect. 2.2):

- Simple linear regression models refer to the case with $p = 1$;
- Ordinary linear regression models have all prior weights set to one (to be distinguished from weighted linear regression models);
- Multiple linear regression models refer to cases where $p > 1$;
- Normal linear regression models refers to models with the additional assumption that $y_i \sim N(\mu_i, \sigma^2/w_i)$ (Sect. 2.8.1).

Matrix notation can be used to write these models compactly (Sect. 2.5.1).

The parameters β_j in the linear regression model are estimated using least-squares estimation, by minimizing the sum of the squared deviations between y_i and μ_i (Sect. 2.4). These estimates are denoted $\hat{\beta}_j$. The residual sum-of-squares is $\text{RSS} = \sum_{i=1}^{n} w_i(y_i - \hat{\mu}_i)^2$, where $\hat{\mu}_i = \hat{\beta}_0 + \sum_{j=1}^{p} \hat{\beta}_j x_{ji}$ are called the fitted values (Sect. 2.4).

For simple linear regression, formulae exist for computing the least-squares estimates of the regression parameters (Sect. 2.3.2). More generally, the values of $\hat{\beta}_0, \ldots, \hat{\beta}_p$ are estimated using matrix algebra (Sect. 2.4). In practice, linear regression models are fitted in R using `lm()` (Sect. 2.6). The estimated regression parameters have standard error $\text{se}(\hat{\beta}_j)$ (Sects. 2.3.4 and 2.5.4).

An unbiased estimate of the variance of the randomness (Sect. 2.4.2) is $s^2 = \text{RSS}/(n - p')$, where $n - p'$ is called the residual degrees of freedom.

To perform inference, it is necessary to also assume that the responses follow a normal distribution, so that $y_i \sim N(\mu_i, \sigma^2/w_i)$. Under this assumption, the $\hat{\beta}_j$ have a normal distribution (Sect. 2.8.2), and a test of H_0: $\beta_j = \beta_j^0$ (for some given value β_j^0) against a one- or two-tailed alternative can be performed using a t-test (Sect. 2.8.3). Furthermore, a $100(1 - \alpha)\%$ confidence interval for β_j can be formed using $\hat{\beta}_j \pm t^*_{\alpha/2,n-p'}\text{se}(\hat{\beta}_j)$, where $t^*_{\alpha/2,n-p'}$ is the value of t on $n - p'$ degrees of freedom such that an area $\alpha/2$ is in each tail (Sect. 2.8.4).

The significance of the regression model as a whole can be assessed by comparing the ratio of the variation due to the systematic component to the variation due to the random component, using an F-test (Sect. 2.9).

Each observation can be separated into a component predicted by the model, and the residual: DATA = FIT + RESIDUAL. In terms of sums of squares, SST = SSREG + RSS. Then, the multiple R^2 measures the proportion of the total variation explained by the systematic component (Sect. 2.9): $R^2 =$ SSREG/SST. The adjusted R^2, denoted $\bar{R}^2$, modifies R^2 to adjust for the number of explanatory variables.

Any two nested models can be compared using an F-test (Sect. 2.10.1). The significance of individual explanatory variables can be tested sequentially using F-tests by partitioning the sum-of-squares due to the systematic component into contributions for each explanatory variable (Sect. 2.10.2). An important application of nested models is testing for parallel and independent regressions (Sect. 2.10.3). For non-nested models, comparisons are possible using the AIC and BIC (Sect. 2.11).

Some tools are available to help with model selection, but must be used with extreme caution (Sect. 2.12.3). The R functions `drop1()` and `add1()` drop or add (respectively) explanatory variables one at a time from a model (Sect. 2.12.1). Forward regression, backward elimination and step-wise selection procedures are automated procedures for choosing models (Sect. 2.12.2).

Finally, any regression coefficients should be interpreted within the limitations of the model and the data (Sect. 2.7).

Problems

Selected solutions begin on p. 530. Problems preceded by an asterisk * refer to the optional sections in the text, and may require matrix manipulations.

2.1. In this problem, we consider two ways of writing the systematic component of a simple linear regression model.

1. Interpret the meaning of the constant term β_0 when the systematic component is written as $\mu = \beta_0 + \beta_1 x$.
2. Interpret the meaning of the constant term α_0 when the systematic component is written as $\mu = \alpha_0 + \beta_1(x - \bar{x})$.

2.2. For simple linear regression, show that the simultaneous solutions to $\partial S/\partial\beta_0 = 0$ and $\partial S/\partial\beta_1 = 0$ in (2.4) and (2.5) produce the solutions shown in (2.6) and (2.7) (p. 37).

* **2.3.** In the case of simple linear regression with all weights set to one, show that

$$\mathrm{X}^T\mathrm{W}\mathrm{X} = \begin{bmatrix} n & \sum x \\ \sum x & \sum x^2 \end{bmatrix},$$

where the summations are over $i = 1, 2, \ldots, n$. Hence, show that

$$\hat{\beta}_1 = \frac{\sum xy - \sum x \sum y/n}{\sum x^2 - (\sum x)^2/n}.$$

* **2.4.** Show that the least-squares estimator of $\boldsymbol{\beta}$ in the linear regression model is $\hat{\boldsymbol{\beta}} = (\mathrm{X}^T\mathrm{W}\mathrm{X})^{-1}\mathrm{X}^T\mathrm{W}\mathbf{y}$, by following these steps.

1. Show that $S = (\mathbf{y}-\mathrm{X}\boldsymbol{\beta})^T\mathrm{W}(\mathbf{y}-\mathrm{X}\boldsymbol{\beta}) = \mathbf{y}^T\mathrm{W}\mathbf{y}-2\boldsymbol{\beta}^T\mathrm{X}^T\mathrm{W}\mathbf{y}+\boldsymbol{\beta}^T\mathrm{X}^T\mathrm{W}\mathrm{X}\boldsymbol{\beta}$. S is the sum of the squared deviations.
2. Differentiate S with respect to $\boldsymbol{\beta}$ to find $\mathrm{d}S/\mathrm{d}\boldsymbol{\beta}$. (HINT: Differentiating $\boldsymbol{\beta}^T\mathrm{M}\boldsymbol{\beta}$ with respect to $\boldsymbol{\beta}$ for any compatible matrix M gives $2\mathrm{M}\boldsymbol{\beta}$.)
3. Use the previous result to find the value of $\hat{\boldsymbol{\beta}}$ minimizing the value of S.

2.5. For simple linear regression, show that $\hat{\beta}_1$ defined by (2.7) is an unbiased estimator of β_1. That is, show that $\mathrm{E}[\hat{\beta}_1] = \beta_1$. (HINT: $\sum w_i(x_i - \bar{x})a = 0$ for any constant a.)

* **2.6.** Show that $\hat{\boldsymbol{\beta}} = (\mathrm{X}^T\mathrm{W}\mathrm{X})^{-1}\mathrm{X}^T\mathrm{W}\mathbf{y}$ is an unbiased estimator of $\boldsymbol{\beta}$. That is, show $\mathrm{E}[\hat{\boldsymbol{\beta}}] = \boldsymbol{\beta}$.

* **2.7.** Show that the variance–covariance matrix of $\hat{\boldsymbol{\beta}}$ is $\mathrm{var}[\hat{\boldsymbol{\beta}}] = (\mathrm{X}^T\mathrm{W}\mathrm{X})^{-1}\sigma^2$, using that $\mathrm{var}[\mathrm{C}\mathbf{y}] = \mathrm{C}\mathrm{var}[\mathbf{y}]\mathrm{C}^T$ for a constant matrix C.

2.8. Show that the F-statistic (2.28) and R^2 (2.29) are related by

$$F = \frac{R^2/(p'-1)}{(1-R^2)/(n-p')}.$$

* **2.9.** Consider a simple linear regression model with systematic component $\mu = \beta_0 + \beta_1 x$. Suppose we wish to design an experiment with $n = 5$ observations, when σ^2 is known to be 1. Suppose three designs for the experiment are considered. In Design A, the values of the explanatory variable are $x = 1$, 1, -1, -1 and 0. In Design B, the values are $x = 1$, 1, 1, 1 and -1. In Design C, the values are $x = 1$, 0.5, 0, -0.5 and -1.

1. Write the model matrix X for each design.
2. Compute $\mathrm{var}[\hat{\mu}]$ for each design.
3. Plot $\mathrm{var}[\hat{\mu}]$ for x_g between -1 and 1. When would Design A be preferred, and why? When would Design B be preferred, and why? When would Design C be preferred, and why?

2.10. Assume that a quantitative response variable y and a covariate x are related by some smooth function f such that $\mu = f(x)$ where $\mu = \mathrm{E}[y]$.

1. Assuming that the necessary derivatives exist, find the first-order Taylor series expansion of $f(x)$ expanded about $\bar{x}$, where $\bar{x}$ is the mean of x.
2. Rearrange this expression into the form of a multiple regression model.
3. Explain how this implies that regression models are locally linear.

2.11. In Sect. 2.7, an interpretation for a model with systematic component $\mu = \mathrm{E}[\log y] = \beta_0 + \beta_1 x$ was discussed.

1. Use a Taylor series expansion of $\log y$ about $\mu = \mathrm{E}[y]$.
2. Find the expected value of both sides of this equation, and hence show that $\mathrm{E}[\log y] \approx \log \mathrm{E}[y] = \log \mu$.
3. Using this information, show that an increase in the value of x by 1 is associated (approximately) with a change in μ by a factor of $\exp(\beta_1)$.

2.12. Using R, produce a vector of 30 random numbers `y` from a standard normal distribution (use `rnorm()`). Generate a second vector of 30 random numbers `x` from a standard normal distribution. Find the P-value for testing if the explanatory variable `x` is significantly related to `y` using the regression model `lm(y ~ x)`.

Repeat the process a large number of times, say 1000 times. What proportion of the P-values are less than 5%? Less than 10%? What is the lesson?

2.13. A study [7] exposed sleeping people (males and females) of various ages to four different fire cues (a crackling noise, a shuffling noise, a flickering light, an unpleasant smell), and recorded the response time (in seconds) for the people to wake. Use the partially complete ANOVA table (Table 2.7) to answer the following questions.

1. Determine the degrees of freedom omitted from Table 2.7.
2. Determine how many observations were used in the analysis.
3. Find an unbiased estimate of σ^2.
4. Determine which explanatory variables are statistically significant for predicting response time, using sequential F-tests.
5. The analysed data are for participants who actually woke during the experiment; some failed to wake at all and were omitted from the analysis. Explain how this affects the interpretation of the results.
6. Compute the AIC for the three nested models implied by Table 2.7. What model is suggested by the AIC?
7. Compute the BIC for the three nested models implied by Table 2.7. What model is suggested by the BIC?
8. Compute R^2 and the adjusted R^2 for the three models implied by Table 2.7. What model is suggested by the R^2 and the adjusted R^2?

Table 2.7 An ANOVA table for fitting a linear regression model to the response time as a function of various fire cues and extraneous variables (Problem 2.13)

Source of variation	df	SS
Cue	?	117,793
Sex	?	2659
Age	3	22,850
Residual	60	177,639

Table 2.8 The parameter estimates and the standard errors in the linear regression model for estimating the systolic blood pressure (in mm Hg) in Ghanaian men aged between 18 and 65 (Problem 2.14)

Explanatory variable	$\hat{\beta}_j$	$\text{se}(\hat{\beta}_j)$
Constant	100.812	13.096
Age (in years)	0.332	0.062
Waist circumference (in cm)	0.411	0.090
Alcohol (yes: 1; no: 0)	−3.003	1.758
Smoking (yes: 1; no: 0)	−0.362	2.732
Ambient temperature (in °C)	−0.521	0.262

9. Compare the models suggested by the ANOVA table, the AIC, the BIC, R^2 and the adjusted R^2. Comment.

2.14. Numerous studies have shown an association between seasonal ambient temperature (in °C) and blood pressure (in mm Hg). A study of 574 rural Ghanaian men aged between 18 and 65 studied this relationship [9] (and also included a number of extraneous variables) using a linear regression model, producing the results in Table 2.8.

1. Compute the P-values for each term in the model, and comment.
2. After adjusting for age, waist circumference, alcohol consumption and smoking habits, describe the relationship between ambient temperature and systolic blood pressure.
3. Plot the line describing the relationship between ambient temperature and systolic blood pressure for 30-year-old men who do not smoke, do drink alcohol and have a waist circumference of 100 cm. The authors state that

   > Daily mean temperatures range between an average minimum of 20°C in the rainy season and an average maximum of 40°C in the dry season. In the dry season, early mornings are usually cool and the afternoons commonly hot with daily maximum temperatures going as high as 45°C (p. 17).

   Use this information to guide your choice of temperature values for your plot.
4. Compute a 95% confidence interval for the regression parameter for ambient temperature.
5. Interpret the relationship between ambient temperature and all the variables in the regression equation.
6. Predict the mean systolic blood pressure for 35 year-old Ghanaian men (who do not smoke, do drink alcohol and have a waist circumference of 100 cm) when the ambient temperature is 30°C.

2.15. An experiment was conducted [11] to determine how to maximize Mermaid meadowfoam flower production (Table 2.9; data set: `flowers`) for extraction as vegetable oil.

Table 2.9 The average number of flowers per meadowfoam plant (based on ten seedlings) exposed to various levels of lighting at two different times: at photoperiodic floral induction (PFI) or 24 days before PFI. These data are consistent with the results in [11] (Problem 2.15)

	Light intensity (in μmol m^{-2} s^{-1})					
Timing	150	300	450	600	750	900
At PFI	62.4 77.1	55.7 54.2	49.5 62.0	39.3 45.3	30.9 45.2	36.8 42.2
Before PFI	77.7 75.4	68.9 78.2	57.2 70.9	62.9 52.1	60.2 45.6	52.5 44.1

1. Plot the average number of flowers produced per plant against the light intensity, distinguishing the two timings. Comment.
2. Suppose a model with the systematic component `Flowers ~ Light + Timing` was needed to model the data. What would such a systematic component imply about the relationship between the variables?
3. Suppose a model with the systematic component `Flowers ~ Light * Timing` was needed to model the data. What would such a systematic component imply about the relationship between the variables?
4. Fit the two linear regression models with the systematic components specified above. Which is the preferred model?
5. The fitted model should use all prior weights as $w_i = 10$ for all i. What difference does it make if the prior weights are not defined (which R interprets as $w_i = 1$ for all i)?
6. Plot the systematic component of the preferred regression model on the data.
7. Interpret the model.

(This problem continues in Problem 3.13.)

2.16. A study of babies [1] hypothesized that babies would take longer to learn to crawl in colder months because the extra clothing restricts their movement. From 1988–1991, the babies' first crawling age and the average monthly temperature six months after birth (when "infants presumably enter the window of locomotor readiness"; p. 72) were recorded. The parents reported the birth month, and age when their baby first crept or crawled a distance of four feet in one minute. Data were collected at the University of Denver Infant Study Center on 208 boys and 206 girls, and summarized by the birth month (Table 2.10; data set: `crawl`).

1. Plot the data. Which assumptions, if any, appear to be violated?
2. Explain why a weighted regression model is appropriate for the data.
3. Fit a weighted linear regression model to the data, and interpret the regression coefficients.
4. Formally test the hypothesis proposed by the researchers.
5. Find a 90% confidence interval for the slope of the fitted line, and interpret.

Table 2.10 The crawling age and average monthly temperature six months after birth for 414 babies (Problem 2.16)

Birth month	Mean age when crawling started (weeks)	Sample size	Monthly average temperature six months after birth (°F)
January	29.84	32	66
February	30.52	36	73
March	29.70	23	72
April	31.84	26	63
May	28.58	27	52
June	31.44	29	39
July	33.64	21	33
August	32.82	45	30
September	33.83	38	33
October	33.35	44	37
November	33.38	49	48
December	32.32	44	57

6. Fit the unweighted regression model, then plot both regression lines on a plot of the data. Comment on the differences.
7. Compute the 95% confidence intervals for the fitted values from the weighted regression line, and also plot these.
8. Interpret the model.

(This problem continues in Problem 3.15.)

2.17. For a sample of 64 grazing Merino castrated male sheep (wethers) [5, 14, 17], the daily energy requirements and weight was recorded (Table 2.11; data set: `sheep`).

1. Fit a linear regression model to model the daily energy requirement from the weight.
2. Plot the data, plus the systematic component of the fitted model and the 95% confidence intervals about the fitted values.
3. Interpret the model.
4. Which assumptions, if any, appear to be violated? Explain.

(This problem continues in Problem 3.17.)

2.18. Children were asked to build towers out of cubical and cylindrical blocks as high as they could [8, 12], and the number of blocks used and the time taken were recorded (Table 2.12; data set: `blocks`). In this Problem, we focus on the time taken to build the towers. (The number of blocks used to build towers is studied in Problem 10.19.)

1. The data were originally examined in Problem 1.9 (p. 28). Using these plots, summarize the possible relationships of the explanatory variables with the time taken. Which assumptions, if any, appear to be violated?

Table 2.11 The energy requirements (in Mcal/day) and weight (in kg) for a sample of 64 Merino wethers (Problem 2.17)

Weight	Energy	Weight	Energy	Weight	Energy	Weight	Energy	Weight	Energy
22.1	1.31	25.1	1.46	25.1	1.00	25.7	1.20	25.9	1.36
26.2	1.27	27.0	1.21	30.0	1.23	30.2	1.01	30.2	1.12
33.2	1.25	33.2	1.32	33.2	1.47	33.9	1.03	33.8	1.46
34.3	1.14	34.9	1.00	42.6	1.81	43.7	1.73	44.9	1.93
49.0	1.78	49.2	2.53	51.8	1.87	51.8	1.92	52.5	1.65
52.6	1.70	53.3	2.66	23.9	1.37	25.1	1.39	26.7	1.26
27.6	1.39	28.4	1.27	28.9	1.74	29.3	1.54	29.7	1.44
31.0	1.47	31.0	1.50	31.8	1.60	32.0	1.67	32.1	1.80
32.6	1.75	33.1	1.82	34.1	1.36	34.2	1.59	44.4	2.33
44.6	2.25	52.1	2.67	52.4	2.28	52.7	3.15	53.1	2.73
52.6	3.73	46.7	2.21	37.1	2.11	31.8	1.39	36.1	1.79
28.6	2.13	29.2	1.80	26.2	1.05	45.9	2.36	36.8	2.31
34.4	1.85	34.4	1.63	26.4	1.27	27.5	0.94		

Table 2.12 The time taken (in s), and the number of blocks used, to build towers out of two shapes of blocks in two trials one month apart. The children's ages are given in decimal years (converted from years and months). The results for the first five children are shown (Prob. 2.18)

		Trial 1				Trial 2			
		Cubes		Cylinders		Cubes		Cylinders	
Child	Age	Number	Time	Number	Time	Number	Time	Number	Time
A	4.67	11	30.0	6	30.0	10	35.0	8	125.0
B	5.00	9	19.0	4	6.0	10	28.0	5	14.4
C	4.42	8	18.6	5	14.2	7	18.0	5	24.0
D	4.33	9	23.0	4	8.2	11	34.8	6	14.4
E	4.33	10	29.0	6	14.0	6	16.2	5	15.0
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮

2. Suppose a model with the systematic component `Time ~ Age * Shape` was needed to model the data. What would such a systematic component imply about the relationship between the variables?
3. Suppose a model with the systematic component `Time ~ Age * Trial` was needed to model the data. What would such a systematic component imply about the relationship between the variables?
4. Suppose a model with the systematic component `Time ~ (Age + Shape) * Trial` was needed to model the data. What would such a systematic component imply about the relationship between the variables?
5. One hypothesis of interest is whether the time taken to build the tower differs between cubical and cylindrical shaped blocks. Test this hypothesis by fitting a linear regression model.

Table 2.13 The sharpener data; the first five cases are shown (Problem 2.13)

y	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}
9.87	0.64	0.22	0.83	0.41	0.64	0.88	0.22	0.41	0.38	0.02
8.86	0.16	0.55	0.71	0.25	0.61	0.68	0.93	0.95	0.15	0.00
7.82	0.14	0.00	0.97	0.54	0.25	0.46	0.71	0.90	0.13	0.18
10.77	0.53	0.45	0.80	0.54	0.84	0.39	0.16	0.06	0.72	0.90
9.53	0.14	0.52	0.13	0.91	0.15	0.52	0.09	0.26	0.12	0.51
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮

6. Another hypothesis of interest is that older children take less time to build the towers than younger children, but the difference would depend on the type of block. Test this hypothesis.
7. Find a suitable linear regression model for the time taken to build the towers. Do you think this model is suitable? Explain.
8. Interpret your final model.

(This problem continues in Problem 3.16.)

2.19. The data in Table 2.13 (data set: `sharpener`) come from a study to make a point.

1. Using the forward regression procedure (Sect. 2.12.2, p. 73), find a suitable linear regression (without interactions) model for predicting y from the explanatory variables, based on using the AIC.
2. Using the backward elimination procedure, find a model (without interactions) for predicting y from the explanatory variables based on using the AIC.
3. Using the step-wise regression procedure, find a model (without interactions) for predicting y from the explanatory variables, based on using the AIC.
4. From the results of the above approaches, deduce a model (without interactions) for the data.
5. Repeat the three procedures, but use the BIC to select a model.
6. After reading the R help for the `sharpener` data (using `?sharpener`), comment on the use of automatic methods for fitting regression models.

References

[1] Benson, J.: Season of birth and onset of locomotion: Theoretical and methodological implications. Infant Behavior and Development **16**(1), 69–81 (1993)

[2] Bland, J.M., Peacock, J.L., Anderson, H.R., Brooke, O.G.: The adjustment of birthweight for very early gestational ages: Two related problems in statistical analysis. Applied Statistics **39**(2), 229–239 (1990)
[3] Davison, A.C., Hinkley, D.V.: Bootstrap Methods and their Application. Cambridge University Press (1997)
[4] Green, P.J.: Iteratively reweighted least squares for maximum likelihood estimation, and some robust and resistant alternatives (with discussion). Journal of the Royal Statistical Society, Series B **46**(2), 149–192 (1984)
[5] Hand, D.J., Daly, F., Lunn, A.D., McConway, K.Y., Ostrowski, E.: A Handbook of Small Data Sets. Chapman and Hall, London (1996)
[6] Harrell Jr, F.: Regression Modeling Strategies: With Applications to Linear Models, Logistic Models, and Survival Analysis. Springer (2001)
[7] Hasofer, A.M., Bruck, D.: Statistical analysis of response to fire cues. Fire Safety Journal **39**, 663–688 (2004)
[8] Johnson, B., Courtney, D.M.: Tower building. Child Development **2**(2), 161–162 (1931)
[9] Kunutsor, S.K., Powles, J.W.: The effect of ambient temperature on blood pressure in a rural West African adult population: A cross-sectional study. Cardiovascular Journal of Africa **21**(1), 17–20 (2010)
[10] Schwarz, G.E.: Estimating the dimension of a model. The Annals of Statistics **6**(2), 461–464 (1978)
[11] Seddigh, M., Joliff, G.D.: Light intensity effects on meadowfoam growth and flowering. Crop Science **34**, 497–503 (1994)
[12] Singer, J.D., Willett, J.B.: Improving the teaching of applied statistics: Putting the data back into data analysis. The American Statistician **44**(3), 223–230 (1990)
[13] Tibshirani, R.: Regression shrinkage and selection via the lasso. Journal of the Royal Statistical Society. Series B (Methodological) **58**(1), 267–288 (1996)
[14] Wallach, D., Goffinet, B.: Mean square error of prediction in models for studying ecological systems and agronomic systems. Biometrics **43**(3), 561–573 (1987)
[15] Woodward, M.: Epidemiology: study design and data analysis, second edn. Chapman & Hall/CRC, Boca Raton, FL (2004)
[16] Woodward, M., Walker, A.R.P.: Sugar consumption and dental caries: Evidence from 90 countries. British Dental Journal **176**, 297–302 (1994)
[17] Young, B.A., Corbett, J.L.: Maintenance energy requirement of grazing sheep in relation to herbage availability. Australian Journal of Agricultural Research **23**(1), 57–76 (1972)

Chapter 3
Linear Regression Models: Diagnostics and Model-Building

Normality is a myth; there never was, and never will be, a normal distribution. This is an over-statement from the practical point of view, but it represents a safer initial mental attitude than any in fashion during the past two decades.
Geary [13, p. 241]

3.1 Introduction and Overview

As the previous two chapters have demonstrated, the process of building a linear regression model, or any regression model, is aided by exploratory plots of the data, by reflecting on the experimental design, and by considering the scientific relationships between the variables. This process should ensure that the model is broadly appropriate for the data. Once a candidate model has been fitted to the data, however, there are specialist measures and plots that can examine the model assumptions and diagnose possible problems in greater detail. This chapter describes these tools for detecting and highlighting violations of assumptions in linear regression models. The chapter goes on to discuss some possible courses of action that might alleviate the identified problems. The process of examining and identifying possible violations of model assumptions is called *diagnostic analysis*. The assumptions of linear regression models are first reviewed (Sect. 3.2), then residuals, the main tools of diagnostic analysis, are defined (Sect. 3.3). We follow with a discussion of the leverage, a measure of the location of an observation relative to the average observation location (Sect. 3.4). The various diagnostic tools for checking the model assumptions are then introduced (Sect. 3.5) followed by techniques for identifying unusual and influential observations (Sect. 3.6). The terminology of residuals is summarized in Sect. 3.7. Techniques for fixing any weaknesses in the models are summarised in Sect. 3.8, and explained in greater detail in Sects. 3.9 to 3.13. Finally, the issue of collinearity is discussed (Sect. 3.14).

P. K. Dunn, G. K. Smyth, *Generalized Linear Models with Examples in R*,
Springer Texts in Statistics, https://doi.org/10.1007/978-1-4419-0118-7_3

3.2 Assumptions from a Practical Point of View

3.2.1 Types of Assumptions

The general form of a linear regression model is given by (2.1) or, assuming normality, by (2.25). The assumptions of the model can be summarized as:

- Lack of outliers: All responses were generated from the same process, so that the same regression model is appropriate for all the observations.
- Linearity: The linear predictor captures the true relationship between μ_i and the explanatory variables, and all important explanatory variables are included.
- Constant variance: The responses y_i have *constant* variance, apart from known weights w_i.
- Independence: The responses y_i are statistically *independent* of each other.
- Distribution: The responses y_i are normally distributed around μ_i.

Failure of the assumptions may lead to inappropriate and incorrect results from hypothesis tests and confidence intervals, potentially leading to incorrect parameter estimation and incorrect interpretations.

The first two assumptions are obviously the most basic. If the linear model doesn't correctly model the systematic trend in the responses, then it will be useless for prediction and interpretation purposes. The other three assumptions affect the precision with which the regression coefficients are estimated, as well as the accuracy of standard errors and the validity of statistical tests.

3.2.2 The Linear Predictor

This chapter generally assumes that all the important explanatory variables are at least available. Methods will be presented for detecting observations that are errors or which do not fit the pattern of the remaining observations. This chapter will also explore ways to improve linearity by changing the scale of the covariate or response, or to accommodate more complex relationships by building new covariates from the existing ones.

3.2.3 Constant Variance

Deviations from constant variance are of two major types. Firstly, it is possible that one group of observations is intrinsically more heterogeneous than another. For example, diseased patients often show more variability than

control patients without the disease, or disease tumour tissue may show more variability than normal tissue. However, by far the most commonly-arising and important scenario leading to non-constant variance is when the response is measured on a scale for which the precision of the observation depends on the size of the observation. Measures of positive physical quantities frequently show more absolute variability when the quantity is large than when the quantity is small. For example, the mass of a heavy object might be measured to a constant relative error over a wide range of values, so that the standard deviation of each measurement is proportional to its mean. The number of people in a group might be counted accurately when there are only a few individuals, but will have to be estimated more approximately when the crowd is large. This sort of mean–variance relationship will be explored extensively in later chapters of this book; in fact it is a major theme of the book. This chapter will examine ways to alleviate any mean–variance relationship by transforming the response.

3.2.4 Independence

Ensuring that the responses y_i are statistically independent is one of the aims of the experimental design or data collection process. Dependence between responses can arise because the responses share a common source or because the data are collected in a hierarchical manner. Examples include:

- Repeated measures. Multiple treatments are applied to same experimental subjects.
- Blocking. A group of observations are drawn close in space or in time so as to minimize their variability. For example, multiple plants are grown in the same plot of ground, or a complex experiment is conducted in a number of separate stages or batches.
- Multilevel sampling. For example, a cost-effective way to sample school children is to take a random sample of school districts; within selected districts, take a random sample of schools; within selected schools, take a random sample of pupils.
- Time series. The responses arise from observing the same process over time. For example, the sales figures of a particular product.

In the simplest cases, the dependence between multiple observations in a block can be accounted for by including the blocking variable as an explanatory factor in the linear model. For example, when multiple treatments are given to the same set of subjects, the subject IDs may be treated as the levels of an explanatory factor. In other cases, dependence can by detected by suitable plots. In more complex cases, when there are multiple levels of variability, *random effects* models may be required to fully represent the data collection process [29]. However, these are beyond the scope of this textbook.

3.2.5 Normality

The assumption of normality underlies the use of F- and t-tests (Sect. 2.8). When the number of observations is large, and there are no serious outliers, t- and F-tests tend to behave well even when the residuals are not normally distributed. This means the assumption of normality is most critical for small sample sizes. Unfortunately, small sample size is exactly the situation when assessing normality is most difficult.

3.2.6 Measurement Scales

A broad consideration that affects many of the assumptions is that of the measurement scales used for the response and the explanatory variables, and especially the range of feasible values that the variables can take on. For example, if the response y_i can take only positive values, then it is clearly mathematically impossible for it to follow a normal distribution. Similarly, a positive response variable may cause problems if the linear predictor can take negative values. A strictly positive random variable is also unlikely to have a constant variance if values near zero are possible. The same sort of considerations apply doubly when the response represents a proportion and is therefore bounded at both zero and one. In this case, constant variance is unlikely if values close to zero or one are possible. In general, linear models for positive or constrained response variables may be fine over a limited range of values, but are likely to be suspect when the values range over several orders of magnitude are possible.

The units of measurement can also guide the process of model building. For the lung capacity data of Example 1.1, the response variable FEV is in units of volume, whereas height is in units of length. If individuals were of the same general shape, volume would tend to be proportional to height cubed.

3.2.7 Approximations and Consequences

As always, a statistical model is a mathematical ideal, and will never be an *exact* representation of any real data set or real physical process. When evaluating the assumptions, we are guided by the likely sensitivity of the conclusions to deviations from the model assumptions. For example, the response variable y may not exactly be a linear function of a covariate x, but a linear approximation may be adequate in a context where are limited range of x values are likely to appear. The assumptions are ordered in the above list from those that effect the first moment of the responses (the mean), to the second moment (variances) to third and higher moments (complete

distribution of y_i). Generally speaking, assumptions that affect the lower moments of y_i are the most basic, and assumptions relating to higher moments are progressively of lower priority.

3.3 Residuals for Normal Linear Regression Models

The *raw residuals* are simply

$$r_i = y_i - \hat{\mu}_i.$$

Recall that RSS $= \sum_{i=1}^{n} w_i r_i^2$.

Since $\hat{\mu}$ is estimated from the data, $\hat{\mu}$ is a random variable. This means that $\text{var}[y_i - \hat{\mu}_i]$ is not the same as $\text{var}[y_i - \mu_i] = \text{var}[y_i] = \sigma^2/w_i$. Instead, as shown in Sect. 3.4.2,

$$\text{var}[r_i] = \sigma^2(1 - h_i)/w_i, \tag{3.1}$$

where h_i is the *leverage* which y_i has in estimating its own fitted value $\hat{\mu}_i$ (Sect. 3.4).

Equation (3.1) means that the raw residuals r_i do not have constant variance, and so may be difficult to interpret in diagnostic plots. A modified residual that does have constant variance can be defined by

$$r_i^* = \frac{\sqrt{w_i}(y_i - \hat{\mu}_i)}{\sqrt{1 - h_i}},$$

with $\text{var}[r_i^*] = \sigma^2$. The modified residual has the interesting interpretation that its square $(r_i^*)^2$ is the reduction in the RSS that results when Observation i is omitted from the data (Problem 3.1).

After estimating σ^2 by s^2, the *standardized residuals* are defined by

$$r_i' = \frac{r_i^*}{s} = \frac{\sqrt{w_i}(y_i - \hat{\mu}_i)}{s\sqrt{1 - h_i}}. \tag{3.2}$$

The standardized residuals estimate the standardized distance between the data y_i about the fitted values $\hat{\mu}_i$. The standardized residuals are approximately standard normal in distribution. More exactly, r_i' follows a t-distribution on $n - p'$ degrees of freedom.

The raw residuals are computed from any fitted linear regression model `fit` in R using `resid(fit)`, and standardized residuals using `rstandard(fit)`.

Example 3.1. In Chaps. 1 and 2, the lung capacity were used (Example 1.5; data set `lungcap`), and log(FEV) was found to be linearly associated with height. For this reason, models in those chapters were considered using the response variable $y = \log(\text{FEV})$.

In this chapter, for the purpose of demonstrating diagnostics for linear regression models, we begin by considering a model for $y = \text{FEV}$ (not $y = \log(\text{FEV})$) to show how the diagnostics reveal the inadequacies of this model. We decide to use a systematic component involving `Ht`, `Gender` and `Smoke`. (preferring `Ht` over `Age` as `Ht` is a physical trait).

```
> library(GLMsData); data(lungcap)
> lungcap$Smoke <- factor(lungcap$Smoke,
                          levels=c(0, 1),
                          labels=c("Non-smoker","Smoker"))
>    ### POOR MODEL!
> LC.lm <- lm( FEV ~ Ht + Gender + Smoke, data=lungcap)
```

To compute the residuals for this model in R, use:

```
> resid.raw <- resid( LC.lm )      # The raw residuals
> resid.std <- rstandard( LC.lm )  # The standardized residuals
> c( Raw=var(resid.raw), Standardized=var(resid.std) )
         Raw Standardized
   0.1812849    1.0027232
```

The standardized residuals have variance close to one, as expected. □

3.4 The Leverages for Linear Regression Models

3.4.1 Leverage and Extreme Covariate Values

To explain the leverages clearly, we need first to standardize the responses so they have constant variance. Write the standardized responses as $z_i = \sqrt{w_i} y_i$. Then $\text{E}[z_i] = \nu_i = \sqrt{w_i}\mu_i$ and $\text{var}[z_i] = \sigma^2$. The fitted values $\hat{\nu}_i = \sqrt{w_i}\hat{\mu}_i$ can be considered to be a linear function of the responses z_i. The *hat-values* are defined as those values h_{ij} that relate the responses z_i to the fitted values $\hat{\nu}_i$, satisfying

$$\hat{\nu}_i = \sum_{j=1}^{n} h_{ij} z_j.$$

The hat-value h_{ij} is the coefficient applied to the standardized observation z_j to obtain the standardized fitted value $\hat{\nu}_i$. When the weights w_i are all one,

$$\hat{\mu}_i = h_{i1} y_1 + h_{i2} y_2 + \cdots + h_{in} y_n = \sum_{j=1}^{n} h_{ij} y_j.$$

This shows that the hat-value h_{ij} is the coefficient applied to y_j to obtain $\hat{\mu}_i$. Colloquially, the hat-values put the "hat" on μ_i.

Of particular interest are the diagonal hat-values h_{ii}, which we will call *leverages*, written $h_i = h_{ii}$. The leverages h_i measure the weight that response y_i (or z_i) receives in computing its own fitted value: $h_i = \sum_{j=1}^{n} h_{ij}^2$. The leverages h_i depend on the values of the explanatory variables and weights, not on the values of the responses. The n leverages satisfy $1/n \leq h_i \leq 1$ (Problem 3.3), and have total sum equal to p'. This shows that the mean of the hat-values is $\bar{h} = p'/n$.

In the case of simple linear regression without weights (Problem 3.3),

$$h_i = \frac{1}{n} + \frac{(x_i - \bar{x})^2}{\text{SS}_x},$$

showing that leverage increases quadratically as x_i is further from the mean $\bar{x}$. It is a good analogy to think of $\bar{x}$ as defining the fulcrum of a lever through which each observation contributes to the regression slope, with $x_i - \bar{x}$ the distance of the point from the fulcrum.

For an unweighted linear regression with a factor as the single explanatory variable, the leverages are $h_i = 1/n_j$, where n_j is the total number of observations in the same group as observation i.

In general, a small leverage for Observation i indicates that many observations, not just one, are contributing to the estimation of the fitted value. In the extreme case that $h_i = 1$, the ith fitted value will be entirely determined by the ith observation, so that $\hat{\mu}_i = y_i$. In practice, this means that large values of h_i (perhaps two or three times the mean value of the h_i) identify unusual combinations of the explanatory variables.

The leverages in R for a linear regression model called `fit` are computed using the command `hatvalues(fit)`.

Example 3.2. For the poor model fitted in Example 3.1 to the `lungcap` data, the leverages are found using `hatvalues()`:

```
> h <- hatvalues( LC.lm )            # Produce the leverages
> sort( h, decreasing=TRUE) [1:2]    # The largest two leverages
       629        631
0.02207842 0.02034224
```

The two largest leverages are for Observations 629 and 631. Compare these leverages to the mean value of the leverages:

```
> mean(h); length(coef(LC.lm))/length(lungcap$FEV)  # Mean leverage
[1] 0.006116208
[1] 0.006116208
> sort( h, decreasing=TRUE) [1:2] / mean(h)
     629      631
3.609822 3.325956
```

Observations 629 and 631 are many times greater than the mean value of the leverages. Note that both of these large leverages correspond to male smokers:

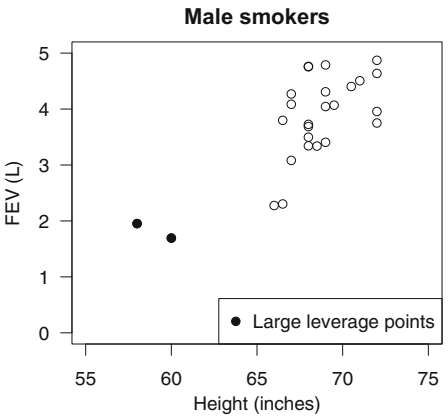

Fig. 3.1 FEV plotted against height for males smokers. The leverages h_i are shown for two observations as filled dots (Example 3.2)

```
> sort.h <- sort( h, decreasing=TRUE, index.return=TRUE)
> large.h <- sort.h$ix[1:2]   # Provide the index where these occur
> lungcap[ large.h, ]
    Age   FEV Ht Gender  Smoke
629   9 1.953 58      M Smoker
631  11 1.694 60      M Smoker
```

Consider the plot of `FEV` against `Ht` for just male smokers then:

```
> plot( FEV ~ Ht, main="Male smokers",
    data=subset( lungcap, Gender=="M" & Smoke=="Smoker"),
    # Only male smokers las=1, xlim=c(55, 75), ylim=c(0, 5),
    xlab="Height (inches)", ylab="FEV (L)" )
> points( FEV[large.h] ~ Ht[large.h], data=lungcap, pch=19) # Large vals
> legend("bottomright", pch=19, legend=c("Large leverage points") )
```

The two largest leverages correspond to the two unusual observations in the bottom left corner of the plot (Fig. 3.1). □

* 3.4.2 The Leverages Using Matrix Algebra

For simplicity, consider first the case of unweighted regression for which all the $w_i = 1$; in other words $\mathrm{W} = \mathrm{I}_n$. Recall that the least squares estimates of the regression coefficients are given by $\hat{\boldsymbol{\beta}} = (\mathrm{X}^T\mathrm{X})^{-1}\mathrm{X}^T\mathbf{y}$ when $\mathrm{W} = \mathrm{I}_n$. Therefore the fitted values are given by $\hat{\boldsymbol{\mu}} = \mathrm{X}\boldsymbol{\beta} = \mathrm{H}\mathbf{y}$ with

$$\mathrm{H} = \mathrm{X}(\mathrm{X}^T\mathrm{X})^{-1}\mathrm{X}^T.$$

We say that H is the *hat matrix*, because it puts the "hat" on $\mathbf{y}$. The leverages h_i are the diagonal elements of H.

Write $\mathbf{r}$ for the vector of raw residuals from the regression

$$\mathbf{r} = \mathbf{y} - \hat{\boldsymbol{\mu}} = (\mathrm{I}_n - \mathrm{H})\mathbf{y}.$$

It is not hard to show that the covariance matrix of this residual vector is given by

$$\mathrm{var}[\mathbf{r}] = (\mathrm{I}_n - \mathrm{H})\sigma^2.$$

In particular, it follows that $\mathrm{var}[r_i] = (1 - h_i)\sigma^2$.

To incorporate general weights $\mathrm{W} = \mathrm{diag}(w_i)$, it is easiest to transform to an unweighted regression. Write $\mathbf{z} = \mathrm{W}^{1/2}\mathbf{y}$, and define $\mathrm{X}_w = \mathrm{W}^{1/2}\mathrm{X}$. Then $\mathrm{E}[\mathbf{z}] = \boldsymbol{\nu} = \mathrm{X}_w\boldsymbol{\beta}$ and $\mathrm{var}[\mathbf{z}] = \sigma^2\mathrm{I}_n$. The hat matrix for this linear model is

$$\mathrm{H} = \mathrm{X}_w(\mathrm{X}_w^T\mathrm{X}_w)^{-1}\mathrm{X}_w^T = \mathrm{W}^{1/2}\mathrm{X}(\mathrm{X}^T\mathrm{W}\mathrm{X})^{-1}\mathrm{X}^T\mathrm{W}^{1/2}. \tag{3.3}$$

For the transformed regression, $\mathrm{var}[\mathbf{z} - \hat{\boldsymbol{\nu}}] = \sigma^2(\mathrm{I}_n - \mathrm{H}_w)$. The residuals for the weighted regression are $\mathbf{r} = W^{-1/2}(\mathbf{z} - \hat{\boldsymbol{\nu}})$. It follows (Problem 3.2) that the covariance matrix of the residuals for the weighted regression is

$$\mathrm{var}[\mathbf{r}] = \mathrm{var}[\mathbf{y} - \hat{\boldsymbol{\mu}}] = \sigma^2\mathrm{W}^{-1/2}(\mathrm{I}_n - \mathrm{H})^T\mathrm{W}^{-1/2}.$$

In R, the leverages may be computed directly from the model matrix `X` using `hatvalues(X)`.

3.5 Residual Plots

3.5.1 Plot Residuals Against x_j: Linearity

Basic exploratory data analysis usually includes a plot of the response variable against each explanatory variable. Such a plot is complicated by the fact that multiple explanatory variables may have competing effects on the response. Furthermore, some deviations from linearity may be hard to detect. A plot of residuals against a covariate x_j can more easily detect deviations from linearity, because the linear effects of all the explanatory variables have been removed. If the model fits well, the residuals should show no pattern, just constant variability around zero for all values of x_j. Any systematic trend in the residuals, such as a quadratic curve, suggests a need to transform x_j or to include extra terms in the linear model.

Using `scatter.smooth()` in place of `plot()` in R adds a smoothing curve to the plots, which may make trends easier to see.

Example 3.3. Consider again the lung capacity data (Example 1.1; data set: `lungcap`), and model `LC.lm` fitted to the data in Example 3.1. Assume the

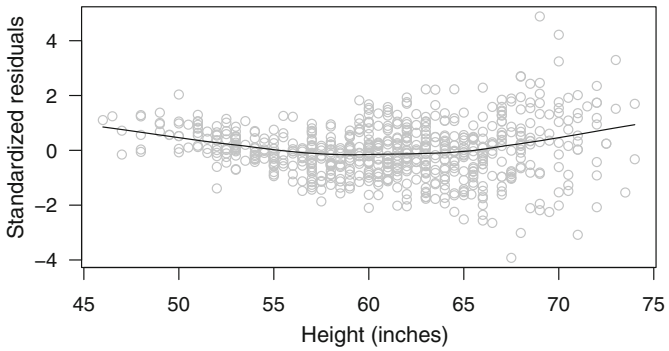

Fig. 3.2 Residuals plotted against the covariate `Ht` for the model `LC.lm` fitted to the lung capacity data (Example 3.3)

data were collected so that the responses are independent. Then, plots of residuals against the covariate can be created:

```
>    # Plot std residuals against Ht
> scatter.smooth( rstandard( LC.lm ) ~ lungcap$Ht, col="grey",
    las=1, ylab="Standardized residuals", xlab="Height (inches)")
```

The plots of residuals against height (Fig. 3.2) are slightly non-linear, and have increasing variance. This suggests that the model is poor. Of course, linearity is not relevant for gender or smoking status, as these variables take only two levels. □

3.5.2 Partial Residual Plots

Partial residuals plots are similar to plotting residuals against x_j, but with the linear trend with respect to x_j added back into the plot. To examine the relationship between the response y and a particular covariate x_j define the *partial residuals* as

$$u_j = r + \hat{\beta}_j x_j. \tag{3.4}$$

The *partial residual plot* is a plot of u_j against x_j. (Here u_j and x_j are variables with n values, and the subscript i has been suppressed.) The partial residual plot shows much the same information as the ordinary residual plot versus x_j but, by showing the linear trend on the same plot, the partial residual plots allows the analyst to judge the relative importance of any linearity relative to the magnitude of the linear trend. When plotting residuals versus x_j, the focus is on existence of any nonlinear trends. With the partial residual plot, the focus is on the relative importance of any nonlinearity in the context of the linear trend.

A partial residual plot can be seen as an attempt to achieve the same effect and simplicity of interpretation as the plot of y against x in simple linear regression, but in the context of multiple regression. With multiple predictors, plots of y against each explanatory variable are generally difficult to interpret because of the competing effects of the multiple variables. The partial residual plot shows the contribution of x_j after adjusting for the other variables currently in the model. The slope of a least-squares line fitted to the partial residual plot gives the coefficient for that explanatory variable in the full regression model. However, the variability of points around the line in the partial residual plot may suggest to the eye that σ^2 is somewhat smaller than it actually is, because the residuals being plotted are from the full regression model with $n - p'$ residual degrees of freedom, rather than from a simple linear regression with $n - 2$ degrees of freedom.

Example 3.4. Consider the `lungcap` data again. Figure 1.1 (p. 6) shows the relationships between FEV and each explanatory variable without adjusting for the other explanatory variables. The partial residuals can be computed using `resid()`:

```
> partial.resid <- resid( LC.lm, type="partial")
> head(partial.resid)
          Ht     Gender       Smoke
1 -1.4958086  0.4026274  0.46481270
2 -1.7288086 -0.0897584 -0.02757306
3 -1.4658086  0.1732416  0.23542694
4 -1.1788086  0.4602416  0.52242694
5 -0.9908086  0.5185487  0.58073406
6 -1.1498086  0.3595487  0.42173406
```

The easiest way to produce the partial residual plots (Fig. 3.3) is to use `termplot()`. We do so here to produce the partial residuals plot for `Ht` only (Fig. 3.3):

```
> termplot( LC.lm, partial.resid=TRUE, terms="Ht", las=1)
```

`termplot()` also shows the ideal linear relationship in the plots. The partial residual plot for `Ht` shows non-linearity, again suggesting the use of $\mu = \mathrm{E}[\log(\mathrm{FEV})]$ as the response variable.

The relationship between `FEV` and `Ht` appears quite strong after adjusting for the other explanatory variables. Note that the slope of the simple regression line is equal to the coefficient in the full model. For example, compare the regression coefficients for `Ht`:

```
> coef( summary(LC.lm) )
               Estimate  Std. Error     t value      Pr(>|t|)
(Intercept) -5.36207814 0.186552603 -28.7429822 7.069632e-118
Ht           0.12969288 0.003105995  41.7556591 3.739216e-186
GenderM      0.12764341 0.034093423   3.7439305  1.972214e-04
SmokeSmoker  0.03413801 0.058581034   0.5827485  5.602647e-01
```

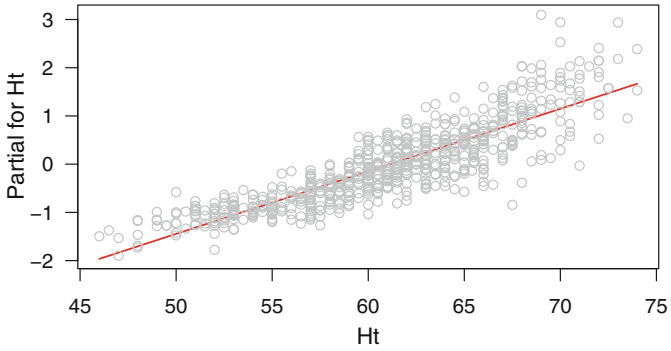

Fig. 3.3 Partial residual plot `Ht` in the model `LC.lm` fitted to the lung capacity data (Example 3.4)

```
> lm.Ht <- lm( partial.resid[, 1]~lungcap$Ht)
> coef( summary(lm.Ht) )
              Estimate  Std. Error   t value      Pr(>|t|)
(Intercept) -7.9298868 0.179532816 -44.16957 3.629602e-198
lungcap$Ht   0.1296929 0.002923577  44.36102 4.369629e-199
```

The coefficients for `Ht` are exactly the same. The full regression gives larger standard errors than the simple linear regression however, because the latter over-estimates the residual degrees of freedom. □

3.5.3 Plot Residuals Against $\hat{\mu}$: Constant Variance

Plotting the residuals against $\hat{\mu}$ is primarily used to check for constant variance (Fig. 3.4). An increasing or decreasing trend in the variability of the residuals about the zero line suggests the need to transform or change the scale of the response variable to achieve constant variance. For example, if the response variable is a positive quantity, and the plot of residuals versus $\hat{\mu}$ shows an increasing spread of the residuals for larger fitted values, this would suggest a need to transform the response variable to compress the larger values, by taking logarithms or similar. Standardized residuals r' (rather than the raw residuals r) are preferred in these plots, as standardized residuals have approximately constant variance if the model fits well.

Example 3.5. Returning to the lung capacity data, Fig. 3.5 shows that the plot of residuals against fitted values has a variance that is not constant, but is increasing as the mean increases. In other words, there appears to be an increasing mean–variance relationship. The plot also shows non-linearity, again suggesting that the model can be improved:

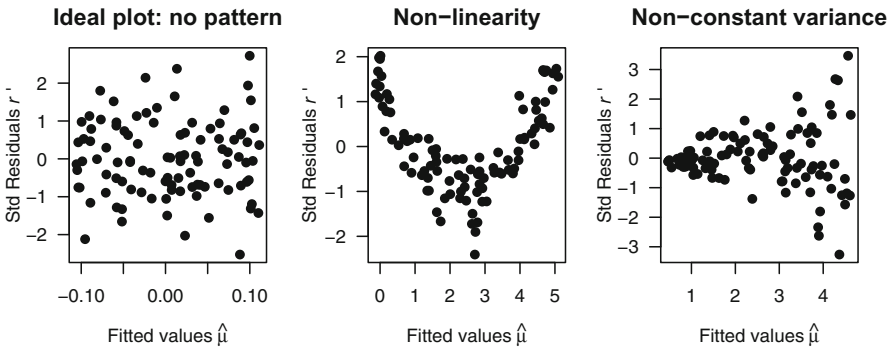

Fig. 3.4 Some example plots of the standardized residuals r' plotted against the fitted values $\hat{\mu}$. The effects are exaggerated from what is usually seen in practice (Sect. 3.5.1)

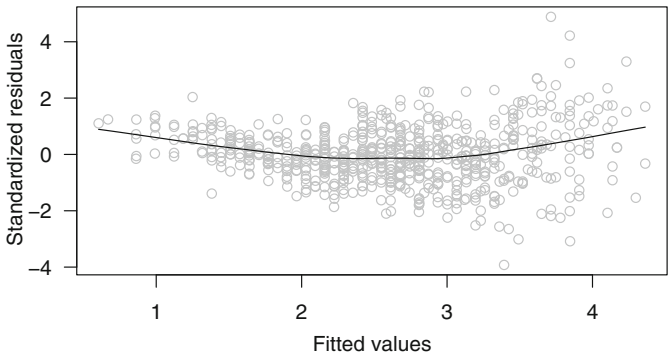

Fig. 3.5 Standardized residual plotted against the fitted values for the model `LC.lm` fitted to the lung capacity data (Example 3.5)

```
>    # Plot std residuals against the fitted values
> scatter.smooth( rstandard( LC.lm ) ~ fitted( LC.lm ), col="grey",
     las=1, ylab="Standardized residuals", xlab="Fitted values")
```

□

3.5.4 Q–Q Plots and Normality

The assumption of normality can be checked using a normal *quantile–quantile* plot, or normal *Q–Q plot*, of the residuals. A Q–Q plot, in general, graphs the quantiles of the data against the quantiles of given distribution; a normal Q–Q plot graphs the quantiles of the data against the quantiles of a standard normal distribution. For example, the value below which 30% of the data lie is plotted against the value below which 30% of a standard normal distribution lies. If the residuals have a normal distribution, the points will lie on a straight

line in the Q–Q plot. For this reason, a straight line is often added to the Q–Q plot to assist in assessing normality. For small sample sizes, Q–Q plots may be hard to assess (Problem 3.5).

Non-normality may appear as positive skewness (which is quite common); negative skewness; as having too many observations in the tails of the distribution; or as having too few observations in the tails of the distribution (Fig. 3.6). Q–Q plots are also a convenient way to check for the presence of large residuals (Sect. 3.6.2). Since standardized residuals r' are more normally distributed than raw residuals, Q–Q plots are more appropriate and outliers are easier to identify using standardized residuals.

In R, Q–Q plots of residuals can be produced from a fitted model `fit` using `qqnorm()`, using either `resid(fit)` or `rstandard(fit)` as the input. A reference line for assessing normality of the points is added by following the `qqnorm()` command with the corresponding `qqline()` command, as shown in the following example.

Example 3.6. Consider the `lungcap` data again (Example 1.1), and model `LC.lm` fitted to the data. The Q–Q plot (Fig. 3.7) suggests that the normality assumption is suspect:

```
>    # Q-Q probability plot
> qqnorm( rstandard( LC.lm ), las=1, pch=19)
> qqline( rstandard( LC.lm ) )      # Add reference line
```

The distribution of residuals appears to have heavier tails than the normal distribution in both directions, because the residuals curve above the line on the right and below the line on the left. The plot also shows a number of large residuals, both positive and negative, suggesting the model can be improved. □

3.5.5 Lag Plots and Dependence over Time

Dependence is not always easy to detect, if not already obvious from the data collection process. When data are collected over time, dependence between successive response can be detected by plotting each residual against the previous residual in time, often called the *lagged* residual. If the responses are independent, the plots should show no pattern under (Fig. 3.8, left panel).

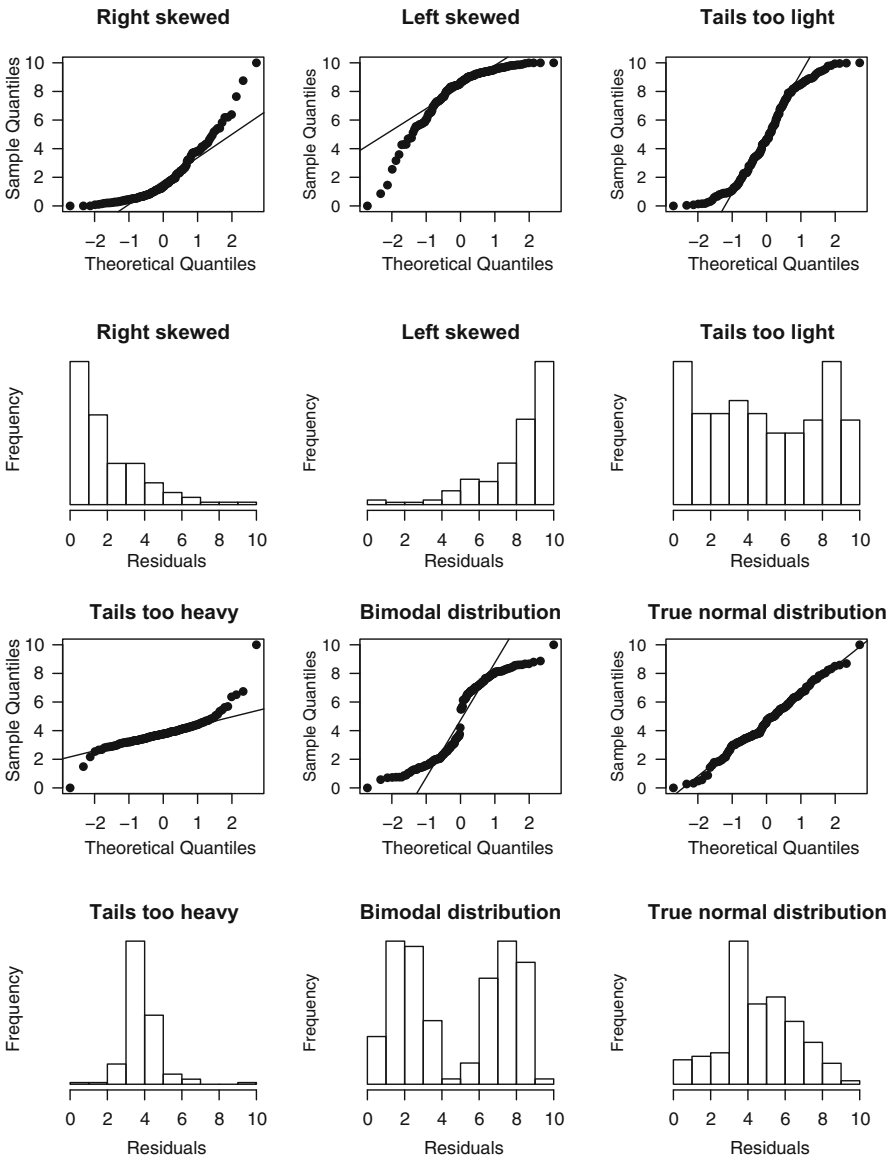

Fig. 3.6 Typical Q–Q plots of standardized residuals. In all cases, the sample size is 150. The solid line is added as a reference to aid is assessing linearity of the points (Sect. 3.5.4)

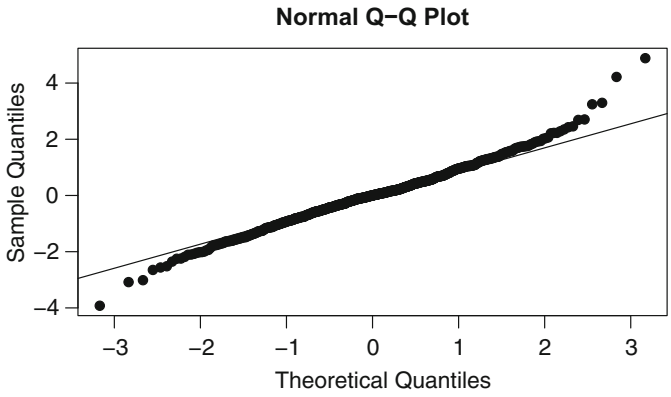

Fig. 3.7 The Q–Q plot for model `LC.lm` fitted to the lung capacity data (Example 3.6)

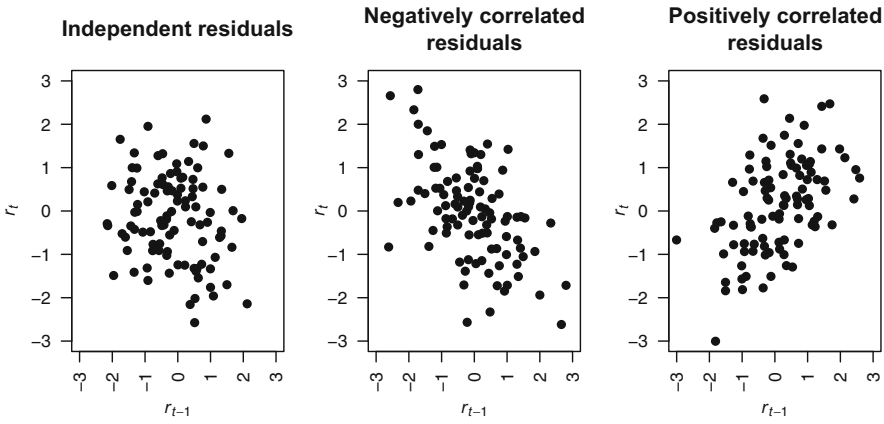

Fig. 3.8 Some example plots of the residuals at time t, denoted r_t, plotted against the previous residual in time r_{t-1} (Sect. 3.5.5)

3.6 Outliers and Influential Observations

3.6.1 Introduction

The previous section presented tools for assessing overall model assumptions. This section discusses methods for detecting problems with individual observations. The two issues may be related: an incorrect model specification may indicate problems with a particular observation. Consequently, the methods in Sect. 3.5 should be used in conjunction with the methods in this section.

3.6.2 Outliers and Studentized Residuals

Outliers are observations inconsistent with the rest of the data set. Inconsistent observations are located by identifying the corresponding residual as unusually large (positive or negative). This may be done by using Q–Q plots or other plots already produced for assessing the model assumptions. As a guideline, potential outliers might be flagged as observations with standardized residual r' greater than, say, 2.5 in absolute value. This is naturally only a guideline to guide further investigation, as approximately 1.2% of observations will have absolute standardized residuals exceeding 2.5 just by chance even when there are no outliers and all the model assumptions are correct.

Standardized residuals are computed using s^2, which is computed from the entire data set. An observation with a large raw residual is actually used to compute s^2 and perhaps inflating its value, in turn making the unusual observation hard to detect. This suggests *omitting* Observation i from the calculation of s^2 when computing the residual for Observation i. These residuals are called *Studentized residuals*.

To find the Studentized residual r_i'', first fit a linear regression model to all the data except case i. Then compute the estimate of the variance $s^2_{(i)}$ from this model based on the remaining $n-1$ observations, the subscript (i) indicating that Observation i has been omitted in computing the estimate. Then, the Studentized residuals are

$$r_i'' = \frac{\sqrt{w_i}(y_i - \hat{\mu}_{i(i)})}{s_{(i)}\sqrt{1-h_i}}, \tag{3.5}$$

where $\hat{\mu}_{i(i)}$ is the fitted value for Observation i computed from the model fitted without Observation i. This definition appears to be cumbersome to compute, since computing r_i'' for all n observations apparently requires fitting $n+1$ models (the original with all observations, plus a model when each observation is omitted). However, numerical identities are available for computing r_i'' without the need for repeated linear regressions. Using R, Studentized residuals are easily found using `rstudent()`.

Example 3.7. For the `lungcap` data, the residual plot in Fig. 3.2 (p. 102) shows no outliers (but does shows some large residuals, both positive and negative), so r' and r'' are expected to be similar:

```
> summary( cbind( Standardized = rstandard(LC.lm),
                  Studentized = rstudent(LC.lm) ) )
  Standardized        Studentized
 Min.   :-3.922299   Min.   :-3.966502
 1st Qu.:-0.596599   1st Qu.:-0.596304
 Median : 0.002062   Median : 0.002061
 Mean   : 0.000213   Mean   : 0.000387
 3rd Qu.: 0.559121   3rd Qu.: 0.558826
 Max.   : 4.885392   Max.   : 4.973802
```

□

Example 3.8. For the model `LC.lm` fitted to the `lungcap` data in Example 3.1, the Studentized residuals can be computed by manually deleting each observation. For example, deleting Observation 1 and refitting produces the Studentized residual for Observation 1:

```
>    # Fit the model *without* Observation 1:
> LC.no1 <- lm( FEV ~ Ht + Gender + Smoke,
                data=lungcap[-1,]) # The negative index *removes* row 1
>    # The fitted value for Observation 1, from the original model:
> mu <- fitted( LC.lm )[1]
>    # The estimate of  s  from the new model, without Obs. 1:
> s <- summary(LC.no1)$sigma
> h <- hatvalues( LC.lm )[1] # Hat value, for Observation 1
> resid.stud <- ( lungcap$FEV[1] - mu ) / ( s * sqrt(1-h) )
> resid.stud
       1
1.104565
> rstudent(LC.lm)[1]  # The easy way
       1
1.104565
```

□

3.6.3 Influential Observations

Influential observations are observations that substantially change the fitted model when omitted from the data set. Influential observations necessarily have moderate to large residuals, but are not necessarily outliers. Similarly, outliers may or may not be influential.

More specifically, influential observations are those that combine large residuals with high leverage (Fig. 3.9). That is, influential observations are outliers with high leverage. A popular measure of influence for observation i is *Cook's distance*:

$$D = \frac{(r')^2}{p'} \left(\frac{h}{1-h} \right). \tag{3.6}$$

(The subscript i has been omitted here from all quantities for brevity.) Problem 3.4 develops another interpretation. The values of Cook's distance are found in R using `cooks.distance()`.

Approximately, D has an F-distribution with $(p', n-p')$ degrees of freedom [9], so a conservative approach for identifying influential observations uses the 50th percentile point of the F-distribution as a guideline [39]. This guideline is used by R. For most F-distributions, the 50th percentile is near 1, so a useful rule-of-thumb is that observations with $D > 1$ may be flagged as potentially influential. Other guidelines also exist for identifying high-influence outliers [10, 12].

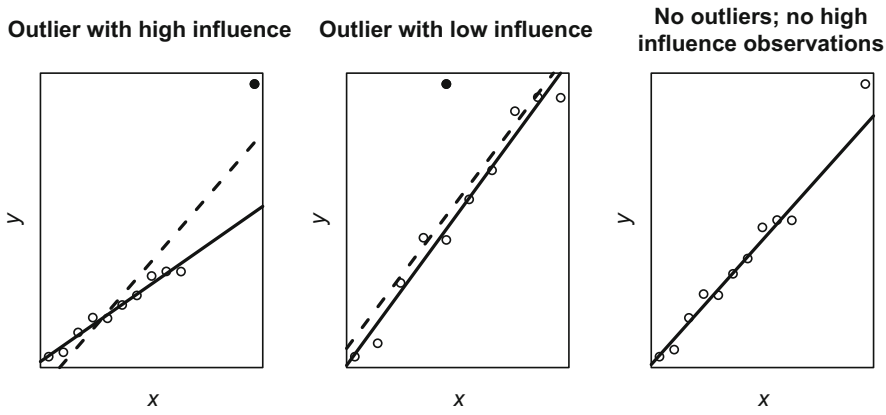

Fig. 3.9 Three examples showing the relationship between outliers and influential observations. The solid circle is the outlier, the solid line is the regression line including the outlier; the dashed line is the regression line omitting the outlier (Sect. 3.6.3)

Another measure of the influence of Observation i, very similar to Cook's distance, is DFFITS. DFFITS measures how much the fitted value of Observation i changes between the model fitted with all the data and the model fitted when Observation i is omitted:

$$\text{DFFITS}_i = \frac{\hat{\mu}_i - \hat{\mu}_{i(i)}}{s_{(i)}} = r_i'' \sqrt{\frac{h_i}{1-h_i}},$$

where $\hat{\mu}_{i(i)}$ is the estimate of μ_i from the model fitted after omitting Observation i from the data. DFFITS_i is essentially equivalent to the square root of Cook's distance. DFFITS_i^2 differs from Cook's distance only by a factor of $1/p'$ and by replacing s_i with $s_{(i)}$. DFFITS are computed in R using `dffits()`.

DFBETA is a coefficient-specific version of DFFITS, which measures how much the estimates of each individual regression coefficient change between the model fitted using all observations and the model fitted with Observation i omitted:

$$\text{DFBETAS}_i = \frac{\hat{\beta}_j - \hat{\beta}_{j(i)}}{\text{se}(\hat{\beta}_{j(i)})},$$

where $\hat{\beta}_{j(i)}$ is the estimate of β_j after omitting Observation i and $\text{se}(\hat{\beta}_{j(i)})$ is the standard error of $\hat{\beta}_j$ using $s_{(i)}$ to estimate the error standard deviation. One set of DFBETAS is produced for each model coefficient. The DFBETAS are computed in R using `dfbetas()`.

Yet another measure of influence, the *covariance ratio* (CR), measures the increase in uncertainty about the regression coefficients when Observation i is omitted. Mathematically, CR is the ratio by which the volume of the confidence ellipsoid for the coefficient vector increases when Observation i is

omitted. More simply, the square root of CR can be interpreted as the average factor by which the confidence intervals for the regression coefficients become wider when Observation i is omitted. A convenient computational formula for CR is:

$$\text{CR} = \frac{1}{1-h}\left\{\frac{n-p}{n-p'+(r'')^2}\right\}^p,$$

where r'' is the Studentized residual (3.5). In R, the function `covratio()` computes CR.

The R function `influence.measures()` produces a table of the influence measures DFBETAS, DFFITS, CR and D, plus the leverages h. Observations identified as influential with respect to any of these statistics (or having high leverage in the case of h) are flagged with a `*` according to the following criteria:

- DFBETAS: Observation i is declared influential when $|\text{DFBETAS}_i| > 1$.
- DFFITS: Observation i is declared influential when

$$|\text{DFFITS}_i| > 3/\sqrt{p'/(n-p')}.$$

- Covariance ratio CR: Observation i is declared influential when $\text{CR}_i > 3p'/(n-p')$.
- Cook's distance D: Observation i is declared influential when D exceeds the 50th percentile of the F distribution with $(p', n-p')$ degrees of freedom.
- Leverages h: Observations are declared high leverage if $h > 3p'/n$.

Different observations may be declared as influential by the different criteria. The covariance ratio has a tendency to declare more observations as influential than the other criteria.

Example 3.9. Consider the lung capacity data again (Example 1.1; data set: `lungcap`), and model `LC.lm` (Example 3.1, p. 97). The observations with the smallest and largest values of Cook's distance are:

```
> cd.max <- which.max( cooks.distance(LC.lm))   # Largest D
> cd.min <- which.min( cooks.distance(LC.lm))   # Smallest D
> c(Min.Cook = cd.min, Max.Cook = cd.max)
 Min.Cook.69 Max.Cook.613
          69          613
```

The values of DFFITS, CV and Cook's distance for these observations can be found as follows:

```
> out <- cbind( DFFITS=dffits(LC.lm),
                Cooks.distance=cooks.distance(LC.lm),
                Cov.ratio=covratio(LC.lm))
```

These statistics for the observations `cd.max` and `cd.min` are:

```
> round( out[c(cd.min, cd.max),], 5)  # Show the values for these obs only
      DFFITS Cooks.distance Cov.ratio
69   0.00006        0.00000   1.01190
613 -0.39647        0.03881   0.96737
```

From these three measures, Observation 613 is more influential than Observation 69 according to DFFITS and Cook's distance (but not CV). Now examine influence of Observation 613 and 69 on each of the regression parameters:

```
> dfbetas(LC.lm)[cd.min,]  # Show DBETAS for cd.min
  (Intercept)            Ht       GenderM   SmokeSmoker
 4.590976e-05 -3.974922e-05 -2.646158e-05 -1.041249e-06
> dfbetas(LC.lm)[cd.max,]  # Show DBETAS for cd.max
(Intercept)          Ht     GenderM SmokeSmoker
 0.05430730 -0.06394615  0.10630441 -0.31682958
```

Omitting Observation 69 (`cd.min`) makes almost no difference to the regression coefficients. Observation 613 is clearly more influential than Observation 69, as expected. The R function `influence.measures()` is used to identify potentially influential observations according to R's criteria:

```
> LC.im <- influence.measures( LC.lm ); names(LC.im)
[1] "infmat" "is.inf" "call"
```

The object `LC.im` contains the influence statistics (as `LC.im$infmat`), and whether or not they are influential according to R's criteria (`LC.im$is.inf`):

```
> head( round(LC.im$infmat, 3) )  # Show for first few observations only
  dfb.1_ dfb.Ht dfb.GndM dfb.SmkS  dffit cov.r cook.d   hat
1  0.117 -0.109   -0.024    0.015  0.127 1.012  0.004 0.013
2 -0.005  0.005    0.001   -0.001 -0.006 1.017  0.000 0.010
3  0.051 -0.047   -0.014    0.005  0.058 1.015  0.001 0.010
4  0.113 -0.104   -0.031    0.012  0.127 1.007  0.004 0.010
5  0.116 -0.106   -0.036    0.010  0.133 1.004  0.004 0.009
6  0.084 -0.077   -0.026    0.007  0.097 1.009  0.002 0.009
> head( LC.im$is.inf )
  dfb.1_ dfb.Ht dfb.GndM dfb.SmkS dffit cov.r cook.d   hat
1  FALSE  FALSE    FALSE    FALSE FALSE FALSE  FALSE FALSE
2  FALSE  FALSE    FALSE    FALSE FALSE FALSE  FALSE FALSE
3  FALSE  FALSE    FALSE    FALSE FALSE FALSE  FALSE FALSE
4  FALSE  FALSE    FALSE    FALSE FALSE FALSE  FALSE FALSE
5  FALSE  FALSE    FALSE    FALSE FALSE FALSE  FALSE FALSE
6  FALSE  FALSE    FALSE    FALSE FALSE FALSE  FALSE FALSE
```

To determine how many entries in the columns of `LC.im$is.inf` are `TRUE`, sum over the columns (this works because R treats `FALSE` as 0 and `TRUE` as 1):

```
> colSums( LC.im$is.inf )
  dfb.1_   dfb.Ht dfb.GndM dfb.SmkS    dffit    cov.r   cook.d      hat
       0        0        0        0       18       56        0        7
```

Seven observations have high leverage, as identified by the column labelled `hat`, 56 observations are identified by the covariance ratio as influential, but Cook's distance does not identify any observation as influential.

We can also determine how many criteria declare observations as influential by summing the relevant columns of `LC.im$is.inf` over the rows:

```
> table( rowSums( LC.im$is.inf[, -8] ) )   # Omitting leverages (col 8)
  0   1   2
590  54  10
```

This shows that most observations are not declared influential on any of the criteria, and 54 observations declared as influential on just one criterion.

For Observations 69 and 613 explicitly:

```
> LC.im$is.inf[c(cd.min, cd.max), ]
    dfb.1_ dfb.Ht dfb.GndM dfb.SmkS dffit cov.r cook.d   hat
69   FALSE  FALSE    FALSE    FALSE FALSE FALSE  FALSE FALSE
613  FALSE  FALSE    FALSE    FALSE  TRUE  TRUE  FALSE FALSE
```

Observation 613 is significantly influential based on DFFITS and CV.

A plot of these influence diagnostics is often useful (Fig. 3.10), using `type="h"` to draw histogram-like (or high-density) plots:

```
>    # Cooks' Distance
> plot( cooks.distance( LC.lm ), type="h", main="Cook's distance",
    ylab="D", xlab="Observation number", las=1  )
>    # DFFITS
> plot( dffits( LC.lm ), type="h", main="DFFITS",
    ylab="DFFITS", xlab="Observation number", las=1 )
```

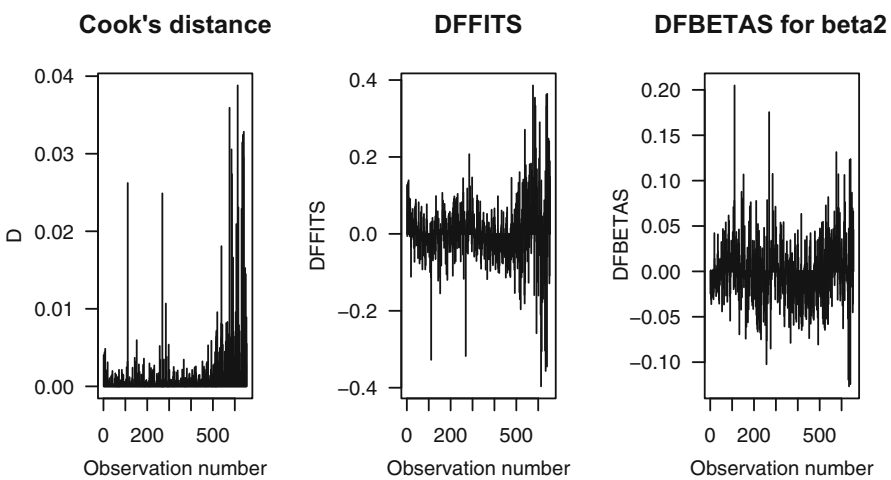

Fig. 3.10 Influence diagnostics for model `LC.lm` fitted to the lung capacity data. Left panel: Cook's distance D_i; centre panel: DFFITS; right panel: DFBETAS for β_2 (Example 3.9)

```
>    # DFBETAS for beta_2 only (that is, column 3)
> dfbi <- 2
> plot( dfbetas( LC.lm )[, dfbi + 1], type="h", main="DFBETAS for beta2",
    ylab="DFBETAS", xlab="Observation number", las=1 )
```

□

3.7 Terminology for Residuals

The terminology used for residuals is confusingly inconsistent. Generally in statistics, dividing some quantity by an estimate of its standard deviation is called *standardizing*. More specifically, dividing a quantity which follows a normal distribution by the sample standard deviation to produce a quantity which follows a t-distribution is called *Studentizing*, following the approach used by Student [37] when introducing the t-distribution.

Under these commonly-used definitions, both r' and r'' are standardized and Studentized residuals, and various authors use the terms for describing both residuals. Following R and Belsley et al. [3], we call r'' the *Studentized residual* (Sect. 3.6.2; `rstudent()` in R) because it follows a Student's t-distribution exactly, whereas r' will simply be called the *standardized residual* (Sect. 3.3; `rstandard()` in R).

An alternative convention [39] is to call r' the *internally Studentized residual* and r'' the *externally Studentized residual*. These labels are perhaps more specific and descriptive of the differences between the two types of residuals, but have not become widely used in the literature.

3.8 Remedies: Fixing Identified Problems

The past few sections have described a variety of diagnostics for identifying different types of weaknesses in the fitted model. The next few sections will consider some standard strategies for modifying the fitted model in order to remedy or ameliorate specific problems.

One commonly-occurring problem is that the response is recorded on a measurement scale for which the variance increases or decreases with the mean. If this is the case, the variance can often be stabilized by transforming the response to a different scale (Sect. 3.9).

Sometimes a nonlinear relationship between y and x can be fixed by a simple transformation of x (Sect. 3.10). More generally, a complex relationship between a covariate and the response signals the need to build further terms into the model to capture this relationship (Sections 3.11 and 3.12). Usually the measurement scale of y should be settled before transforming

the covariates, because any transformation of y will obviously impact on the shape of its relationships with the covariates.

Often the above steps will solve structural problems and hence also tend to reduce the number of apparent outliers or dangerously influential observations. If some remain, however, decisions must be made to remove the outliers or to accommodate them into a modified model. Section 3.13 discusses these issues.

One possible problem that will not be discussed in detail later is that of correlated residuals. Dependence between responses can arise from common causes shared between observations, or from a carryover effect from one observation to another, or from other causes. When the responses fail to be independent, there are a variety of more complex models that can be developed to accommodate this dependence, including generalized least squares [8], mixed models [40] or spatial models [5]. All of these possibilities however would take us outside the scope of this book.

3.9 Transforming the Response

3.9.1 Symmetry, Constraints and the Ladder of Powers

The idea of a transformation is to convert the response variable to a different measurement scale. For example, consider the acidity of swimming pool water. From a chemical point of view, acidity is measured by the concentration of hydrogen ions. However acidity is more commonly expressed in terms of pH-level. If y is hydrogen ion concentration, then the pH level is defined by $\text{pH} = -\log_{10} y$. This serves as an alternative and, for many purposes, more useful scale on which to measure the same quantity. In mathematical terms, a new response variable $y^* = h(y)$ is computed from y, where $h()$ is some invertible function, and then a linear regression model is built for y^* instead of y. In the case of the pH-level, $h(y) = -\log_{10} y$. After transforming the response, the basic linear regression model structure remains the same, the new variable y^* simply replacing y. The model becomes

$$\begin{cases} y_i^* \sim N(\mu_i, \sigma^2) \\ \mu_i = \beta_0 + \sum_{j=1}^{p} \beta_j x_{ij}. \end{cases} \tag{3.7}$$

Note that now $\mu_i = \mathrm{E}[y_i^*]$ rather than $\mathrm{E}[y_i]$. After transforming the response, one will typically refit the model and produce new residual plots to recheck assumptions for the new model. This may be done iteratively until a satisfactory transformation is found.

There are three main reasons why one might choose to transform the response variable. First, transforming the measurement scale so that it covers

the whole real line can avoid difficulties with constraints on the linear regression coefficients. In the lung capacity study, for example, ideally we would like to ensure that our model will never predict a negative value for FEV. The difficulty with predicting negative values for FEV can be avoiding by building a linear regression model for $y^* = \log(\text{FEV})$ instead of for FEV itself, because any predicted value for the logarithm of FEV, whether negative or positive, translates into a positive value for FEV itself.

When y is a count for which zero is a possible value, the starred log-transformations $y^* = \log(y + 0.5)$ or $y^* = \log(y + 1)$ have been used to avoid taking the logarithm of zero. When y is a count out of a possible total n, then the empirical logistic transformation $y^* = \log\{(y+0.5)/(n+0.5)\}$ has sometimes been used. In both cases the motivation is the same: to convert the response to a scale for which the linear predictor is unconstrained. These transformations can be successful if the counts are not too small or too near the boundary values.

A second possible reason for transforming the response is to cause its distribution to be more nearly normally distributed. Typically this means trying to make the distribution of y-values more symmetric. For example, consider the acidity of swimming pool water again. The concentration of hydrogen ions is a strictly positive quantity, usually very close to zero but varying by orders of magnitude from one circumstance to another. Hence hydrogen concentration is likely have a highly right-skewed distribution. By contrast, the pH-levels are usually more symmetrically distributed. In other words, the pH-level is likely to be more nearly normally distributed than is the hydrogen ion concentration itself. Right skew distributions arise most commonly when the response measures a physical quantity that can only take positive values. In such a case, a log-transformation, $y^* = \log y$, or a power transformation, $y^* = y^\lambda$ with $\lambda < 1$, will reduce the right skewness. Common values for λ make up what is sometimes called a ladder of powers (Table 3.1). The smaller λ is chosen, the stronger the transformation. A too small value for λ will reverse a right skew distribution to one that is left skew. The usual procedure is to start with a transformation with λ near one, then decrease λ until symmetry of the residuals from the regression is roughly achieved.

If y is left skewed, then a power transformation $y^* = y^\lambda$ with $\lambda > 1$ might be used (Table 3.1). Such situations are less common however.

3.9.2 Variance-Stabilizing Transformations

There is a third and even more fundamental motivation for transforming the response variable, which is to try to achieve close to constant variance across all observations. Again we focus on the commonly-occurring situation in which y measures some physical quantity that can only take on positive values. For such a variable, it is almost inevitable that the variance of y will be smaller when μ is close to zero than when μ is large, because of

Table 3.1 The 'ladder of powers'. Variance increasing with mean is more common than variance decreasing with the mean, hence the transformations on the right-hand side are more commonly used. Note that $\lambda = 1$ produces no transformation of the response (Sect. 3.9)

Transformation:	$\leftarrow \ldots$	y^3	y^2	y	$\sqrt{y}$	$\log y$	$1/\sqrt{y}$	$1/y$	$1/y^2$	$\cdots \rightarrow$
Box–Cox λ:	$\leftarrow \ldots$	3	2	1	1/2	0	$-1/2$	-1	-2	$\cdots \rightarrow$
Primary use:	• When variance *decreases* with increasing mean				• When variance *increases* with increasing mean					
Other uses:	• When y left-skewed				• When y right-skewed					

the requirement that the range of y is restricted to positive values. This phenomenon will become readily apparent in practical terms when the values of y vary by orders of magnitude in a single data set. In these cases, we say that y shows a positive mean–variance relationship.

In the scientific literature, the uncertainty of physical measurements of positive quantities are often expressed in terms of the coefficient of variation (standard deviation divided by the mean) instead of in terms of variance or standard deviation. This is because the coefficient of variation often tends to be more nearly constant across cases than is the standard deviation, so it is more useful to express variability in relative terms rather than in absolute terms. Mathematically, this means that the standard deviation σ of y is proportional to the mean μ or, equivalently, the variance is proportional to the mean squared, $\text{var}[y] = \phi\mu^2$ for some ϕ. In such cases, y is said to have a quadratic mean–variance relationship. The strongest motivation for transforming the response is usually to try to remove the mean–variance relationship.

If y takes positive values, then the ladder of powers may be used to remove or mitigate a mean–variance relationship (Table 3.1). A power transformation with $\lambda < 1$ will reduce or remove an increasing mean–variance relationship, while $\lambda > 1$ will reduce or remove a decreasing mean–variance relationship.

More generally, we consider the class of *variance-stabilizing transformations*. Suppose that y has a mean–variance relationship defined by the function $V(\mu)$, with $\text{var}[y] = \phi V(\mu)$. Then, consider a transformation $y^* = h(y)$. A first-order Taylor series expansion of $h(y)$ about μ gives $y^* = h(y) \approx h(\mu) + h'(\mu)(y - \mu)$, from which it can be inferred that

$$\text{var}[y^*] = \text{var}[h(y)] \approx h'(\mu)^2 \text{var}[y].$$

Hence the transformation $y^* = h(y)$ will approximately stabilize the variance if $h'(\mu)$ is proportional to $\text{var}[y]^{-1/2} = V(\mu)^{-1/2}$. When $V(\mu) = \mu^2$ (standard deviation proportional to the mean), the variance-stabilizing transformation is the logarithm, because then $h'(\mu) = 1/\mu$. When $V(\mu) = \mu$, the variance-stabilizing transformation is the square root, because $h'(\mu) = 1/\mu^{1/2}$.

The most common variance-stabilizing transformations appear on a ladder of powers (Table 3.1). To use this ladder, note that the milder transformations are closer to $\lambda = 1$ (no transformation). It is usual to start with mild transformations and progressively try more severe transformations as necessary. For example, if a logarithmic transformation still produces increasing variance as the mean increases, try the next transformation on the ladder: $1/\sqrt{y}$. The most commonly-used transformation is the logarithmic transformation.

When y is a proportion or percentage (taking values from zero to one, or zero to 100%), the mean–variance relationship is likely to be unimodal. In such cases, the possible values for y have two boundaries, one at zero and the other at one, and the variance of y is likely to decrease as the mean approaches either boundary. Proportions often show a quadratic mean–variance relationship of the form $V(\mu) \propto \mu(1-\mu)$, with $0 < \mu < 1$. In such cases, the variance-stabilizing transformation is the arc-sin–square root transformation $y^* = \sin^{-1}\sqrt{y}$.

Transformations with $\lambda \leq 0$ can only be applied to positive values of y. If negative values are present, then power transformations should not be used. If y is positive except for a few exact zeros, one has the choice between using a positive value of λ, for example a small positive value such as $\lambda = 1/4$ instead of a log-transformation, or else offsetting y to be positive before transforming. For example, a response variable such as rainfall is positive and continuous on days when rain has occurred, but is zero otherwise. In such cases, the starred logarithmic transformation, $y^* = \log(y+c)$ where c is a small positive constant, has sometimes been used. Such transformations should be used with caution, as they are sensitive to the choice of offset c. Choosing c too small can easily introduce outliers into the data.

Example 3.10. For the `lungcap` data, we have established that the model `LC.lm` is inadequate (Example 3.3). For example, a plot of r' against $\hat{\mu}_i$ (Fig. 3.5) shows non-constant variance. Various transformations of the response can be used to determine which, if any, transformation of the response is appropriate (Fig. 3.11). Since the variance increases with increasing mean, try the first transformation suggested on the ladder of powers (Table 3.1, p. 118), the square root transformation:

```
> LC.sqrt <- update( LC.lm, sqrt(FEV) ~ .)
> scatter.smooth( rstandard(LC.sqrt)~fitted(LC.sqrt), las=1, col="grey",
    ylab="Standardized residuals", xlab="Fitted values",
    main="Square-root transformation")
```

This transformation (Fig. 3.11, top right panel) produces slightly increasing variance. Try the next transformation on the ladder, the commonly-used logarithmic transformation:

```
> LC.log <- update( LC.lm, log(FEV) ~ .)
> scatter.smooth( rstandard(LC.log)~fitted(LC.log), las=1, col="grey",
    ylab="Standardized residuals", xlab="Fitted values",
    main="Log transformation")
```

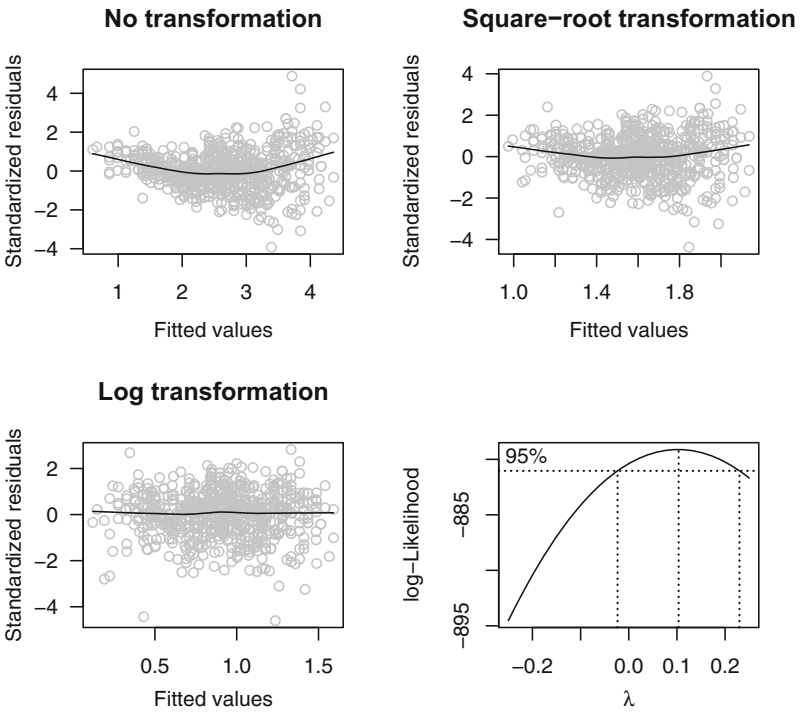

Fig. 3.11 Transformations of the FEV in the data frame `lungcap`. The original data (top left panel); using a square root transformation (top right panel); using a logarithmic transformation (bottom left panel); a plot to find the Box–Cox transformation (bottom right panel) (Examples 3.10 and 3.11)

This plot show approximately constant variance and no trend. The logarithmic transformation appears suitable, and also allows easier interpretations than using the square root transformation. A logarithmic transformation of the response is required to produce almost constant variance, as used in Chap 2. □

3.9.3 Box–Cox Transformations

Notice that the transformations in Table 3.1 have the form of y raised to some power, except for the logarithmic transformation. The logarithmic transformation also fits the general power-transformation form if we define

$$y^* = \begin{cases} \dfrac{y^\lambda - 1}{\lambda} & \text{for } \lambda \neq 0 \\ \log y & \text{for } \lambda = 0. \end{cases} \tag{3.8}$$

This family of transformations is called the *Box–Cox transformation* [7]. The form of the Box–Cox transformation (3.8) is continuous in λ when natural logarithms are used, since $(y^\lambda - 1)/\lambda \to \log y$ as $\lambda \to 0$. The Box–Cox transformation (3.8) has the same impact as the transformation $y^* = y^\lambda$, but the results differ numerically. For example, $\lambda = 1$ transforms the responses y to $(y-1)$, which has no impact on the model structure, but the numerical values of the response change.

Computationally, various values of λ are chosen, and the transformation producing the response y^* with approximately constant variance is then chosen. This approach can be implemented in R directly, or by using the function `boxcox()` (in the package **MASS**). The `boxcox()` function uses the maximum likelihood criterion, discussed in the next chapter of this book. It finds the optimal λ to achieve linearity, normality and constant variance simultaneously.

Example 3.11. Continuing using the `lungcap` data from the previous example, we use the `boxcox()` function to estimate the optimal Box–Cox transformation. In the plot produced, higher log-likelihood values are preferable. The maximum of the Box–Cox plot is achieved when λ is just above zero, confirming that a logarithmic transformation is close to optimal for achieving linearity, normality and constant variance (Fig. 3.11, bottom right panel):

```
> library(MASS) # The function  boxcox()  is in the MASS package
> boxcox( FEV ~ Ht + Gender + Smoke,
          lambda=seq(-0.25, 0.25, length=11), data=lungcap)
```

□

3.10 Simple Transformations of Covariates

Sometimes, to achieve linearity or to reduce the influence of influential observations, transformations of the covariates are required (Fig. 3.12). Using transformed covariates still produces a model linear in the parameters. Transformations may apply to any or all of the covariates. (Transforming factors makes no sense.)

Example 3.12. The wind velocity and corresponding direct current (DC) output from windmills (Table 3.2; data set: `windmill`) was recorded [18, 19]. A plot of the data (Fig. 3.13, left panels) shows non-linearity, but little evidence of non-constant variance (so a transformation of the response is not recommended):

```
> data(windmill); names(windmill)
[1] "Wind" "DC"
```

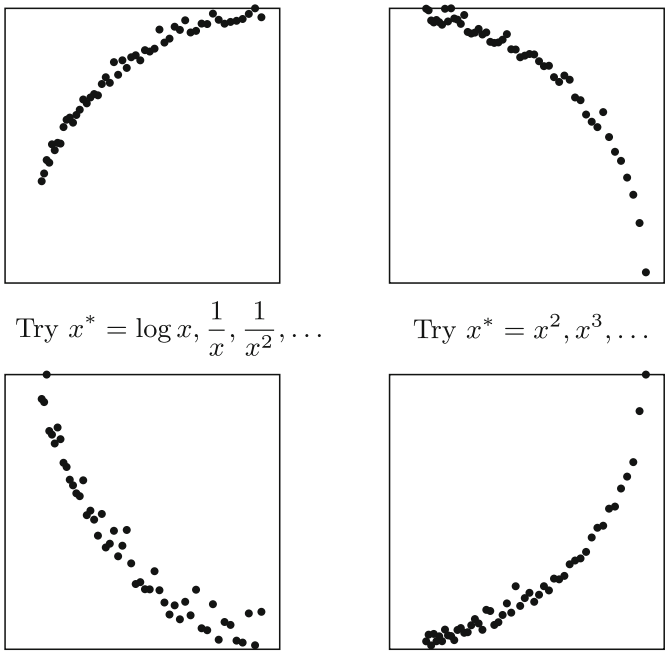

Fig. 3.12 Transformations of covariates to achieve linearity (Sect. 3.10)

Table 3.2 The DC output from windmills at various wind velocities (in miles/h) (Example 3.2)

Wind velocity	DC output	Wind velocity	DC output	Wind velocity	DC output
2.45	0.123	4.60	1.562	7.85	2.179
2.70	0.500	5.00	1.582	8.15	2.166
2.90	0.653	5.45	1.501	8.80	2.112
3.05	0.558	5.80	1.737	9.10	2.303
3.40	1.057	6.00	1.822	9.55	2.294
3.60	1.137	6.20	1.866	9.70	2.386
3.95	1.144	6.35	1.930	10.00	2.236
4.10	1.194	7.00	1.800	10.20	2.310
		7.40	2.088		

```
> scatter.smooth( windmill$DC ~ windmill$Wind, main="No transforms",
    xlab="Wind speed", ylab="DC output", las=1)
> wm.m1 <- lm( DC ~ Wind, data=windmill )
> scatter.smooth( rstandard(wm.m1) ~ fitted(wm.m1), main="No transforms",
    xlab="Standardized residulas", ylab="Fitted values", las=1)
```

To alleviate the non-linearity, we try some transformations of the wind-speed. Based on Fig. 3.12, we initially try a logarithmic transformation of `Wind`, the most common transformation (Fig. 3.13, centre panels):

```
> scatter.smooth( windmill$DC ~ log(windmill$Wind), main="Log(Wind)",
    xlab="log(Wind speed)", ylab="DC output", las=1)
> wm.m2 <- lm( DC ~ log(Wind), data=windmill )
> scatter.smooth( rstandard(wm.m2) ~ fitted(wm.m2), main="Log(Wind)",
    ylab="Standardized residuals", xlab="Fitted values", las=1)
```

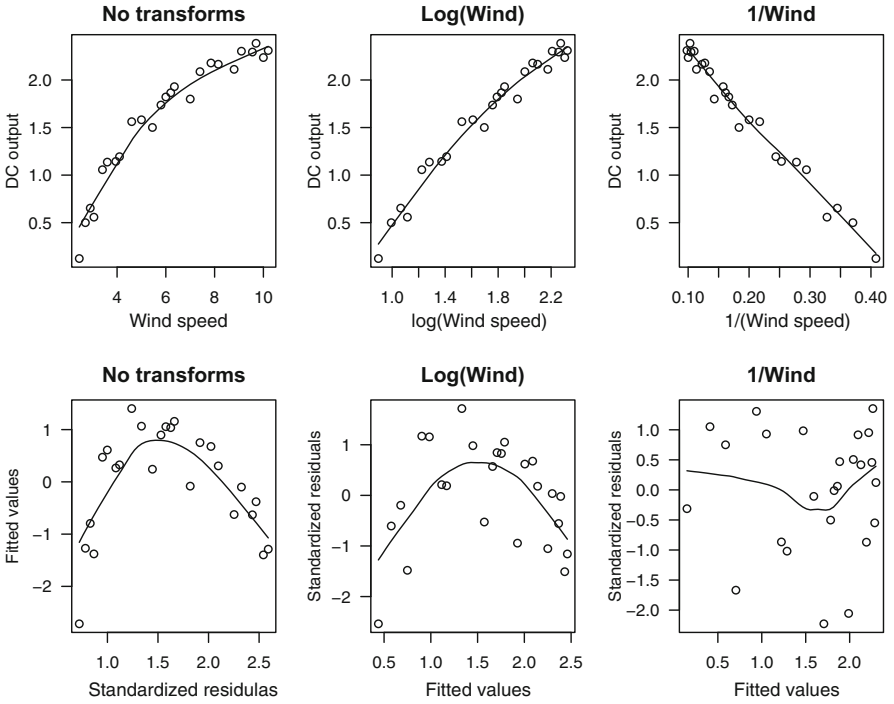

Fig. 3.13 The windmill data. Left panels: the original data; centre panels: using the logarithm of `Wind`; right panels: using the inverse of `Wind`; top panels: `DC` against the covariate or transformed covariate; bottom panels: the standardized residuals against the covariate or transformed covariate (Example 3.12)

The relationship is still non-linear, so try a more extreme transformation, such as a reciprocal transformation of `Wind` (Fig. 3.13, right panels):

```
> scatter.smooth( windmill$DC ~ (1/windmill$Wind), main="1/Wind",
    xlab="1/(Wind speed)", ylab="DC output", las=1)
> wm.m3 <- lm( DC ~ I(1/Wind), data=windmill )
> scatter.smooth( rstandard(wm.m3) ~ fitted(wm.m3), main="1/Wind",
     ylab="Standardized residuals", xlab="Fitted values", las=1)
```

Note the use of `I()` when using `lm()`. This is needed because `1/Wind` has a different meaning in an R formula than what is intended here. The term `1/Wind` would mean to fit a model with `Wind` nested within the intercept, an interpretation which makes no sense here. To tell R to interpret `1/Wind` as an arithmetic expression rather than as a formula we *insulate* it (or *inhibit* interpretation as a formula operator) by surrounding it with the function `I()`. (For another example using `I()`, see Example 3.15, p. 128.)

The relationship is now approximately linear, and the variance is approximately constant. The diagnostics show the model is mostly adequate (Fig. 3.14):

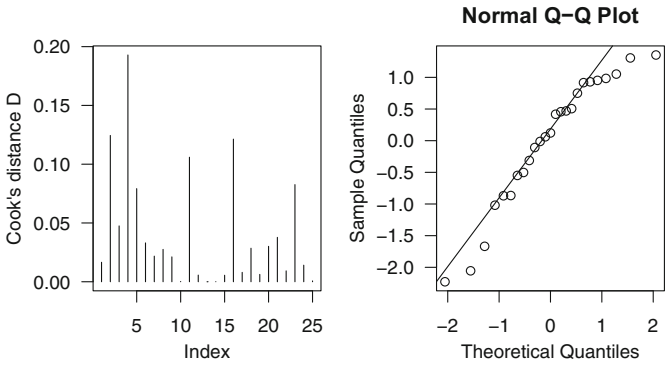

Fig. 3.14 Diagnostic plots from fitting a model with the inverse of `Wind` to the windmill data. Left: Cook's distance; right: the Q–Q plot of standardized residuals (Example 3.12)

```
> plot( cooks.distance( wm.m3 ), type="h", las=1, ylab="Cook's distance D")
> qqnorm( rstandard( wm.m3), las=1 ); qqline( rstandard( wm.m3 ), las=1 )
```

No observations appear influential; no standardized residuals appear large (though the normality of the residuals may be a little suspect). The systematic component is

```
> coef( wm.m3 )
(Intercept)   I(1/Wind)
   2.978860   -6.934547
```

□

A special case where simultaneous log-transformations of both x and y can be useful is that where physical quantities may be related through power laws. If y is proportional to some power of x such that $\mathrm{E}[y] = \alpha x^{\beta}$, the relationship may be linearized by logging both x and y, since $\mathrm{E}[\log y] \approx \log \alpha + \beta \log x$.

Example 3.13. In the lung capacity study (data set: `lungcap`), FEV is a volume measure and hence is in units of length cubed, whereas height is in ordinary units of length. Other things being equal, one would expect volume to be proportional to a length measure (like height) cubed. On the log-scale, we would expect log(`FEV`) to be linearly related to log(`Ht`) with a slope close to 3, and this turns out to be so (Fig. 3.15):

```
> LC.lm.log <- lm(log(FEV)~log(Ht), data=lungcap)
> printCoefmat(coef(summary(LC.lm.log)))
             Estimate Std. Error t value  Pr(>|t|)
(Intercept) -11.921103   0.255768 -46.609 < 2.2e-16 ***
log(Ht)       3.124178   0.062232  50.202 < 2.2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
> plot( log(FEV) ~ log(Ht), data=lungcap, las=1)
```

□

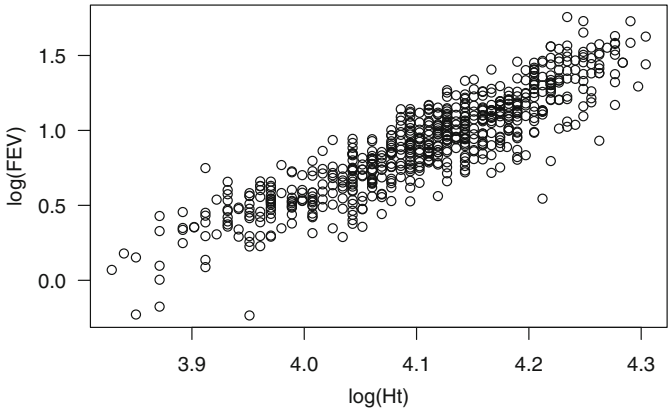

Fig. 3.15 The logarithm of FEV plotted against the logarithm of height for the lung capacity data (Example 3.13)

Example 3.14. The volume y (in cubic feet) of 31 black cherry trees was measured [2, 28, 34] as well as the height (in feet) and the girth, or diameter, at breast height (in inches) (Table 3.3; data set: `trees`):

```
> data(trees)   # The  trees  data frame comes with R
> plot( Volume ~ Height, data=trees, las=1, pch=19, xlab="Height (feet)",
    ylab="Volume (cubic feet)", main="Volume vs height", las=1)
> plot(Volume ~ Girth,  data=trees, las=1, pch=19, xlab="Girth (inches)",
    ylab="Volume (cubic feet)", main="Volume vs girth", las=1)
```

The volume of the tree is related to the volume of timber, which is important economically. The relationships between the tree volume and height, and tree volume and girth, both appear non-linear (Fig. 3.16, top panels).

An appropriate systematic component can be developed by approximating the cherry trees as either cones or cylinders in shape. For these shapes, formulae for computing the timber volume y in cubic feet from the height in feet h and the girth (diameter) in feet $d/12$ (recall the girth is given in inches, not feet; 12 inches in one foot) are:

$$\text{Cone:} \qquad y = \frac{\pi(d/12)^2 h}{12};$$
$$\text{Cylinder:} \qquad y = \frac{\pi(d/12)^2 h}{4}.$$

Taking logarithms and simplifying,

$$\text{Cone:} \qquad \mu = \log(\pi/1728) + 2\log d + \log h$$
$$\text{Cylinder:} \qquad \mu = \log(\pi/576) \ + 2\log d + \log h$$

Table 3.3 The volume, height and girth (diameter) for 31 felled black cherry trees in the Allegheny National Forest, Pennsylvania (Example 3.3)

Girth (in inches)	Height (in feet)	Volume (in cubic feet)	Girth (in inches)	Height (in feet)	Volume (in cubic feet)
8.3	70	10.3	12.9	85	33.8
8.6	65	10.3	13.3	86	27.4
8.8	63	10.2	13.7	71	25.7
10.5	72	16.4	13.8	64	24.9
10.7	81	18.8	14.0	78	34.5
10.8	83	19.7	14.2	80	31.7
11.0	66	15.6	14.5	74	36.3
11.0	75	18.2	16.0	72	38.3
11.1	80	22.6	16.3	77	42.6
11.2	75	19.9	17.3	81	55.4
11.3	79	24.2	17.5	82	55.7
11.4	76	21.0	17.9	80	58.3
11.4	76	21.4	18.0	80	51.5
11.7	69	21.3	18.0	80	51.0
12.0	75	19.1	20.6	87	77.0
12.9	74	22.2			

where $\mu = \mathrm{E}[\log y]$. Plotting the logarithm of volume against the logarithm of girth and height (Fig. 3.16, bottom panels) shows approximately linear relationships:

```
> plot( log(Volume)~log(Height), data=trees, pch=19, xlab="log(Height)",
    ylab="log(Volume)", main="Log(Volume) vs log(Height)", las=1)
> plot( log(Volume)~log(Girth), data=trees, pch=19, xlab="log(Girth)",
    ylab="log(Volume)", main="Log(Volume) vs log(Girth)", las=1)
```

Since the cone and cylinder are only approximations, enforcing the parameters to the above values may be presumptuous. Instead, consider the more general model with the form

$$\log \mu = \beta_0 + \beta_1 \log d + \beta_2 \log h.$$

If the assumptions about the tree shapes are appropriate, expect $\beta_1 \approx 2$ and $\beta_2 \approx 1$. The value of β_0 may give an indication of whether the cone ($\beta_0 \approx \log(\pi/1728) = -6.310$) or the cylinder ($\beta_0 \approx \log(\pi/576) = -5.211$) is a better approximation to the shape.

To fit the suggested model in R:

```
> m.trees <- lm( log(Volume)~log(Girth)+log(Height), data=trees)
> printCoefmat( coef(summary(m.trees)))
             Estimate Std. Error t value  Pr(>|t|)
(Intercept) -6.631617   0.799790 -8.2917 5.057e-09 ***
log(Girth)   1.982650   0.075011 26.4316 < 2.2e-16 ***
log(Height)  1.117123   0.204437  5.4644 7.805e-06 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

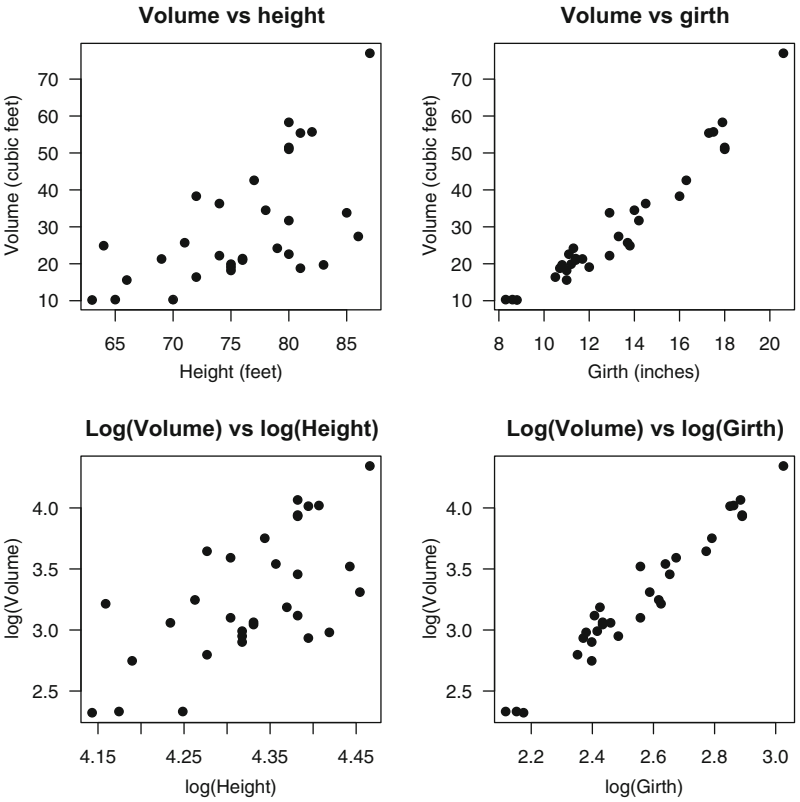

Fig. 3.16 The volume of timber from 31 cherry trees plotted against the tree height (top left panel) and against tree girth (top right panel). The bottom panels show the logarithm of volume against logarithm of height (bottom left panel) and logarithm of volume against logarithm of girth (bottom right panel) (Example 3.14)

Observe that $\hat{\beta}_0 = -6.632$ is close to the value expected if trees were approximated as cones. In addition, $\hat{\beta}_1 \approx 2$ and $\hat{\beta}_2 \approx 1$ as expected. □

3.11 Polynomial Trends

The covariate transformations discussed in the previous section are simple and commonly used. Sometimes, however, the relationship between the response and the covariates is more complicated than can be described by simple transformations of the covariates. A more general possibility is to build a polynomial trend as a function of one of the covariates. The higher the

Table 3.4 The heat capacity C_p of hydrogen bromide (in calories/(mole.K)) and the temperature (in K) (Example 3.15)

C_p	Temperature	C_p	Temperature	C_p	Temperature
10.79	118.99	10.98	132.41	11.40	158.03
10.80	120.76	11.03	135.89	11.61	162.72
10.86	122.71	11.08	139.02	11.69	167.67
10.93	125.48	11.10	140.25	11.91	172.86
10.99	127.31	11.19	145.61	12.07	177.52
10.96	130.06	11.25	153.45	12.32	182.09

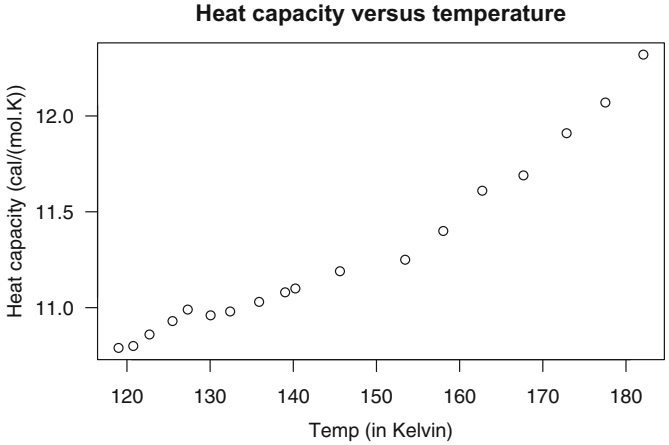

Fig. 3.17 The heat capacity of hydrogen bromide plotted against temperature (Example. 3.15)

degree of the polynomial, the greater the complexity of the trend that can be fitted. Unlike covariate transformations, which do not increase the number of covariates in the model, polynomial trends involve adding new terms to linear predictor, such as x^2 and x^3, which are powers of the original covariate.

Example 3.15. Consider the heat capacity (C_p) of solid hydrogen bromide (HBr) [17, 31] as a function of temperature (Table 3.4; data set: `heatcap`). The relationship between heat capacity and temperature is clearly non-linear (Fig. 3.17):

```
> data(heatcap)
> plot( Cp ~ Temp, data=heatcap, main="Heat capacity versus temperature",
    xlab="Temp (in Kelvin)", ylab="Heat capacity (cal/(mol.K))", las=1)
```

First note that the variation in the responses appears approximately constant, and that the relationship is nonlinear. However, simple transformations like $\log x$ are unlikely to work well for these data as the relationship is more

complex; polynomials may be suitable. Care is needed when adding powers of covariates to the systematic component in R. For example, this command does *not* produce the required result:

```
> lm( Cp ~ Temp + Temp^2, data=heatcap)  ###   INCORRECT!
```

The above command fails, because the `^` symbol is interpreted in a formula as crossing terms in the formula, and not as the usual arithmetic instruction to raise `Temp` to a power. To tell R to interpret `^` arithmetically, we *insulate* the terms (or *inhibit* interpretation as a formula operator) by using `I()`:

```
> hc.col <- lm( Cp ~ Temp + I(Temp^2), data=heatcap)
```

Observe that the correlations between the two predictors are extremely close to plus or minus one.

```
> summary(hc.col, correlation=TRUE)$correlation
             (Intercept)       Temp  I(Temp^2)
(Intercept)   1.0000000 -0.9984975  0.9941781
Temp         -0.9984975  1.0000000 -0.9985344
I(Temp^2)     0.9941781 -0.9985344  1.0000000
```

This is not uncommon when x, x^2, x^3 and similar higher powers (referred to as the *raw polynomials*) are used as model explanatory variables. Correlated covariates may cause difficulties and confusion in model selection, and are discussed more generally in Sect. 3.14. More numerically stable polynomials are usually fitted, called orthogonal polynomials, using `poly()` in R. For the heat capacity data, we can fit four polynomial models using `poly()`, and compare:

```
> hc.m1 <- lm( Cp ~ poly(Temp, 1), data=heatcap)   # Linear
> hc.m2 <- lm( Cp ~ poly(Temp, 2), data=heatcap)   # Quadratic
> hc.m3 <- lm( Cp ~ poly(Temp, 3), data=heatcap)   # Cubic
> hc.m4 <- lm( Cp ~ poly(Temp, 4), data=heatcap)   # Quartic
```

The correlations between the estimated regression parameters are now zero to computer precision. For example:

```
> summary(hc.m2, correlation=TRUE)$correlation
                 (Intercept) poly(Temp, 2)1 poly(Temp, 2)2
(Intercept)     1.000000e+00   3.697785e-32  -3.330669e-16
poly(Temp, 2)1  3.697785e-32   1.000000e+00  -1.110223e-16
poly(Temp, 2)2 -3.330669e-16  -1.110223e-16   1.000000e+00
> zapsmall( summary(hc.m2,correlation=TRUE)$correlation )
              (Intercept) poly(Temp, 2)1 poly(Temp, 2)2
(Intercept)             1              0              0
poly(Temp, 2)1          0              1              0
poly(Temp, 2)2          0              0              1
```

Because the polynomials are orthogonal, the coefficients of each fitted polynomial do not change when higher order polynomials are added to the model, unlike the coefficients when using the raw polynomials 1, x and x^2.

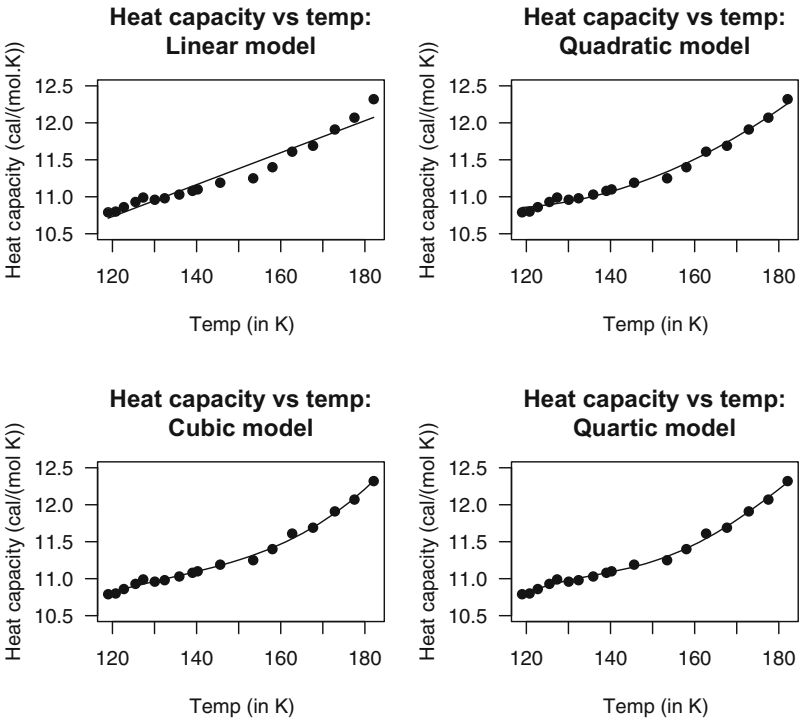

Fig. 3.18 Four models fitted to the heat capacity data (Example 3.15)

```
> coef( hc.m1 )
  (Intercept) poly(Temp, 1)
    11.275556      1.840909
> coef( hc.m2 )
   (Intercept) poly(Temp, 2)1 poly(Temp, 2)2
     11.275556       1.840909       0.396890
> coef( hc.m3 )
   (Intercept) poly(Temp, 3)1 poly(Temp, 3)2 poly(Temp, 3)3
    11.2755556      1.8409086      0.3968900      0.1405174
```

Significance tests show that the fourth order coefficient is not required, so the third-order polynomial is sufficient (Fig. 3.18):

```
> printCoefmat(coef(summary(hc.m4)))
                Estimate Std. Error   t value  Pr(>|t|)
(Intercept)    11.2755556  0.0077737 1450.4766 < 2.2e-16 ***
poly(Temp, 4)1  1.8409086  0.0329810   55.8173 < 2.2e-16 ***
poly(Temp, 4)2  0.3968900  0.0329810   12.0339  2.02e-08 ***
poly(Temp, 4)3  0.1405174  0.0329810    4.2606 0.0009288 ***
poly(Temp, 4)4 -0.0556088  0.0329810   -1.6861 0.1156150
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

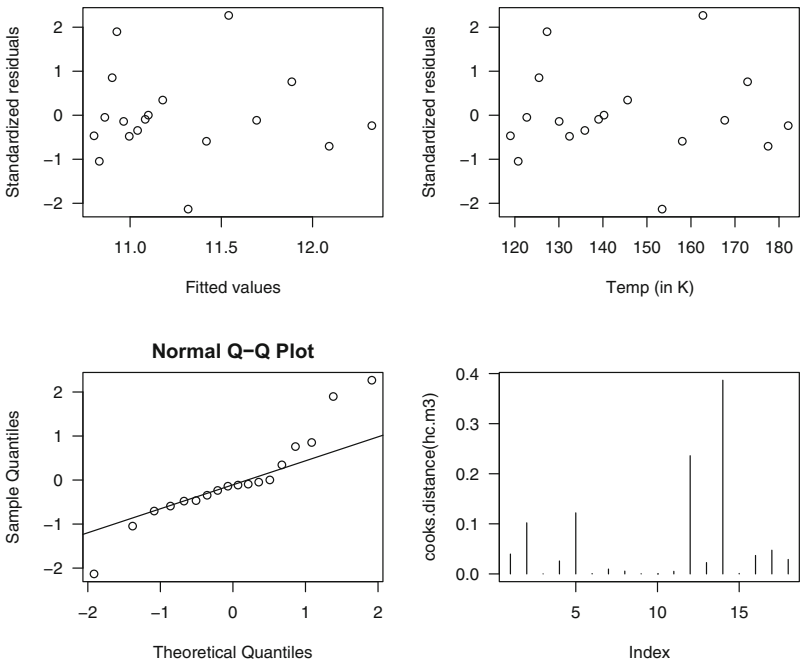

Fig. 3.19 The diagnostic plots for the third-order polynomial model fitted to the heat capacity data (Example 3.15)

The diagnostics suggest no major problems with the cubic model, though normality is perhaps suspect (Fig. 3.19):

```
> plot( rstandard(hc.m3) ~ fitted(hc.m3), las=1,
    ylab="Standardized residuals", xlab="Fitted values" )
> plot( rstandard(hc.m3) ~ heatcap$Temp, las=1,
    ylab="Standardized residuals", xlab="Temp (in K)" )
> qqnorm( rstandard( hc.m3 ), las=1 ); qqline( rstandard( hc.m3 ) )
> plot( cooks.distance(hc.m3), type="h", las=1)
```

□

3.12 Regression Splines

A more flexible alternative to polynomial trends is to fit a general-purpose smooth curve which can take almost any shape. The simplest way to do this is to use *regression splines*. Splines provide an objective and flexible means to fit general but unknown curves.

A spline represents the relationship between y and x as a *series* of polynomials, usually cubic polynomials, joined together at locations called *knots*, in such a way to ensure a continuous relationship and continuous first and second derivatives (to ensure the polynomials join smoothly). The number of

polynomials to join together, and the degree of those polynomials (quadratic, cubic, and so on) can be chosen by the user, depending on the type of spline used. For each spline, the fit is local to a subset of the observations; fewer polynomials means a smoother curve and a simpler model.

The simplest approach to specify a spline curve is to specify a convenient number of knots, depending on the complexity of the curve required, then fit the spline curve to the data by least squares. This approach is called *regression splines*. It is a type of linear regression with specially chosen covariates that serve as a basis for the fitted cubic polynomial curve. The number of regression coefficients used to a fit a regression spline is known as the *degrees of freedom* of the curve. The higher the degrees of freedom, the more complex the trend that the curve can follow.

In R, splines may be fitted using either `bs()` or `ns()`, both in the R package **splines** which comes with R distributions. The function `ns()` fits *natural* cubic splines, which are splines with the second derivatives forced to zero at the endpoints of the given interval, which are by default at the minimum and maximum values of x. For a natural cubic spline, the degrees of freedom are one more than the number of knots. `bs()` generates a B-spline basis for a cubic spline. For a cubic B-spline, the degrees of freedom is one plus the number of knots including the boundary knots at the minimum and maximum values of x; in other words the number of internal knots plus three.

For either `bs()` or `ns()`, the complexity of the fitted curve can be specified by specifying the degrees of freedom or by explicitly specifying the locations of the (internal) knots. The number of degrees of freedom is given using `df`. For `bs()`, the number of internal knots is `df` − `degree` under the default settings, where `degree` is the degree of the polynomial (three by default). For `ns()`, the number of internal knots is `df` − 1 under the default settings. (This is different to `bs()` since the two functions treat the boundary conditions differently.)

The location of the knots is given using the input `knots`. A common way to do this is to use, for example,

```
bs(Temp, knots=quantile(Temp, c(.3, 0.6)), degree=2),
```

where the construct `quantile(Temp, c(0.3, 0.6 )` locates the knots at the 30% and 60% quantiles of the data. (The Q% quantile is that value larger than Q% of the observations.) By default, the knots are chosen at the quantiles of x corresponding to equally spaced proportions.

Natural smoothing splines are linear at the end points, and hence can be extrapolated in a predictable way outside the interval of the data used to estimate the curve, unlike polynomials or B-splines which have relatively unpredictable behaviour outside the interval. For this reason, natural smoothing splines are a good practical choice in most cases for fitting data-driven curves.

Example 3.16. Consider fitting splines to the heat capacity data set (Example 3.15; data set: `heatcap`). Fit a B-spline of `degree=3` (that is, cubic) and a natural cubic spline. Compare to the cubic polynomial fitted using `poly()` chosen in Sect. 3.15 (p. 128), and use the same number of degrees of freedom for all models:

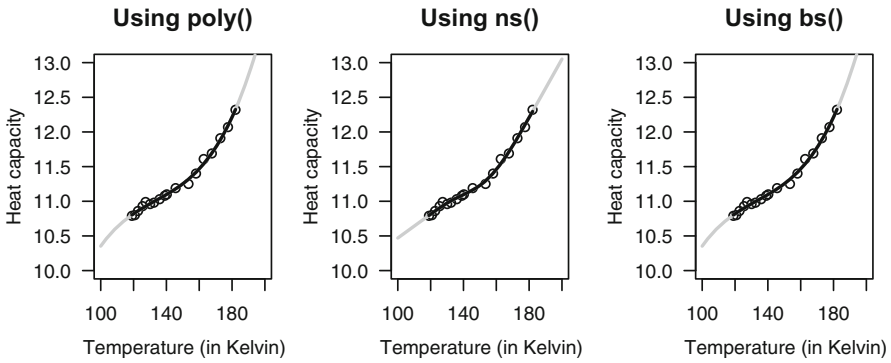

Fig. 3.20 The three cubic models fitted to the heat capacity data (Example 3.16)

```
> library(splines)
> lm.poly <- lm( Cp ~ poly(Temp, 3),  data=heatcap )
> lm.ns   <- lm( Cp ~ ns(Temp, df=3), data=heatcap )
> lm.bs   <- lm( Cp ~ bs(Temp, df=3), data=heatcap )
```

The models are not nested, so we use the AIC to compare the models:

```
> extractAIC(lm.poly); extractAIC(lm.ns); extractAIC(lm.bs)
[1]    4.0000 -117.1234
[1]    4.0000 -119.2705
[1]    4.0000 -117.1234
```

The first output from `extractAIC()` indicates that all models use the same effective number of parameters and so have the same level of complexity. Of these three models, `lm.ns` has the smallest (closest to $-\infty$) AIC. The fitted models (Fig. 3.20) are reasonably similar over the range of the data as expected. However, the behaviour of `ns()` near the endpoints is different. Recall `ns()` fits natural cubic splines, forcing the second derivatives to zero at the endpoints (Fig. 3.20, centre panel). □

Example 3.17. As more cubic polynomials are joined together in the spline curve (and hence each is fitted to fewer observations), the fitted models become more complex. Figure 3.21 is constructed using natural cubic splines and the function `ns()`, but the fitted splines are almost identical to those produced with `bs()` and the same degrees of freedom. The dashed vertical lines show the location of the knots partitioning the data; a cubic polynomial is fitted in each partition. By default the knots are located so that approximately equal numbers of observations are between the knots, so where the data are more concentrated around smaller values of `Temp` the knots are closer together. □

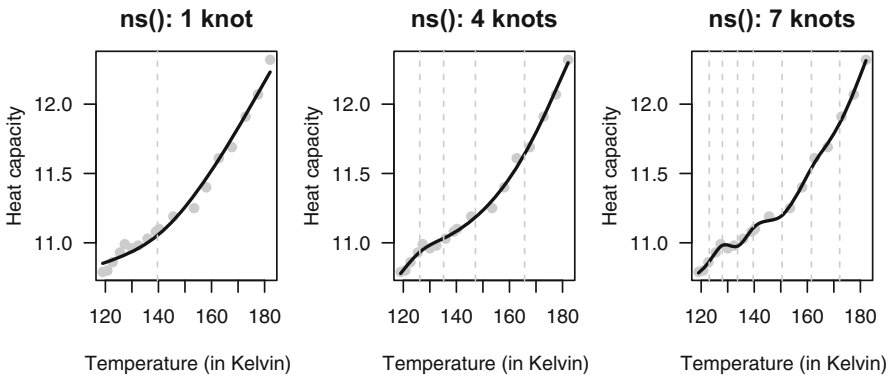

Fig. 3.21 The heat capacity data, plotting C_p against temperature using natural cubic splines `ns()`. The dashed vertical lines are the locations of the knots on the horizontal axis (Example 3.17)

3.13 Fixing Identified Outliers

After applying remedies to ensure linearity and constant variance, observations previously identified as outliers or as influential may no longer be identified as such. Sometimes outliers or influential observations do remain, however, or new ones may become apparent.

The first step in dealing with outliers is to try to identify their cause. This will lead to one of following conclusions:

- The observation is a known mistake. For example, too much herbicide was accidentally used, the operator made a mistake using the machine, or the observation was simply mis-recorded.
- The observation is known to come from a different population. For example, in an analysis of hospital admission rates, the outlier turns out on closer examination to correspond to a hospital much larger than others in the study.
- There is no known reason for why the observation might be an outlier.

When the outlier arises from an identifiable mistake, the ideal solution is obviously to correct the mistake. For example, if a number was mis-recorded and the correct value can still be recovered, then the data can be repaired. If the mistake cannot be corrected, for example because it would require re-running the experiment, then the offending observation can be discarded.

This assumes that the occurrence of the mistake did not depend on the value of the observation. If, for example, mistakes are more common for larger values of the response than for smaller values, after a machine has been run for some time perhaps, then more complex considerations come into play. Little and Rubin [22] consider to what extent missing data or errors can be accommodated into a statistical analysis when the errors depend on the response variable of interest.

If the outlier arises from a different population (such as 'large hospitals') than the rest of the observations ('small- and medium-sized hospitals'), then again the outlier may safely be discarded. Any reporting of the results must make it clear that the results do not apply to large hospitals, since that population of hospitals is not represented in the analysis. If there are a number of observations from the secondary population ('large hospitals'), not just one or two, then the model might be augmented to allow separate parameter values for the two populations, so that these observations could be retained.

When the cause of an outlier cannot be identified, the analyst is faced with a dilemma. Simply discarding the observation is often unwise, since that observation may be a real, genuine observation for which an alternative model would be appropriate. An outlier that is not a mistake suggests that a different or more complex model may be necessary. One strategy to evaluate the influence of the outlier is to fit the model to the data with and without the outlier. If the two models produce similar interpretations and conclusions for the researcher, then the outlier is unimportant, whether discarded or not. If the two models are materially different, perhaps other types of models should be considered. At the very least, note the observation and discuss the effect of the observation on the model.

3.14 Collinearity

Collinearity, sometimes called *multicollinearity*, occurs when some of the covariates are highly correlated with each other, implying that they measure almost the same information.

Collinearity means that different combinations of the covariates may lead to nearly the same fitted values. Collinearity is therefore mainly a problem for interpretation rather than prediction (Sect. 1.9). Very strong collinearity can theoretically cause numerical problems during the model fitting, but this is seldom a problem in practice with modern numerical software. Collinearity does cause the estimated regression coefficients to be highly dependent on other variables in the linear predictor, making direct interpretation virtually impossible.

A symptom of collinearity is that the standard errors of the affected regression coefficients become large. If two covariates are very highly correlated, typically only one of them needs to be retained in the model, but either one would do equally well from a statistical point of view. In these cases, there will exist many different linear predictors all of which compute virtually the same predictions, but with quite different coefficients for individual variables. Collinearity means that separating causal variables from associated (passenger) variables is especially difficult, perhaps impossible.

Collinearity is most easily identified by examining the correlations between the covariates. Correlations close to one in absolute value are of concern. Other methods also exist for identifying collinearity.

A special case of collinearity occurs when a covariate and a power of the covariate are included in the same model, such as x and x^2 (Example 3.15): x and x^2 are almost inevitably highly correlated. Using orthogonal polynomials or regression splines (Sect. 3.12) avoids this problem.

If collinearity is detected or suspected, remedies include:

- Omitting some explanatory variables from the analysis, since collinearity implies the explanatory variables contain almost the same information. Favour omitting explanatory variables with less theoretical basis for belonging in the model, whose interpretation is less clear, or are harder to collect or measure. However, in practice, researchers tend to be reluctant to throw away data.
- Combine explanatory variables in the model provided the combination makes sense. For example, if height and weight are highly correlated, consider combining the explanatory variables as the body mass index, or BMI, and use this explanatory variable in the model in place of height and weight. (BMI is weight (in kg), divided by the square of height (in m).)
- Collect more data, if there are observations that can be made that better distinguish the correlated covariates. Sometimes the covariates are intrinsically correlated, so collinearity is difficult to remove regardless of data collection.
- Use special methods, such as ridge regression [39, §11.2], which are beyond the scope of this book.

Example 3.18. The monthly maintenance hours associated with maintaining the anaesthesiology service for twelve naval hospitals in the USA was collected (Table 3.5; data set: `nhospital`) together with some possible explanatory variables [26]. All explanatory variables appear strongly related to the response (Fig. 3.22):

Table 3.5 Naval hospital maintenance data. `MainHours` is the monthly maintenance hours; `Eligible` is the eligible population per thousand; `OpRooms` is the number of operating rooms; `Cases` is the number of surgical cases (Example 3.18)

MainHours	Eligible	OpRooms	Cases	MainHours	Eligible	OpRooms	Cases
304.37	25.5	4	89	383.78	43.4	4	82
2616.32	294.3	11	513	2174.27	165.2	10	427
1139.12	83.7	4	231	845.30	74.3	4	193
285.43	30.7	2	68	1125.28	60.8	5	224
1413.77	129.8	6	319	3462.60	319.2	12	729
1555.68	180.8	6	276	3682.33	376.2	12	951

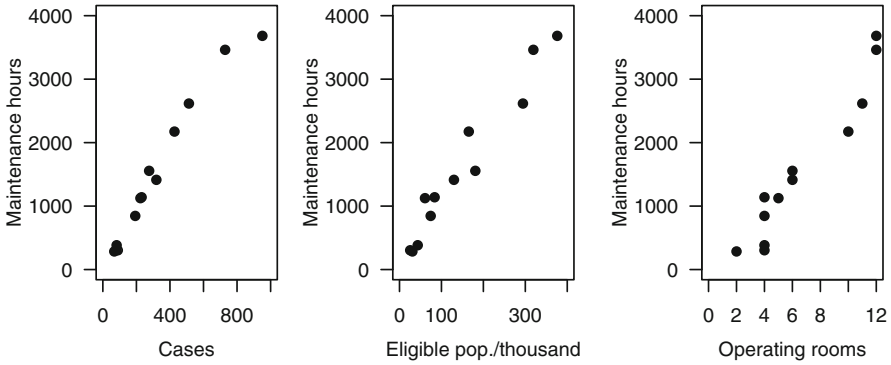

Fig. 3.22 Plots of the naval hospital data (Example 3.18)

```
> data(nhospital); names(nhospital)
[1] "Cases"     "Eligible"  "OpRooms"   "MainHours"
> plot( MainHours~Cases, data=nhospital, las=1, pch=19,
    ylim=c(0, 4000), xlim=c(0, 1000),
    xlab="Cases", ylab="Maintenance hours")
> plot( MainHours~Eligible, data=nhospital, las=1, pch=19,
    ylim=c(0, 4000), xlim=c(0, 400),
    xlab="Eligible pop./thousand", ylab="Maintenance hours")
> plot( MainHours~OpRooms, data=nhospital, las=1, pch=19,
    ylim=c(0, 4000), xlim=c(0, 12),
    xlab="Operating rooms", ylab="Maintenance hours")
```

The variables are all highly correlated:

```
> cor( nhospital)
              Cases  Eligible   OpRooms MainHours
Cases     1.0000000 0.9602926 0.9264237 0.9802365
Eligible  0.9602926 1.0000000 0.9399181 0.9749010
OpRooms   0.9264237 0.9399181 1.0000000 0.9630730
MainHours 0.9802365 0.9749010 0.9630730 1.0000000
```

The correlations are all very close to one, implying many models exists which give very similar predictions (Problem 3.7).

Consider fitting the model:

```
> nh.m1 <- lm( MainHours ~ Eligible + OpRooms + Cases, data=nhospital)
```

Since the correlations are very high between the response and explanatory variables, strong relationships between `MainHours` and each covariate are expected after fitting the model. However, the results of the t-tests for this model show no evidence of strong relationships:

```
> printCoefmat( coef( summary( nh.m1 ) ) )
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -114.58953  130.33919 -0.8792  0.40494
Eligible       2.27138    1.68197  1.3504  0.21384
```

```
OpRooms       99.72542   42.21579  2.3623  0.04580 *
Cases          2.03154    0.67779  2.9973  0.01714 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The t-tests suggest `OpRooms` and `Cases` are mildly significant in the model after adjusting for `Eligible`, but `Eligible` is not significant after adjusting for the other explanatory variables. In contrast, consider the sequential ANOVA F-tests:

```
> anova( nh.m1 )
Analysis of Variance Table

Response: MainHours
          Df   Sum Sq  Mean Sq  F value    Pr(>F)
Eligible   1 14346071 14346071 523.7574 1.409e-08 ***
OpRooms    1   282990   282990  10.3316   0.01234 *
Cases      1   246076   246076   8.9839   0.01714 *
Residuals  8   219125    27391
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

In the ANOVA table, `Eligible` is highly significant, and shows a very small P-value. Since these F-tests are *sequential*, this test has not adjusted for any other explanatory variable, so the result is strong as expected. After `Eligible` is in the model, the other explanatory variable have little contribution to make because the explanatory variables are highly correlated. □

3.15 Case Studies

3.15.1 Case Study 1

Consider the DMFT data (data set: `dental`) first seen in Sect. 2.13 (p. 76). In that section, the model fitted to the data was:

```
> data(dental)
> dental.lm <- lm( DMFT ~ Sugar * Indus, data=dental)
```

Consider some diagnostic plots (Fig. 3.23, top panels):

```
> scatter.smooth( rstandard(dental.lm) ~ fitted(dental.lm),
    xlab="Fitted values", ylab="Standardized residuals", las=1)
> qqnorm( rstandard( dental.lm ), las=1 ); qqline( rstandard( dental.lm ) )
> plot( cooks.distance(dental.lm), type="h", las=1)
```

The plots are acceptable, though the Q–Q plot is not ideal. However, one observation has a large residual of $r' = 3.88$ (top left panel; top centre panel).

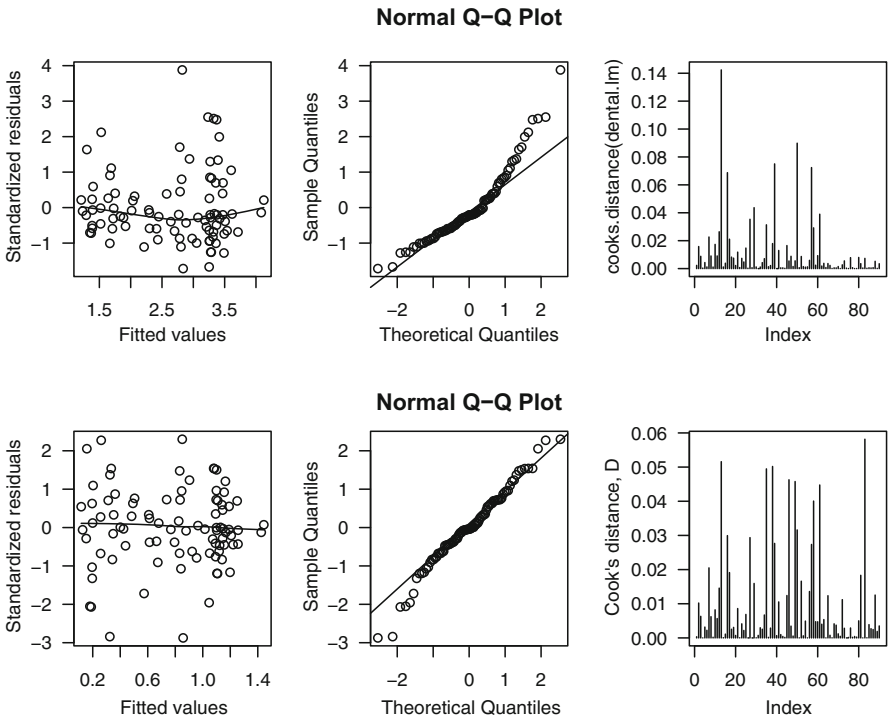

Fig. 3.23 Diagnostic plots of the model fitted to the DMFT data. Top panels: using DMFT as the response; bottom panels: using the logarithm of DMFT as the response (Sect. 3.15.1)

The influence diagnostics reveal that two observations are influential according to DFFITS, but none are influential according to Cook's distance or DFBETAS:

```
> im <- influence.measures(dental.lm)
> colSums(im$is.inf)
  dfb.1_ dfb.Sugr dfb.InNI dfb.S:IN    dffit    cov.r   cook.d      hat
       0        0        0        0        2       11        0        2
```

DMFT is a strictly positive response variable that varies over an order of magnitude between countries, so a log-transformation may well be helpful:

```
> dental.lm.log <- update(dental.lm, log(DMFT) ~ .)
> anova(dental.lm.log)
Analysis of Variance Table

Response: log(DMFT)
            Df  Sum Sq Mean Sq F value    Pr(>F)
Sugar        1 10.9773 10.9773 36.8605 3.332e-08 ***
Indus        1  0.6183  0.6183  2.0761   0.15326
Sugar:Indus  1  1.3772  1.3772  4.6245   0.03432 *
```

```
Residuals   86 25.6113  0.2978
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Now examine the diagnostics of this new model (Fig. 3.23, bottom panels):

```
> scatter.smooth( rstandard(dental.lm.log) ~ fitted(dental.lm.log),
    xlab="Fitted values", ylab="Standardized residuals", las=1)
> qqnorm( rs <- rstandard( dental.lm.log ), las=1 ); qqline( rs )
> plot( cooks.distance(dental.lm.log), type="h", las=1,
    ylab="Cook's distance, D")
```

Each diagnostic plot is improved: the variance of the standardized residuals appears approximately constant and the slight curvature is gone; the residuals appear more normally distributed; and the largest absolute residual is much smaller. Furthermore, the two observations identified as influential according to DFFITS are no longer declared influential:

```
> im <- influence.measures(dental.lm.log); colSums(im$is.inf)
  dfb.1_ dfb.Sugr dfb.InNI dfb.S:IN    dffit    cov.r   cook.d      hat
       0        0        0        0        0       11        0        2
```

The final model is:

```
> printCoefmat(coef( summary(dental.lm.log)) )
                    Estimate Std. Error t value Pr(>|t|)
(Intercept)        1.3871066  0.5102055  2.7187 0.007926 **
Sugar             -0.0058798  0.0119543 -0.4919 0.624075
IndusNonInd       -1.2916000  0.5253985 -2.4583 0.015964 *
Sugar:IndusNonInd  0.0272742  0.0126829  2.1505 0.034325 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

`Sugar` is retained due to the marginality principle. The two fitted models are shown in Fig. 3.24. The model can be written as

$$\begin{cases} y_i \sim N(\mu_i, s^2 = 0.298) \\ \mu_i = 1.387 - 0.005880x_1 - 1.292x_2 + 0.02727x_1x_2, \end{cases}$$

where $\mathrm{E}[\log y_i] = \mu_i$, x_1 is the mean annual sugar consumption (in kg/person/year) and $x_2 = 1$ for industrialized countries (and is 0 otherwise). More directly, the systematic component is

$$\mathrm{E}[\log y_i] = \mu_i = \begin{cases} 1.387 \quad - 0.005880x_1 \text{ for industrialized countries} \\ 0.09551 + 0.02139x_1 \quad \text{for non-industrialized countries.} \end{cases}$$

The two models (using the response as `DMFT` or `log(DMFT)`) can be compared using the AIC and BIC:

```
> # AIC
> c( "AIC (DMFT)" = extractAIC(dental.lm)[2],
     "AIC (log-DMFT)" = extractAIC(dental.lm.log)[2] )
```

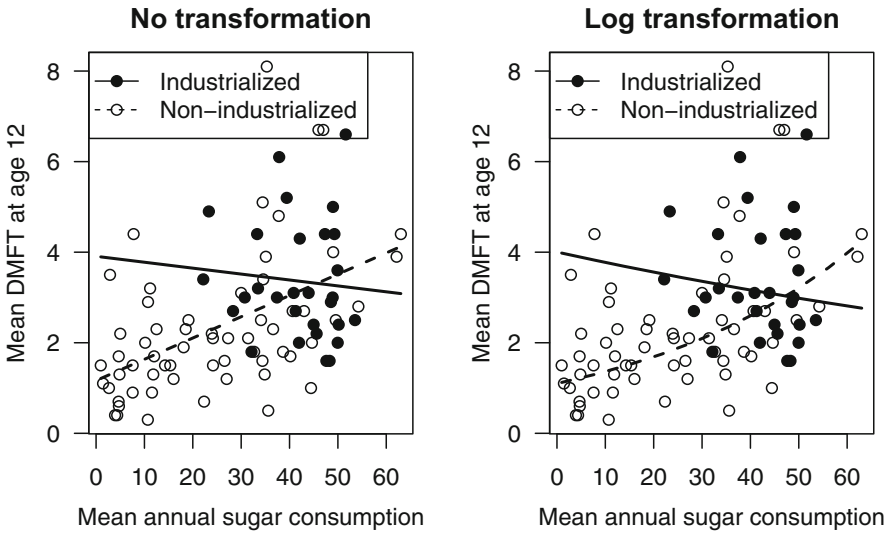

Fig. 3.24 Two models fitted to the DMFT data. Left panel: using DMFT as the response; right panel: using the logarithm of DMFT as the response (Sect. 3.15.1)

```
    AIC (DMFT) AIC (log-DMFT)
      61.36621     -105.10967
> # BIC
> k <- nobs(dental.lm)  # The penalty to compute the BIC
> c( "BIC (DMFT)" = extractAIC(dental.lm,     k=k )[2],
     "BIC (log-DMFT)" = extractAIC(dental.lm.log, k=k )[2])
    BIC (DMFT) BIC (log-DMFT)
      413.3662       246.8903
```

In both cases, the model using `log(DMFT)` as the response variable is preferred.

For industrialized countries, the mean number of DMFT at age 12 increases approximately by a factor of $\exp(-0.005880) = 0.9941$ for each 1 kg/person/year increase in sugar consumption, which is not statistically significant. For non-industrialized countries, the mean number of DMFT at age 12 increases by approximately a factor of $\exp(0.02139) = 1.022$ for each 1 kg/person/year increase in sugar consumption.

The limitations in the study (identified in Sec. 2.13) remain, though the fitted model is now slightly better according to the diagnostics.

3.15.2 Case Study 2

To understand the how the chemical composition of cheese is related to its taste, a study [25, 34] from the La Trobe Valley in Victoria (Australia) had

Table 3.6 The chemical composition and tastes of samples of cheddar cheese (Sect. 3.15.2)

Taste	Acetic	H2S	Lactic	Taste	Acetic	H2S	Lactic
12.3	94	23	0.86	40.9	581	14,589	1.74
20.9	174	155	1.53	15.9	120	50	1.16
39.0	214	230	1.57	6.4	224	110	1.49
47.9	317	1801	1.81	18.0	190	480	1.63
5.6	106	45	0.99	38.9	230	8639	1.99
25.9	298	2000	1.09	14.0	96	141	1.15
37.3	362	6161	1.29	15.2	200	185	1.33
21.9	436	2881	1.78	32.0	234	10,322	1.44
18.1	134	47	1.29	56.7	349	26,876	2.01
21.0	189	65	1.58	16.8	214	39	1.31
34.9	311	465	1.68	11.6	421	25	1.46
57.2	630	2719	1.90	26.5	638	1056	1.72
0.7	88	20	1.06	0.7	206	50	1.25
25.9	188	140	1.30	13.4	331	800	1.08
54.9	469	856	1.52	5.5	481	120	1.25

samples of cheddar cheese chemically analysed. For each cheese, the acetic acid concentration (`Acetic`), the lactic acid concentration (`Lactic`), and the H_2S concentration (`H2S`) were measured. The cheeses were also scored for their taste (Table 3.6; data set: `cheese`), and the final `Taste` score combines the taste scores from several judges.

Plotting the response `Taste` against the explanatory variables shows possible relationships between the variables (Fig. 3.25):

```
> data(cheese); names(cheese)
[1] "Taste"  "Acetic" "H2S"    "Lactic"
> plot( Taste ~ Acetic, data=cheese, las=1, pch=19,
    xlab="Acetic acid concentration", ylab="Taste score")
> plot( Taste ~ H2S, data=cheese, las=1, pch=19,
    xlab="H2S concentration", ylab="Taste score")
> plot( Taste ~ Lactic, data=cheese, las=1, pch=19,
    xlab="Lactic acid concentration", ylab="Taste score")
```

First consider the variance of y. The plot of `Taste` against `Lactic` shows little evidence of non-constant variance (Fig. 3.25, bottom left panel); the plot of `Taste` against `Acetic` suggests the variance slightly increases as the mean taste score increases (top left panel). The plot of `Taste` against `H2S` is difficult to interpret (top right panel) as most values of `H2S` are small, but some are very large.

The relationships between `Taste` and `Acetic`, and also between `Taste` and `Lactic`, appear approximately linear. The relationship between `Taste` against `H2S` is non-linear, and the observations with large values of `H2S` will

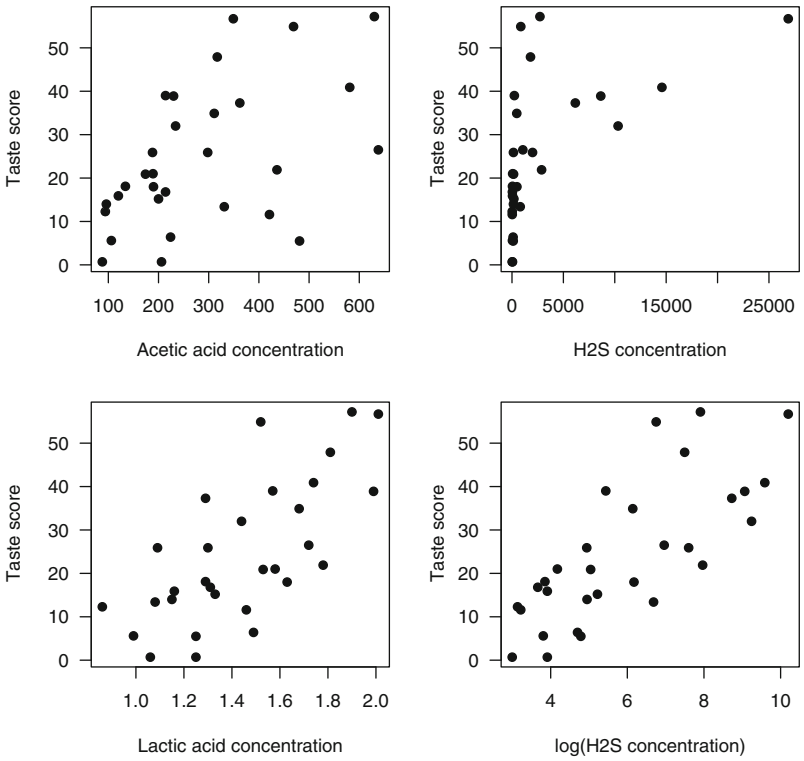

Fig. 3.25 The cheese data. The mean taste score plotted against the acetic acid concentration (top left panel); the mean taste score plotted against the H_2S concentration (top right panel); the mean taste score plotted against the lactic acid concentration (bottom left panel); the mean taste score plotted against the logarithm of H_2S concentration (bottom right panel) (Sect. 3.15.2)

certainly be influential. Since `H2S` covers many orders of magnitude (from 20 to 26,880), consider taking logarithms (Fig. 3.25, bottom right panel):

```
> plot( Taste ~ log(H2S), data=cheese, las=1, pch=19,
    xlab="log(H2S concentration)", ylab="Taste score")
```

The relationship between `Taste` and `log(H2S)` now appears approximately linear. The variance of `Taste` appears to be slightly increasing as `log(H2S)` increases. Some, but not all, evidence suggests the variation is slightly increasing for increasing taste scores. For the moment, we retain `Taste` as the response without transforming, and examine the diagnostics to determine if a transformation is necessary.

Begin with the full model, including all interactions:

```
> cheese.m1 <- lm( Taste ~ Acetic * log(H2S) * Lactic, data=cheese )
> drop1(cheese.m1, test="F")
Single term deletions
```

```
Model:
Taste ~ Acetic * log(H2S) * Lactic
                      Df Sum of Sq    RSS    AIC F value Pr(>F)
<none>                               2452.3 148.11
Acetic:log(H2S):Lactic  1    36.467 2488.8 146.55  0.3272 0.5731
```

The three-way interaction is not needed. Then consider dropping each two-way interaction in turn:

```
> cheese.m2 <- update( cheese.m1, . ~ (Acetic + log(H2S): + Lactic)^2 )
> drop1(cheese.m2, test="F")
Single term deletions

Model:
Taste ~ Acetic + log(H2S):Lactic + Acetic:log(H2S):Lactic
                      Df Sum of Sq    RSS    AIC F value Pr(>F)
<none>                               2679.1 142.76
Acetic:log(H2S):Lactic  1    24.269 2703.4 141.03  0.2355 0.6315
```

No two-way interactions are needed either. Finally, consider dropping each main effect term:

```
> cheese.m3 <- lm( Taste ~ log(H2S) + Lactic + Acetic, data=cheese )
> drop1(cheese.m3, test="F")
Single term deletions

Model:
Taste ~ log(H2S) + Lactic + Acetic
         Df Sum of Sq    RSS    AIC F value   Pr(>F)
<none>                2660.9 142.56
log(H2S)  1   1012.39 3673.3 150.23  9.8922 0.004126 **
Lactic    1    527.53 3188.4 145.98  5.1546 0.031706 *
Acetic    1      8.05 2668.9 140.65  0.0787 0.781291
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The most suitable model appears to be:

```
> cheese.m4 <- lm( Taste ~ log(H2S) + Lactic, data=cheese )
> coef( summary(cheese.m4) )
             Estimate Std. Error   t value    Pr(>|t|)
(Intercept) -27.591089   8.981801 -3.071888 0.004813785
log(H2S)      3.946425   1.135722  3.474817 0.001742652
Lactic       19.885953   7.959175  2.498494 0.018858866
```

While all three covariates appear associated with `Taste` (Fig. 3.25, p. 143), only two are necessary in the model. This implies the covariates are correlated:

```
> with(cheese, cor( cbind(Taste, Acetic, logH2S=log(H2S), Lactic) ) )
          Taste    Acetic    logH2S    Lactic
Taste  1.0000000 0.5131983 0.7557637 0.7042362
Acetic 0.5131983 1.0000000 0.5548159 0.5410837
logH2S 0.7557637 0.5548159 1.0000000 0.6448351
Lactic 0.7042362 0.5410837 0.6448351 1.0000000
```

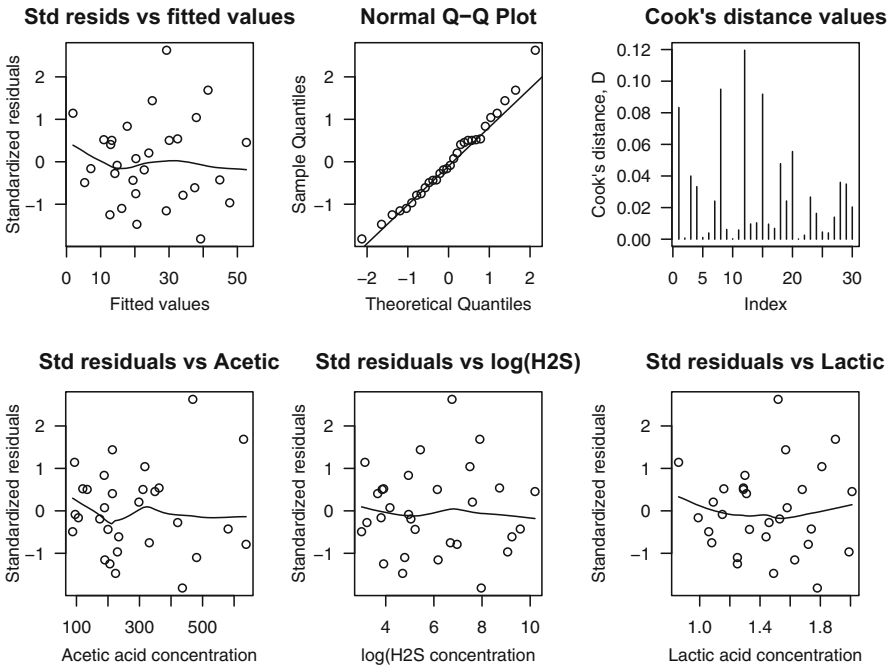

Fig. 3.26 The diagnostics from the model fitted to the cheese-tasting data (Sect. 3.15.2)

Clearly, the relationships between `Taste` and `Lactic`, and between `Taste` and `log(H2S)`, are stronger than that between `Taste` and `Acetic`. Furthermore, `Acetic` is correlated with both `Lactic` and `log(H2S)`, so once `Lactic` and `log(H2S)` are in the model `Acetic` has almost nothing further to contribute:

```
> cor( cbind(rstandard(cheese.m3), cheese$Acetic))
             [,1]         [,2]
[1,]  1.000000000 -0.002230637
[2,] -0.002230637  1.000000000
```

Consider the diagnostics of the final model (Fig. 3.26):

```
> scatter.smooth( rstandard(cheese.m4) ~ fitted(cheese.m4), las=1,
    main="Std resids vs fitted values",
    xlab="Fitted values", ylab="Standardized residuals")
> qqnorm( rstandard(cheese.m4), las=1); qqline( rstandard(cheese.m4) )
> plot( cooks.distance(cheese.m4), type="h", las=1,
    main="Cook's distance values", ylab="Cook's distance, D")
> scatter.smooth( rstandard(cheese.m4) ~ cheese$Acetic,
    main="Std residuals vs Acetic", las=1,
    xlab="Acetic acid concentration", ylab="Standardized residuals")
> scatter.smooth( rstandard(cheese.m4) ~ log(cheese$H2S),
    main="Std residuals vs log(H2S)", las=1,
    xlab="log(H2S concentration", ylab="Standardized residuals")
```

```
> scatter.smooth( rstandard(cheese.m4) ~ cheese$Lactic,
    main="Std residuals vs Lactic", las=1,
    xlab="Lactic acid concentration", ylab="Standardized residuals")
```

The model diagnostics suggest the model `cheese.m4` is adequate, although a single observation with a standardized residual just larger than 2 makes the variance appear larger in the centre of some plots. No observation appears substantially more influential than the others based on the Cook's distance, DFFITS or DFBETAS:

```
> im <- influence.measures(cheese.m4); colSums(im$is.inf)
  dfb.1_ dfb.l(H2 dfb.Lctc    dffit    cov.r   cook.d      hat
       0        0        0        0        4        0        0
```

The fitted model `cheese.m4` shows that the taste improves, on average, with increasing concentrations of lactic acid and H_2S. Because of the high correlations between `Lactic` and `H2S`, interpreting the individual contributions of each chemical to the taste is not straightforward.

3.16 Using R for Diagnostic Analysis of Linear Regression Models

An introduction to using R is given in Appendix A. For fitting linear regression models, the function `lm()` is used (see Sect. 2.14, p. 79 for more on the use of `lm()`). This section summarizes and collates R commands relevant to diagnostic analysis of linear regression models.

Three types of residuals may be computed from a fitted model, say `fit`, using R:

- Raw residuals (Sect. 3.3): Use `resid(fit)` or `residuals(fit)`.
- Standardized residuals r' (Sect. 3.3): Use `rstandard(fit)`.
- Studentized residuals r'' (Sect. 3.6.2): Use `rstudent(fit)`.

Different measures of influence may be computed in R (Sect. 3.6.3):

- Cook's distance D: Use `cooks.distance(fit)`.
- DFBETAS: Use `dfbetas(fit)`.
- DFFITS: Use `dffits(fit)`.
- Covariance ratio CR: Use `covratio(fit)`.

All these measures of influence, together with the leverages h, are returned using `influence.measures(fit)`. Observations of potential interest are flagged according to the criteria explained in Sect. 3.6.3 (p. 110). Other useful R commands for diagnostics analysis include:

- Q–Q plots: Use `qqnorm()`, where the input is a function to produce residuals from a fitted model `fit`, such as `rstandard(fit)`. Add a reference line by following the `qqnorm()` call with `qqline()` with the same input.
- Fitted values $\hat{\mu}$: Use `fitted(fit)`.
- Leverages h: Use `hatvalues(fit)`.

A fitted model can be plotted also; for example:

```
> model <- lm( y ~ x);  plot( model )
```

These commands produce four residual plots by default; see `?plot.lm`.

R commands useful for remedying problems include:

- The `poly()` function (Sect. 3.12) is used to add orthogonal polynomials to the systematic component. To use `poly()`, supply the name of the covariate `x`, and the `degree` of the polynomial to fit. Typical use: `poly(Ht, degree=4)` which fits a quartic in `Ht`.
- The spline functions `ns()` (to fit natural cubic splines) and `bs()` (to fit splines of any degree) are in package **splines** which comes with R (Sect. 3.12).
  To use `ns()`, supply the name of the covariate, and either the degrees of freedom using `df` or the location of the internal knots using `knots`. Typical use: `ns(Ht, df=3)`, which fits a natural cubic spline with three degrees of freedom.
  To use `bs()`, supply the name of the covariate, the `degree` of the polynomials to use, and either the degrees of freedom using `df` or the location of the internal knots using `knots`. Typical use: `bs(Ht, df=3, degree=2)`, which fits quadratic splines with three degrees of freedom.
- Transformations of the responses (Sect. 3.9) or the covariates (Sect. 3.10) are computed using standard R functions, such as `sqrt(x)`, `log(y)`, `1/x`, `asin(sqrt(y))`, and `y^(-2)`. When used with covariates in `lm()`, the transformation should be insulated using `I()`; for example, `I(1/x)`.
- The Box–Cox transformation may be chosen using the `boxcox()` function in package **MASS** (which comes with R), designed to identify the transformation most suitable for achieving linearity, normality and constant variance simultaneously. Typical use: `boxcox(FEV ~ Age + Ht + Gender + Smoke)`.

3.17 Summary

Chapter 3 discusses methods for identifying possible violations of assumptions in multiple regression models, and remedying these issues. The assumptions for linear regression models are, in order of importance (Sect. 3.2):

- Lack of outliers: The model is appropriate for all observations.
- Linearity: The linear predictor captures the true relationship between μ_i and the explanatory variables, and all important explanatory variables are included.
- Constant variance: The responses y_i have *constant* variance, apart from known weights w_i.
- Independence: The responses y_i are *independent* of each other.

In addition, *normal* linear regression models assume the responses y come from a normal distribution.

Diagnostic analysis is used to identify any deviations from these assumptions that are likely to affect conclusions (Sect. 3.2), and the main tool for diagnostic analysis is residuals. The three main types of residuals (Sects. 3.3 and 3.6.2) are raw residuals r_i, standardized residuals r_i', and Studentized residuals r_i''. The standardized and Studentized residuals have approximately constant variance of one, and are preferred in residual plots for this reason (Sect. 3.3; Sect. 3.6.2). The terminology used for residuals is confusingly inconsistent (Sect. 3.7). In addition to residuals, the leverages h_i identify unusual combinations of the explanatory variable (Sects. 3.4).

A strategy for assessing models is (Sect. 3.5):

- Check for independence of the responses when possible. This assumption can be hard to check, as this may be depend on the method of data collection. However, if the data are collected over time, dependence may be identified by plotting residuals against the previous residual in time. Likewise, if the data are spatial, check for dependence by plotting residuals against spatial variables (Sect. 3.5.5).
- Check for linearity between the responses and all covariates using plots of the residuals against each explanatory variable (Sect. 3.5.1). Linearity between the response and explanatory variables after adjusting for the effects of the other explanatory variables can also be assessed using partial residual plots (Sect. 3.5.2).
- Check for constant variance in the response using plots of the residuals against $\hat{\mu}$ (Sect. 3.5.3).
- Check for normality of the responses using a Q–Q plot (Sect. 3.5.4).

Outliers are observations inconsistent with the rest of the observations (Sect. 3.6.2), when the corresponding residuals are unusually large, positive or negative. Outliers should be identified and, if necessary, appropriately managed (Sect. 3.13).

Influential observations are outliers that substantially change the fitted model when omitted from the data set (Sect. 3.6.2). Numerical means for identifying influence include Cook's distance D, DFFITS, DFBETAS, or the covariance ratio CR (Sect. 3.6.3).

Some strategies for solving model weaknesses are (Sect. 3.8):

- If the responses are not independent, use other methods.
- If the variance of the response is not approximately constant, transform y as necessary (Sect. 3.9).
- Then, if the relationship is not linear, transform the covariates using simple transformations (Sect. 3.10), polynomials in the covariates (Sect. 3.11), or regression splines (Sect. 3.12).

Finally, collinearity occurs when at least some of the covariates are highly correlated with each other (Sect. 3.14).

Problems

Selected solutions begin on p. 532. Problems preceded by an asterisk * refer to the optional sections in the text, and may require matrix manipulations.

3.1. The standardized residual r'_i measures the reduction in the RSS (divided by s^2) when Observation i is omitted from the data. Demonstrate this in R using the `lungcap` data as follows.

- Fit the model `LC.lm` (Example 3.1, p. 97). Compute the RSS, s^2 and the standardized residuals from this model.
- Omit observation 1 from `lungcap`, and refit the model without Observation 1. Call this model `LC.omit1`.
- Compute the difference between the RSS for the full model `LC.lm` and for model `LC.omit1`. Show that this difference divided by s^2 is the standardized residuals squared for Observation 1.

Repeat the above process for every observation i, and show that the n differences divided by s^2 are the standardized residuals squared.

*** 3.2.** Consider the hat matrix as defined in (3.3) (p. 101).

1. Show that H is *idempotent*; that is, $\mathrm{H}^2 = \mathrm{H}$.
2. Show that H is *symmetric*; that is, $\mathrm{H}^T = \mathrm{H}$.
3. Show $\mathrm{I}_n - \mathrm{H}$ is idempotent and symmetric.

*** 3.3.** Consider a simple linear regression model, with all prior weights set to one and including a constant term in the linear predictor.

1. Show that
$$h_i = \frac{1}{n} + \frac{(x_i - \bar{x})^2}{\sum_{j=1}^{n}(x_j - \bar{x})^2}.$$
2. Use this expression to show that $h_i \geq (1/n)$
3. Show that $h_i \leq 1$. HINT: Since H is idempotent (Problem 3.2), first show $h_i = \sum_{j=1}^{n} h_{ij}^2 = h_i^2 + \sum_{j \neq i} h_{ij}^2$.

*** 3.4.** Equation (3.6) (p. 110) gives an expression for Cook's distance, which can also be written as

$$D_i = \frac{(\hat{\boldsymbol{\mu}} - \hat{\boldsymbol{\mu}}_{(i)})^T (\hat{\boldsymbol{\mu}} - \hat{\boldsymbol{\mu}}_{(i)})}{p' s^2}. \tag{3.9}$$

Interpret Cook's distance using this form.

3.5. To gain experience reading Q–Q plots, use R to produce Q–Q plots of data known to be generated randomly from a standard normal distribution using `rnorm()`. Generate ten Q–Q plots based on 100 random numbers, and comment on using Q–Q plots when $n = 100$. Repeat the exercise for $n = 50$, 20 and 10, and comment further.

3.6. Show that the partial residual plot for a simple linear regression model is simply a plot of y against x.

3.7. For the naval hospital data (data set: `nhospital`) (Example 3.18, p. 136), fit the three models that contain two of the explanatory variables. Show that the fitted values are very similar for all three models.

3.8. The lung capacity data [21] in Example 1.1 (data set: `lungcap`) have been used often in Chaps. 2 and 3.

1. Fit the model with FEV as the response and smoking status as the only explanatory variable. Interpret the meaning of the coefficient for smoking.
2. Fit the model with FEV as the response and all other variables as explanatory variables (but do not use any interactions). Interpret the coefficient for smoking status.
3. Fit the model with the logarithm of FEV as the response and all other variables as explanatory variables (but do not use any interactions). Interpret the coefficient for smoking status.
4. Determine a suitable model for the data.

3.9. In Chap. 2, the lung capacity data (data set: `lungcap`) was analysed using `log(FEV)` as the response variable, with `Ht` as one of the explanatory variables. In Example 3.13, a model was proposed for analysing `log(FEV)` using `log(Ht)` in place of `Ht` as one of the covariates. Compare these two models using a diagnostic analysis, and comment.

3.10. In Sect. 3.15.2 (p. 141), a model is fitted to the cheese tasting data (data set: `cheese`). However, before fitting this model, the plot of `Taste` against `log(H2S)` suggested slightly non-constant variance. An alternative model might suggest using `log(Taste)` as the response rather than `Taste`. Show that using `log(Taste)` as the response results in a poor model.

3.11. A study [27] compiled information about the food consumption habits of various fish species (data set: `fishfood`). The fitted linear regression model has the form

$$\log \hat{\mu} = \beta_0 + \beta_1 \log \texttt{MaxWt} + \beta_2 \log \texttt{Temp} + \beta_3 \log \texttt{AR} + \beta_4 \texttt{Food},$$

where $\mu = \mathrm{E}[\texttt{FoodCon}]$ is the predicted daily food consumption as a percentage of biomass, $\texttt{F} = 0$ for carnivores, and $\texttt{F} = 1$ for herbivores, and the other variables are defined in Table 3.7.

1. Fit the model used in original study.
2. Perform a diagnostic analysis of this model.
3. Interpret the model.
4. Determine if a better model can be found by considering interaction terms.

Table 3.7 The daily food consumption (as a percentage of biomass) `FoodCon`, maximum weight (in g) `MaxWt`, mean habitat temperature (in °C) `Temp`, aspect ratio `AR`, and food type `Food` (where C means carnivore and H means herbivore) for various fish `Species`. The first six observations are shown (Problem 3.11)

Species	MaxWt	Temp	AR	Food	FoodCon
Brevoortia patronus	362	25	1.69	C	2.22
Brevoortia tyrannus	1216	18	2.31	H	8.61
Engraulis encrasicholus	28	15	1.42	C	2.50
Hygophum proximum	2	25	1.65	C	9.28
Hygophum reindhardtii	1	25	1.05	C	6.66
Lampanyctus alatus	2	25	1.62	C	3.32
⋮	⋮	⋮	⋮	⋮	⋮

Table 3.8 Energy and digestibilities ('Digest.') of diets for sheep (Problem 3.12)

Dry matter digest. (%)	Energy digest. (%)	Digestible energy (cal/gram)	Dry matter digest. (%)	Energy digest. (%)	Digestible energy (cal/gram)
30.5	27.8	1.243	68.5	66.8	3.016
63.0	61.5	2.750	71.6	70.7	3.149
62.8	60.4	2.701	71.5	69.8	3.131
50.0	49.5	2.213	75.4	73.5	3.396
60.3	58.7	2.681	71.7	69.8	3.131
64.1	63.0	2.887	73.2	72.1	3.226
63.7	62.8	2.895	56.6	55.2	2.407
63.4	62.8	2.895	49.7	48.1	2.098
65.4	64.2	2.952	54.7	53.4	2.331
68.1	66.5	3.059	58.7	57.0	2.488
72.1	70.4	3.239	64.3	62.3	2.761
68.8	68.7	3.154	67.7	65.5	2.904
52.8	50.7	2.229	68.3	66.2	2.933
60.3	58.1	2.550	66.4	64.8	2.869
52.8	50.7	2.226	68.1	66.3	2.963
66.1	64.2	2.823	72.2	70.8	3.164
62.5	61.3	2.768	76.3	74.2	3.314
65.8	64.0	2.768	70.4	69.0	3.081

3.12. In a study [24] of the feed of ruminants, the data in Table 3.8 were collected (data set: `ruminant`). The purpose of the study was to model the digestible energy content, and explore the relationships with percentage dry matter digestibility and percentage energy digestibility.

1. Plot the digestible energy content against the other two variables, and comment on the relationships.
2. Compute the correlations between the three variables, and comment.
3. Fit a suitable *simple* linear regression model.
4. Perform a diagnostic analysis. In particular, one observation is different to the others: does the observation have a large residual or a high leverage?

Table 3.9 The pH and wound size of for 20 lower-leg wounds on 17 patients (Problem 3.14)

Start		End	
Size (in cm^2)	pH	Size (in cm^2)	pH
4.3	7.26	4.0	7.15
2.4	7.63	1.5	7.15
7.3	7.63	2.9	7.50
4.3	7.18	1.4	7.15
3.5	7.75	0.1	6.69
10.3	7.94	6.0	7.56
0.6	7.60	0.6	5.52
0.7	7.90	1.1	7.70
18.3	7.60	13.1	7.76
16.1	7.70	18.1	7.42
2.5	7.98	1.0	7.15
20.0	7.35	16.5	6.55
2.4	7.89	2.3	7.28
3.7	8.00	3.5	7.40
2.4	7.10	1.0	7.48
61.0	8.30	72.0	7.95
17.7	7.66	9.6	7.32
2.1	8.20	3.0	7.24
0.9	8.25	2.0	7.71
22.0	7.63	23.5	7.52

3.13. An experiment was conducted [30] to determine how to maximize meadowfoam flower production. The data and a fuller description are given in Problem 2.15 (data set: `flowers`). In that problem, a linear regression model was fitted to the data.

1. Perform a diagnostic analysis on the fitted linear regression model.
2. Identify any influential observations or outliers.
3. Interpret the final model.

3.14. A study [15] of the effect of Manuka honey of the healing of wounds collected data from 20 wounds from 17 individuals (Table 3.9; data set: `manuka`).

1. Plot the percentage reduction in wound size over 2 weeks against the initial pH.
2. Fit the corresponding regression equation, and draw the regression line on the plot.
3. Write down the regression model. Interpret the model. (This led to one of the main conclusions of the paper.)

Later, a retraction notice was issued for the article [16] which stated that:

> The regression results presented... are strongly influenced by a high outlying value... When the results for this patient are omitted, the association is no longer statistically significant... As this relationship is pivotal to the conclusions of the paper, it is felt that the interests of patient care would be best served by a retraction.

4. Perform a diagnostic analysis of the model fitted above. Identify the observation that is influential.
5. Refit the regression model without this influential observation, and write down the model. Interpret the model, and compare to your interpretation of the previous model.
6. Plot this regression line on the plot generated above. Compare the two regression lines, and comment.

3.15. A study of babies [4] hypothesized that babies would take longer to learn to crawl in colder months because the extra clothing restricts their movement (data set: `crawl`). The data and a fuller description are given in Problem 2.16 (p. 87). In that problem, a linear regression model was fitted to the data.

1. Perform a diagnostic analysis of the fitted linear regression model.
2. Identify any influential observations or outliers.
3. Suppose some of the babies were twins. Which assumption would be violated by the inclusion of these babies in the study? Do you think this would have practical implications?

3.16. Children were asked to build towers out of cubical and cylindrical blocks as high as they could [20, 33], and the number of blocks used and the time taken were recorded. The data (data set: `blocks`) and a fuller description are given in Problem 2.18 (p. 88). In that problem, a linear regression model was fitted to model the time to build the towers, based on the initial examination in Problem 1.9 (p. 28).

1. Perform a diagnostic analysis of the linear regression model fitted in Problem 2.18 (p. 88), and show a transformation of the response is necessary.
2. Fit an appropriate linear regression model to the data after applying the transformation, ensuring a diagnostic analysis.

3.17. In Problem 2.17, the daily energy requirements and weight of 64 wethers (Table 2.11; data set: `sheep`) were analysed [18, 38, 42].

1. Using the model fitted in Problem 2.17, perform a diagnostic analysis.
2. Fit another linear regression model using the logarithm of energy requirements as the response variable. Perform a diagnostic analysis of this second model, and show this is a superior model.
3. Interpret the model that was fitted using the logarithm of energy requirements.

Table 3.10 Age, percent body fat and BMI (in kg/m^2) for 18 normal adults aged between 23 and 61 years, for males (M) and females (F) (Problem 3.18)

Age (years)	Percent body fat	Gender	BMI	Age (years)	Percent body fat	Gender	BMI
23	9.5	M	17.8	56	32.5	F	28.4
23	27.9	F	22.5	57	30.3	F	31.8
27	7.8	M	24.6	58	33.0	F	25.2
27	17.8	M	20.5	53	34.7	F	23.8
39	31.4	F	25.1	53	42.0	F	22.8
41	25.9	F	21.4	54	29.1	F	26.4
45	27.4	M	26.0	58	33.8	F	28.3
49	25.2	F	22.3	60	41.1	F	23.2
50	31.1	F	21.8	61	34.5	F	23.2

3.18. A study [23] measured the body fat percentage and BMI of adults aged between 23 and 61 (Table 3.10; data set: `humanfat`).

1. Plot the data, distinguishing between males and females. Which assumptions, if any, appear to be violated?
2. Fit the linear regression model with systematic component `Percent.Fat ~ Age * Gender` to the data.
3. Write down the two systematic components corresponding to females and males.
4. Interpret each coefficient in this model.
5. Use a t-test to determine if the interaction term is significant.
6. Use an F-test to determine if the interaction term is significant.
7. Show that the P-values for the t- and F-tests are the same for the interaction term, and explain why. Also show that the square of the t-statistic is the F-statistic (within the limitations of computer arithmetic).
8. To the earlier plot, add the separate regression lines for males and females.
9. Compute and plot the 90% confidence intervals about the fitted values for both males and females, and comment
10. Argue that only using the females in the study is sensible. Furthermore, argue that only using females aged over 38 is sensible.
11. Using this subset of the data, find a model using age and BMI as explanatory variables.
12. Using this model, compute Cook's distance, leverages, Studentized residuals and standardized residuals to evaluate the model. Identify any outliers and influential observations, and discuss the differences between the Studentized and standardized residuals.

3.19. A study of urethral length L and mass M of various mammals [41] expected to find *isometric scaling*; that is, proportional relationships being maintained as the size of animals increases. For these data (Table 3.11; data set: `urinationL`) then, one postulated relationship is $L = kM^{1/3}$ for some

Table 3.11 The urethral length of 47 mammals (Problem 3.19)

Animal	Sex	Mean mass (in kg)	Mean urethral length (in mm)	Sample size
Mouse	F	0.02	10.0	1
Wister rat	F	0.20	9.5	20
Rat	F	0.20	17.0	1
Sprague-Dawley rat	F	0.30	20.0	61
Dunkin Hartley guinea pig	M	0.40	20.0	1
Normal adult cat	F	2.30	49.4	1
⋮	⋮	⋮	⋮	⋮

Table 3.12 The mean annual rainfall, altitude, latitude and longitude for 24 cities in the wheat-growing region of eastern Australia. Only the first six observations are shown (Problem 3.20)

Station name	Altitude (in m)	Latitude (°S)	Longitude (°E)	Mean annual rainfall (in mm)	Region
Goondiwindi	216.0	28.53	150.30	529	3
Condobolin	199.0	33.08	147.15	447	1
Coonamble	180.0	30.97	148.38	505	1
Gilgandra	278.0	31.72	148.67	563	2
Nyngan	177.0	31.56	147.20	440	1
Trangie	219.0	32.03	147.99	518	1
⋮	⋮	⋮	⋮	⋮	⋮

proportionality constant k. By using a transformation, fit an appropriate weighted linear regression model, and test the hypothesis using both a t-test and an F-test. Interpret your model.

3.20. A study of the annual rainfall between 1916 and 1990 in a wheat-growing region of eastern Australia [6] explored the relationships between mean annual rainfall `AR` and region `Region`, altitude `Alt`, latitude `Lat` and longitude `Lon` (Table 3.12; data set: `wheatrain`).

1. Plot the annual rainfall against the region and altitude, and identify any important features.
2. Interpret a regression model with systematic component `AR ~ Alt * Region`.
3. Fit the model with systematic component `AR ~ Alt * Region`. Show that the interaction term is not necessary in the model, but both main effect terms are necessary.
4. Produce diagnostic plots and evaluate the fitted model. Use both standardized and Studentized residuals, and compare. Identify the observation that appears to be an outlier.

Table 3.13 The strength of Kraft paper measured for various percentages of hardwood concentration (Problem 3.21)

Strength	% Hardwood	Strength	% Hardwood	Strength	% Hardwood
6.3	1.0	33.8	5.0	52.0	10.0
11.1	1.5	34.0	5.5	52.5	11.0
20.0	2.0	38.1	6.0	48.0	12.0
24.0	3.0	39.9	6.5	42.8	13.0
26.1	4.0	42.0	7.0	27.8	14.0
30.0	4.5	46.1	8.0	21.9	15.0
		53.1	9.0		

5. The data are spatial, so examine the independence of the data by plotting the residuals against `Lon` and against `Lat`. Comment.
6. Summarize the diagnostic analysis of the fitted model.

3.21. The tensile strength of Kraft paper (a strong, coarse and usually brownish type of paper) was measured [18, 19] for different percentages of hardwood concentrations (Table 3.13; data set: `paper`).

1. Plot the data, and show that the data have a non-linear relationship.
2. Determine a suitable polynomial model for the data using `poly()`.
3. Determine a suitable model using a regression spline.
4. Plot the two models (one using `poly()`; one using a regression spline) on the data, and comment.

3.22. An experiment was conducted [11] to measure the heat developed by setting cement with varying constituents (Table 3.14; data set: `setting`).

1. Plot each explanatory variable against heat evolved, and decide which constituents appear to be related to heat evolved.
2. Fit the linear regression model predicting heat evolved from the explanatory variables `A`, `B`, `C` and `D` (that is, no interactions). Using t-tests, determine which explanatory variables appear statistically significant. Compare to your decisions in the previous part of this question.
3. Show that collinearity may be a problem. Explain why this may be the case, and propose a solution.
4. Fit the amended model, and compare the t-test results to the t-test results from the initial model above.

3.23. A compilation of data [1] from various studies of Gopher tortoises linked the mean clutch size to environmental variables for 19 populations of the tortoises (Table 3.15; data set: `gopher`).

1. Plot the mean clutch size against the temperature and evapotranspiration. Comment on the relationships.

Table 3.14 The amount of heat evolved (in calories/gram of cement) `Heat` by setting cement for given percentages of four constituents: `A` refers to tricalcium aluminate; `B` to tricalcium silicate; `C` to tetracalcium alumino ferrite; `D` to dicalcium silicate (Problem 3.22)

A	B	C	D	Heat	A	B	C	D	Heat	A	B	C	D	Heat
7	26	6	60	78.5	11	55	9	22	109.2	21	47	4	26	115.9
1	29	15	52	74.3	3	71	17	6	102.7	1	40	23	34	83.8
11	56	8	20	104.3	1	31	22	44	72.5	11	66	9	12	113.3
11	31	8	47	87.6	2	54	18	22	93.1	10	68	8	12	109.4
7	52	6	33	95.9										

Table 3.15 Results from 19 studies of Gopher tortoises. `Lat` is the latitude at which the study was conducted; `Evap` is the mean total annual actual evapotranspiration (in mm); `Temp` is the mean annual temperature (in °C); `ClutchSize` is the mean clutch size; `SampleSize` is the sample size used in the study (Problem 3.23)

Site	Latitude	Evap	Temp	ClutchSize	SampleSize
1	26.8	1318	24.0	8.2	23
2	27.3	1193	22.2	6.5	8
3	27.7	1112	22.7	7.6	32
4	28.0	1171	22.6	7.1	19
5	28.5	1116	21.4	4.8	12
6	28.5	1116	21.4	5.8	16
7	28.5	1116	21.4	8.0	19
8	28.6	1198	22.2	7.5	24
9	29.5	1091	20.4	5.8	62
10	29.7	1091	20.4	5.8	51
11	30.3	1037	20.4	5.0	23
12	30.7	1039	20.0	4.6	11
13	30.8	1030	19.2	5.5	19
14	30.9	1036	19.3	7.0	47
15	31.2	995	19.2	5.6	36
16	31.3	992	18.8	4.8	87
17	31.9	1018	19.7	6.5	25
18	32.5	965	18.6	3.8	23
19	32.6	911	18.6	4.5	23

2. Explain why a weighted linear regression model is appropriate.
3. Fit a weighted linear regression model for modelling `ClutchSize` using `Evap` and `Temp` as explanatory variables. Produce the t-tests, and comment.
4. Compute the ANOVA table for the fitted model, and comment.
5. Show that collinearity is evident in the data.
6. Perform a diagnostic analysis of this model. Be sure to test spatial independence by plotting the residuals against `Latitude`.

3.24. Consider the (artificial) data in Table 3.16 (based on [14]), and contained in data set `triangle`.

Table 3.16 The data for Problem 3.24

y	x_1	x_2	y	x_1	x_2	y	x_1	x_2	y	x_1	x_2
10.1	5.3	8.5	11.1	4.2	10.3	8.8	4.2	7.7	10.9	5.7	9.3
11.6	5.4	10.3	11.4	5.0	10.2	13.5	5.6	12.3	12.2	4.0	11.6
10.4	4.5	9.4	13.0	5.0	12.1	10.3	3.2	9.8	11.3	4.2	10.4
13.0	4.7	12.2	13.2	6.9	11.2	12.6	6.5	10.8	10.1	5.6	8.5
12.3	6.6	10.4	10.2	4.7	9.0	10.1	4.3	9.1	9.7	5.6	7.9

1. Fit the linear regression model with the systematic component `y ~ x1 + x2` to the data. Show that the interaction term is not necessary.
2. Use appropriate diagnostics to show the model is appropriate.
3. Interpret the fitted model.
4. The data are actually randomly generated so that $\mu = \sqrt{x_1^2 + x_2^2}$; that is, x_1 and x_2 are the lengths of the sides of a right-angled triangle, and μ is the length of the hypotenuse (and some randomness has been added to produce y). What lesson does this demonstrate?
5. Fit the model for modelling $\mu = \mathrm{E}[y^2]$, using the systematic component `I(x1^2) + I(x2^2) - 1`. Then use the t-test to confirm that the parameter estimates suggested by Pythagoras' theorem are supported by the data.

3.25. In an experiment [39, p 122] conducted to investigate the amount of drug retained in the liver of a rat (Table 3.17; data set: `ratliver`), nineteen rats were randomly selected, weighed, and placed under light anesthetic and given an oral dose of the drug. Because large livers were thought to absorb more of a given dose than a small liver, the dose was approximately determined as 40 mg of the drug per kg of body weight. After a fixed length of time, each rat was sacrificed, the liver weighed, and the percentage dose in the liver y determined.

1. Plot `DoseInLiver` against each explanatory variable, and identify important features to be modelled.
2. Fit a linear regression model with systematic component `DoseInLiver ~ BodyWt + LiverWt + Dose`.
3. Using t-tests, show that `BodyWt` and `Dose` are significant for modelling `DoseInLiver`.
4. In the study, the dose was determined as an approximate function of body weight, hence both variables `BodyWt` and `Dose` measure almost the same physical quantity. Why should both covariates be necessary in the model? By computing the appropriate statistics, show that Observation 3 has high leverage and is influential.
5. Plot `BodyWt` against `Dose`, and identify Observation 3 to see the problem.
6. Fit the same linear regression model, after omitting Observation 3. Use t-tests to show that *none* of the covariates are now statistically significant.

Table 3.17 Drug doses retained in the liver of rats. See the text for an explanation of the data. `BodyWt` is the body weight of each rat (in g); `LiverWt` is liver weight (in g); `Dose` is the dose relative to largest dose; `DoseInLiver` is the proportion of the dose in liver, as percentage of liver weight (Problem 3.25)

BodyWt	LiverWt	Dose	DoseInLiver	BodyWt	LiverWt	Dose	DoseInLiver
176	6.5	0.88	0.42	158	6.9	0.80	0.27
176	9.5	0.88	0.25	148	7.3	0.74	0.36
190	9.0	1.00	0.56	149	5.2	0.75	0.21
176	8.9	0.88	0.23	163	8.4	0.81	0.28
200	7.2	1.00	0.23	170	7.2	0.85	0.34
167	8.9	0.83	0.32	186	6.8	0.94	0.28
188	8.0	0.94	0.37	146	7.3	0.73	0.30
195	10.0	0.98	0.41	181	9.0	0.90	0.37
176	8.0	0.88	0.33	149	6.4	0.75	0.46
165	7.9	0.84	0.38				

Table 3.18 Inorganic and organic phosphorus in 18 soil samples, tested at 20°C. `Inorg` is the amount of inorganic phosphorus (in ppm); `Org` is the amount of organic phosphorus (in ppm); `PA` is the amount of plant-available phosphorus (in ppm) (Problem 3.26)

Sample	Inorg	Org	PA	Sample	Inorg	Org	PA	Sample	Inorg	Org	PA
1	0.4	53	64	7	9.4	44	81	13	23.1	50	77
2	0.4	23	60	8	10.1	31	93	14	21.6	44	93
3	3.1	19	71	9	11.6	29	93	15	23.1	56	95
4	0.6	34	61	10	12.6	58	51	16	1.9	36	54
5	4.7	24	54	11	10.9	37	76	17	26.8	58	168
6	1.7	65	77	12	23.1	46	96	18	29.9	51	99

3.26. The amount of organic, inorganic and plant-available phosphorus was chemically determined [35] in eighteen soil samples (Table 3.18; data set: `phosphorus`), all tested at 20°C.

1. Plot the plant-available phosphorous against both inorganic and organic phosphorus. Comment.
2. Fit the linear regression model with systematic component `PA ~ Inorg + Org`.
3. Use t-tests to identify which covariates are statistically significant.
4. Use appropriate statistics to identify any influential observations, and any observations with high leverage.

3.27. Thirteen American footballers punted a football [26], and had their leg strengths measured (Table 3.19; data set: `punting`).

1. Plot punting distance y against left leg strength x_1, and then against right leg strength x_2. Comment.
2. Show that collinearity is likely to be a problem.
3. Propose a sensible solution to the collinearity problem.

Table 3.19 Leg strength (in lb) and punting distance (in feet, using the right foot) for 13 American footballers. Leg strengths were determined using a weight lifting test (Problem 3.27)

Left-leg strength	Right-leg strength	Punting distance	Left-leg strength	Right-leg strength	Punting distance
170	170	162.50	110	110	104.83
130	140	144.00	110	120	105.67
170	180	174.50	120	130	117.58
160	160	163.50	140	120	140.25
150	170	192.00	130	140	150.17
150	150	171.75	150	160	165.17
180	170	162.00			

Table 3.20 The age and salary (including bonuses) of CEOs of small companies. The first six observations are shown (Problem 3.28)

Age (in years)	Salary (in $'000)
53	145
43	621
33	262
45	208
46	362
55	424
⋮	⋮

4. Determine a suitable model for the data, ensuring a diagnostics analysis.
5. Interpret the final model.

3.28. The age and salary of the chief executive officers (CEO) of small companies in 1993 (Table 3.20; data set: `ceo`) were published by *Forbes* magazine [34]. (Small companies were defined as those with annual sales greater than $5 million and less than $350 million, according to 5-year average return on investment.) Find a suitable model for the data, and supply appropriate diagnostics to show the model is appropriate.

3.29. A study of computer tomography (CT) interventions [32, 43] in the abdomen measured the total procedure time and the total radiation dose received (Table 3.21; data set: `fluoro`). During these procedures, "one might postulate that the radiation dose received is related to... the total procedure time" [43, p. 619].

1. Plot the dose against the exposure time, and comment.
2. Fit the linear regression model for modelling dose from exposure time. Produce the residual plots, and show that the variance is not constant.

Table 3.21 Total exposure time and radiation dose for nineteen patients undergoing CT fluoroscopy in the abdomen (Problem 11.13)

Time (in min)	Dose (in rad)	Time (in min)	Dose (in rad)	Time (in min)	Dose (in rad)
37	4.39	66	9.39	90	34.81
48	3.46	67	6.36	92	16.61
52	8.00	75	17.12	97	58.56
57	5.47	75	50.91	98	84.77
58	8.00	83	20.70	100	23.57
61	18.92	83	25.28	114	66.02
		86	47.94		

Table 3.22 Percentage butterfat for various pure-bred cattle taken from Canadian records. There are five breeds, and ten 2-year old cows have been randomly selected plus ten mature (older than 4 years) cows (Problem 3.30)

Ayrshire		Canadian		Guernsey		Holstein–Fresian		Jersey	
Mature	2 years	Mature	2 years	Mature	2 years	Mature	2 years	Mature	2 years
3.74	4.44	3.92	4.29	4.54	5.30	3.40	3.79	4.80	5.75
4.01	4.37	4.95	5.24	5.18	4.50	3.55	3.66	6.45	5.14
3.77	4.25	4.47	4.43	5.75	4.59	3.83	3.58	5.18	5.25
3.78	3.71	4.28	4.00	5.04	5.04	3.95	3.38	4.49	4.76
4.10	4.08	4.07	4.62	4.64	4.83	4.43	3.71	5.24	5.18
4.06	3.90	4.10	4.29	4.79	4.55	3.70	3.94	5.70	4.22
4.27	4.41	4.38	4.85	4.72	4.97	3.30	3.59	5.41	5.98
3.94	4.11	3.98	4.66	3.88	5.38	3.93	3.55	4.77	4.85
4.11	4.37	4.46	4.40	5.28	5.39	3.58	3.55	5.18	6.55
4.25	3.53	5.05	4.33	4.66	5.97	3.54	3.43	5.23	5.72

3. Try using various transformations of the response variable. Fit these model, and re-examine the residual plots to determine a suitable transformation.
4. Test the hypothesis implied by the quote given original article.
5. Interpret the final model.

3.30. The average butterfat content of milk from dairy cows was recorded for each of five breeds of cattle [18, 36]. Random samples of ten mature (older than 4 years) and ten 2-year olds were taken (Table 3.22; data set: `butterfat`).

1. Plot the percentage butterfat against breed, and also against age. Discuss any features of the data that are apparent.
2. Use various transformation to make the variance of the response approximately constant. Which transformation appears appropriate? Does using `boxcox()` help with the decision?
3. Fit an appropriate linear regression model, and interpret the appropriate diagnostics.

References

[1] Ashton, K.G., Burke, R.L., Layne, J.N.: Geographic variation in body and clutch size of Gopher tortoises. Copeia **2007**(2), 355–363 (2007)

[2] Atkinson, A.C.: Regression diagnostics, transformations and constructed variables. Journal of the Royal Statistical Society, Series B **44**(1), 1–36 (1982)

[3] Belsley, D.A., Kuh, E., Welsch, R.E.: Regression Diagnostics: Identifying Influential Data and Sources of Collinearity. John Wiley & Sons, New York (2004)

[4] Benson, J.: Season of birth and onset of locomotion: Theoretical and methodological implications. Infant Behavior and Development **16**(1), 69–81 (1993)

[5] Bivand, R.S., Pebesma, E.J., Gómez-Rubio, V.: Applied Spatial Data Analysis with R. Springer (2008)

[6] Boer, R., Fletcher, D.J., Campbell, L.C.: Rainfall patterns in a major wheat-growing region of Australia. Australian Journal of Agricultural Research **44**, 609–624 (1993)

[7] Box, G.E.P., Cox, D.R.: An analysis of transformations (with discussion). Journal of the Royal Statistical Society, Series B **26**, 211–252 (1964)

[8] Cochran, D., Orcutt, G.H.: Application of least squares regression to relationships containing auto-correlated error terms. Journal of the American Statistical Association **44**(245), 32–61 (1949)

[9] Cook, D.R.: Detection of influential observations in linear regression. Technometrics **19**(1), 15–18 (1977)

[10] Davison, A.C.: Statistical Models. Cambridge University Press, UK (2003)

[11] Draper, N., Smith, H.: Applied Regression Analysis. John Wiley and Sons, New York (1966)

[12] Fox, J.: An R and S-Plus Companion to Applied Regression Analysis. Sage Publications, Thousand Oaks, CA (2002)

[13] Geary, R.C.: Testing for normality. Biometrics **34**(3/4), 209–242 (1947)

[14] Gelman, A., Nolan, D.: Teaching Statistics: A Bag of Tricks. Oxford University Press, Oxford (2002)

[15] Gethin, G.T., Cowman, S., Conroy, R.M.: The impact of Manuka honey dressings on the surface pH of chronic wounds. International Wound Journal **5**(2), 185–194 (2008)

[16] Gethin, G.T., Cowman, S., Conroy, R.M.: Retraction: The impact of Manuka honey dressings on the surface pH of chronic wounds. International Wound Journal **11**(3), 342–342 (2014)

[17] Giauque, W.F., Wiebe, R.: The heat capacity of hydrogen bromide from 15°K. to its boiling point and its heat of vaporization. The entropy from spectroscopic data. Journal of the American Chemical Society **51**(5), 1441–1449 (1929)
[18] Hand, D.J., Daly, F., Lunn, A.D., McConway, K.Y., Ostrowski, E.: A Handbook of Small Data Sets. Chapman and Hall, London (1996)
[19] Joglekar, G., Scheunemyer, J.H., LaRiccia, V.: Lack-of-fit testing when replicates are not available. The American Statistician **43**, 135–143 (1989)
[20] Johnson, B., Courtney, D.M.: Tower building. Child Development **2**(2), 161–162 (1931)
[21] Kahn, M.: An exhalent problem for teaching statistics. Journal of Statistical Education **13**(2) (2005).
[22] Little, R.J.A., Rubin, D.B.: Statistical analysis with missing data (2nd ed.). Wiley, New York (2002)
[23] Mazess, R.B., Peppler, W.W., Gibbons, M.: Total body composition by dualphoton (^{153}Gd) absorptiometry. American Journal of Clinical Nutrition **40**, 834–839 (1984)
[24] Moir, R.J.: A note on the relationship between the digestible dry matter and the digestable energy content of ruminant diets. Australian Journal of Experimental Agriculture and Animal Husbandry **1**, 24–26 (1961)
[25] Moore, D.S., McCabe, G.P.: Introduction to the Practice of Statistics, second edn. W. H. Freeman and Company, New York (1993)
[26] Myers, R.H.: Classical and Modern Regression with Applications, second edn. Duxbury, Belmont CA (1990)
[27] Palomares, M.L., Pauly, D.: A multiple regression model for predicting the food consumption of marine fish populations. Australian Journal of Marine and Freshwater Research **40**(3), 259–284 (1989)
[28] Ryan, T.A., Joiner, B.L., Ryan, B.F.: Minitab Student Handbook. Duxbury Press, North Scituate, Mass. (1976)
[29] Searle, S.R., Casella, G., McCulloch, C.E.: Variance Components. John Wiley and Sons, New York (2006)
[30] Seddigh, M., Joliff, G.D.: Light intensity effects on meadowfoam growth and flowering. Crop Science **34**, 497–503 (1994)
[31] Shacham, M., Brauner, N.: Minimizing the effects of collinearity in polynomial regression. Industrial and Engineering Chemical Research **36**, 4405–4412 (1997)
[32] Silverman, S.G., Tuncali, K., Adams, D.F., Nawfel, R.D., Zou, K.H., Judy, P.F.: CT fluoroscopy-guided abdominal interventions: Techniques, results, and radiation exposure. Radiology **212**, 673–681 (1999)
[33] Singer, J.D., Willett, J.B.: Improving the teaching of applied statistics: Putting the data back into data analysis. The American Statistician **44**(3), 223–230 (1990)

[34] Smyth, G.K.: Australasian data and story library (OzDASL) (2011). URL http://www.statsci.org/data
[35] Snapinn, S.M., Small, R.D.: Tests of significance using regression models for ordered categorical data. Biometrics **42**, 583–592 (1986)
[36] Sokal, R.R., Rohlf, F.J.: Biometry: The Principles and Practice of Statistics in Biological Research, third edn. W. H. Freeman and Company, New York (1995)
[37] Student: The probable error of a mean. Biometrika **6**(1), 1–25 (1908)
[38] Wallach, D., Goffinet, B.: Mean square error of prediction in models for studying ecological systems and agronomic systems. Biometrics **43**(3), 561–573 (1987)
[39] Weisberg, S.: Applied Linear Regression. Wiley Series in Probability and Mathematical Statistics. John Wiley and Sons, New York (1985)
[40] West, B.T., Welch, K.B., Galecki, A.T.: Linear Mixed Models: A Practical Guide using Statistical Software. CRC, Boca Raton, Fl (2007)
[41] Yang, P.J., Pham, J., Choo, J., Hu, D.L.: Duration of urination does not change with body size. Proceedings of the National Academy of Sciences **111**(33), 11 932–11 937 (2014)
[42] Young, B.A., Corbett, J.L.: Maintenance energy requirement of grazing sheep in relation to herbage availability. Australian Journal of Agricultural Research **23**(1), 57–76 (1972)
[43] Zou, K.H., Tuncali, K., Silverman, S.G.: Correlation and simple linear regression. Radiology **227**, 617–628 (2003)

Chapter 4
Beyond Linear Regression: The Method of Maximum Likelihood

Just as the ability to devise simple but evocative models is the signature of the great scientist so overelaboration and overparameterization is often the mark of mediocrity.
Box [2, p. 792]

4.1 Introduction and Overview

The linear regression model introduced in Chap. 2 assumes the variance is constant, possibly from a normal distribution. Many data types exist for which the randomness is not constant, and so other methods are necessary. This chapter demonstrates situations where the linear regression model fails. In these cases, least-squares estimation, as used in Chap. 2, is no longer appropriate. Instead, maximum likelihood estimation is appropriate. In Chap. 4, we discuss three specific situations in which linear regression models fail (Sect. 4.2) and then consider a general approach to modelling such data (Sect. 4.3). To fit these models, maximum likelihood estimation is needed and is reviewed in Sect. 4.4. We then examine maximum likelihood estimation in the case of one parameter (Sect. 4.5) and more than one parameter (Sect. 4.6), and then using matrix algebra (Sect. 4.7). Fitting models using maximum likelihood is discussed in Sect. 4.8, followed by a review of the properties of maximum likelihood estimators (Sect. 4.9). Results concerning hypothesis tests (Sect. 4.10) and confidence intervals (Sect. 4.11) are then presented, followed by a discussion of comparing non-nested models (Sect. 4.12).

4.2 The Need for Non-normal Regression Models

4.2.1 When Linear Models Are a Poor Choice

The random component of the regression models in Chap. 2 has constant variance, possibly from a normal distribution. Three common situations exist where the variation is not constant, and so linear regression models are a poor choice for modelling such data:

P. K. Dunn, G. K. Smyth, *Generalized Linear Models with Examples in R*,
Springer Texts in Statistics, https://doi.org/10.1007/978-1-4419-0118-7_4

1. The response is a *proportion*, ranging between 0 and 1 inclusive, of a total number of counts. As the modelled proportion approaches these boundaries of 0 and 1, the variance of the responses must approach zero. The variance must be smaller near 0 and 1 than the variation of proportions near 0.5 (where the observations can spread equally in both directions toward the boundaries). Thus, the variance is not, and cannot be, constant. Furthermore, because the response is between 0 and 1, the randomness cannot be normally distributed. For proportions of a total number of counts, the *binomial* distribution may be appropriate (Sect. 4.2.2; Chap. 9).
   A specific example of binomial data is *binary* data (Example 4.6) where the response takes one of two outcomes (such as 'success' and 'failure', or 'present' and 'absent').
2. The response is a *count*. As the modelled count approaches zero, the variance of the responses must approach zero. Furthermore, the normal distribution is a poor choice for modelling the randomness because counts are discrete and non-negative. For count data, the *Poisson* distribution may be appropriate (Example 1.5; Sect. 4.2.3; Chap. 10).
3. The response is *positive continuous*. As the modelled response approaches zero, the variance of the responses must approach zero. Furthermore, the normal distribution is a poor choice because positive continuous data are often right skewed, and because the normal distribution permits negative values. For positive continuous data, distributions such as the *gamma* and *inverse Gaussian* distributions may be appropriate (Sect. 4.2.4; Chap. 11).

In these circumstances, the relationship between y and the explanatory variables is usually non-linear also: the response has boundaries in all cases, so a linear relationship cannot apply for all values of the response.

4.2.2 Binary Outcomes and Binomial Counts

First consider binary regression. There are many applications in which the response is a binary variable, taking on only two possible states. In this situation, a transformation to normality is out of the question.

Example 4.1. (Data set: `gforces`) Military pilots sometimes black out when their brains are deprived of oxygen due to G-forces during violent manoeuvres. A study [7] produced similar symptoms by exposing volunteers' lower bodies to negative air pressure, likewise decreasing oxygen to the brain. The data record the ages of eight volunteers and whether they showed syncopal blackout-related signs (pallor, sweating, slow heartbeat, unconsciousness) during an 18 min period. Does resistance to blackout decrease with age?

```
> data(gforces); gforces
  Subject Age Signs
1      JW  39     0
2      JM  42     1
3      DT  20     0
4      LK  37     1
5      JK  20     1
6      MK  21     0
7      FP  41     1
8      DG  52     1
```

The explanatory variable is `Age`. The response variable is `Signs`, coded as `1` if the subject showed blackout-related signs and `0` otherwise. The response variable is binary, taking only two distinct values, and no transformation can change that. A regression approach that directly models the *probability* of a blackout response given the age of the subject is needed. □

The same principles apply to situations where a number of binary outcomes are tabulated to make a binomial random variable, as in the following example.

Example 4.2. (Data set: `shuttles`) After the explosion of the space shuttle *Challenger* on January 28, 1986, a study was conducted [3, 4] to determine if previously-collected data about the ambient air temperature at the time of launch could have been used to foresee potential problems with the launch (Table 4.1). In this example, the response variable is the number of damaged O-rings out of six for each of the previous 23 launches with data available, so only seven values are possible for the response. No transformation can change this.

A more sensible model would be to use a binomial distribution with mean proportion μ for modelling the proportion y of O-rings damaged out of m at various temperatures x. (Here, $m = 6$ for every launch.) Furthermore, a linear relationship between temperature and the proportion of damaged O-rings cannot be linear, as proportions are restricted to the range $(0, 1)$. Instead, a systematic relationship of the form

$$\log \frac{\mu}{1-\mu} = \beta_0 + \beta_1 x$$

may be more suitable, since $\log\{\mu/(1-\mu)\}$ has a range over the entire real line. □

Combining the systematic and random components, a possible model for the data is:

$$\begin{cases} ym \sim \text{Bin}(\mu, m) & \text{(random component)} \\ \log \dfrac{\mu}{1-\mu} = \beta_0 + \beta_1 x & \text{(systematic component).} \end{cases} \tag{4.1}$$

Table 4.1 The ambient temperature and the number of O-rings (out of six) damaged for 23 of the 24 space shuttle launches before the launch of *Challenger*; *Challenger* was the 25th shuttle. One engine was lost at sea and so its O-rings could not be examined (Example 4.2)

Temperature (in °F)	O-rings damaged	Temperature (in °F)	O-rings damaged	Temperature (in °F)	O-rings damaged
53	2	68	0	75	0
57	1	69	0	75	2
58	1	70	0	76	0
63	1	70	0	76	0
66	0	70	1	78	0
67	0	70	1	79	0
67	0	72	0	81	0
67	0	73	0		

4.2.3 Unrestricted Counts: Poisson or Negative Binomial

Count data is another situation where linear regression models are inadequate.

Example 4.3. (Data set: `nminer`) A study [9] of the habitats of the noisy miner (a small but aggressive native Australian bird) counted the number of noisy miners y and the number of eucalypt trees x in two-hectare buloke woodland transects (Table 1.2, p. 15). Buloke woodland patches with more eucalypts tend to have more noisy miners (Fig. 1.4, p. 15).

The number of noisy miners is more variable where more eucalypts are present. Between 0 and 10 eucalypts, the number of noisy miners is almost always zero; between 10 and 20 eucalypts, the number of noisy miners increases. This shows that the systematic relationship between the number of eucalypts and the number of noisy miners is not linear. A possible model for the systematic component is $\log \mu = \beta_0 + \beta_1 x$, where x is the number of eucalypt trees at a given site, and μ is the expected number of noisy miners. Using the logarithm ensures $\mu > 0$ even when β_0 and β_1 range between $-\infty$ and ∞, and also models the non-linear form of the relationship between μ and x.

Between 0 and 10 eucalypts, the number of noisy miners varies little. Between 10 and 20 eucalypts, a larger amount of variation exists in the number of noisy miners. This shows that the randomness does not have constant variance. Instead, the variation in the data may be modelled using a *Poisson distribution*, $y \sim \text{Pois}(\mu)$, where $y = 0, 1, 2, \ldots$, and $\mu > 0$.

Combining the systematic and random components, a possible model for the data is:

$$\begin{cases} y \sim \text{Pois}(\mu) & \text{(random component)} \\ \log \mu = \beta_0 + \beta_1 x & \text{(systematic component).} \end{cases} \tag{4.2}$$

□

Table 4.2 The time for delivery to soft drink vending machines (Example 4.4)

Time (in mins)	Cases	Distance (in feet)	Time (in mins)	Cases	Distance (in feet)	Time (in mins)	Cases	Distance (in feet)
16.68	7	560	79.24	30	1460	19.00	7	132
11.50	3	220	21.50	5	605	9.50	3	36
12.03	3	340	40.33	16	688	35.10	17	770
14.88	4	80	21.00	10	215	17.90	10	140
13.75	6	150	13.50	4	255	52.32	26	810
18.11	7	330	19.75	6	462	18.75	9	450
8.00	2	110	24.00	9	448	19.83	8	635
17.83	7	210	29.00	10	776	10.75	4	150
			15.35	6	200			

4.2.4 Continuous Positive Observations

A third common situation where linear regressions are unsuitable is for positive continuous data.

Example 4.4. (Data set: `sdrink`) A soft drink bottler is analyzing vending machine service routes in his distribution system [11, 13]. He is interested in predicting the amount of time y required by the route driver to service the vending machines in an outlet. This service activity includes stocking the machine with beverage products and minor maintenance or housekeeping. The industrial engineer responsible for the study has suggested that the two most important variables affecting the delivery time are the number of cases of product stocked x_1 and the distance walked by the route driver x_2. The engineer has collected 25 observations on delivery time, the number of cases and distance walked (Table 4.2).

In this case, the delivery times are strictly positive values. They are likely to show an increasing mean–variance relationship with standard deviation roughly proportional to the mean, so a log-transformation might be approximately variance stabilizing. However the dependence of time on the two covariates is likely to be directly linear, because time should increase linearly with the number of cases or the distance walked (Fig. 4.1); that is, a sensible systematic component is $\mu = \beta_0 + \beta_1 x_1 + \beta_2 x_2$. No normal linear regression approach can achieve these conflicting aims, because any transformation to stabilize the variance would destroy linearity. A regression approach that directly models the delivery times using an appropriate probability distribution for positive numbers (such as a gamma distribution) is desirable. Combining the systematic and random components, a possible model for the data is:

$$\begin{cases} y \sim \text{Gamma}(\mu; \phi) & \text{(random component)} \\ \mu = \beta_0 + \beta_1 x & \text{(systematic component)} \end{cases} \tag{4.3}$$

where ϕ is related to the variance of the gamma distribution. □

Fig. 4.1 A plot of the soft drink data: time against the number of cases of product sold (left panel) and time against the distance walked by the route driver (right panel)

Table 4.3 The time to death (in weeks) and white blood cell count (WBC) for leukaemia patients, grouped according to AG type (Example 4.5)

AG positive patients				AG negative patients			
WBC	Time to death	WBC	Time to death	WBC	Time to death	WBC	Time to death
2300	65	7000	143	4400	56	28000	3
750	156	9400	56	3000	65	31000	8
4300	100	32000	26	4000	17	26000	4
2600	134	35000	22	1500	7	21000	3
6000	16	100000	1	9000	16	79000	30
10500	108	100000	1	5300	22	100000	4
10000	121	52000	5	10000	3	100000	43
17000	4	100000	65	19000	4	27000	2
5400	39						

Example 4.5. (Data set: `leukwbc`) The times to death (in weeks) of two groups of leukaemia patients (grouped according to a morphological variable called the AG factor) were recorded (Table 4.3) and their white blood cell counts were measured (Fig. 4.2). The authors originally fitted a model using the exponential distribution [5, 6].

We would like to model the survival times on a log-linear scale, building a linear predictor for $\log \mu_i$, where $\mu_i > 0$ is the expected survival time. However the log-survival times are not normally distributed, as the logarithm of an exponentially distributed random variable is markedly left-skewed. Hence normal linear regression with the log-survival times as response is less than desirable. Furthermore, linear regression would estimate the variance of the residuals, whereas the variance of an exponential random variable is known once the mean is specified. An analysis that uses the exponential distribution explicitly is needed. □

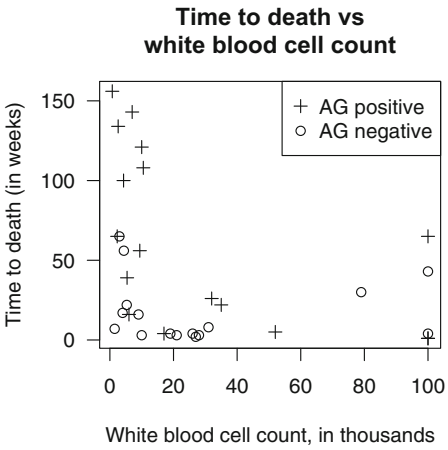

Fig. 4.2 A plot of the leukaemia data: time to death against the white blood cell count (Example 4.5)

Table 4.4 Different models discussed so far, all of which are generalized linear models. In all cases $\eta = \beta_0 + \sum_{j=1}^{p} \beta_j x_j$ for the appropriate explanatory variables x_j (Sect. 4.3)

Data	Reference	Random component	Systematic component
FEV data	Example 1.1 (p. 1)	Normal	$\mu = \eta$
Challenger data	Example 4.2 (p. 167)	Binomial	$\log\{\mu/(1-\mu)\} = \eta$
Noisy miner data	Example 4.3 (p. 168)	Poisson	$\log \mu = \eta$
Soft drink data	Example 4.4 (p. 169)	Gamma	$\mu = \eta$
Leukaemia data	Example 4.5 (p. 170)	Exponential	$\log \mu = \eta$

4.3 Generalizing the Normal Linear Model

For the data in Sect. 4.2, different models are suggested (Table 4.4): a variety of random and systematic components appear. The theory in Chaps. 2 and 3, based on linearity and constant variance, no longer applies.

To use each of the models listed in Table 4.4 requires the development of separate theory: fitting algorithms, inference procedures, diagnostic tools, and so on. An alternative approach is to work more generally. For example, later we consider a *family* of distributions which has the normal, binomial, Poisson and gamma distributions as special cases. Using this general family of distributions, any estimation algorithms, inference procedures and diagnostic tools that are developed apply to *all* distributions in this family of distributions. Implementation for any one specific model would be a special case of the general theory. In addition, later we allow systematic components of the form $f(\mu) = \eta$ for certain functions $f()$.

This is the principle behind generalized linear models (GLMs). GLMs unify numerous models into one general theoretical framework, incorporating all the models in Table 4.4 (and others) under one structure. Common estimation algorithms (Chap. 6), inference methods (Chap. 7), and diagnostic tools (Chap. 8) are possible under one common framework. The family of distributions used for GLMs is called the *exponential dispersion model* (or EDM) family, which includes common distributions such as the normal, binomial, Poisson and gamma distributions, among others.

Why should the random component be restricted to distributions in the EDM family? For example, distributions such as the Weibull distribution and von Mises distribution are not EDMs, but may be useful for modelling certain types of data. GLMs are restricted to distributions in the EDM family because the general theory is developed by taking advantage of the structure of EDMs. Using the structure provided by the EDM family enables simple fitting algorithms and inference procedures, which share similarities with the normal linear regression models. The theory does not apply to distributions that are not EDMs. Naturally, if a non-EDM distribution really is appropriate it should be used (and the model will not be a GLM). However, EDMs are useful for most common types of data:

- Continuous data over the entire real line may be modelled by the normal distribution (Chaps. 2 and 3).
- Proportions of a total number of counts may be modelled by the binomial distribution (Example 4.2; Chap. 9).
- Discrete count data may be modelled by the Poisson or negative binomial distributions (Example 4.3; Chap. 10).
- Continuous data over the positive real line may be modelled by the gamma and inverse Gaussian distributions (Example 4.4; Chap. 11).
- Positive data with exact zeros may be modelled by a special case of the Tweedie distributions (Chap. 12).

The advantages of GLMs are two-fold. Firstly, the mean–variance relationship can be chosen separately from the appropriate scale for the linear predictor. Secondly, by choosing a response distribution that matches the natural support of the responses, we can expect to achieve a better approximation to the probability distribution.

4.4 The Idea of Likelihood Estimation

Chapter 2 developed the principle of least-squares as a criterion for estimating the parameters in the linear predictor of linear regression models. Least-squares is an appropriate criterion for fitting regression models to response data that are approximately normally distributed. In the remainder of this chapter, we develop a much more general estimation methodology called

maximum likelihood. Maximum likelihood is appropriate for estimating the parameters of non-normal models such as those based on the binomial, Poisson or gamma distributions discussed earlier in this chapter, and includes least-squares as a special case. Maximum likelihood tools will be used extensively for fitting models and testing hypotheses in the remaining chapters of this book.

Maximum likelihood can be applied whenever a specific probability distribution has been proposed for the data at hand. The idea of maximum likelihood is to choose those estimates for the unknown parameters that maximize the probability density of the observed data.

Suppose for example that $y_1, \ldots, y_n$ are independent observations from an exponential distribution with scale parameter θ. The probability density function, or probability function, of the exponential distribution is

$$\mathcal{P}(y;\theta) = \theta \exp(-y\theta).$$

The joint probability density function of $y_1, \ldots, y_n$ therefore is

$$\mathcal{P}(y_1, \ldots, y_n; \theta) = \prod_{i=1}^{n} \mathcal{P}(y_i;\theta) = \theta^n \exp(-n\bar{y}\theta)$$

where $\bar{y}$ is the arithmetic mean of the y_i. This quantity is called the *likelihood function*, $\mathcal{L}(\theta; y_1, \ldots, y_n)$. This is often written more compactly as $\mathcal{L}(\theta; y)$, so that

$$\mathcal{L}(\theta; y) = \prod_{i=1}^{n} \mathcal{P}(y_i;\theta) = \theta^n \exp(-n\bar{y}\theta).$$

The maximum likelihood principle is to estimate θ by that value $\hat{\theta}$ that maximizes this joint probability function. The value of the parameter θ that maximizes the likelihood function is the *maximum likelihood estimate* (MLE) of that parameter. In this book, MLEs will be represented by placing a 'hat' over the parameter estimated, so the MLE of θ is denoted $\hat{\theta}$. For the exponential distribution example above, it is easy to show that $\mathcal{L}(\theta; y)$ is maximized with respect to θ at $1/\bar{y}$ (Problem 4.5). Hence we say that the *maximum likelihood estimator* of θ is $\hat{\theta} = 1/\bar{y}$.

Ordinarily, the probability function is viewed a function of $y_1, \ldots, y_n$ for a given parameter θ. Likelihood theory reverses the roles of the observations and the parameters, considering the probability function as a function of the parameters for a given set of observations. In practice, the *log-likelihood function* $\ell(\theta; y_1, \ldots, y_n)$, often written more compactly as $\ell(\theta; y)$, is usually more convenient to work with:

$$\ell(\theta; y) = \log \mathcal{L}(\theta; y) = \sum_{i=1}^{n} \log \mathcal{P}(y_i;\theta).$$

Obviously, maximizing the log-likelihood is equivalent to maximizing the likelihood itself. For the exponential distribution example discussed above, the log-likelihood function for θ is

$$\ell(\theta; y) = n(\log\theta - \bar{y}\theta).$$

It is easy to show that least squares is a special case of maximum likelihood. Consider a normal linear regression model, $y_i \sim N(\mu_i, \sigma^2)$, with $\mu_i = \beta_0 + \beta_1 x_{1i} + \cdots + \beta_p x_{pi}$. The normal distribution has the probability density function

$$\mathcal{P}(y_i; \mu_i, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left\{-\frac{(y_i - \mu_i)^2}{2\sigma^2}\right\}.$$

Hence the log-probability density function for y_i is

$$\log\mathcal{P}(y_i; \mu_i, \sigma^2) = -\frac{1}{2}\log(2\pi\sigma^2) - \frac{1}{2\sigma^2}(y_i - \mu_i)^2.$$

The log-likelihood function for the unknown parameters is

$$\begin{aligned}\ell(\beta_0, \ldots, \beta_p, \sigma^2; y) &= -\frac{n}{2}\log(2\pi\sigma^2) - \frac{1}{2\sigma^2}\sum_{i=1}^{n}(y_i - \mu_i)^2 \\ &= -\frac{n}{2}\log(2\pi\sigma^2) - \frac{1}{2\sigma^2}\text{RSS},\end{aligned}$$

where RSS is the sum of squares. The likelihood depends on $\beta_0, \ldots, \beta_p$ only through the RSS and so, for any fixed value of σ^2, the likelihood is maximized by minimizing the RSS. Hence maximizing the likelihood with respect to the regression coefficients β_j is the same as minimizing the sum of squares. Hence maximum likelihood is the same as least-squares for normal regression models.

Example 4.6. The total July rainfall (in millimetres) at Quilpie, Australia, has been recorded (Table 4.5; data set: `quilpie`), together with the value of the monthly mean southern oscillation index (SOI). The SOI is the standardized difference between the air pressures at Darwin and Tahiti, and is known to have relationships with rainfall in parts of Australia [10, 14]. Some Australian farmers may delay planting crops until a certain amount of rain has fallen (a 'rain threshold') within a given time frame (a 'rain window') [12]. Accordingly, we define the response variable y as

$$y = \begin{cases} 1 \text{ if the total July rainfall exceeds } 10\,\text{mm} \\ 0 \text{ otherwise.} \end{cases} \tag{4.4}$$

The unknown parameter here is the *probability* that the rainfall exceeds 10 mm, which we will write as μ because $\text{E}[y] = \mu = \Pr(y = 1)$. We will

Table 4.5 The total July rainfall (in millimetres) at Quilpie, and the corresponding SOI and SOI phase. The first six observations are shown (Example 4.6)

i	Year	Rainfall (in mm)	Rainfall exceeds 10 mm?	SOI	SOI phase
1	1921	38.4	Yes	2.7	2
2	1922	0.0	No	2.0	5
3	1923	0.0	No	−10.7	3
4	1924	24.4	Yes	6.9	2
5	1925	0.0	No	−12.5	3
6	1926	9.1	No	−1.0	4
⋮	⋮	⋮	⋮	⋮	⋮

be interested in the relationship between μ and SOI, but for the moment we ignore the SOI and consider all the observations as equivalent.

The probability function of y is defined by $\Pr(y = 1) = \mu$ and $\Pr(y = 0) = 1 - \mu$ or, more compactly, by

$$\mathcal{P}(y; \mu) = \mu^y (1 - \mu)^{1-y}, \tag{4.5}$$

for $y = 0$ or 1. This is known as a *Bernoulli distribution* with probability μ, denoted Bern(μ). The R function `dbinom()` evaluates the probability function for the binomial distribution, and when `size=1` the binomial distribution corresponds to the Bernoulli distribution. Evaluating the log-likelihood for a few test values of μ shows that the MLE of μ is near 0.5, and certainly between 0.4 and 0.6:

```
> data(quilpie); names(quilpie)
[1] "Year"   "Rain"   "SOI"    "Phase"  "Exceed" "y"
> mu <- c(0.2, 0.4, 0.5, 0.6, 0.8) # Candidate values to test
> ll <- rep(0, 5)   # A place-holder for the log-likelihood values
> for (i in 1:5)
   ll[i] <- sum( dbinom(quilpie$y, size=1, prob=mu[i], log=TRUE))
> data.frame(Mu=mu, LogLikelihood=ll)
   Mu LogLikelihood
1 0.2     -63.69406
2 0.4     -48.92742
3 0.5     -47.13401
4 0.6     -48.11649
5 0.8     -60.92148
```

Figure 4.3 plots the likelihood and log-likelihood functions for a greater range of μ values. Visually, the MLE of μ appears to be just above 0.5. □

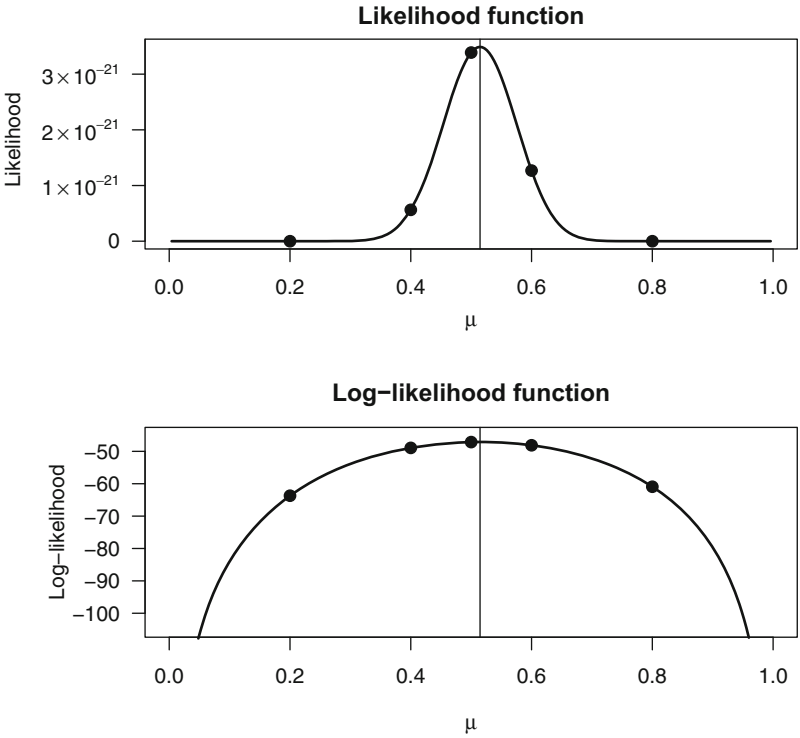

Fig. 4.3 The likelihood function (top panel) and the log-likelihood function (bottom panel) for the Quilpie rainfall data. The solid dots correspond to the five test values. The vertical line is at $\hat{\mu} = 0.5147$

4.5 Maximum Likelihood for Estimating One Parameter

4.5.1 Score Equations

A systematic approach to maximizing the log-likelihood is to use calculus, finding that value of the parameter where the derivative of the log-likelihood is zero. If there is a single parameter ζ, the derivative of the log-likelihood is called the *score function*, denoted $U(\zeta) = \mathrm{d}\ell/\mathrm{d}\zeta$, and the equation to be solved for $\hat{\zeta}$ is the *score equation* $U(\hat{\zeta}) = 0$. When there are p' unknown regression parameters, there are p' corresponding score equations.

In general in calculus, a stationary point of a function is not necessarily the global maximum—it could be merely a local maximum or even a local minimum. The log-likelihood functions considered in this book however are always unimodal and continuously differentiable in the parameters, so the score equations always yield the maximum likelihood estimators.

The score function has the important property that it has zero expectation, $\mathrm{E}[U(\zeta)] = 0$, when evaluated at the true parameter value (Problem 4.3). It follows that $\mathrm{var}[U(\zeta)] = \mathrm{E}[U(\zeta)^2]$.

Example 4.7. The log-probability function of the Bernoulli distribution (4.5) is

$$\log \mathcal{P}(y; \mu) = y \log \mu + (1 - y) \log(1 - \mu), \tag{4.6}$$

so that

$$\frac{\mathrm{d} \log \mathcal{P}(y; \mu)}{\mathrm{d}\mu} = \frac{y - \mu}{\mu(1 - \mu)}.$$

The log-likelihood function is

$$\ell(\mu; y) = \sum_{i=1}^{n} y_i \log \mu + (1 - y_i) \log(1 - \mu).$$

Hence the score function is

$$U(\mu) = \frac{\mathrm{d}\ell(\mu; y)}{\mathrm{d}\mu} = \sum_{i=1}^{n} \frac{y_i - \mu}{\mu(1 - \mu)} = \frac{\sum_{i=1}^{n} y_i - n\mu}{\mu(1 - \mu)} = \frac{n(\bar{y} - \mu)}{\mu(1 - \mu)}, \tag{4.7}$$

where $\bar{y} = (1/n) \sum_{i=1}^{n} y_i$ is the sample mean of the y_i or, in other words, the proportion of cases for which $y = 1$. Setting $U(\hat{\mu}) = 0$ and solving produces $\hat{\mu} = \bar{y}$ (Fig. 4.3); that is, the MLE of μ is the sample mean. In R:

```
> muhat <- mean(quilpie$y); muhat
[1] 0.5147059
```

□

4.5.2 Information: Observed and Expected

The previous section focused on the derivative of the log-likelihood. We now focus on the second derivative, as a measure of how well determined the MLE is. For simplicity of notation, we assume a single parameter ζ for this section.

Write $\mathcal{J}(\zeta)$ for *minus* the second derivative of the log-likelihood with respect to ζ:

$$\mathcal{J}(\zeta) = -\frac{\mathrm{d}^2 \ell(\zeta; y)}{\mathrm{d}\zeta^2} = -\frac{\mathrm{d}U(\zeta)}{\mathrm{d}\zeta}.$$

$\mathcal{J}(\zeta)$ must be positive near the MLE $\hat{\zeta}$. If it is large, then U is changing rapidly near the MLE and the peak of the log-likelihood is very sharp and hence the estimate is well-defined. In this situation, changing the estimate of ζ by a

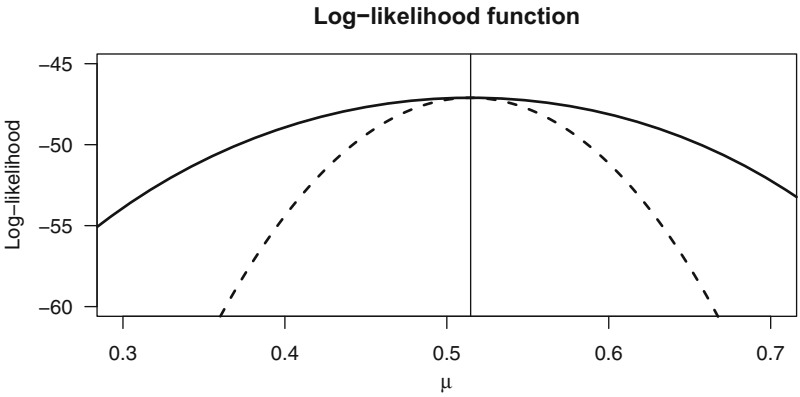

Fig. 4.4 A plot of the likelihood function for the Quilpie rainfall data (solid line), and a hypothetical log-likelihood that contains more information (dashed line). In both cases, the MLE is the same (as shown by the thin vertical line). The log-likelihood function is sharper with more information (dashed line), so that a small change in the estimate causes larger changes in the value of the log-likelihood

small amount will substantially change the value of the log-likelihood. This means that $\hat{\zeta}$ is a very precise estimate of ζ (Fig. 4.4). On the other hand, if $\mathcal{J}(\zeta)$ is close to zero, then the log-likelihood is relatively flat around $\hat{\zeta}$ and the peak is less defined. This means that $\hat{\zeta}$ is not so well determined and is a less precise estimator of ζ. All this shows that $\mathcal{J}(\zeta)$ is a measure of the precision of the estimate $\hat{\zeta}$; that is, $\mathcal{J}(\zeta)$ measures how much *information* is available for estimating ζ.

The expression $\mathcal{J}(\zeta) = -\mathrm{d}U(\zeta)/\mathrm{d}\zeta$ is called the *observed information*. We also define the *expected information* $\mathcal{I}(\zeta) = \mathrm{E}[\mathcal{J}(\zeta)]$, also called *Fisher information*. Whereas $\mathcal{J}(\zeta)$ is a function of the observed data, $\mathcal{I}(\zeta)$ is a property of the model. It measures the average information that will be observed for this parameter from this model and the specified parameter value.

The expected information $\mathcal{I}(\zeta)$ has some advantages over the observed information $\mathcal{J}(\zeta)$. First, expected information is much simpler to evaluate for the models that will be considered in this book. Second, $\mathcal{J}(\zeta)$ can only be guaranteed to be positive at $\zeta = \hat{\zeta}$, whereas $\mathcal{I}(\zeta)$ is positive for any parameter value. Third, $\mathcal{I}(\zeta)$ has a very neat relationship to the variance of the score function and to that of the MLE itself, as shown in the next section.

Example 4.8. We continue the example fitting the Bernoulli distribution to the `quilpie` data introduced in Example 4.6. The second derivative of the log-probability function (for an individual observation) is

$$\frac{\mathrm{d}^2\ell(\mu; y)}{\mathrm{d}\mu^2} = \frac{\mathrm{d}U(\mu)}{\mathrm{d}\mu} = \frac{-\mu(1-\mu) - (y-\mu)(1-2\mu)}{\mu^2(1-\mu)^2},$$

and so the observed information for μ is

$$\mathcal{J}(\mu) = -\frac{\mathrm{d}^2\ell(\mu; y)}{\mathrm{d}\mu^2} = -\sum_{i=1}^{n} \frac{\mathrm{d}^2 \log \mathcal{P}(y; \mu)}{\mathrm{d}\mu^2}$$
$$= n\frac{\mu(1-\mu) - (\hat{\mu} - \mu)(1 - 2\mu)}{\mu^2(1-\mu)^2}.$$

When we evaluate at $\mu = \hat{\mu}$, the second term in the numerator is zero, so that

$$\mathcal{J}(\hat{\mu}) = \frac{n}{\hat{\mu}(1-\hat{\mu})}.$$

Note that $\mathcal{J}(\hat{\mu})$ is positive, confirming that the second derivative is negative and hence that the log-likelihood has a maximum at $\hat{\mu}$. In fact, $\hat{\mu}$ is a global maximum of the likelihood. The expected information is

$$\mathcal{I}(\mu) = \mathrm{E}[\mathcal{J}(\mu)] = \frac{n}{\mu(1-\mu)} \tag{4.8}$$

because $\mathrm{E}[\hat{\mu}] = \mu$. Hence the observed and expected information coincide when μ is evaluated at $\mu = \hat{\mu}$. Note that the expected information increases proportionally with the sample size n. Evaluating (4.8) in R gives Fisher information:

```
> n <- length( quilpie$y )
> Info <- n / (muhat *(1-muhat))
> c(muhat=muhat, FisherInfo=Info)
      muhat  FisherInfo
  0.5147059 272.2354978
```

□

4.5.3 Standard Errors of Parameters

It can be shown that $\mathcal{I}(\zeta) = \mathrm{E}[U(\zeta)] = \mathrm{var}[U(\zeta)]$ (Problem 4.3). This states exactly how the expected information measures the rate of change in the score function around the true parameter value. A Taylor's series expansion of the log-likelihood around $\zeta = \hat{\zeta}$ shows furthermore that

$$\mathrm{var}[\hat{\zeta}] \approx 1/\mathcal{I}(\zeta). \tag{4.9}$$

Hence the expected information is a measure of the precision of the MLE; specifically, the variance of the MLE is inversely proportion to the Fisher information for the parameter. The estimated standard deviation (standard error) of $\hat{\zeta}$ is $1/\mathcal{I}(\hat{\zeta})^{1/2}$.

Example 4.9. Based on the Fisher information found in Example 4.8, the estimated standard error for $\hat{\mu}$ can be found:

```
> 1/sqrt(Info)
[1] 0.06060767
```

□

4.6 Maximum Likelihood for More Than One Parameter

4.6.1 Score Equations

Our discussion of likelihood functions so far has not included covariates and explanatory variables. The normal and non-normal regression models developed in this book will assume that each response observation y_i follows a probability distribution that is parametrised by a location parameter μ_i, actually the mean $\mu_i = \mathrm{E}[y_i]$, and dispersion parameter ϕ that specifies the variance of y_i. The mean μ_i will be assumed to be a function of explanatory variables x_{ij} and regression parameters β_j. Specifically, we will assume a linear predictor

$$\eta_i = \beta_0 + \beta_1 x_{i1} + \cdots + \beta_p x_{ip}.$$

The mean μ_i depends on the linear predictor; more precisely, $g(\mu_i) = \eta_i$ for some known function $g()$. The function $g()$ links the means to the linear predictor, and so is known as the *link function*.

For regression models, the log-likelihood function is

$$\ell(\beta_0, \beta_1, \ldots, \beta_p; y) = \sum_{i=1}^{n} \log \mathcal{P}(y_i; \mu_i, \phi).$$

The score functions have the form

$$U(\beta_j) = \frac{\partial \ell(\beta_0, \beta_1, \ldots, \beta_p; y)}{\partial \beta_j} = \sum_{i=1}^{n} \frac{\mathcal{P}(y_i; \mu_i, \phi)}{\partial \mu_i} \frac{\partial \mu_i}{\partial \beta_j},$$

with one score function corresponding to each unknown regression parameter β_j.

Example 4.10. (Data set: `quilpie`) We return to the Quilpie rainfall example (Example 4.6, p. 174), now relating the SOI to the probability that the rainfall exceeds the 10 mm threshold. Plots of the data suggest that the probability of exceeding 10 mm increases with increasing values of the SOI (Fig. 4.5):

```
> boxplot( SOI ~ Exceed, horizontal=TRUE, data=quilpie, las=2,
    xlab="July average SOI", ylab="Rainfall exceeds threshold" )
```

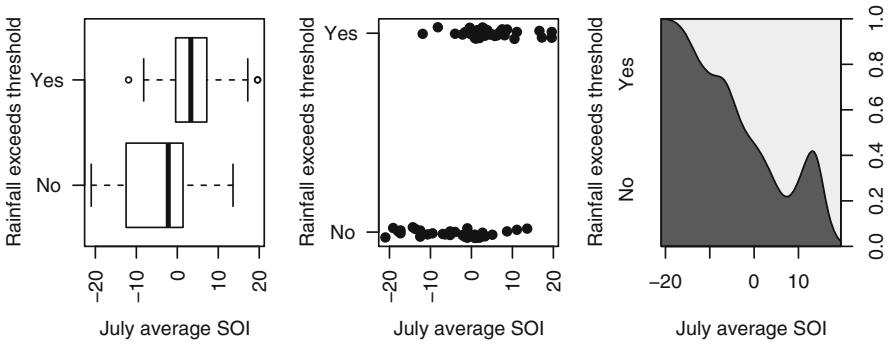

Fig. 4.5 The relationship between the SOI and exceeding the rainfall threshold of 10 mm in July at Quilpie and the SOI (Example 4.6)

```
> plot( jitter(y, 0.15) ~ SOI, data=quilpie, pch=19, axes=FALSE, las=2,
    xlab="July average SOI", ylab="Rainfall exceeds threshold" )
> axis(side=1, las=2)
> axis(side=2, at=0:1, labels=c("No", "Yes"), las=2); box()
> cdplot( Exceed ~ SOI, data=quilpie,
    xlab="July average SOI", ylab="Rainfall exceeds threshold" )
```

The left panel of Fig. 4.5 shows the distribution of the SOI in years when the rainfall exceeded and did not exceed the threshold. The centre panel of Fig. 4.5 uses the `jitter()` command to add a small amount of randomness to `y` to avoid overplotting. The right panel using a conditional density plot for the data.

Recall that $\mu = \Pr(y = 1)$ is the probability that the 10 mm threshold is exceeded. A direct linear model would assume

$$\mu = \beta_0 + \beta_1 x. \tag{4.10}$$

This, however, is not sensible for the Quilpie rainfall data. Since μ is a probability, it cannot be smaller than 0, nor larger than 1. The systematic component (4.10) cannot ensure this without imposing difficult-to-enforce constraints on the β_j. A different form of the systematic component is needed to ensure μ remains between 0 and 1.

One possible systematic component is

$$\log \frac{\mu}{1-\mu} = \eta = \beta_0 + \beta_1 x, \tag{4.11}$$

which ensures $0 < \mu < 1$. The systematic component (4.11) has two parameters to be estimated, β_0 and β_1, so there are two score functions: $U(\beta_0)$ and $U(\beta_1)$. Note that, from (4.11),

$$\frac{\partial \mu}{\partial \beta_0} = \mu(1-\mu) \qquad \text{and} \qquad \frac{\partial \mu}{\partial \beta_1} = \mu(1-\mu)x.$$

Then, working with just one observation, the score functions are

$$U(\beta_0) = \frac{\partial \log \mathcal{P}(y;\mu)}{\partial \beta_0} = \frac{\mathrm{d} \log \mathcal{P}(y;\mu)}{\mathrm{d}\mu} \times \frac{\partial \mu}{\partial \beta_0} = \; y - \mu;$$
$$U(\beta_1) = \frac{\partial \log \mathcal{P}(y;\mu)}{\beta_1} = \frac{\mathrm{d} \log \mathcal{P}(y;\mu)}{\mathrm{d}\mu} \times \frac{\partial \mu}{\partial \beta_1} = (y - \mu)x.$$

Hence the two score equations are

$$U(\hat{\beta}_0) = \sum_{i=1}^{n} y_i - \hat{\mu}_i = 0 \quad \text{and}$$
$$U(\hat{\beta}_1) = \sum_{i=1}^{n} (y_i - \hat{\mu}_i)x_i = 0,$$

where $\log\{\hat{\mu}_i/(1-\hat{\mu}_i)\} = \hat{\beta}_0 + \hat{\beta}_1 x_i$. Solving these simultaneous equations for $\hat{\beta}_0$ and $\hat{\beta}_1$ is, in general, best achieved using iterative matrix algorithms (Sect. 4.8). □

4.6.2 Information: Observed and Expected

The second derivatives of the log-likelihood, as seen earlier (Sect. 4.5.2), quantify the amount of information available for estimating parameters. For more than one parameter to be estimated, the second derivatives are

$$\mathcal{J}_{jk}(\beta) = -\frac{U(\beta_j)}{\partial \beta_k} = -\frac{\mathrm{d}U(\beta_j)}{\mathrm{d}\mu}\frac{\partial \mu}{\partial \beta_k}.$$

The expected information is, as always, $\mathcal{I}_{jk}(\beta) = \mathrm{E}[\mathcal{J}_{jk}(\beta)]$. Note that the expected information relating to regression parameter β_j is $\mathcal{I}_{jj}(\beta)$.

Example 4.11. Returning again to the Quilpie rainfall data (Example 4.6, p. 174), we can compute:

$$\mathcal{J}_{00}(\beta) = -\frac{\partial U(\beta_0)}{\partial \beta_0} = -\frac{\mathrm{d}U(\beta_0)}{\mathrm{d}\mu}\frac{\partial \mu}{\partial \beta_0} \quad = \sum_{i=1}^{n} \mu_i(1-\mu_i);$$
$$\mathcal{J}_{11}(\beta) = -\frac{\partial U(\beta_1)}{\partial \beta_1} = -\frac{\mathrm{d}U(\beta_1)}{\mathrm{d}\mu}\frac{\partial \mu}{\partial \beta_1} \quad = \sum_{i=1}^{n} \mu_i(1-\mu_i)x_i^2;$$
$$\mathcal{J}_{01}(\beta) = \mathcal{J}_{10}(\beta) = -\frac{\partial U(\beta_1)}{\partial \beta_0} = -\frac{\mathrm{d}U(\beta_1)}{\mathrm{d}\mu}\frac{\partial \mu}{\partial \beta_0} \quad = \sum_{i=1}^{n} \mu_i(1-\mu_i)x_i.$$

□

4.6.3 Standard Errors of Parameters

Similar to before,

$$\text{var}[\hat{\beta}_j] \approx 1/\mathcal{I}_{jj}(\beta),$$

so that the standard errors are $\text{se}(\hat{\beta}_j) \approx 1/\mathcal{I}_{jj}(\hat{\beta})^{1/2}$.

* 4.7 Maximum Likelihood Using Matrix Algebra

* 4.7.1 Notation

Now assume that the responses come from a probability distribution with probability function $\mathcal{P}(\mathbf{y};\boldsymbol{\zeta})$, where $\boldsymbol{\zeta} = [\zeta_1, \ldots, \zeta_q]$ is a vector of unknown parameters of the distribution. The *likelihood function* is the same as the joint probability function, only viewed as a function of the parameters:

$$\mathcal{L}(\zeta_1, \ldots, \zeta_p; y_1, \ldots, y_n) = \mathcal{L}(\boldsymbol{\zeta};\mathbf{y}) = \mathcal{P}(\mathbf{y};\boldsymbol{\zeta}). \tag{4.12}$$

In practice, the *log-likelihood function*

$$\ell(\boldsymbol{\zeta};\mathbf{y}) = \log \mathcal{L}(\boldsymbol{\zeta};\mathbf{y})$$

is usually more convenient to work with. Obviously, maximizing the log-likelihood is equivalent to maximizing the likelihood itself.

The values of the parameters $\zeta_1, \ldots, \zeta_p$ that maximize the likelihood function are the *maximum likelihood estimates* (MLE) of those parameters. In this book, MLEs will be represented by placing a 'hat' over the parameter estimated, so the MLE of $\boldsymbol{\zeta}$ is denoted $\hat{\boldsymbol{\zeta}} = [\hat{\zeta}_1, \ldots, \hat{\zeta}_p]$.

* 4.7.2 Score Equations

The first derivative of the log-likelihood with respect to $\boldsymbol{\zeta}$ is called the *score function* or *score vector* $U(\boldsymbol{\zeta})$:

$$U(\boldsymbol{\zeta}) = \frac{\partial \ell(\boldsymbol{\zeta};\mathbf{y})}{\partial \boldsymbol{\zeta}} = \sum_{i=1}^{n} \frac{\partial \log \mathcal{P}(y_i;\boldsymbol{\zeta})}{\partial \boldsymbol{\zeta}},$$

where $U(\boldsymbol{\zeta})$ is a vector of partial first derivatives, one for each parameter in $\boldsymbol{\zeta}$ such that $U(\zeta_j) = \partial \ell(\boldsymbol{\zeta};\mathbf{y})/\partial \zeta_j$. Thus, the MLE of $\boldsymbol{\zeta}$ is usually the unique solution to the score equation

$$U(\hat{\boldsymbol{\zeta}}) = \mathbf{0}. \tag{4.13}$$

In some cases, several solutions exist to (4.13), or the log-likelihood may be maximized at a boundary of the parameter space. In these cases, the log-likelihood is evaluated at all solutions to (4.13) and the boundary values, and the solution giving the maximum value is chosen. For all situations in this book, a unique maximum occurs at the solution to (4.13), unless otherwise noted. Solving (4.13) usually requires numerical methods (Sect. 4.8).

In the specific case of regression models, the parameter of interest is μ which is usually a function of explanatory variables, so estimates of μ are not of direct interest. For example, for the Quilpie rainfall data μ is assumed to be some function of the SOI x. In these situations, the estimates of the regression parameters β_j are of primary interest, so we need to evaluate the derivatives of the log-likelihood with respect to the regression parameters. For the models in this book, the linear predictor is written as

$$\boldsymbol{\eta} = \beta_0 + \beta_1 x_{i1} + \cdots + \beta_p x_{ip} = \mathrm{X}\boldsymbol{\beta}$$

where X is an $n \times p'$ matrix, and $\boldsymbol{\beta}$ is a vector of regression parameters of length p'. There will be p' score functions, one for each unknown parameter β_j, of the form:

$$U(\beta_j) = \frac{\partial \ell(\boldsymbol{\beta}; \mathbf{y})}{\partial \beta_j} = \frac{\mathrm{d}\ell(\boldsymbol{\beta}; \mathbf{y})}{\mathrm{d}\mu} \frac{\partial \mu}{\partial \beta_j}.$$

Then, $\boldsymbol{\mu} = g(\boldsymbol{\eta})$ for some known function $g()$.

Simultaneously solving the score equations is not trivial in general, and usually requires iterative numerical methods (Sect. 4.8).

Example 4.12. For the Quilpie rainfall example (data set: `quilpie`), the score equations were given in Example 4.10 for estimating the relationship between SOI and the probability that rainfall exceeds the 10 mm threshold. In matrix form, $\boldsymbol{\mu} = g(\boldsymbol{\eta}) = g(\mathrm{X}\boldsymbol{\beta})$.

The MLE $\hat{\boldsymbol{\beta}} = [\hat{\beta}_0, \ \hat{\beta}_1]$ is the solution to the *score equation*

$$U(\hat{\boldsymbol{\beta}}) = \begin{bmatrix} \sum_{i=1}^{n} \ y_i - \hat{\mu}_i \\ \sum_{i=1}^{n} (y_i - \hat{\mu}_i) x_i \end{bmatrix} = \mathbf{0}, \tag{4.14}$$

where $\log\{\hat{\boldsymbol{\mu}}/(1 - \hat{\boldsymbol{\mu}})\} = \mathrm{X}\hat{\boldsymbol{\beta}}$. Solving this score equation is not trivial. □

* 4.7.3 Information: Observed and Expected

Under certain conditions, which hold for models in this book, the *information matrix* (or the *expected information matrix*) $\mathcal{I}(\boldsymbol{\zeta})$ is defined as the negative of the expected value of the matrix of second derivatives (Problem 4.3):

$$\mathcal{I}(\boldsymbol{\zeta}) = -\mathrm{E}\left[\frac{\partial^2 \ell(\boldsymbol{\zeta}; \mathbf{y})}{\partial \boldsymbol{\zeta} \partial \boldsymbol{\zeta}^T}\right] = \mathrm{E}[\mathcal{J}(\boldsymbol{\zeta})].$$

$\mathcal{J}(\boldsymbol{\zeta})$ is called the *observed information matrix*, where element (j,k) of this matrix, denoted $\mathcal{J}_{jk}(\boldsymbol{\zeta})$, is

$$\mathcal{J}_{jk}(\boldsymbol{\zeta}) = -\frac{\partial^2 \ell(\boldsymbol{\zeta};\mathbf{y})}{\partial\zeta_j \partial\zeta_k}.$$

For the models in this book, $\boldsymbol{\eta} = \beta_0 + \beta_1 x_{i1} + \cdots + \beta_p x_{ip} = \mathrm{X}\boldsymbol{\beta}$. Then, in matrix form, the observed information for each parameter is

$$\frac{\partial^2 \ell(\boldsymbol{\beta};\mathbf{y})}{\partial\beta_j^2} = \frac{\mathrm{d}}{\mathrm{d}\mu}\left(\frac{\partial\ell(\boldsymbol{\beta};\mathbf{y})}{\partial\beta_j}\right)\frac{\partial\mu}{\partial\beta_j} = \frac{\mathrm{d}U(\beta_j)}{\mathrm{d}\mu}\frac{\partial\mu}{\partial\beta_j}.$$

The mixed derivatives are

$$\frac{\partial^2 \ell(\boldsymbol{\beta};\mathbf{y})}{\partial\beta_j\partial\beta_k} = \frac{\partial^2 \ell(\boldsymbol{\beta};\mathbf{y})}{\partial\beta_k\partial\beta_j} = \frac{\mathrm{d}}{\mathrm{d}\mu}\left(\frac{\partial\ell(\boldsymbol{\beta};\mathbf{y})}{\partial\beta_k}\right)\frac{\partial\mu}{\partial\beta_j} = \frac{\mathrm{d}U(\beta_k)}{\mathrm{d}\mu}\frac{\partial\mu}{\partial\beta_j}.$$

These derivatives can be assembled into a matrix, called the *observed information matrix*, $\mathcal{J}(\boldsymbol{\beta})$. The expected information matrix (or Fisher information matrix) is $\mathcal{I}(\boldsymbol{\beta}) = \mathrm{E}[\mathcal{J}(\boldsymbol{\beta})]$. When necessary, element (j,k) of the information matrix is denoted $\mathcal{I}_{jk}(\boldsymbol{\zeta})$.

Using these results, two important properties of the score vector (Problem 4.3) are:

1. The expected value of the score vector is zero: $\mathrm{E}[U(\boldsymbol{\zeta})] = \mathbf{0}$.
2. The variance of the score vector is $\mathrm{var}[U(\boldsymbol{\zeta})] = \mathcal{I}(\boldsymbol{\zeta}) = \mathrm{E}[U(\boldsymbol{\zeta})U(\boldsymbol{\zeta})^T]$.

Example 4.13. For the Quilpie rainfall example, expressions for the information were given in Example 4.11. Using matrices and vectors, compute (for example)

$$\frac{\partial^2 \ell(\boldsymbol{\beta};\mathbf{y})}{\partial\beta_0^2} = \frac{\mathrm{d}}{\mathrm{d}\mu}\left(\frac{\partial\ell(\boldsymbol{\beta};\mathbf{y})}{\partial\beta_0}\right)\frac{\partial\mu}{\partial\beta_0} = -\sum_{i=1}^{n}\mu_i(1-\mu_i).$$

Computing all second derivatives (Problem 4.2), the 2×2 observed information matrix $\mathcal{J}(\boldsymbol{\beta})$ is

$$\mathcal{J}(\boldsymbol{\beta}) = -\frac{\partial^2 \ell(\boldsymbol{\beta};\mathbf{y})}{\partial\boldsymbol{\beta}\partial\boldsymbol{\beta}^T} = \begin{bmatrix} \sum\mu_i(1-\mu_i) & \sum\mu_i(1-\mu_i)x_i \\ \sum\mu_i(1-\mu_i)x_i & \sum\mu_i(1-\mu_i)x_i^2 \end{bmatrix}, \tag{4.15}$$

where the summations are over $i = 1,\ldots n$, and μ_i is defined by (4.11). The expected information matrix is

$$\mathcal{I}(\boldsymbol{\beta}) = \mathrm{E}[\mathcal{J}(\boldsymbol{\beta})] = \begin{bmatrix} \sum\mu_i(1-\mu_i) & \sum\mu_i(1-\mu_i)x_i \\ \sum\mu_i(1-\mu_i)x_i & \sum\mu_i(1-\mu_i)x_i^2 \end{bmatrix}. \tag{4.16}$$

For this example, the expected information $\mathcal{I}(\boldsymbol{\beta})$ and the observed information matrix $\mathcal{J}(\boldsymbol{\beta})$ are identical, since $\mathcal{J}(\boldsymbol{\beta})$ does not contain any random components. This is not true in general. □

* 4.7.4 Standard Errors of Parameters

The variances of each parameter are found from the corresponding diagonal elements of the inverse of the information matrix:

$$\text{var}[\hat{\beta}_j] \approx \mathcal{I}_{jj}^{-1}(\boldsymbol{\beta}),$$

where $\mathcal{I}_{jk}^{-1}(\boldsymbol{\beta})$ is element (j,k) of $\mathcal{I}^{-1}(\boldsymbol{\beta})$. Hence, the standard error of each parameter is

$$\text{se}(\hat{\beta}_j) \approx \mathcal{I}_{jj}^{-1/2}(\hat{\boldsymbol{\beta}}).$$

If the off-diagonal elements of the information matrix are zero, then estimates of the corresponding parameters, or sets of parameters, are independent and can be computed separately.

Example 4.14. For the Bernoulli model fitted to the Quilpie rainfall data, use the information matrix in (4.16) to find

$$\mathcal{I}^{-1}(\hat{\boldsymbol{\zeta}}) = \frac{1}{\Delta}\begin{bmatrix} \sum \mu_i(1-\mu_i)x_i^2 & -\sum \mu_i(1-\mu_i)x_i \\ -\sum \mu_i(1-\mu_i)x_i & \sum \mu_i(1-\mu_i) \end{bmatrix},$$

where $\Delta = \sum \mu_i(1-\mu_i) \sum \mu_i(1-\mu_i)x_i^2 - \left(\sum \mu_i(1-\mu_i)x_i\right)^2$, and the summations are over $i = 1, \ldots n$. For example, the variance of $\hat{\beta}_0$ is

$$\text{var}[\hat{\beta}_0] = \frac{\sum_{i=1}^{n} \mu_i(1-\mu_i)x_i^2}{\Delta}.$$

The standard error of $\hat{\beta}_0$ is the square root of $\text{var}[\hat{\beta}_0]$ after replacing μ with $\hat{\mu}$. □

* 4.8 Fisher Scoring for Computing MLEs

By definition, the MLE occurs when $U(\hat{\boldsymbol{\zeta}}) = \mathbf{0}$ (ignoring situations where the maxima occur on the boundaries of the parameter space). Many methods exist for solving such an equality. In general, an iterative technique is needed, such as the Newton–Raphson method. In matrix form, the Newton–Raphson iteration is

$$\hat{\boldsymbol{\zeta}}^{(r+1)} = \hat{\boldsymbol{\zeta}}^{(r)} + \mathcal{J}(\hat{\boldsymbol{\zeta}}^{(r)})^{-1} U(\hat{\boldsymbol{\zeta}}^{(r)}),$$

where $\hat{\boldsymbol{\zeta}}^{(r)}$ is the estimate of $\boldsymbol{\zeta}$ at iteration r. In practice, the observed information matrix $\mathcal{J}(\boldsymbol{\zeta})$ may be difficult to compute, so the expected (Fisher) information matrix $\mathcal{I}(\boldsymbol{\zeta}) = \text{E}[\mathcal{J}(\boldsymbol{\zeta})]$ is used in place of the observed information because $\mathcal{I}(\boldsymbol{\zeta})$ usually has a simpler form than $\mathcal{J}(\boldsymbol{\zeta})$. This leads to the *Fisher scoring* iteration:

$$\hat{\boldsymbol{\zeta}}^{(r+1)} = \hat{\boldsymbol{\zeta}}^{(r)} + \mathcal{I}(\hat{\boldsymbol{\zeta}}^{(r)})^{-1} U(\hat{\boldsymbol{\zeta}}^{(r)}).$$

Example 4.15. For the Quilpie rainfall data, the score vector is given in (4.14) and the expected information matrix in (4.16). Solving the score equation is an iterative process. Start the process assuming no relationship between `y` and `SOI` (that is, setting $\hat{\beta}_1^{(0)} = 0$) and setting $\hat{\beta}_0^{(0)} = 0.5147$ (the MLE of μ computed in Example 4.6). R code for implementing the algorithm explicitly using the Fisher scoring algorithm is shown in Sect. 4.14 (p. 204). The output is shown below. The iterations converge rapidly:

```
>    # Details of the iterations, using an R function  FitModelMle()
>    # that was specifically written for this example (see Sect 4.14)
> m1.quilpie <- FitModelMle(y=quilpie$y, x=quilpie$SOI)
> m1.quilpie$coef.vec       # Show the estimates at each iteration
           [,1]      [,2]
[1,] 0.51470588 0.0000000
[2,] 0.04382413 0.1146656
[3,] 0.05056185 0.1422438
[4,] 0.04820676 0.1463373
[5,] 0.04812761 0.1464183
[6,] 0.04812757 0.1464184
[7,] 0.04812757 0.1464184
[8,] 0.04812757 0.1464184
```

The output indicates that the algorithm has converged quickly, and that the fitted model has the systematic component

$$\log \frac{\hat{\mu}}{1-\hat{\mu}} = 0.04813 + 0.1464x, \tag{4.17}$$

where x is the monthly average SOI. Figure 4.6 displays the model plotted with the data. The linear regression model with the linear systematic component (4.10) is also shown. The linear regression model is inappropriate: negative probabilities of exceeding the rainfall threshold are predicted for large

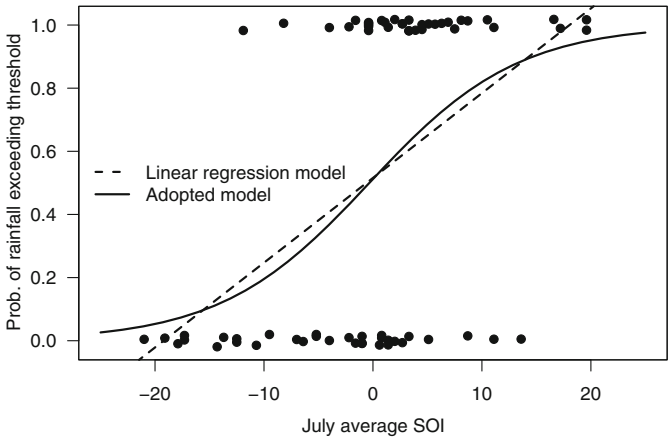

Fig. 4.6 The fitted linear regression model (4.10) and the adopted model (4.17). The points have a small amount of added randomness in the vertical direction to avoid overplotting (Example 4.10)

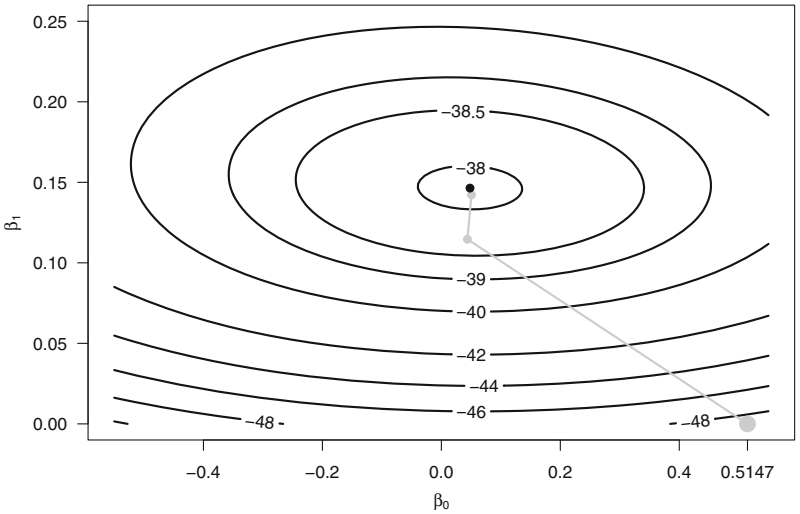

Fig. 4.7 A contour plot showing the log-likelihood function for the Quilpie rainfall data (note the contours are not equally spaced). The solid point in the centre is the maximum likelihood estimate. The gray lines and gray points show the path of the estimates on the likelihood surface; the larger gray point in the bottom right corner is the starting point (Example 4.15)

negative values of the SOI, and probabilities exceeding one are predicted for large positive values of the SOI. Figure 4.7 shows the log-likelihood surface for the example, and the progress of the iterations. □

The fitted model explains the relationship between the SOI and the probability of exceeding 10 mm of total July rainfall at Quilpie. Rearranging (4.17),

$$\hat{\mu} = \frac{1}{1 + \exp(-0.04813 - 0.1464x)}.$$

Then, $\hat{\mu} \to 0$ as $x \to -\infty$, and $\hat{\mu} \to 1$ as $x \to \infty$. This shows that larger values of the SOI are associated with higher probabilities of exceeding 10 mm, and lower values of the SOI are associated with lower probabilities of exceeding 10 mm (as seen in Fig. 4.6). When the SOI is zero, the probability of exceeding 10 mm is computed as approximately 51%.

Example 4.16. For the Bernoulli model fitted to the Quilpie rainfall data, we can continue Example 4.15. Since the values of μ_i are unknown, the diagonal elements of the inverse of the information matrix evaluated at $\hat{\boldsymbol{\mu}}$ (at the final iteration) give the *estimated* variance of the parameter estimates:

```
> inf.mat.inverse <- solve( m1.quilpie$inf.mat )
>    # Note:  'solve'  with one matrix input computes a matrix inverse
> inf.mat.inverse
             Constant             x
Constant  0.0775946484 -0.0006731683
x        -0.0006731683  0.0018385219
```

Hence the standard errors are:

```
> std.errors <- sqrt( diag( inf.mat.inverse ) )
> std.errors
  Constant          x
0.27855816 0.04287799
```

□

The Fisher scoring iteration is used for parameter estimation with GLMs used later in this book. However, writing corresponding R functions for each different model, as for the Quilpie rainfall example and shown in Sect. 4.14 (p. 204), is clearly time-consuming, error-prone and tedious. In Chap. 5, the structure of GLMs is established that enables the Fisher scoring iteration to be written in a general form applicable to all types of GLMs, and hence a common algorithm is established for fitting the models. Because of the structure established in Chap. 5, a simple-to-use R function (called `glm()`) is used to fit the generalized linear models in this book, avoiding the need to develop problem-specific R code (as in the example above).

4.9 Properties of MLEs

4.9.1 Introduction

Maximum likelihood estimators have many appealing properties, which we state in this section without proof. The properties in this section hold under standard conditions that are true for models in this book. The main assumption is that information about the unknown parameters increases with the number of observations n.

4.9.2 Properties of MLEs for One Parameter

The MLE of ζ, denoted $\hat{\zeta}$, has the following appealing properties.

1. MLEs are *invariant*. This means that if $s(\zeta)$ is a one-to-one function of ζ, then $s(\hat{\zeta})$ is the MLE of $s(\zeta)$.

2. MLEs are *asymptotically unbiased*. This means that $\mathrm{E}[\hat{\zeta}] = \zeta$ as $n \to \infty$. For small samples, the bias may be substantial. In some situations (such as the parameter estimates $\hat{\beta}_j$ in normal linear regression models), the MLE is unbiased for all n.
3. MLEs are *asymptotically efficient*. This means that no other asymptotically unbiased estimator exists with a smaller variance. Furthermore, if an efficient estimator of ζ exists, then it must be asymptotically equivalent to $\hat{\zeta}$.
4. MLEs are *consistent*. This means that the MLE converges to the true value of ζ for increasing n: $\hat{\zeta} \to \zeta$ as $n \to \infty$.
5. MLEs are *asymptotically normally distributed*. This means that if ζ_0 is the true value of ζ,
$$\hat{\zeta} \sim N\left(\zeta_0, 1/\mathcal{I}(\zeta_0)\right), \tag{4.18}$$
as $n \to \infty$, where N denotes the normal distribution. Importantly, this shows that the reciprocal of the information is the variance $\hat{\zeta}$ as $n \to \infty$:
$$\mathrm{var}[\hat{\zeta}] = 1/\mathcal{I}(\zeta_0). \tag{4.19}$$
Consequently, the standard error of $\hat{\zeta}_j$ is $\sqrt{\mathcal{I}(\zeta_0)}$.

* 4.9.3 Properties of MLEs for Many Parameters

The properties of MLEs described above can be extended to more than one parameter, using vector notation. The MLE of $\boldsymbol{\zeta}$, denoted $\hat{\boldsymbol{\zeta}}$, has the following appealing properties, which are stated without proof but which hold under standard conditions that are true for models in this book. The main assumption is that information about $\hat{\boldsymbol{\zeta}}$ (as measured by the eigenvalues of $\mathcal{I}(\boldsymbol{\zeta})$) increases with the number of observations n.

1. MLEs are *invariant*. This means that if $s(\boldsymbol{\zeta})$ is a one-to-one function of $\boldsymbol{\zeta}$, then $s(\hat{\boldsymbol{\zeta}})$ is the MLE of $s(\boldsymbol{\zeta})$.
2. MLEs are *asymptotically unbiased*. This means that $\mathrm{E}[\hat{\boldsymbol{\zeta}}] = \boldsymbol{\zeta}$ as $n \to \infty$. For small samples, the bias may be substantial. In some situations (such as the parameter estimates $\hat{\beta}_j$ in normal linear regression models), the MLE is unbiased for all n.
3. MLEs are *asymptotically efficient*. This means that no other asymptotically unbiased estimator exists with a smaller variance. Furthermore, if an efficient estimator of $\boldsymbol{\zeta}$ exists, then it must be asymptotically equivalent to $\hat{\boldsymbol{\zeta}}$.
4. MLEs are *consistent*. This means that the MLE converges to the true value of $\boldsymbol{\zeta}$ for increasing n: $\hat{\boldsymbol{\zeta}} \to \boldsymbol{\zeta}$ as $n \to \infty$.

5. MLEs are *asymptotically normally distributed*. This means that if $\boldsymbol{\zeta}_0$ is the true value of $\boldsymbol{\zeta}$,

$$\hat{\boldsymbol{\zeta}} \sim N_q\left(\boldsymbol{\zeta}_0, \mathcal{I}(\boldsymbol{\zeta}_0)^{-1}\right), \tag{4.20}$$

as $n \to \infty$, where N_q denotes the multivariate normal distribution of dimension q, and q is the length of $\boldsymbol{\zeta}$. Importantly, this shows that the inverse of the information matrix is the covariance matrix of $\hat{\boldsymbol{\zeta}}$ as $n \to \infty$:

$$\text{var}[\hat{\boldsymbol{\zeta}}] = \mathcal{I}(\boldsymbol{\zeta}_0)^{-1}. \tag{4.21}$$

Consequently, the standard error of $\hat{\zeta}_j$ is the corresponding diagonal element of $\mathcal{I}(\boldsymbol{\zeta}_0)^{-1/2}$. Equation (4.20) may be written equivalently as

$$(\hat{\boldsymbol{\zeta}} - \boldsymbol{\zeta}_0)^T \mathcal{I}(\boldsymbol{\zeta}_0)(\hat{\boldsymbol{\zeta}} - \boldsymbol{\zeta}_0) \sim \chi^2_q \tag{4.22}$$

as $n \to \infty$.

4.10 Hypothesis Testing: Large Sample Asymptotic Results

4.10.1 Introduction

After fitting a model, asking questions and testing hypotheses about the model is natural. Start by considering models with only one parameter, and hypotheses concerning this single parameter. Specifically, we test the null hypothesis that H_0: $\zeta = \zeta^0$ for some postulated value ζ^0 against the two-tailed alternative H_A: $\zeta \neq \zeta^0$.

Three methods for testing the null hypothesis H_0: $\zeta = \zeta^0$ are possible (Fig. 4.8). A *Wald test* is based on the distance between $\hat{\zeta}$ and ζ^0 (Fig. 4.8, left panel). After normalizing by an estimate of the variance of $\hat{\zeta}$, write

$$W = \frac{(\hat{\zeta} - \zeta^0)^2}{\widehat{\text{var}}[\hat{\zeta}]},$$

where $\widehat{\text{var}}[\hat{\zeta}] = 1/\mathcal{I}(\hat{\zeta})$ from (4.9). If H_0 is true, then W follows a χ^2_1 distribution as $n \to \infty$. If W is small, the distance $\hat{\zeta} - \zeta^0$ is small, which means the estimate $\hat{\zeta}$ is close to the hypothesized value ζ^0 and is evidence to support H_0.

When testing about one parameter, the square root of W is often used as the test statistic, when we write $Z = \sqrt{W}$. Then, $Z \sim N(0,1)$ as $n \to \infty$. Using Z enables testing with one-sided alternative hypotheses.

The *score test* examines the slope of the log-likelihood near ζ^0 (Fig. 4.8, centre panel). By definition, the slope of the log-likelihood is zero at $\hat{\zeta}$, so if the

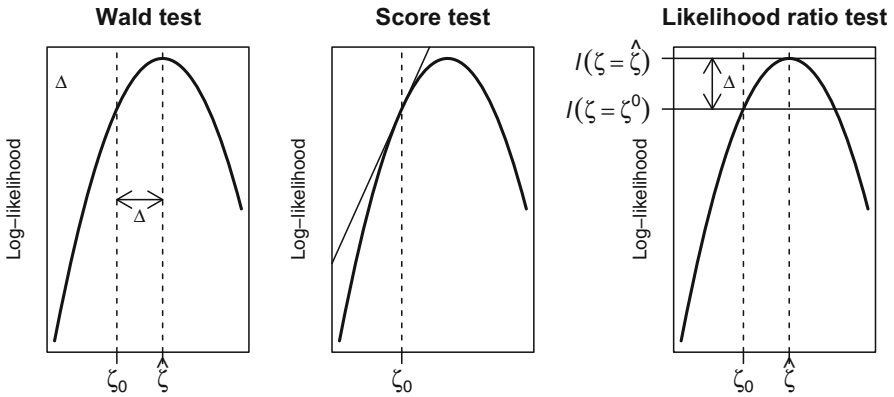

Fig. 4.8 Three ways of testing the hypothesis that $\zeta = \zeta^0$. The Wald test measures the change in the ζ dimension; the score test measures the slope of the likelihood function at ζ^0; the likelihood ratio test measures the change in the likelihood dimension. The likelihood curve is actually computed using the Quilpie rainfall data (Sect. 4.10.1)

slope of the log-likelihood at ζ^0 is near zero, then ζ^0 is near $\hat{\zeta}$. Normalizing by the variance of the slope, using $\text{var}[U(\zeta^0)] = \mathcal{I}(\zeta^0)$ from Sect. 4.5.3 (p. 179), write

$$S = \frac{U(\zeta^0)^2}{\mathcal{I}(\zeta^0)}.$$

If H_0 is true, then S follows a χ^2_1 distribution as $n \to \infty$. If S is small, then the slope at ζ^0 is close to zero, and the estimate $\hat{\zeta}$ is close to the hypothesized value ζ^0 which is evidence to support H_0. Notice that computing S does not require knowledge of $\hat{\zeta}$; instead, S is evaluated at ζ^0, so the estimate of ζ is not needed. For this reason, score tests are often simpler than Wald tests. When testing about one parameter, the square root of S is often used, where $\sqrt{S} \sim N(0, 1)$ as $n \to \infty$. Using $\sqrt{S}$ enables testing with one-sided alternative hypotheses.

The *likelihood ratio test* is based on the distance between the maximum possible value of the log-likelihood (evaluated at $\hat{\zeta}$) and the likelihood evaluated at ζ^0 (Fig. 4.8, right panel):

$$L = 2\{\ell(\hat{\zeta}; y) - \ell(\zeta^0; y)\}.$$

Twice the difference between the log-likelihoods is used, because then L follows a χ^2_1 distribution as $n \to \infty$. If L is small, then the difference between the log-likelihoods is small, and the estimate $\hat{\zeta}$ is close to the hypothesized value ζ^0 which is evidence to support H_0.

Note that W, S and L all have approximate χ^2_1 distributions. To compute P-values corresponding to each statistic, refer to a χ^2_1 distribution. As $n \to \infty$, all three test statistics are equivalent.

Example 4.17. For the Quilpie rainfall data (data file: `quilpie`), and the model based on estimating μ (and ignoring SOI), consider testing H_0: $\mu = 0.5$ using all three tests (that is, use $\mu^0 = 0.5$). For reference, recall that

$$U(\mu) = \frac{\sum_{i=1}^{n} y_i - n\mu}{\mu(1-\mu)} = \frac{n(\hat{\mu} - \mu)}{\mu(1-\mu)} \quad \text{and} \quad \mathcal{I}(\mu) = \frac{\mu(1-\mu)}{n},$$

from Examples 4.7 and 4.8. Also, $\hat{\mu} = 0.5147$ and $n = 68$. For the Wald test, compute

$$W = \frac{(\hat{\mu} - \mu^0)^2}{\hat{\mu}(1-\hat{\mu})/n},$$

where $W \sim \chi^2_1$ as $n \to \infty$. Using R:

```
> muhat <- mean( quilpie$y )
> mu0 <- 0.5
> n <- length(quilpie$y)
> varmu <- muhat*(1-muhat)/n
> W <- (muhat - mu0)^2 / varmu; W
[1] 0.05887446
```

The score statistic is

$$S = \frac{U(\mu^0)^2}{\mathcal{I}(\mu^0)} = \frac{\left(n\hat{\mu} - n\mu^0\right)^2}{n\mu^0(1-\mu^0)},$$

where $S \sim \chi^2_1$ as $n \to \infty$. Notice that

$$\sqrt{S} = \frac{\hat{\mu} - \mu^0}{\sqrt{\mu^0(1-\mu^0)/n}},$$

where $\sqrt{S} \sim N(0,1)$ as $n \to \infty$. This expression for $\sqrt{S}$ is the usual test statistic for a one-sample proportion problem. Using R:

```
> S <- (muhat - mu0)^2 / ( mu0*(1-mu0)/n ); S
[1] 0.05882353
```

For the likelihood ratio test statistic, compute the log-likelihood at μ^0 and at $\hat{\mu}$, then compute $L = 2\left\{\ell(\hat{\mu}; y) - \ell(\mu^0; y)\right\}$. Using R:

```
> Lmu0   <- sum( dbinom(quilpie$y, 1, mu0,   log=TRUE ) )
> Lmuhat <- sum( dbinom(quilpie$y, 1, muhat, log=TRUE ) )
> L <- 2*(Lmuhat - Lmu0); L
[1] 0.05883201
```

In this example, W, S and L have similar values:

```
> c( Wald=W, score=S, LLR=L)
      Wald      score        LLR
0.05887446 0.05882353 0.05883201
```

For each statistic, the asymptotic theory suggests referring to a χ^2_1 distribution. Assuming the likelihood-theory approximations are sound, the corresponding two-tailed P-values are:

```
> P.W <- pchisq(W, df=1, lower.tail=FALSE)     # Wald
> P.S <- pchisq(S, df=1, lower.tail=FALSE)     # Score
> P.L <- pchisq(L, df=1, lower.tail=FALSE)     # Likelihood ratio
> round(c(Wald=P.W, Score=P.S, LLR=P.L), 5)
   Wald   Score     LLR
0.80828 0.80837 0.80835
```

(The function `pchisq` computes the cumulative distribution function for the chi-square distribution with `df` degrees of freedom.) The two-tailed P-values and conclusions are similar in all cases: the data are consistent with the null hypothesis that $\mu = 0.5$. Recall that none of these P-values are exact; each statistic follows a χ^2_1 distribution as $n \to \infty$. □

* 4.10.2 Global Tests

The three tests used in the last section were applied when only one parameter appears in the model. These tests can also be used to test hypotheses for all parameters $\boldsymbol{\zeta}$ simultaneously in situations where more than one parameter appears. Consider testing the hypothesis H_0: $\boldsymbol{\zeta} = \boldsymbol{\zeta}^0$, where $\boldsymbol{\zeta}^0$ is the postulated value of $\boldsymbol{\zeta}$. In this context, the three test statistics are:

$$\begin{aligned} \text{Wald: } W &= (\hat{\boldsymbol{\zeta}} - \boldsymbol{\zeta}^0)^T \mathcal{I}(\hat{\boldsymbol{\zeta}})(\hat{\boldsymbol{\zeta}} - \boldsymbol{\zeta}^0); \\ \text{Score: } S &= U(\boldsymbol{\zeta}^0)^T \mathcal{I}(\boldsymbol{\zeta}^0)^{-1} U(\boldsymbol{\zeta}^0); \\ \text{Likelihood ratio: } L &= 2\{\ell(\hat{\boldsymbol{\zeta}}; \mathbf{y}) - \ell(\boldsymbol{\zeta}^0; \mathbf{y})\}. \end{aligned} \tag{4.23}$$

Large values are evidence against H_0. Each statistic follows a χ^2_q distribution as $n \to \infty$, where q is the length of $\boldsymbol{\zeta}$. This result can be used to find the corresponding two-tailed P-values.

Example 4.18. For the Quilpie rainfall data (data set: `quiplie`), consider the model with $\log\{\mu/(1-\mu)\} = \beta_0 + \beta_1 x$ where x is the value of the SOI (Example 4.10, p. 180). If $\mu = 0.5$ regardless of the SOI, then $\log\{\mu/(1-\mu)\} = 0$ for all values of the SOI. This means that $\beta_0 = \beta_1 = 0$. Hence, consider testing $\boldsymbol{\beta} = [0, 0]^T$, where $\hat{\boldsymbol{\beta}}$ is:

```
> m1.quilpie$coef
[1] 0.04812757 0.14641837
```

Note that $\boldsymbol{\beta}^0 = [0, 0]^T$, and so $(\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}^0) = \hat{\boldsymbol{\beta}}$. Also, the inverse of the information matrix is given in Example 4.14 (p. 186). Using R:

```
> beta0 <- c(0, 0); betahat <- m1.quilpie$coef
> betahat.minus.beta0 <- betahat - beta0
> W.global <- t(betahat.minus.beta0) %*% m1.quilpie$inf.mat %*%
                betahat.minus.beta0
> p.W.global <- pchisq( W.global, df=2, lower.tail=FALSE)
> round(c(W.stat=W.global, P=p.W.global), 6)
   W.stat         P
11.794457  0.002747
```

For the score test, all quantities must be computed under H_0, so the information matrix must be recomputed at $\mu = 0.5$ (the value of μ when $\boldsymbol{\beta} = [0, 0]^T$):

```
> U <- MakeScore(cbind(1, quilpie$SOI), quilpie$y, beta0)
>    # Note: MakeScore()  was written for this example (Sect. 4.14)
> inf.mat.score <- MakeExpInf( cbind(1, quilpie$SOI), 0.5)
> inf.mat.inverse <- solve( inf.mat.score )
> S.global <- t(U) %*% inf.mat.inverse %*% U
> p.S.global <- pchisq( S.global, df=2, lower.tail=FALSE)
> round(c(score.stat=S.global, P=p.S.global), 6)
score.stat          P
 15.924759   0.000348
```

For the likelihood ratio test, first compute the two likelihoods:

```
> mu <- m1.quilpie$mu
> Lbeta0 <-   sum( dbinom(quilpie$y, 1, 0.5,   log=TRUE ) )
> Lbetahat <- sum( dbinom(quilpie$y, 1, mu,  log=TRUE ) )
> L.global <- 2*(Lbetahat - Lbeta0)
> p.L.global <- pchisq( L.global, df=2, lower.tail=FALSE)
> round(c(LLR.stat=L.global, P=p.L.global), 6)
 LLR.stat         P
18.367412  0.000103
```

Recall each statistic follows a χ^2_2 distribution as $n \to \infty$. Nonetheless, the three different tests produce different two-tailed P-values:

```
> test.info <- array(dim=c(3, 2))    # Array to hold the information
> rownames(test.info) <- c("Wald","Score","Likelihood ratio")
> colnames(test.info) <- c("Test statistic","P-value")
> test.info[1,] <- c(W.global, p.W.global)
> test.info[2,] <- c(S.global, p.S.global)
> test.info[3,] <- c(L.global, p.L.global)
> round(test.info, 6)
                Test statistic  P-value
Wald                  11.79446 0.002747
Score                 15.92476 0.000348
Likelihood ratio      18.36741 0.000103
```

The conclusions will almost certainly be the same here whichever test statistic is used: the evidence is *not* consistent with H_0: $\boldsymbol{\beta} = [0, 0]^T$. The P-values from the score and likelihood ratio tests are similar, but the Wald test P-value is about ten times larger. □

* 4.10.3 Tests About Subsets of Parameters

So far, the Wald, score and likelihood ratio testing procedures have considered tests about all the parameters in the model, either the single parameter (Sect. 4.10.1) or all of the many parameters (Sect. 4.10.2). However, commonly tests are performed about *subsets* of the parameters.

To do this, partition $\boldsymbol{\zeta}$ so that $\boldsymbol{\zeta}^T = [\boldsymbol{\zeta}_1^T, \boldsymbol{\zeta}_2^T]$, where $\boldsymbol{\zeta}_1$ has length q_1 and $\boldsymbol{\zeta}_2$ has length q_2, and the null hypotheses H_0: $\boldsymbol{\zeta}_2 = \boldsymbol{\zeta}_2^0$ is to be tested. Partition the information matrix correspondingly as

$$\mathcal{I}(\hat{\boldsymbol{\zeta}}) = \begin{bmatrix} \mathcal{I}_{11} & \mathcal{I}_{12} \\ \mathcal{I}_{21} & \mathcal{I}_{22} \end{bmatrix}$$

so that $\mathcal{I}_{11}$ is a $q_1 \times q_1$ matrix, and $\mathcal{I}_{22}$ is a $q_2 \times q_2$ matrix. Then write

$$\mathcal{I}(\hat{\boldsymbol{\zeta}})^{-1} = \begin{bmatrix} \mathcal{I}^{11} & \mathcal{I}^{12} \\ \mathcal{I}^{21} & \mathcal{I}^{22} \end{bmatrix}.$$

(Note that $\mathcal{I}^{22} = (\mathcal{I}_{22} - \mathcal{I}_{21}\mathcal{I}_{11}^{-1}\mathcal{I}_{12})^{-1}$.) Consider testing H_0: $\boldsymbol{\zeta}_2 = \boldsymbol{\zeta}_2^0$ against the two-tailed alternative, where $\boldsymbol{\zeta}_2^0$ is some postulated value. $\boldsymbol{\zeta}_1$ is a *nuisance parameter*, and is free to vary without restriction. Now define $\boldsymbol{\zeta}^{*T} = [\hat{\boldsymbol{\zeta}}_1^T, \boldsymbol{\zeta}_2^{0T}]$. In other words, $\boldsymbol{\zeta}^*$ is the vector of the MLE for $\boldsymbol{\zeta}_1$ under H_0, and the value of $\boldsymbol{\zeta}_2^0$ defined in H_0. Then the three test statistics are:

$$\text{Wald: } W = (\hat{\boldsymbol{\zeta}}_2 - \boldsymbol{\zeta}_2^0)^T (\mathcal{I}^{22})^{-1} (\hat{\boldsymbol{\zeta}}_2 - \boldsymbol{\zeta}_2^0);$$

$$\text{Score: } S = U(\boldsymbol{\zeta}^*)^T \mathcal{I}(\boldsymbol{\zeta}^*)^{-1} U(\boldsymbol{\zeta}^*); \tag{4.24}$$

$$\text{Likelihood ratio: } L = 2\left\{\ell(\hat{\boldsymbol{\zeta}}; \mathbf{y}) - \ell(\boldsymbol{\zeta}^*; \mathbf{y})\right\}. \tag{4.25}$$

Each statistic follows a $\chi^2_{q_2}$ distribution as $n \to \infty$. Large values are evidence against H_0.

Example 4.19. For the Quilpie rainfall data (data file: `quiplie`), possibly SOI is not significantly related to the probability of the rainfall exceeding the threshold, and is not necessary in the model. An appropriate hypothesis to test is H_0: $\beta_1 = 0$, so that β_0 plays the role of $\boldsymbol{\zeta}_1$ and β_1 plays the role of $\boldsymbol{\zeta}_2$.

We can test the hypothesis using the score test (the Wald and likelihood ratio tests for this hypothesis will be demonstrated in Example 4.20). First, evaluate the log-likelihood where $\beta_1 = 0$:

```
> m2.quilpie <- FitModelMle(quilpie$y); m2.quilpie$coef
[1] 0.0588405
> zeta.star <- c(m2.quilpie$coef, 0) # Add the coefficient for beta1 = 0
> Xvars <- cbind( rep(1, length(quilpie$y)),  # Constant
                  quilpie$SOI )
> U.vec <- MakeScore( Xvars, y=quilpie$y, zeta.star); U.vec
              [,1]
[1,] -2.331468e-15
[2,]  1.477353e+02
```

Note that since ${\boldsymbol{\zeta}^*}^T = [\hat{\beta}_0, 0]^T$, the first element of $U(\boldsymbol{\zeta}^*)$ is zero (to computer precision) since the MLE is computed for this first parameter. Effectively, since $U(\boldsymbol{\zeta}^*)$ has only one non-zero component, the matrix computation (4.24) simplifies considerably:

```
> inf.mat2 <- MakeExpInf( Xvars, m2.quilpie$mu )
> inf.mat.inv2 <- solve( inf.mat2 )
> scoretest <- t( U.vec ) %*% inf.mat.inv2 %*% U.vec
> drop(scoretest)
[1] 15.87967
```

Since the score statistic has an approximate chi-square distribution with one degree of freedom, the two-tailed P-value is approximately

```
> p.score <- pchisq( scoretest, df=1, lower.tail=FALSE)
> drop(p.score)
[1] 6.749985e-05
```

The evidence is not consistent with $\beta_1 = 0$. □

4.10.4 Tests About One Parameter in a Set of Parameters

A common situation is to test the hypothesis H_0: $\beta_j = \beta_j^0$ when a group of parameters are in the model. This is a special case of the situation in Sect. 4.10.3 when $q_2 = 1$. While the Wald, score and likelihood ratio test statistics can all be used in this situation, the Wald statistic conveniently reduces to

$$W = \frac{(\hat{\zeta}_j - \zeta_j^0)^2}{\text{var}[\hat{\zeta}_j]}, \tag{4.26}$$

which is distributed as χ_1^2 as $n \to \infty$. In this situation, working with $Z = \sqrt{W}$ is more common (and permits one-sided alternative hypotheses), giving

$$Z = \frac{\hat{\zeta}_j - \zeta_j^0}{\sqrt{\text{var}[\hat{\zeta}_j]}}, \tag{4.27}$$

where $Z \sim N(0, 1)$ as $n \to \infty$.

The likelihood ratio test is conducted by evaluating the log-likelihood under H_0, say $\ell(\beta_j^0; y)$ (that is, setting β_j to β_j^0) and evaluating the likelihood under the alternative hypothesis, say $\ell(\hat{\beta}_j; y)$ (that is, setting β_j to $\hat{\beta}_j$), and computing $L = 2\{\ell(\hat{\beta}_j; y) - \ell(\beta_j^0; y)\}$. L follows a χ_1^2 distribution as $n \to \infty$.

Example 4.20. For the Quilpie rainfall data (data file: `quiplie`), possibly SOI is not significantly related to the probability of the rainfall exceeding the threshold. An appropriate hypothesis to test is H_0: $\beta_1 = 0$. A Wald test is conducted using either

$$W = \frac{(\hat{\beta}_1 - 0)^2}{1/\sum \mu_i(1-\mu_i)x_i^2} \quad \text{or} \quad Z = \frac{\hat{\beta}_1 - 0}{\sqrt{1/\sum \mu_i(1-\mu_i)x_i^2}},$$

using results from Examples 4.14 and 4.16. In R:

```
> m1.quilpie <- FitModelMle(y=quilpie$y, x=quilpie$SOI) # Refit
> mu <- m1.quilpie$mu
> var.beta1 <- 1 / sum( mu * (1-mu) * quilpie$SOI^2 )
> se.beta1 <- sqrt(var.beta1); Z <- m1.quilpie$coef[2] / se.beta1; Z
[1] 3.420204
```

Since $Z \sim N(0, 1)$ as $n \to \infty$, the two-tailed P-value is approximately

```
> p.Z <- 2 * pnorm( Z, lower.tail=FALSE )  # Two-tailed P-value
> round( c(Z=Z, P=p.Z), 6)
       Z        P
3.420204 0.000626
```

Exactly the same two-tailed P-value results if $W = Z^2$ is used as the test statistic, after referring to a χ_1^2 distribution:

```
> W <- Z^2; p.W <- ( pchisq( W, df=1, lower.tail=FALSE ) )
> round( c(W=W, P=p.W), 6)
        W         P
11.697796  0.000626
```

Consider testing the same hypothesis using the likelihood ratio test statistic. For the fitted model, the log-likelihood is

```
> llh.full <- sum( dbinom( quilpie$y, size=1, prob=m1.quilpie$mu) )
> llh.full
[1] 42.16348
```

Under H_0, when $\beta_1 = 0$, the model must be fitted again:

```
> ### Fit reduced model:
> m2.quilpie <- FitModelMle(quilpie$y); m2.quilpie$coef
[1] 0.0588405
```

Then the log-likelihood for this reduced model is

```
> llh.reduced <- sum( dbinom( quilpie$y, size=1, prob=m2.quilpie$mu) )
> llh.reduced
[1] 34.02941
```

The values of L and the corresponding two-tailed P-value are

```
> L <- 2*( llh.full - llh.reduced )
> p.lrt <- pchisq( L, df=1, lower.tail=FALSE)
> round( c(L=L, P=p.lrt), 6)
        L         P
16.268137  0.000055
```

The three test statistics and corresponding P-values are very similar, but different (the score test was performed in Example 4.19):

```
> test.info <- array(dim=c(3, 2))
> rownames(test.info) <- c("Wald","Score","Likelihood ratio")
> colnames(test.info) <- c("Test statistic","P-value")
> test.info[1,] <- c(W, p.W); test.info[2,] <- c(scoretest, p.score)
> test.info[3,] <- c(L, p.lrt); round(test.info, 6)
                 Test statistic  P-value
Wald                    11.69780 0.000626
Score                   15.87967 0.000067
Likelihood ratio        16.26814 0.000055
```

The data are inconsistent with the null hypothesis, and suggest SOI is necessary in the model. Again, the P-values from the score and likelihood ratio tests are similar, but the Wald test P-value is about ten times larger. □

4.10.5 Comparing the Three Methods

Three methods have been discussed for testing H_0: $\beta_1 = 0$ for the Quilpie rainfall data (Example 4.20): the Wald, score and likelihood ratio tests. While the conclusions drawn from these tests are probably the same here, the P-values are different for the three tests. The P-value from the Wald test is larger than the others by a factor of 10 approximately. Referring the statistics to a χ^2_1 distribution in each case only gives approximate P-values, as the χ^2 assumption applies asymptotically as $n \to \infty$. In practice, the asymptotic results apply when n is much larger than the number of parameters, so that all unknown parameters become well estimated. (In some cases, such as when y follows a normal distribution, the χ^2 approximations are exact even for small sample sizes.)

Of the three tests, the Wald test is usually the easiest to perform, because the necessary information (the parameter estimates and the standard errors of the parameters) are computed as a direct result of fitting the model using the algorithm in Sect. 4.8. This means that a simple explicit formula

exists for testing hypotheses about a single parameter (4.26). However, W has undesirable statistical properties, particularly with binomial distributions (Sect. 9.9). Under some circumstances, as $\hat{\zeta}_j - \zeta_j$ increases the test statistic W approaches zero, in contrast to the expectations of Fig. 4.8. This is sometimes called the Hauck–Donner effect [8]. The results from the score and likelihood ratio tests are more reliable.

Score tests often require less computational effort. For example, score tests concerning β_j do not require the estimate $\hat{\beta}_j$. Likelihood ratio tests require two models to be fitted: the model under the null hypothesis and the model under the alternative hypothesis.

4.11 Confidence Intervals

* 4.11.1 Confidence Regions for More Than One Parameter

For the Wald, score and likelihood ratio statistics, confidence intervals can be formed for parameters. A joint $100(1-\alpha)\%$ confidence *region* for all the unknown parameters $\boldsymbol{\zeta}$ simultaneously can be obtained from the Wald, score or likelihood ratio statistics, as the two vector solutions to

$$\text{Wald:} \qquad (\hat{\boldsymbol{\zeta}} - \boldsymbol{\zeta})^T \mathcal{I}(\hat{\boldsymbol{\zeta}})(\hat{\boldsymbol{\zeta}} - \boldsymbol{\zeta}) \leq \chi^2_{q,1-\alpha} \tag{4.28}$$

$$\text{Score:} \qquad U(\boldsymbol{\zeta})^T \mathcal{I}(\boldsymbol{\zeta})^{-1} U(\boldsymbol{\zeta}) \leq \chi^2_{q,1-\alpha} \tag{4.29}$$

$$\text{Likelihood ratio:} \qquad 2\left\{\ell(\hat{\boldsymbol{\zeta}}; \mathbf{y}) - \ell(\boldsymbol{\zeta}; \mathbf{y})\right\} \leq \chi^2_{q,1-\alpha} \tag{4.30}$$

where $\boldsymbol{\zeta}$ is the true value, and q is the length of $\boldsymbol{\zeta}$. General solutions to these equations are difficult to find. The intervals are only approximate in general, as they are based on the distributional assumptions which apply as $n \to \infty$.

4.11.2 Confidence Intervals for Single Parameters

A confidence interval for a single parameter ζ_j (Fig. 4.9) has the limits of the confidence interval as the two values of ζ_j satisfying the appropriate condition (4.28)–(4.30). Wald confidence intervals are based on the values of ζ at a given distance either side of $\hat{\zeta}$. Score confidence intervals are based on the values of ζ at which the *slope* of the likelihood function meets appropriate criteria. Likelihood-ratio confidence intervals are based on the values of ζ such that difference between the maximum value of the likelihood and the likelihood function meet appropriate criteria.

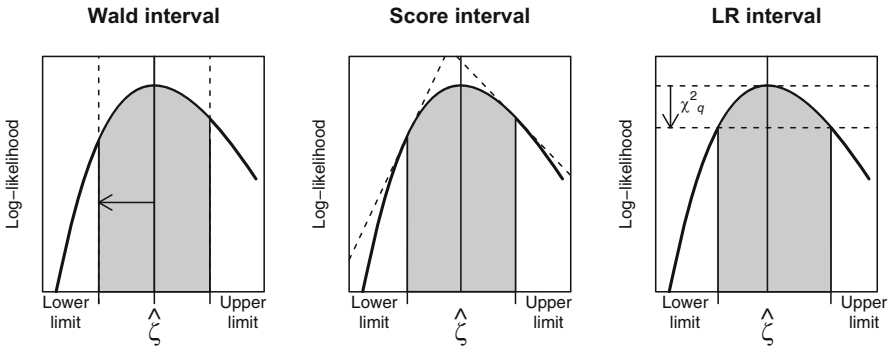

Fig. 4.9 Three ways of computing confidence intervals for a one-dimensional situation. The Wald confidence interval is symmetric by definition; the score and the likelihood ratio confidence intervals are not necessarily symmetric (Sect. 4.11)

For a single parameter, the approximate $100(1-\alpha)\%$ confidence interval based on the Wald statistic is obtained directly from (4.27):

$$\hat{\zeta}_j - z^* \sqrt{\text{var}[\hat{\zeta}_j]} < \zeta_j < \hat{\zeta}_j + z^* \sqrt{\text{var}[\hat{\zeta}_j]}$$

where z^* is the quantile of the standard normal distribution such that an area $\alpha/2$ is in each tail. Wald confidence intervals are most commonly used, because this explicit solution is available, and because $\hat{\zeta}_j$ and $\sqrt{\text{var}[\hat{\zeta}_j]}$ are found directly from the fitting algorithm (Sect. 4.8). Note the confidence interval is necessarily symmetric for the Wald statistic.

Confidence intervals for single parameters based on the score and likelihood statistics are harder to find, as they require numerically solving the corresponding equations that come from the relevant statistics. The limits of the confidence interval are the two solutions to

$$\text{Score:} \qquad U(\zeta)^2 / \mathcal{I}(\zeta) \leq \chi^2_{1,1-\alpha} \tag{4.31}$$

$$\text{Likelihood ratio:} \qquad 2\left\{\ell(\hat{\zeta}; y) - \ell(\zeta; y)\right\} \leq \chi^2_{1,1-\alpha} \tag{4.32}$$

Example 4.21. Consider the model fitted to the Quilpie rainfall data (data file: `quiplie`) using SOI as an explanatory variable (Example 4.6, p. 174), and finding a confidence interval for β_1. The log-likelihood evaluated at the MLEs of β_0 and β_1 is $\ell(\hat{\beta}_0, \hat{\beta}_1; y) = -37.95$ and $\chi^2_{1,1-\alpha} = 3.841$ for a 95% confidence interval. Then, from (4.30), the limits of the confidence interval are the two solutions to

$$2\left\{-37.95 - \ell(\hat{\beta}_0, \beta_1; y)\right\} = 3.841, \tag{4.33}$$

Table 4.6 Confidence intervals for β_1, using the Wald, score and likelihood ratio statistics. Note that $\hat{\beta}_1 = 0.1464$ (Sect. 4.11)

Type of interval	Lower	Upper
Wald:	0.06238	0.2305
Score:	0.06552	0.2289
Likelihood-ratio:	0.07191	0.2425

a non-linear equation which must be solved numerically. One solution will be less than $\hat{\beta}_1 = 0.1464$, and one solution greater than $\hat{\beta}_1 = 0.1464$.

In Fig. 4.9, confidence intervals are shown based on the Wald, score and likelihood-ratio statistics. The Wald confidence interval is symmetric, by definition. The confidence intervals based on the score and log-likelihood functions are not necessarily symmetric (Table 4.6), since the log-likelihood function is not exactly symmetric about $\hat{\beta}_1$. □

4.12 Comparing Non-nested Models: The AIC and BIC

In Sect. 2.11, the AIC and BIC were used to compare non-nested linear regression models. More generally, the AIC and BIC can be used to compare any non-nested models based on a specific probability distribution, by using the log-likelihood and penalizing the complexity of models. Formally, the AIC is defined [1] in terms of the log-likelihood as

$$\begin{aligned} \text{AIC} = &-2\ell(\hat{\zeta}_1, \ldots, \hat{\zeta}_p; y) + \\ &2 \times (\text{Number of unknown parameters}), \end{aligned} \tag{4.34}$$

where $\ell(\hat{\zeta}_1, \ldots, \hat{\zeta}_p; y)$ is the log-likelihood evaluated at the MLEs for the model under consideration. The AIC penalizes the log-likelihood by the number of unknown parameters using $k = 2$. Using this definition, smaller values of the AIC (closer to $-\infty$) represent better models.

Similarly, the BIC is defined as

$$\begin{aligned} \text{BIC} = &-2\ell(\hat{\zeta}_1, \ldots, \hat{\zeta}_p; y) + \\ &(\log n) \times (\text{Number of unknown parameters}). \end{aligned} \tag{4.35}$$

The BIC penalizes the log-likelihood by the number of unknown parameters using $k = 2 \log n$. The results in Sect 2.11 (p. 70) are simply those for (4.34) and (4.35) applied to normal linear regression models (Problem 4.10), ignoring all constants.

Example 4.22. Consider the model `quilpie.m1` fitted to the Quilpie rainfall data `quiplie` in Example 4.15 (p. 187). The AIC and BIC are:

```
> LLH <- m1.quilpie$LLH
> m1.aic <- -2 * LLH + 2 * length(m1.quilpie$coef)
> m1.bic <- -2 * LLH + log(length(quilpie$y)) * length(m1.quilpie$coef)
> c(AIC=m1.aic, BIC=m1.bic)
     AIC      BIC
79.90060 84.33962
```

Rather than using the SOI as an explanatory variable, an alternative is to use the SOI *phases* [14]. The SOI can be classified into one of five phases, depending on the SOI in the current and previous months (see `?quilpie` for more details). For five SOI phases, four dummy variables are needed, so the total number of estimated parameters is five (including the constant). The fitted model is:

```
> quilpie$Phase <- factor( quilpie$Phase )
> Xvars <- with( quilpie, model.matrix( ~ Phase ) ) # Create dummy vars
> head(Xvars)
  (Intercept) Phase2 Phase3 Phase4 Phase5
1           1      1      0      0      0
2           1      0      0      0      1
3           1      0      1      0      0
4           1      1      0      0      0
5           1      0      1      0      0
6           1      0      0      1      0
> phase.quilpie <- FitModelMle(quilpie$y, x=Xvars, add.constant=FALSE )
```

(Notice the use of `model.matrix()` to automatically define the dummy variables for SOI phases.) The two models `m1.quilpie` and `phase.quilpie` are not nested, so comparing the models using the likelihood ratio test is inappropriate. Instead, the AIC and BIC are:

```
> LLH <- phase.quilpie$LLH
> m2.aic <- -2 * LLH + 2 * length(phase.quilpie$coef)
> m2.bic <- -2 * LLH + log(length(quilpie$y)) * length(phase.quilpie$coef)
> c( "AIC (SOI model)"=m1.aic, "AIC (SOI Phase model)"=m2.aic)
      AIC (SOI model) AIC (SOI Phase model)
              79.90060              75.79902
> c( "BIC (SOI model)"=m1.bic, "BIC (SOI Phase model)"=m2.bic)
      BIC (SOI model) BIC (SOI Phase model)
              84.33962              86.89656
```

The AIC suggests that the model using the SOI phases makes better predictions than using the SOI, as the AIC for the SOI model is closer to $-\infty$. In contrast, the BIC suggests that the model using the SOI is a superior model.

□

4.13 Summary

Chapter 4 discusses situations where linear regression models do not apply, and explores the theory of likelihood methods for estimation in these contexts.

We considered three important cases for which linear regression models fail (Sect. 4.2):

- The response y is a *proportion* of a total number of counts, where $0 \leq y \leq 1$.
- The response y is a *count*, where $y = 0, 1, 2, \ldots$.
- The response y is *positive continuous*, where $y > 0$.

A more general approach to regression models assumes the responses belong to a *family* of distributions (Sect. 4.3).

For these models, maximum likelihood methods (Sect. 4.4) can be used for estimation and hypothesis testing. We consider the one parameter (Sect. 4.5) and two-parameter (Sect. 4.6) cases separately, and then the case of many parameters using matrix algebra (Sect. 4.7).

Estimation using maximum likelihood includes a discussion of the score equations (Sect. 4.5.1) the observed and expected information (Sect. 4.5.2) and standard errors (Sect. 4.5.3). Then, the Fisher scoring algorithm for finding the maximum likelihood estimates was detailed (Sect. 4.8). Maximum likelihood estimators are invariant, asymptotically unbiased, asymptotically efficient, consistent, and asymptotically normally distributed (Sect. 4.9).

Three types of inference are suggested by maximum likelihood methods: Wald, score and likelihood ratio (Sect. 4.10 for hypothesis testing;. Sect. 4.11 for confidence intervals). Asymptotic results are available for describing the distribution of the Wald, score and likelihood ratio statistics, which apply as $n \to \infty$ (Sect. 4.10). Non-nested models can be compared using the AIC or the BIC (Sect. 4.12).

* 4.14 Appendix: R Code to Fit Models to the Quilpie Rainfall Data

In Example 4.15 (p. 187), a model was fitted to the Quilpie rainfall data using the ideas in Sect. 4.8 (p. 186). The R code used to fit these models is shown below. The purpose of the code is to demonstrate the application of the ideas and formulae, and is not optimal R programming (for example, there is no error checking). Later (Chap. 6), built-in R functions are described to fit these models without the need to use these functions. Notes on writing R functions are given in Sect. A.3.11.

```
# Function for computing the information matrix:
MakeExpInf <- function(x, mu){
  # Args:
```

```
  #   x: The matrix of explanatory variables
  #   mu: The fitted values
  #
  # Returns:
  #   The expected information matrix
  if ( length(mu) == 1 ) mu <- rep( mu, dim(x)[1])
  mu <- as.vector(mu)
  return( t(x) %*% diag(  mu * (1 - mu) ) %*% x )
}
# Function for computing mu:
MakeMu <- function(x, beta){
  # Args:
  #   x: The matrix of explanatory variables
  #   beta: The linear model parameter estimates
  #
  # Returns:
  #   The value of mu
  eta <- x %*% beta
  return( 1 / ( 1 + exp( -eta ) ) )
}
# Function for computing the score vector:
MakeScore <- function(x, y, beta){
  # Args:
  #   x: The matrix of explanatory variables
  #   y: The response variable
  #   beta: The linear model parameter estimates
  #
  # Returns:
  #   The score matrix
  mu <- MakeMu(x, beta)
  return( t(x) %*% (y - mu) )
}
FitModelMle <- function(y, x=NULL, maxits=8, add.constant=TRUE){
  # Args:
  #   y:    The response variable
  #   x:    The matrix of explanatory variables
  #   maxits: The maximum number of iteration for the algorithm
  #   add.constant: If TRUE, a constant is added to the x matrix
  #      (All models must have a constant term.)
  #
  # Returns:
  #   Information about the fitted glm
  if ( is.null(x)){ # If no x given, ensure constant appears
     allx <- cbind( Constant=rep( 1, length(y) ) )
  } else {
    allx <- x
    if( add.constant ){
      allx <- cbind( Constant=rep(1, length(y)), x)
    }
  }
  num.x.vars <- dim(allx)[2] - 1  # Take one, because of constant

  # Find initials: beta_0 = mean(y), and the other beta_j are zero
  beta <- c( mean(y), rep( 0, num.x.vars ) )
```

```
  # Set up
  beta.vec <- array( dim=c(maxits, length(beta) ) )
  beta.vec[1,] <- beta
  mu <- MakeMu( allx, beta )
  score.vec <- MakeScore(allx, y, beta)
  inf.mat <- MakeExpInf( allx, mu )

  # Now iterate to update
  for (i in (2:maxits)){
     beta <- beta + solve( inf.mat ) %*% score.vec
     beta.vec[i,] <- beta

     mu <- MakeMu( allx, beta )
     score.vec <- MakeScore(allx, y, beta)
     inf.mat <- MakeExpInf( allx, mu )
  }

  # Compute log-likelihood
  LLH <- sum( y*log(mu) + (1-y)*log(1-mu) )

  return( list(coef = beta.vec[maxits,],  # MLE of parameter estimates
               coef.vec = beta.vec,       # Estimates at each iteration
               LLH = LLH,                 # The maximum log-likelihood
               inf.mat = inf.mat,         # The information matrix
               score.vec = score.vec,     # The score vector
               mu = mu) )                 # The fitted values
}
```

Problems

Selected solutions begin on p. 534. Problems preceded by an asterisk * refer to the optional sections in the text, and may require matrix manipulations.

4.1. Show that an approximation to the Wald statistic can be developed from the second-order Taylor expansion of the log-likelihood as follows. For this problem, focus on just one of the regression parameters, say β_j.

1. Write the first three terms of the Taylor series expansion of $\ell(\beta_j; y)$ expanded about $\hat{\beta}_j$.
2. Rearrange to show that the Wald statistic is approximately equal to $2\{\ell(\beta_j; y) - \ell(\hat{\beta}_j; y)\}$, and hence show that the Wald statistic is approximately equivalent to a likelihood ratio test when $\beta_j - \hat{\beta}_j$ is small.

* **4.2.** In Example 4.10 (p. 180), the information matrix was given for the Bernoulli model fitted to the Quilpie rainfall data. Prove the result in (4.15).

* **4.3.** In Sect. 4.7.3 (p. 184), two statements were made concerning the log-likelihood, which we now prove. In this question, assume y is continuous.

1. Working with just one observation, use the definition of the expected value to show that

$$\mathrm{E}[U(\boldsymbol{\zeta})] = \int_{-\infty}^{\infty} \frac{\partial \mathcal{P}(y;\boldsymbol{\zeta})}{\partial \boldsymbol{\zeta}}\, dy. \tag{4.36}$$

   Then use (4.36) to show that $\mathrm{E}[U(\boldsymbol{\zeta})] = \mathbf{0}$.
2. Using that $\mathrm{E}[U(\boldsymbol{\zeta})] = \mathbf{0}$ and the definition of the variance, show that $\mathrm{var}[U(\boldsymbol{\zeta})] = \mathrm{E}[U(\boldsymbol{\zeta})U(\boldsymbol{\zeta})^T]$, which is $\mathcal{I}(\boldsymbol{\zeta})$ (assuming the order of the integration and differentiation can be reversed).

4.4. The normal distribution $N(\mu, \sigma^2)$ has the probability function

$$\mathcal{P}(y;\mu,\sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left\{-\frac{(y-\mu)^2}{2\sigma^2}\right\},$$

for $\sigma > 0$, $-\infty < \mu < \infty$ and $-\infty < y < \infty$. Consider estimating the mean μ for the normal distribution when σ^2 is known, based on a sample $y_1, \ldots, y_n$.

1. Determine the likelihood function and the log-likelihood function.
2. Find the score function.
3. Using the score function, find the MLE of μ.
4. Find the observed and expected information for μ.
5. Find the standard error for $\hat{\mu}$.
6. Find the Wald test statistic W for testing H_0: $\mu = 0$.
7. Find the score test statistic S for testing H_0: $\mu = 0$.
8. Find the likelihood ratio test statistic L for testing H_0: $\mu = 0$.
9. Show that $W = S = L$ in this example.

4.5. The exponential distribution has the probability function

$$\mathcal{P}(y;\mu) = \exp(-y/\mu)/\mu, \tag{4.37}$$

for $\mu > 0$ and $y > 0$. Consider estimating the mean μ for the exponential distribution based on a sample $y_1, \ldots, y_n$.

1. Determine the likelihood function and the log-likelihood function.
2. Find the score function.
3. Using the score function, find the MLE of μ.
4. Find the observed and expected information for μ.
5. Show that the standard error for $\hat{\mu}$ is $\mathrm{se}(\hat{\mu}) = \hat{\mu}/\sqrt{n}$.
6. Show that the Wald test statistic for testing H_0: $\mu = 1$ is $W = (\hat{\mu} - 1)^2/(\hat{\mu}^2/n)$.
7. Show that the score test statistic for testing H_0: $\mu = 1$ is $S = n(\hat{\mu} - 1)^2$.
8. Show that the likelihood ratio test statistic for testing H_0: $\mu = 1$ is $L = 2n(\hat{\mu} - \log\hat{\mu} - 1)$.
9. Plot W, S and L for values of μ between 0.5 and 2, for $n = 10$. Comment.
10. Plot W, S and L for values of μ between 0.5 and 2, for $n = 100$. Comment.

4.6. Use the R function `rexp()` to generate $n = 100$ random numbers from the exponential distribution (4.37) with $\mu = 1$. (In R, the parameter of the exponential distribution is the rate where the `rate` is $1/\mu$.)

1. Use R to plot the likelihood function for the randomly generated data from $\mu = 0.75$ to $\mu = 1.25$. Use vertical lines to show the location of $\hat{\mu}$ and $\mu^0 = 1$.
2. Test the hypothesis H_0: $\mu = 1$ using the Wald, score and likelihood ratio statistics developed in Problem 4.5.
3. Plot the Wald, score and likelihood ratio test statistics against possible values of μ. Use a horizontal line to show the location of the critical value of χ^2_1. Compare the values of the test statistics for various values of $\hat{\mu}$.
4. Find the standard error of $\hat{\mu}$.
5. Find a 95% confidence interval for μ using the Wald statistic.

* **4.7.** Consider a model based on the exponential distribution (4.37), where $\log \mu = \beta_0 + \beta_1 x$. Consider estimating the regression parameters based on a sample $y_1, \ldots, y_n$.

1. Show that the score vector has elements

$$\frac{\partial \ell}{\partial \beta_0} = \sum_{i=1}^{n} \frac{y_i - \mu_i}{\mu_i} \qquad \text{and} \qquad \frac{\partial \ell}{\partial \beta_1} = \sum_{i=1}^{n} \frac{(y_i - \mu_i)x_i}{\mu_i}.$$

2. Show that the second derivatives of the log-likelihood are

$$\frac{\partial^2 \ell}{\partial \beta_0^2} = -\sum_{i=1}^{n} \frac{y_i}{\mu_i}; \qquad \frac{\partial^2 \ell}{\partial \beta_1^2} = -\sum_{i=1}^{n} \frac{y_i x_i^2}{\mu_i}; \qquad \frac{\partial^2 \ell}{\partial \beta_0 \partial \beta_1} = -\sum_{i=1}^{n} \frac{y_i x_i}{\mu_i}.$$

3. Using the results above, determine an expression for $\text{se}(\hat{\beta}_1)$.
4. Define the Wald test statistic for testing H_0: $\beta_1 = 0$.

4.8. The Poisson distribution has the probability function

$$\mathcal{P}(y; \mu) = \frac{\exp(-\mu)\mu^y}{y!},$$

for $\mu > \infty$ and where y is a non-negative integer. Initially, consider estimating the mean μ for the Poisson distribution, based on a sample $y_1, \ldots, y_n$.

1. Determine the likelihood function and the log-likelihood function.
2. Find the score function $U(\mu)$.
3. Using the score function, find the MLE of μ.
4. Find the observed and expected information for μ.
5. Find the standard error for $\hat{\mu}$.

* **4.9.** Following Problem 4.8, now consider the case where $\log \mu = \beta_0 + \beta_1 x$.

1. Find the score functions $U(\beta_0)$ and $U(\beta_1)$.

2. Find the observed and expected information matrices.
3. Hence find the standard errors of β_0 and β_1.

4.10. Using the definition of the AIC in (4.34), show that the formulae for computing the AIC in normal linear regression models is given by AIC $= n\log(\text{RSS}/n) + 2p'$, as shown in (2.35) (p. 71), after ignoring all constants.

References

[1] Akaike, H.: A new look at the statistical model identification. IEEE Transactions on Automatic Control **19**(6), 716–723 (1974)

[2] Box, G.E.P.: Science and statistics. Journal of the American Statistical Association **71**, 791–799 (1976)

[3] Chatterjee, S., Handcock, M.S., Simonoff, J.S.: A Casebook for a First Course in Statistics and Data Analysis. John Wiley and Sons, New York (1995)

[4] Dala, S.R., Fowlkes, E.B., Hoadley, B.: Risk analysis of the space shuttle: pre-Challenger prediction of failure. Journal of the American Statistical Association **84**(408), 945–957 (1989)

[5] Dunn, P.K., Smyth, G.K.: Randomized quantile residuals. Journal of Computational and Graphical Statistics **5**(3), 236–244 (1996)

[6] Feigl, P., Zelen, M.: Estimation of exponential survival probabilities with concomitant information. Biometrics **21**, 826–838 (1965)

[7] Glaister, D.H., Miller, N.L.: Cerebral tissue oxygen status and psychomotor performance during lower body negative pressure (LBNP). Aviation, Space and Environmental Medicine **61**(2), 99–105 (1990)

[8] Hauck Jr., W.W., Donner, A.: Wald's test as applied to hypotheses in logit analysis. Journal of the American Statistical Association **72**, 851–853 (1977)

[9] Maron, M.: Threshold effect of eucalypt density on an aggressive avian competitor. Biological Conservation **136**, 100–107 (2007)

[10] McBride, J.L., Nicholls, N.: Seasonal relationships between Australian rainfall and the southern oscillation. Monthly Weather Review **111**(10), 1998–2004 (1983)

[11] Montgomery, D.C., Peck, E.A.: Introduction to Regression Analysis. Wiley, New York (1992)

[12] Pook, M., Lisson, S., Risbey, J., Ummenhofer, C.C., McIntosh, P., Rebbeck, M.: The autumn break for cropping in southeast Australia: trends, synoptic influences and impacts on wheat yield. International Journal of Climatology **29**, 2012–2026 (2009)

[13] Smyth, G.K.: Australasian data and story library (OzDASL) (2011). URL http://www.statsci.org/data

[14] Stone, R.C., Auliciems, A.: SOI phase relationships with rainfall in eastern Australia. International Journal of Climatology **12**, 625–636 (1992)

Chapter 5
Generalized Linear Models: Structure

Models are useful distillations of reality. Although wrong by definition, they are the wind that blows away the fog and cuts through the untamed masses of data to let us see answers to our questions.
Keller [4, p. 97]

5.1 Introduction and Overview

Chapters 2 and 3 considered linear regression models. These models assume constant variance, which demonstrably is not true for all data, as shown in Chap. 4. Generalized linear models (GLMs) assume the responses come from a distribution that belongs to a more general *family* of distributions, and also permit more general systematic components. We first review the two components of a GLM (Sect. 5.2) then discuss in greater detail the family of distributions upon which the random component is based (Sect. 5.3), including writing the probability functions in the useful dispersion model form (Sect. 5.4). The systematic component of the GLM is then considered in greater detail (Sect. 5.5). Having discussed the two components of the GLM, GLMs are then formally defined (Sect. 5.6), and the important concept of the deviance function is introduced (Sect. 5.7). Finally, using a GLM is compared to using a regression model after transforming the response (Sect. 5.8).

5.2 The Two Components of Generalized Linear Models

Generalized linear models (GLMs) are regression models (Sect. 1.6), and so consist of a random component and a systematic component. The random and systematic components take specific forms for GLMs, which depend on the answers to the following questions:

1. What probability distribution is appropriate? The answer determines the random component of the model. The choice of probability distribution may be suggested by the response data (for example, proportions of a total suggest a binomial distribution), or knowledge of how the variance changes with the mean.

P. K. Dunn, G. K. Smyth, *Generalized Linear Models with Examples in R*,
Springer Texts in Statistics, https://doi.org/10.1007/978-1-4419-0118-7_5

2. How are the explanatory variables related to the mean of the response μ? The answer suggests the systematic component of the model. GLMs assume a function linking the *linear predictor* $\eta = \beta_0 + \sum_{j=1}^{p} \beta_j x_j$ to the mean μ, such as $\log \mu = \eta$ for example. That is, GLMs are regression models linear in the parameters.

5.3 The Random Component: Exponential Dispersion Models

5.3.1 Examples of EDMs

GLMs assume the responses come from a distribution that belongs to a *family* of distributions called the *exponential dispersion model* family (or EDM family, or just EDMs). *Continuous* EDMs include the normal and gamma distributions. *Discrete* EDMs include the Poisson, binomial and negative binomial distributions. The EDM family of distributions enables GLMs to be fitted to a wide range of data types, including binary data (Chap. 4), proportions (Chap. 9), counts (Chap. 10), positive continuous data (Chap. 11), and positive continuous data with exact zeros (Chap. 12).

5.3.2 Definition of EDMs

Distributions in the EDM family have a probability function (a probability *density* function if y is continuous; a probability *mass* function if y is discrete) of the form

$$\mathcal{P}(y; \theta, \phi) = a(y, \phi) \exp\left\{ \frac{y\theta - \kappa(\theta)}{\phi} \right\} \tag{5.1}$$

where

- θ is called the *canonical parameter*.
- $\kappa(\theta)$ is a known function, and is called the *cumulant function*.
- $\phi > 0$ is the *dispersion parameter*.
- $a(y, \phi)$ is a normalizing function ensuring that (5.1) is a probability function. That is, $a(y, \phi)$ is the function of ϕ and y ensuring that $\int \mathcal{P}(y; \theta, \phi)\, dy = 1$ over the appropriate range if y is continuous, or the function ensuring that $\sum_y \mathcal{P}(y; \theta, \phi)\, dy = 1$ if y is discrete. The function $a(y, \phi)$ cannot always be written in closed form.

The mean μ is a known function of the canonical parameter θ (Sect. 5.3.5). The notation $y \sim \text{EDM}(\mu, \phi)$ indicates that the responses come from a distribution in the EDM family (5.1), with mean μ and dispersion parameter ϕ. Definition (5.1) writes the form of an EDM in *canonical form*. Other parameterizations are also possible, and the dispersion model form (Sect. 5.4) is particularly important.

The support of y (the set of possible values for y) is denoted by S, where S does not depend on the parameters θ and ϕ. The domain of θ, denoted Θ, is an open interval of values satisfying $\kappa(\theta) < \infty$ that includes zero. The corresponding domain of μ is denoted Ω.

Example 5.1. The probability density function for the normal distribution with mean μ and variance σ^2 is

$$\begin{aligned}\mathcal{P}(y;\mu,\sigma^2) &= \frac{1}{\sqrt{2\pi\sigma^2}}\exp\left\{-\frac{(y-\mu)^2}{2\sigma^2}\right\} \\ &= \frac{1}{\sqrt{2\pi\sigma^2}}\exp\left\{\frac{y\mu-(\mu^2/2)}{\sigma^2}-\frac{y^2}{2\sigma^2}\right\}.\end{aligned} \tag{5.2}$$

Comparing to (5.1), $\theta = \mu$ is the canonical parameter, $\kappa(\theta) = \mu^2/2 = \theta^2/2$ is the cumulant function, $\phi = \sigma^2$ is the dispersion parameter, and $a(y,\phi) = (2\pi\sigma^2)^{-1/2}\exp\{-y^2/(2\sigma^2)\}$ is the normalizing function. The normal distribution is an EDM. □

Example 5.2. The Poisson probability function is usually written

$$\mathcal{P}(y;\mu) = \frac{\exp(-\mu)\mu^y}{y!}$$

for $\mu > 0$ and $y = 0, 1, 2, \ldots$. In the form of (5.1),

$$\mathcal{P}(y;\mu) = \exp\{y\log\mu - \mu - \log(y!)\},$$

showing that $\theta = \log\mu$ is the canonical parameter, $\kappa(\theta) = \mu$, and $\phi = 1$. The normalizing function is $a(y,\phi) = 1/y!$. The Poisson distribution is an EDM. □

Example 5.3. The binomial probability function is

$$\begin{aligned}\mathcal{P}(y;\mu,m) &= \binom{m}{my}\mu^y(1-\mu)^{m(1-y)} \\ &= \binom{m}{my}\exp\left[m\left\{y\log\frac{\mu}{1-\mu}+\log(1-\mu)\right\}\right],\end{aligned} \tag{5.3}$$

where $y = 0, 1/m, 2/m, \ldots 1$, and $0 < \mu < 1$. Comparing to (5.1), $\theta = \log\{\mu/(1-\mu)\}$ is the canonical parameter, $\kappa(\theta) = -\log(1-\mu)$, $\phi = 1/m$ and $a(y,\phi) = \binom{m}{my}$. The binomial distribution is an EDM when m is known. □

Example 5.4. The Weibull distribution has the probability function

$$\mathcal{P}(y;\alpha,\gamma) = \frac{\alpha}{\gamma}\left(\frac{y}{\gamma}\right)^{\alpha-1}\exp\left\{-\left(\frac{y}{\gamma}\right)^{\alpha}\right\}$$

for $y > 0$ with $\alpha > 0$ and $\gamma > 0$. Rewriting,

$$\mathcal{P}(y;\alpha,\gamma) = \exp\left\{-\left(\frac{y}{\gamma}\right)^{\alpha} + \log\left(\frac{\alpha}{\gamma}\right) + (\alpha-1)\log\frac{y}{\gamma}\right\}.$$

Inside the exponential function, a term of the form $y\theta$ cannot be extracted unless $\alpha = 1$. Hence, the Weibull distribution is not an EDM in general. When $\alpha = 1$, the probability function is

$$\mathcal{P}(y;\gamma) = \exp(-y/\gamma)/\gamma = \exp\left\{-(y/\gamma) - \log\gamma\right\},$$

which is the exponential distribution (4.37) with mean γ. The exponential distribution written in this form is an EDM where $\theta = -1/\gamma$ is the canonical parameter, $\kappa(\theta) = \log\gamma$ and $\phi = 1$. □

5.3.3 Generating Functions

EDMs have many important and useful properties. One useful property is that the moment generating function (MGF) always has a simple form, even if the probability function cannot be written in closed form. The mean and variance may be found from this simple MGF.

The *moment generating function*, denoted $M(t)$, for some variable y with probability function $\mathcal{P}(y)$ is

$$M(t) = \mathrm{E}[e^{ty}] = \begin{cases} \displaystyle\int_S \mathcal{P}(y)e^{ty}\,dy & \text{for } y \text{ continuous} \\ \displaystyle\sum_{y\in S} \mathcal{P}(y)e^{ty} & \text{for } y \text{ discrete,} \end{cases}$$

for all values of t for which the expectation exists. The *cumulant generating function* (or CGF) is then defined as

$$K(t) = \log M(t) = \log \mathrm{E}[e^{ty}],$$

for all values of t for which the expectation exists. The CGF is used to derive the *cumulants* of a distribution, such as the mean (first cumulant, κ_1) and the variance (second cumulant, κ_2). The rth cumulant, κ_r, is

$$\kappa_r = \left.\frac{\mathrm{d}^r K(t)}{\mathrm{d}t^r}\right|_{t=0} \tag{5.4}$$

where the notation means to evaluate the indicated derivative at $t = 0$. Using the CGF, the mean and variance are (Problem 5.4):

$$\mathrm{E}[y] = \kappa_1 = \left.\frac{\mathrm{d}K(t)}{\mathrm{d}t}\right|_{t=0} \quad \text{and} \quad \mathrm{var}[y] = \kappa_2 = \left.\frac{\mathrm{d}^2K(t)}{\mathrm{d}t^2}\right|_{t=0}. \tag{5.5}$$

5.3.4 The Moment Generating and Cumulant Functions for EDMs

The MGF, and hence CGF, for an EDM has a very simple form. The MGF is developed here for a continuous response, but the results also hold for discrete distributions (Problem 5.6).

Using (5.1), the MGF for an EDM is

$$\begin{aligned} M(t) &= \mathrm{E}[\exp(ty)] \\ &= \int_S \exp(ty)a(y,\phi)\exp\left\{\frac{y\theta - \kappa(\theta)}{\phi}\right\} dy \\ &= \exp\left\{\frac{\kappa(\theta') - \kappa(\theta)}{\phi}\right\}\int_S a(y,\phi)\exp\left\{\frac{y\theta' - \kappa(\theta')}{\phi}\right\} dy, \end{aligned}$$

where $\theta' = \theta + t\phi$. The integral on the right is one, since the integrand is an EDM density function (5.1) written in terms of θ' rather than θ. This means that the MGF and cumulant generating function (CGF) for an EDM are

$$M(t) = \exp\left\{\frac{\kappa(\theta + t\phi) - \kappa(\theta)}{\phi}\right\}; \tag{5.6}$$

$$K(t) = \frac{\kappa(\theta + t\phi) - \kappa(\theta)}{\phi}. \tag{5.7}$$

Using (5.7), the rth cumulant for an EDM is (Problem 5.5)

$$\kappa_r = \phi^{r-1}\frac{\mathrm{d}^r\kappa(\theta)}{\mathrm{d}\theta^r}. \tag{5.8}$$

For this reason, $\kappa(\theta)$ is called the *cumulant function.*

Example 5.5. For the normal distribution, the results in Example 5.1 can be used with (5.7) to obtain

$$K(t) = \frac{(\mu + t\sigma^2)^2}{2\sigma^2} - \frac{\mu^2}{2\sigma^2} = \mu t + \frac{\sigma^2 t^2}{2}.$$

□

Example 5.6. For the Poisson distribution, the results in Example 5.2 can be used to obtain $K(t) = \mu(\exp t - 1)$. □

5.3.5 The Mean and Variance of an EDM

The mean and variance of an EDM are found by applying (5.8) to (5.5):

$$\mathrm{E}[y] = \mu = \frac{\mathrm{d}\kappa(\theta)}{\mathrm{d}\theta} \quad \text{and} \quad \mathrm{var}[y] = \phi \frac{\mathrm{d}^2\kappa(\theta)}{\mathrm{d}\theta^2}. \tag{5.9}$$

Observe that

$$\frac{\mathrm{d}^2\kappa(\theta)}{\mathrm{d}\theta^2} = \frac{\mathrm{d}}{\mathrm{d}\theta}\left(\frac{\mathrm{d}\kappa(\theta)}{\mathrm{d}\theta}\right) = \frac{\mathrm{d}\mu}{\mathrm{d}\theta}.$$

Since $\mathrm{d}^2\kappa(\theta)/\mathrm{d}\theta^2 > 0$ is a variance, then $\mathrm{d}\mu/\mathrm{d}\theta > 0$. This means that μ must be a monotonically increasing function of θ, so μ and θ are one-to-one functions of each other. Hence, define

$$V(\mu) = \frac{\mathrm{d}\mu}{\mathrm{d}\theta}, \tag{5.10}$$

called the *variance function*. Then the variance of y can be written as

$$\mathrm{var}[y] = \phi V(\mu). \tag{5.11}$$

The variance is a product of the dispersion parameter ϕ and $V(\mu)$. Table 5.1 (p. 221) gives the variance function for common EDMs.

Example 5.7. For the normal distribution (Example 5.1; Table 5.1), $\kappa(\theta) = \theta^2/2$, and so $\mathrm{E}[y] = \mathrm{d}\kappa(\theta)/\mathrm{d}\theta = \theta$. Since $\theta = \mu$ for the normal distribution, $\mathrm{E}[y] = \theta = \mu$ (as expected). For the variance, compute $V(\mu) = \mathrm{d}^2\kappa(\theta)/\mathrm{d}\theta^2 = 1$, and so $\mathrm{var}[y] = \phi V(\mu) = \sigma^2$ as expected. □

Example 5.8. For the Poisson distribution (Example 5.2; Table 5.1), $\kappa(\theta) = \mu$ and $\theta = \log \mu$. The mean is

$$\mathrm{E}[y] = \frac{\mathrm{d}\kappa}{\mathrm{d}\theta} = \frac{\mathrm{d}\kappa}{\mathrm{d}\mu} \times \frac{\mathrm{d}\mu}{\mathrm{d}\theta} = \mu$$

as expected. For the variance function, $V(\mu) = \mathrm{d}\mu/\mathrm{d}\theta = \mu$. Since $\phi = 1$ for the Poisson distribution, $\mathrm{var}[y] = \mu$ for the Poisson distribution. □

5.3.6 The Variance Function

The variance function $V(\mu)$ *uniquely* determines the distribution within the class of EDMs since the variance function determines $\kappa(\theta)$, up to an additive constant. This in turn specifies $K(t)$, which uniquely characterizes the distribution.

To demonstrate, consider EDMs with $V(\mu) = \mu^2$. Since $V(\mu) = \mathrm{d}\mu/\mathrm{d}\theta$ from (5.10), solve $\mathrm{d}\theta/\mathrm{d}\mu = \mu^{-2}$ for θ to obtain $\theta = -1/\mu$, setting the integration constant to zero. Then using that $\mu = \mathrm{d}\kappa(\theta)/\mathrm{d}\theta$ from (5.9) together with $\theta = -1/\mu$ shows that $\kappa(\theta) = -\log(-\theta) = \log\mu$. Using these forms for θ and $\kappa(\theta)$, the EDM uniquely corresponding to $V(\mu) = \mu^2$ has the probability function

$$\mathcal{P}(y) = a(y,\phi)\exp\left\{\frac{y(-1/\mu) - \log\mu}{\phi}\right\},$$

for an appropriate normalizing function $a(y;\phi)$. The constants of integration are not functions of μ, so are absorbed into $a(y,\phi)$ if not set to zero. This probability function is the probability function for a gamma distribution. Hence, the variance function $V(\mu) = \mu^2$ *uniquely* refers to a gamma distribution within the EDM class of distributions.

This result means that if the mean–variance relationship can be established for a given data set, and quantified using the variance function, the corresponding EDM is uniquely identified.

In general, (5.11) states that, in general, the variance of an EDM depends on the mean. The normal distribution is unique in the family of EDMs, as its variance does not depend on the mean since $V(\mu) = 1$. For other EDMs, the variance is a function of the mean, and the role of the variance function is to specify exactly that function.

Example 5.9. For the noisy miner data [6] in Table 1.2 (data set: `nminer`), divide the data into five approximately equal-sized groups:

```
> data(nminer)
> breaks <- c(-Inf, 4, 11, 15, 19, Inf) + 0.5   # Break points
> Eucs.cut <- cut(nminer$Eucs, breaks ); summary(Eucs.cut)
 (-Inf,4.5]  (4.5,11.5] (11.5,15.5] (15.5,19.5] (19.5, Inf]
          9           6           5           6           5
```

For each group, compute the mean and variance of the number of noisy miners:

```
> mn <- tapply( nminer$Minerab, Eucs.cut, "mean" )    # Mean of each group
> vr <- tapply( nminer$Minerab, Eucs.cut, "var" )     # Var of each group
> sz <- tapply( nminer$Minerab, Eucs.cut, "length" )  # Num. in each group
> cbind("Group size"=sz, "Group mean"=mn, "Group variance"=vr)
            Group size Group mean Group variance
(-Inf,4.5]          9  0.1111111      0.1111111
(4.5,11.5]          6  0.5000000      1.5000000
(11.5,15.5]         5  3.8000000     11.2000000
```

```
(15.5,19.5]          6  4.3333333      7.8666667
(19.5, Inf]          5  7.0000000     48.5000000
```

The command `tapply(nminer$Minerab, Eucs.cut, "mean")` computes the `mean()` of `nminer$Minerab` for each level of `Eucs.cut`. More generally, `tapply(X, INDEX, FUN)` applies the function `FUN()` to the data `X`, for each group of values in the unique combination of factors in `INDEX`.

A plot of the logarithm of each group mean against the logarithm of each group variance (Fig. 5.1, right panel) shows that, in general, the variance increases as the mean increases:

```
> plot(jitter(Minerab)~(Eucs), pch=1, las=1, data=nminer, ylim=c(0, 20),
    xlab="Number of eucalypts/2 ha.", ylab="Number of noisy miners")
>    #  Draw the dashed vertical lines
> abline(v=breaks, lwd=1, lty=2, col="gray")
> plot( log( vr ) ~ log ( mn ), pch=19, las=1, cex=0.45*sqrt(sz),
    xlab="Log of means", ylab="Log of variances" )
```

(The points are plotted so that the area is proportional to the sample size. The scaling factor `0.45` is chosen by trial-and-error.) More specifically, an approximate linear a relationship of the form

$$\log(\text{group variance}) = a + b\log(\text{group mean})$$

may be reasonable (Fig. 5.1, right panel). This is equivalent to (group variance) $\propto$ (group mean)^b. This is the form of the variance of an EDM: $\text{var}[y] = \phi V(\mu)$, where $V(\mu) = \mu^b$ and where b is the slope of the linear relationship:

```
> hm.lm <- lm( log( vr ) ~ log ( mn ), weights=sz )
> coef(hm.lm); confint(hm.lm)
(Intercept)      log(mn)
   0.802508     1.295222
                   2.5 %    97.5 %
(Intercept) 0.007812159 1.597204
log(mn)     0.821058278 1.769386
```

For the data, the slope of the linear regression line (weighted by the number of observations in each group) is $b \approx 1.3$, suggesting the mean is approximately proportional to the variance. In addition, the estimate of ϕ is approximately 1 as needed for the Poisson distribution. In other words, $V(\mu) = \mu$ approximately. Since this is the variance function for a Poisson distribution (Table 5.1), a Poisson distribution may be suitable for the data. Of course, the Poisson distribution is also suggested because the data are counts. □

5.4 EDMs in Dispersion Model Form

5.4.1 The Unit Deviance and the Dispersion Model Form

We have shown that μ and θ are one-to-one functions of each another. As a result, it must be possible to write the probability function (5.1) as a function

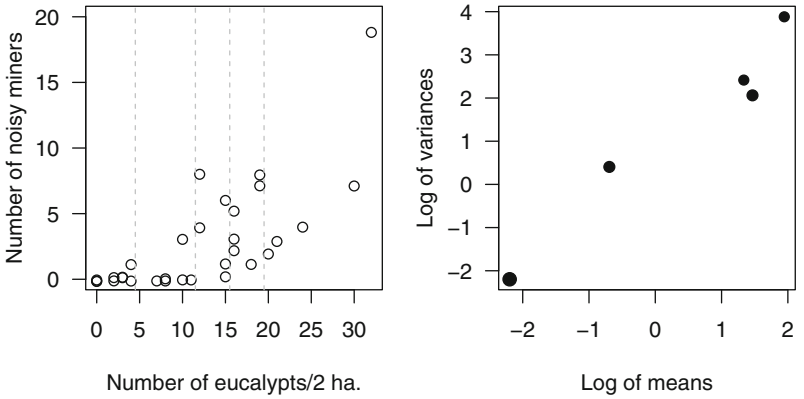

Fig. 5.1 Plots of the noisy miner data. Left: the number of noisy miners plotted against the number of eucalypt trees (a small amount of randomness is added in the vertical direction to the number of noisy miners to avoid over-plotted observations). The dashed vertical lines break the data into five groups of similar size. Right panel: the logarithm of sample variances for each group plotted against the logarithm of the sample means for each group in the data; the area of the plotted points are proportional to the number of observations in each group (Example 5.9)

of μ instead of θ. We will see that this version has some advantages because μ has such a clear interpretation as the mean of the distribution. To do this, start by writing

$$t(y, \mu) = y\theta - \kappa(\theta)$$

for that part of the probability function which depends on θ. There must be some function $t(\cdot, \cdot)$ for which this is true. Now consider $t(y, \mu)$ as a function of μ. See that

$$\frac{\partial t(y, \mu)}{\partial \theta} = y - \frac{\mathrm{d}\kappa(\theta)}{\mathrm{d}\theta} = y - \mu$$

and

$$\frac{\partial^2 t(y, \mu)}{\partial \theta^2} = \frac{\mathrm{d}^2 \kappa(\theta)}{\mathrm{d}\theta^2} = V(\mu) > 0.$$

The second derivative is always positive, and the first derivative is zero at $y = \mu$, so $t(y, \mu)$ must have a unique maximum with respect to μ at $\mu = y$. This allows us to define a very important quantity, the *unit deviance*:

$$d(y, \mu) = 2\left\{t(y, y) - t(y, \mu)\right\}. \tag{5.12}$$

Notice that $d(y, \mu) = 0$ only when $y = \mu$ and otherwise $d(y, \mu) > 0$. In fact, $d(y, \mu)$ increases as μ moves away from y in either direction. This shows that $d(y, \mu)$ can be interpreted as a type of distance measure between y and μ.

In terms of the unit deviance, the probability function (5.1) for an EDM is

$$\mathcal{P}(y;\mu,\phi) = b(y,\phi)\exp\left\{-\frac{1}{2\phi}d(y,\mu)\right\}, \tag{5.13}$$

where $b(y,\phi) = a(y,\phi)\exp\{t(y,y)/\phi\}$, which cannot always be written in closed form. This is called the *dispersion model* form of the probability function for EDMs, and is invaluable for much of what follows.

Example 5.10. For the normal distribution (Example 5.1), deduce that $t(y,\mu) = y\mu - \mu^2/2$ and so $t(y,y) = y^2 - y^2/2 = y^2/2$. The unit deviance then is $d(y,\mu) = (y-\mu)^2$. Hence the normal distribution written as (5.2) is in dispersion model form. □

The above definition for the unit deviance assumes that we can always set μ equal to y. However, cases exist when values of y are not allowable values for μ. The important cases occur when y is on the boundary of the support of the distribution. For example, the binomial distribution requires $0 < \mu < 1$, so setting $\mu = y$ is not possible when $y = 0$ or $y = 1$. However μ can still take values arbitrarily close to y. To cover these cases, we generalize the definition of the unit deviance to

$$d(y,\mu) = 2\left\{\lim_{\epsilon\to 0} t(y+\epsilon, y+\epsilon) - t(y,\mu)\right\}. \tag{5.14}$$

If y is on the lower boundary of S, the right limit will be taken. If y is at the upper bound (such as $y = 1$ for the binomial), then the left limit is taken. This definition covers all the distributions considered in this book. For simplicity, the unit deviance is usually written as (5.12), on the understanding that (5.14) is used when necessary. The unit deviances for common EDMs are in Table 5.1 (p. 221).

Example 5.11. Consider the Poisson distribution in Example 5.2 (p. 213), for which $\mu > 0$. Deduce that $t(y,\mu) = y\log\mu - \mu$. If $y \neq 0$, then $t(y,y) = y\log y - y$, so that

$$d(y,\mu) = 2\left\{y\log\frac{y}{\mu} - (y-\mu)\right\}. \tag{5.15}$$

If $y = 0$ we need the limit form (5.14) of the unit deviance instead. It is easily seen that $\lim_{\epsilon\downarrow 0} t(y+\epsilon, y+\epsilon) = 0$ so that

$$d(0,\mu) = 2\mu. \tag{5.16}$$

The unit deviance is commonly written as (5.15) on the understanding that the limit form (5.16) is used when $y = 0$. The other terms in the dispersion model form (5.13) are $b(y) = (y\log y - y)/y!$ and $\phi = 1$. □

As already noted, the unit deviance is a measure of the discrepancy between y and μ. For normal distributions, the unit deviance $d(y,\mu) = (y-\mu)^2$ (Example 5.10) is symmetric about μ as a function of y. For other EDMs, the

Table 5.1 Common EDMs, showing their variance function $V(\mu)$, cumulant function $\kappa(\theta)$, canonical parameter θ, dispersion parameter ϕ, unit deviance $d(y,\mu)$, support S (the permissible values of y), domain Ω for μ and domain Θ for θ. For the Tweedie distributions, the case $\xi = 2$ is the gamma distribution, and $\xi = 1$ with $\phi = 1$ is the Poisson distribution. $\mathbb{R}$ refers to the real line; $\mathbb{N}$ refers to the natural numbers $1, 2, \ldots$; superscript $+$ means positive values only; superscript $-$ means negative values only; subscript 0 means zero is included in the space (Sect. 5.3.5)

EDM	$V(\mu)$	$\kappa(\theta)$	θ	ϕ	$d(y,\mu)$	S	Ω	Θ	Reference
Normal	1	$\theta^2/2$	μ	σ^2	$(y-\mu)^2$	$\mathbb{R}$	$\mathbb{R}$	$\mathbb{R}$	Chaps. 2 and 3
Binomial	$\mu(1-\mu)$	$\dfrac{\exp\theta}{1+\exp\theta}$	$\log\dfrac{\mu}{1-\mu}$	$\dfrac{1}{m}$	$2\left\{y\log\dfrac{y}{\mu}+(1-y)\log\dfrac{1-y}{1-\mu}\right\}$	$\dfrac{0,1,\ldots m}{m}$	$(0,1)$	$\mathbb{R}$	Chap. 9
Negative binomial	$\mu+\dfrac{\mu^2}{k}$	$-\log(1-\exp\theta)$	$\log\dfrac{\mu}{\mu+k}$	1	$2\left\{y\log\dfrac{y}{\mu}-(y+k)\log\dfrac{y+k}{\mu+k}\right\}$	$\mathbb{N}_0$	$\mathbb{R}^+$	$\mathbb{R}^-$	Chap. 10
Poisson	μ	$\exp\theta$	$\log\mu$	1	$2\left\{y\log\dfrac{y}{\mu}-(y-\mu)\right\}$	$\mathbb{N}_0$	$\mathbb{R}^+$	$\mathbb{R}$	Chap. 10
Gamma	μ^2	$-\log(-\theta)$	$-\dfrac{1}{\mu}$	ϕ	$2\left\{-\log\dfrac{y}{\mu}+\dfrac{y-\mu}{\mu}\right\}$	$\mathbb{R}^+$	$\mathbb{R}^+$	$\mathbb{R}$	Chap. 11
Inverse Gaussian	μ^3	$-\sqrt{-2\theta}$	$-\dfrac{1}{2\mu^2}$	ϕ	$\dfrac{(y-\mu)^2}{\mu^2 y}$	$\mathbb{R}^+$	$\mathbb{R}^+$	$\mathbb{R}_0^-$	Chap. 11
Tweedie ($\xi \le 0$ or $\xi \ge 1$)	μ^ξ	$\dfrac{\{(1-\xi)\theta\}^{(2-\xi)/(1-\xi)}}{2-\xi}$ for $\xi \ne 2$	$\dfrac{\mu^{1-\xi}}{1-\xi}$ for $\xi \ne 1$	ϕ	$2\left\{\dfrac{\max(y,0)^{2-\xi}}{(1-\xi)(2-\xi)}-\dfrac{y\mu^{1-\xi}}{1-\xi}+\dfrac{\mu^{2-\xi}}{2-\xi}\right\}$ for $\xi \ne 1, 2$	$\xi < 0$: $\mathbb{R}$ $1 < \xi < 2$: $\mathbb{R}_0^+$ $\xi > 2$: $\mathbb{R}^+$	$\mathbb{R}^+$ $\mathbb{R}^+$ $\mathbb{R}^+$	$\mathbb{R}_0^+$ $\mathbb{R}^-$ $\mathbb{R}_0^-$	Chap. 12

unit deviance is asymmetric (Fig. 5.2), because differences relative to the variance are important. For example, consider the unit deviance for the gamma distribution (which has $V(\mu) = \mu^2$) with $\mu = 3$ (Fig. 5.2, bottom left panel). The unit deviance is greater at $y = 1$ than at $y = 5$ even though the absolute difference $|y - \mu| = 2$ is the same in both cases. This is because the variance is smaller at $y = 1$ than at $y = 5$, so the difference between y and μ is greater in standard deviation terms.

Technical note. All the EDM distributions used for examples in this book have the property that the domain Ω for μ is the same as the support for y, at least in a limiting sense. (Technically, the support for y is contained in the closure of the domain for μ.) However, EDMs exist for which the allowable values for μ are far more restricted than those for y. Chapter 12 will discuss Tweedie models with power variance functions $V(\mu) = \mu^{\xi}$. When $\xi < 0$, the resulting distributions can take all values y on the real line, whereas the mean is restricted to be positive, $\mu > 0$. To cover such distributions, the definition of the unit deviance can be generalized further to

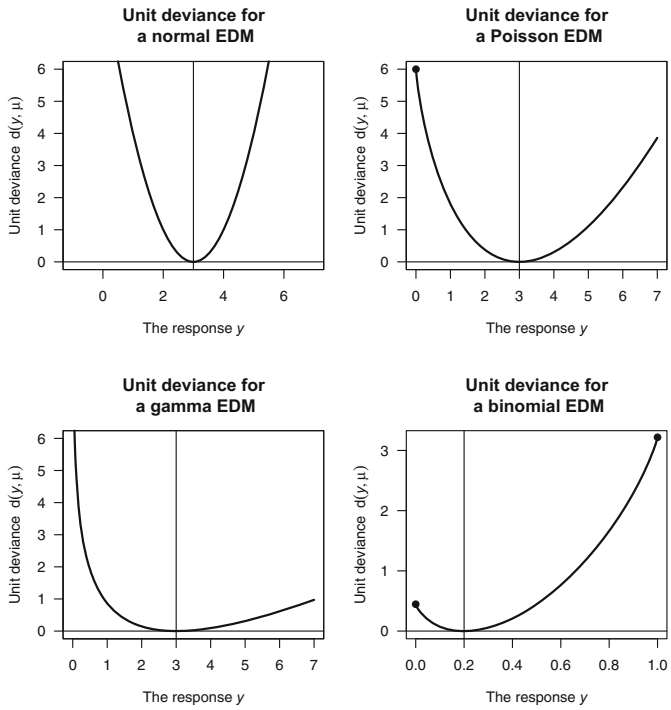

Fig. 5.2 The unit deviance $d(y, \mu)$ for four EDMs. Top left panel: the unit deviance for the normal distribution when $\mu = 3$; top right panel: the unit deviance for the Poisson distribution when $\mu = 3$; bottom left panel: the unit deviance for the gamma distribution when $\mu = 3$ and $\phi = 1$; bottom right: the unit deviance for the binomial distribution when $\mu = 0.2$. The solid points show where the limit form of the unit deviance (5.14) has been used (Sect. 5.11)

$$d(y,\mu) = 2\left\{\sup_{\mu\in\Omega} t(y,\mu) - t(y,\mu)\right\} \tag{5.17}$$

where the notation 'sup' is short for 'supremum'. However such distributions do not have any useful applications for modelling real data, as least not yet, so we can ignore this technicality in practice. The limiting definition (5.14) given previously is adequate for all applications considered in this book.

5.4.2 The Saddlepoint Approximation

The *saddlepoint approximation* to the EDM density function $\mathcal{P}(y;\mu,\phi)$ is defined by

$$\tilde{\mathcal{P}}(y;\mu,\phi) = \frac{1}{\sqrt{2\pi\phi V(y)}} \exp\left\{-\frac{d(y,\mu)}{2\phi}\right\}. \tag{5.18}$$

The saddlepoint approximation is often remarkably accurate, even in the extreme tails of the distribution. As well as being computationally useful in some cases, the approximation aids our theoretical understanding of the properties of EDMs.

For practical use, the term $V(y)$ in the denominator of (5.18) is usually modified slightly so that it can never take the value zero [7]. For example, the saddlepoint approximation to the Poisson or negative binomial distributions can be improved by replacing $V(y)$ with $V(y+\epsilon)$ where $\epsilon = 1/6$. The saddlepoint approximation, adjusted in this way, has improved accuracy everywhere as well as having the advantage of being defined at $y = 0$. This is called the *modified saddlepoint approximation*.

Comparing to the dispersion model form (5.13) (p. 220), the saddlepoint approximation (5.18) is equivalent to writing $b(y,\phi) \approx 1/\sqrt{2\pi\phi V(y)}$. Observe that $b(y,\phi)$, which for some EDMs isn't available in any closed form, is approximated by a simple analytic function.

Example 5.12. For the normal distribution, $V(\mu) = 1$ so that $V(y) = 1$. Applying (5.18) simply reproduces the probability function for the normal distribution in dispersion model form (5.2). This shows that the saddlepoint approximation is exact for the normal distribution. □

Example 5.13. For the Poisson distribution, $V(\mu) = \mu$ so that $V(y) = y$. The saddlepoint approximation is therefore

$$\tilde{\mathcal{P}}(y;\mu) = \frac{1}{\sqrt{2\pi y}} \exp\{-y\log(y/\mu) + (y-\mu)\}. \tag{5.19}$$

□

5.4.3 The Distribution of the Unit Deviance

The saddlepoint approximation has an important consequence. If the saddlepoint approximation to the probability function of an EDM is accurate, then it follows that the unit deviance $d(y, \mu)$ follows a χ^2_1 distribution.

To prove this, we use the fact that the χ^2_1 distribution is determined by its MGF. Consider a random variable y whose probability function is an EDM. If the saddlepoint approximation to its probability function is accurate, then the MGF of the unit deviance is

$$\begin{aligned} M_{d(y,\mu)}(t) &= \mathrm{E}[\exp\{d(y,\mu)t\}] \quad \text{(by definition)} \\ &= \int_S \exp\{d(y,\mu)t\} \frac{1}{\sqrt{2\pi\phi V(y)}} \exp\left\{-\frac{d(y,\mu)}{2\phi}\right\} dy. \end{aligned}$$

(Recall that $y \in S$.) Rearranging:

$$\begin{aligned} M_{d(y,\mu)}(t) &= \int_S \exp\left\{-d(y,\mu)\left(\frac{1-2\phi t}{2\phi}\right)\right\} \frac{1}{\sqrt{2\pi\phi V(y)}} dy \\ &= (1-2\phi t)^{-1/2} \int_S \frac{(1-2\phi t)^{1/2}}{\{2\pi\phi V(y)\}^{1/2}} \exp\left\{-d(y,\mu)\left(\frac{1-2\phi t}{2\phi}\right)\right\} dy. \end{aligned}$$

Let $\phi' = \phi/(1-2\phi t)$. Then

$$\begin{aligned} M_{d(y,\mu)}(t) &= (1-2\phi t)^{-1/2} \int_S \frac{1}{\{2\pi\phi' V(y)\}^{-1/2}} \exp\left\{-\frac{d(y,\mu)}{2\phi'}\right\} dy \\ &= (1-2\phi t)^{-1/2}, \end{aligned} \tag{5.20}$$

since the integrand is the (saddlepoint) density of the distribution with $\phi' = \phi/(1-2\phi t)$. The MGF (5.20) identifies a χ^2_1 distribution, showing that

$$d(y,\mu)/\phi \sim \chi^2_1 \tag{5.21}$$

whenever the saddlepoint approximation is accurate. This result forms the basis of small-dispersion asymptotic theory used in Chap. 7. Note that (5.21) implies that $\mathrm{E}[d(y,\mu)] = \phi$ whenever the saddlepoint approximation is accurate.

Example 5.14. The saddlepoint approximation is exact for the normal distribution (Example 5.12), implying that the unit deviance has an exact χ^2_1 distribution for the normal distribution. The unit deviance for the normal distribution, found in Example 5.10, is $d(y,\mu) = (y-\mu)^2$. This means $d(y,\mu)/\phi = \{(y-\mu)/\sigma\}^2$, which defines a χ^2_1 random variate when y comes from the $N(\mu, \sigma^2)$ distribution. □

5.4.4 Accuracy of the Saddlepoint Approximation

The saddlepoint approximation is exact for the normal and inverse Gaussian distributions (Example 5.12; Problem 5.9). For other two-parameter distributions, the accuracy is such that $\tilde{\mathcal{P}}(y;\mu,\phi) = \mathcal{P}(y;\mu,\phi)\{1+O(\phi)\}$, where $O(\phi)$ means "order ϕ", an expression which is like a constant times ϕ as $\phi \to 0$ [3]. This shows that the error is relative, so that the density is approximated equally well even in the tails of the distribution where the density is low. This expression also shows that the approximation becomes nearly exact for ϕ small.

For the gamma distribution, the saddlepoint approximation is equivalent to approximating the gamma function $\Gamma(1/\phi)$ in the probability function with Stirling's formula

$$n! \approx n^n \exp(-n)\sqrt{2\pi n} \qquad \text{as } n \to \infty. \tag{5.22}$$

For the gamma distribution, the relative accuracy of the approximation is constant for all y.

For the binomial, Poisson and negative binomial distributions, the saddlepoint approximation is equivalent to replacing all factorials in the probability density functions with their Stirling's formula equivalents. This means that the saddlepoint approximation will be good for the Poisson distribution if y is not too small. For the binomial distribution, the saddlepoint approximation will be accurate if my and $m(1-y)$ are both not too small.

Smyth and Verbyla [11] give a guideline for judging when the saddlepoint approximation is sufficiently accurate to be relied on for practical purposes. They define

$$\tau = \frac{\phi V(y)}{(y - \text{boundary})^2}, \tag{5.23}$$

where "boundary" is the nearest boundary of the support S for y. Here τ is a sort of empirical coefficient of variation. Based on a number of heuristic and theoretical justifications, they argue that the saddlepoint approximation should be adequate when $\tau \leq 1/3$. This corresponds to the following guidelines (Problems 5.13 to 5.15):

- Binomial distribution: $my \geq 3$ and $m(1-y) \geq 3$.
- Poisson distribution: $y \geq 3$.
- Gamma distribution: $\phi \leq 1/3$.

These guidelines apply to the ordinary saddlepoint approximation. The modified saddlepoint approximation is often much better, sometimes adequate for any y.

Comparing the saddlepoint approximation with the Central Limit Theorem is revealing. It is true that EDMs converge to normality also as $\phi \to 0$, a result which can be derived from the Central Limit Theorem. However, the saddlepoint approximation is usually far more accurate, because its error

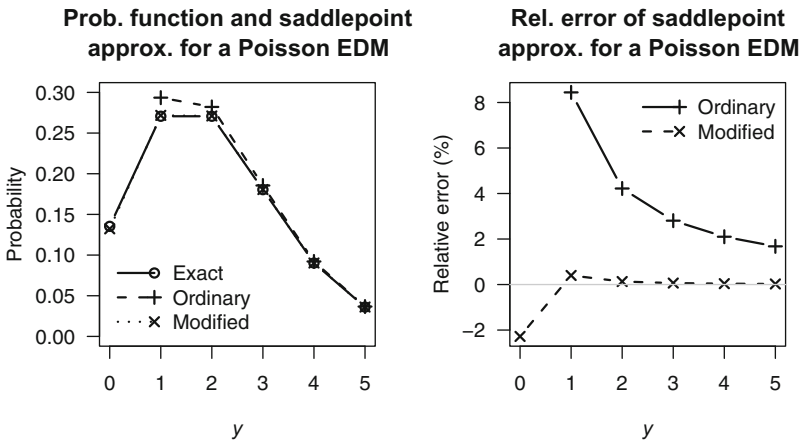

Fig. 5.3 The accuracy of the saddlepoint approximation for the Poisson distribution with $\mu = 2$. For $y = 0$ the ordinary saddlepoint approximation is undefined. The modified saddlepoint is evaluated with $\epsilon = 1/6$. The accuracy of the modified approximation is never worse than 2.3% (Example 5.15)

is relative and $O(\phi)$, whereas the accuracy of the Central Limit Theorem is additive and $O(\sqrt{\phi})$. This means that the saddlepoint approximation applies for larger values of ϕ than the Central Limit Theorem. For continuous EDMs, the saddlepoint approximation holds almost uniformly in the tails of the distribution, whereas the Central Limit Theorem is best near the mean of the distribution and deteriorates rapidly in the tails.

Example 5.15. For the Poisson distribution, $V(\mu) = \mu$, so the modified saddlepoint approximation is

$$\tilde{\mathcal{P}}(y;\mu) = \frac{1}{\sqrt{2\pi(y+\epsilon)}} \exp\{-y\log(y/\mu) + (y-\mu)\}.$$

The ordinary saddlepoint approximation (5.19) corresponds to $\epsilon = 0$. The relative accuracy of the saddlepoint approximation is the same for any μ at given y (Fig. 5.3, right panel). The relative accuracy of the ordinary approximation is less than 3% when $y \geq 3$. The accuracy of the modified approximation is excellent, never worse than 2.3%. □

5.4.5 Accuracy of the χ^2_1 Distribution for the Unit Deviance

In the previous section we considered conditions under which the saddlepoint approximation to the probability function should be accurate. In this section,

we consider what implications this has for the distribution of the unit deviance. We have already noted that the relative accuracy of the saddlepoint approximation does not depend on μ. However, when we consider the distribution of the unit deviance, the saddlepoint approximation needs to hold for all likely values of y. So we need μ and ϕ to be such that values of y close to the boundary of the distribution are not too likely.

For the normal and inverse Gaussian distributions, the unit deviance has an exact χ^2_1 distribution since the saddlepoint approximation is exact for these distributions. For other EDMs, the distribution of the unit deviance approaches χ^2_1 for any μ as $\phi \to 0$.

We will limit our investigation to considering how close the expected value of the unit deviance is to its nominal value ϕ. For continuous distributions, the expected value of the unit deviance is defined by

$$\mathrm{E}[d(y,\mu)] = \int_S d(y,\mu)\mathcal{P}(y;\mu,\phi)\,dy$$

where $\mathcal{P}(y;\mu,\phi)$ is the probability density function of the distribution. Using this expression, the expected value of the unit deviance can be computed for the gamma distribution, and compared to $\mathrm{E}[d(y,\mu)] = \phi$ (Fig. 5.4, top left panel). The relative error is less than about 5% provided $\phi < 1/3$.

For discrete distributions, the expected value of the unit deviance is defined by

$$\mathrm{E}[d(y,\mu)] = \sum_S d(y,\mu)\mathcal{P}(y;\mu,\phi)$$

where $\mathcal{P}(y;\mu,\phi)$ is the probability mass function of the distribution. We now use R to compute the expected value of the unit deviance for the Poisson distribution, and compare it to its nominal value $\mathrm{E}[d(y,\mu)] = 1$ according to the chi-square approximation (Fig. 5.4, top right panel):

```
> Poisson.mu <- c(0.000001, 0.001, 0.01, seq(0.1, 10, by=0.1) )
> DensityTimesDeviance <- function(mu) {
    y  <- seq(0, 100, by=1)
    sum( dpois(y, lambda=mu) * poisson()$dev.resids(y, mu, wt=1) )
 }
> ExpD.psn <- sapply( Poisson.mu, DensityTimesDeviance)
> plot( ExpD.psn ~ Poisson.mu, las=1, type="n",
    main="Poisson distribution", xlab=expression(mu),
    ylab="Exp. value of unit deviance")
> polygon( x=c(-1, -1, 12, 12), y=c(0.95, 1.05, 1.05, 0.95),
          col="gray", border=NA) # Draws the region of 5% rel. accuracy
> lines( ExpD.psn ~ Poisson.mu, lty=2, lwd=2)
> abline(h=1)
```

(The awkward construct `poisson()$dev.resids()` accesses the function `dev.resids()` from the `poisson()` family definition. Despite its name, `dev.resids()` returns the unit deviance.) The plots show that the expected value of the deviance is generally not near one for small μ, but the error is well below 10% provided $\mu > 3$.

For the binomial distribution, plots of the expected value of the deviance against μ for various values of m (Fig. 5.4, bottom panels) show that the expected value of the deviance can be far from one when $m\mu$ or $m(1-\mu)$ are small, but the error is reasonable provided $m\mu > 3$ and $m(1-\mu) > 3$.

In summary, the unit deviance is always chi-square for the normal and inverse Gaussian distributions, and for other common EDMs the unit deviance is roughly chi-square with the correct expected value when

- Binomial distribution: $m\mu \geq 3$ and $m(1-\mu) \geq 3$.
- Poisson distribution: $\mu \geq 3$.
- Gamma distribution: $\phi \leq 1/3$.

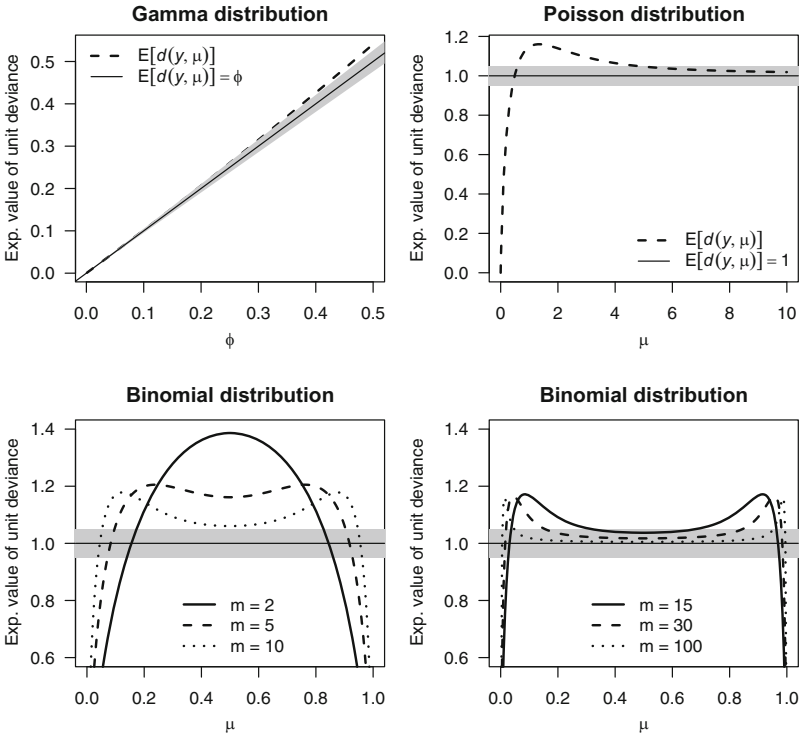

Fig. 5.4 The expected value of the unit deviance (modelled on [12, p. 208]). Top left panel: the gamma distribution for various values of ϕ (the solid line represents the target $\mathrm{E}[d(y,\mu)] = \phi$); top right panel: the Poisson distribution for various values of μ; bottom panels: the binomial distribution for various values of μ and m. The gray regions indicate relative accuracy within 5% (Sect. 5.4.5)

5.5 The Systematic Component

5.5.1 Link Function

In addition to assuming that the responses come from the EDM family, GLMs assume a specific form for the systematic component. GLMs assume a systematic component where the *linear predictor*

$$\eta = \beta_0 + \sum_{j=1}^{p} \beta_j x_j$$

is linked to the mean μ through a *link function* $g()$ so that $g(\mu) = \eta$. This systematic component shows that GLMs are regression models linear in the parameters.

The link function $g(\cdot)$ is a monotonic, differentiable function relating the fitted values μ to the linear predictor η. Monotonicity ensures that any value of η is mapped to only one possible value of μ. Differentiability is required for estimation (Sect. 6.2). The *canonical link function* is a special link function, the function $g(\mu)$ such that $\eta = \theta = g(\mu)$.

Example 5.16. For the normal distribution, $\theta = \mu$ (Table 5.1, p. 221). The canonical link function is the *identity* link function $g(\mu) = \mu$, which implies $\eta = \mu$. □

Example 5.17. For the Poisson distribution, $\theta = \log \mu$ (Table 5.1, p. 221). The canonical link function is $g(\mu) = \log \mu$, so that $\log \mu = \eta$. The Poisson distribution is only defined for positive values of μ, and the logarithmic link function ensures η (which possibly takes any real value) always maps to a positive value of μ. Hence the canonical link function is a sensible link function to use in this case. □

5.5.2 Offsets

In some applications, the linear predictor contains a term that requires no estimation, which is called an *offset*. The offset can be viewed as a term $\beta_j x_{ji}$ in the linear predictor for which β_j is known *a priori*. For example, consider modelling the annual number of hospital births in various cities to facilitate resource planning. The annual *number* of births is discrete, so a Poisson distribution may be appropriate. However, the expected annual number of births μ_i in city i depends on the given populations P_i of the city, since cities with larger population would be expected to have more births each year, in

general. The number of births per unit of population, assuming a logarithmic link function, can be modelled using the systematic component

$$\log(\mu/P) = \eta,$$

for the linear predictor η. Rearranging to model μ:

$$\log(\mu) = \log P + \eta.$$

The first term in the systematic component $\log P$ is completely known: nothing needs to be estimated. The term $\log P$ is called an *offset*. Offsets commonly appear in Poisson GLMs, but may appear in any GLM (Example 5.18).

The offset variable is commonly a measure of *exposure*. For example, the number of cases of a certain disease recorded in various mines depends on the number of workers, and also on the number of years each worker has worked in the mine. The exposure would be the number of person-years worked in each mine, which could be incorporated into a GLM as an offset. That is, a mine with many workers who have been employed for many years would be exposed to a greater likelihood of a worker contracting the disease than a mine with only a few workers who have been employed for short periods of time.

Example 5.18. For the cherry tree data (Example 3.14, p. 125), approximating the shape of the trees as a cone or as a cylinder leads to a model with the systematic component

$$\log \mu = \beta_0 + 2 \log g + \log h, \tag{5.24}$$

where g is the girth and h is the height of each tree, and the value of β_0 is different for cones and cylinders. To fit this model, the term $2 \log g + \log h$ is an offset, as this expression has no terms requiring estimation. □

5.6 Generalized Linear Models Defined

The two components of a generalized linear model (GLM) have been discussed: the random component (Sect. 5.3) and the systematic component (Sect. 5.5). Now a GLM can be formally defined. A GLM consists of two components:

- Random component: The observations y_i come independently from a specified EDM such that $y_i \sim \text{EDM}(\mu_i, \phi/w_i)$ for $i = 1, 2, \ldots, n$. The w_i are known non-negative *prior weights*, which potentially weight each Observation i differently. Commonly, the prior weights all equal one.
- Systematic component: A linear predictor $\eta_i = o_i + \beta_0 + \sum_{j=1}^{p} \beta_j x_{ji}$, where the o_i are offsets (Sect. 5.5.2) that are often equal to zero, and $g(\mu) = \eta$ is a known, monotonic, differentiable link function.

The GLM is

$$\begin{cases} y_i \sim \text{EDM}(\mu_i, \phi/w_i) \\ g(\mu_i) = o_i + \beta_0 + \sum_{j=1}^{p} \beta_j x_{ji}. \end{cases} \tag{5.25}$$

The core structure of a GLM is specified by the choice of distribution from the EDM class and the choice of link function; that is, the answer to the two important questions in Sect. 5.2. The notation

$$\text{GLM}(\text{EDM; Link function})$$

specifies the GLM by giving the EDM used for the random component, and the link function relating the mean μ to the explanatory variables.

Example 5.19. For the Quilpie rainfall data (Example 4.6, p. 174), the model suggested is

$$\begin{cases} y_i m_i \sim \text{Bin}(\mu_i, m_i) & \text{(random component)} \\ \log \dfrac{\mu_i}{1-\mu_i} = \beta_0 + \beta_1 x_i & \text{(systematic component)} \end{cases}$$

where x_i is the SOI, and $y_i = 1$ if the total July rainfall exceeds 10 mm (and $y_i = 0$ otherwise). This is a *binomial* GLM. Algorithms for estimating the values of β_0 and β_1 are discussed in Chap. 6. The GLM is denoted GLM(binomial; logit). In R, the GLM is specified by `family("binomial", link="logit")`. □

5.7 The Total Deviance

The unit deviance has been shown to be a measure of distance between y and μ (Sect. 5.4.1). An overall measure of the distance between all the y_i and all the μ_i can be defined as

$$D(y, \mu) = \sum_{i=1}^{n} w_i d(y_i, \mu_i),$$

called the *deviance function*, and its value called the *deviance* or the *total deviance*. The *scaled deviance* function is defined as

$$D^*(y, \mu) = D(y, \mu)/\phi,$$

and its value is called the *scaled deviance* or the *scaled total deviance*.

If the saddlepoint approximation holds, then the distribution of the scaled deviance follows an approximate chi-square distribution

$$D^*(y, \mu) \sim \chi^2_n,$$

with μ_i (for all i) and ϕ at their true values. As usual, the approximation is exact for normal linear GLMs. However, in practice the μ_i are seldom known. We will return to the distribution of the deviance and scaled deviance functions when the β_j are estimated in Chap. 7.

Note that by using the dispersion model form of the EDM, the log-likelihood function for the GLM in (5.25) can be expressed as

$$\begin{aligned} \ell(\mu; y) &= \sum_{i=1}^{n} \log b(y_i, \phi/w_i) - \frac{1}{2\phi} \sum_{i=1}^{n} w_i d(y_i, \mu_i) \\ &= \sum_{i=1}^{n} \log b(y_i, \phi/w_i) - \frac{D(y, \mu)}{2\phi}. \end{aligned} \tag{5.26}$$

Example 5.20. For a normal linear GLM, $y_i \sim N(\mu_i, \sigma^2)$ (Example 5.10), and $D(y, \mu) = \sum_{i=1}^{n} (y_i - \mu_i)^2$. This is the squared Euclidean distance between the corresponding values of y_i and μ_i. Hence, $D^*(y, \mu) = \sum_{i=1}^{n} \{(y_i - \mu_i)/\sigma\}^2$, which has an exact χ^2_n distribution. □

5.8 Regression Transformations Approximate GLMs

In Chap. 3, variance-stabilizing transformations of y were used to create constant variance in the response for linear regression models. When $V(\mu)$ represents the true mean–variance relationship for the responses, there is a clear relationship between $V(\mu)$ and the variance-stabilizing transformation. Consider the transformation $y^* = h(y)$. A first-order Taylor series expansion about μ gives $h(y) \approx h(\mu) + h'(\mu)(y - \mu)$, so that

$$\text{var}[y^*] = \text{var}[h(y)] \approx h'(\mu)^2 \text{var}[y].$$

Hence the transformation $y^* = h(y)$ will approximately stabilize the variance (that is, ensure var$[y^*]$ is approximately constant) if $h'(\mu)$ is proportional to var$[y]^{-1/2} = V(\mu)^{-1/2}$. Using linear regression after a transformation of y is therefore roughly equivalent to fitting a GLM with variance function $V(\mu) = 1/h'(\mu)^2$ and link function $g(\mu) = h(\mu)$. Almost any variance-stabilizing transformation can be viewed in this way (Table 5.2). Notice that the choice of transformation $h(y)$ influences both the implied variance function (and hence EDM) and the implied link function.

Table 5.2 EDMs and the approximately equivalent variance-stabilizing transformations used with linear regression models (Sect. 5.8)

Variance-stabilizing transformation (with Box–Cox λ)	The GLM being approximated	
	Variance function	Link function
$y^* = \sin^{-1}\sqrt{y}$	$V(\mu) = \mu(1-\mu)$ Binomial GLM (Chap. 9)	$g(\mu) = \sin^{-1}\sqrt{\mu}$
$y^* = \sqrt{y}$ $(\lambda = 0)$	$V(\mu) = \mu$ Poisson GLM (Chap. 10)	$g(\mu) = \sqrt{\mu}$
$y^* = \log y$ $(\lambda = 0)$	$V(\mu) = \mu^2$ gamma GLM (Chap. 11)	$g(\mu) = \log \mu$
$y^* = 1/\sqrt{y}$ $(\lambda = -1/2)$	$V(\mu) = \mu^3$ inverse Gaussian (Chap. 11)	$g(\mu) = 1/\sqrt{\mu}$
$y^* = 1/y$ $(\lambda = -1)$	$V(\mu) = \mu^4$ Tweedie GLM, with $\xi = 4$ (Chap. 12)	$g(\mu) = 1/\mu$

Example 5.21. Consider the square root transformation of the response, when used in a linear regression model. Expanding this transformation about μ using a Taylor series gives $\text{var}[\sqrt{y}] \approx \text{var}[y]/(4\mu)$. This will be constant if $\text{var}[y]$ is proportional to μ, which is true if y follows a Poisson distribution. Using this transformation of y in a linear regression model is roughly equivalent to fitting a Poisson GLM with square root link function. □

Using a transformation to simultaneously achieve linearity and constant variance assumes a relationship between the variance and link functions which in general is overly simplistic. GLMs obviously provide more flexibility: GLMs allow the EDM family and link function to be chosen separately depending on the data. The EDM family is chosen to reflect the support of the data and the mean–variance relationship, then the link function is chosen to achieve linearity. GLMs have the added advantages of modelling the data on the original scale, avoiding artificial transformations, and of giving realistic probability statements when the data are actually non-normal. The normal approximation for $h(y)$, implicit in the transformation approach, is often reasonable when ϕ is small, but may be very poor otherwise.

A GLM enables the impact of the explanatory variables on μ to be interpreted directly. For example, consider a systematic component of GLM using a log-link:

$$\log \mu = \beta_0 + \beta_1 x,$$

which can be written as

$$\mu = \exp(\beta_0)\exp(\beta_1)^x.$$

However, a logarithmic transformation used with a linear regression model gives

$$\mathrm{E}[\log y] = \beta_0 + \beta_1 x,$$

which does not allow direct interpretation in terms of $\mu = \mathrm{E}[y]$. However, since $\mathrm{E}[\log y] \approx \log \mathrm{E}[y] = \log \mu$ (Problem 2.11), then

$$\mu \approx \exp(\beta_0) \exp(\beta_1)^x.$$

5.9 Summary

Chapter 5 introduced the components, structure, notation and terminology of generalized linear models. GLMs are regression models linear in the parameters, and consist of two components (a random component and a systematic component), chosen in separate decisions (Sect. 5.2).

Common distributions that are EDMs include the normal, Poisson, gamma, binomial and negative binomial distributions (Sect. 5.3.1). The probability function for EDMs has the general form (Sect. 5.3.2)

$$\mathcal{P}(y; \theta, \phi) = a(y, \phi) \exp\{[y\theta - \kappa(\theta)]/\phi\}$$

where θ is the called *canonical parameter*, $\kappa(\theta)$ is called the *cumulant function*, and $\phi > 0$ is the *dispersion parameter*. The moment generating function and cumulant generating function for an EDM have simple forms (Sect. 5.3.4), which can be used to show that the mean of an EDM is $\mathrm{E}[y] = \mu = \mathrm{d}\kappa/\mathrm{d}\theta$ (Sect. 5.3.5), and the variance of an EDM is $\mathrm{var}[y] = \phi V(\mu)$, where $V(\mu) = \mathrm{d}^2\kappa(\theta)/\mathrm{d}\theta^2$ is the variance function (Sect. 5.3.5). The variance function uniquely determines the distribution within the class of EDMs (Sect. 5.3.6).

The unit deviance is $d(y, \mu) = 2\{t(y, y) - t(y, \mu)\}$ (Sect. 5.4). Using this, the dispersion model form of an EDM is (Sect. 5.4)

$$\mathcal{P}(y; \mu, \phi) = b(y, \phi) \exp\left\{-\frac{d(y, \mu)}{2\phi}\right\}.$$

For EDMs, the saddlepoint approximation is

$$\tilde{\mathcal{P}}(y; \mu, \phi) = \frac{1}{\sqrt{2\pi\phi V(y)}} \exp\left\{-\frac{d(y, \mu)}{2\phi}\right\}.$$

The approximation is accurate as $\phi \to 0$ (Sect. 5.4.2). The saddlepoint approximation implies $d(y, \mu) \sim \chi^2_1$ as $\phi \to 0$ (Sect. 5.4.3). The approximation is exact for the normal and inverse Gaussian distributions (Sect. 5.4.3).

The *link function* $g(\cdot)$ expresses the functional relationship between the mean μ and the linear predictor η as $g(\mu) = \eta = \beta_0 + \sum_{j=1}^{n} \beta_j x_j$, where $g(\mu)$

is a differentiable, monotonic function (Sect. 5.5.1). Offsets are components of the linear predictor with no unknown parameters (Sect. 5.5.2).

A GLM is defined by two components (Sect. 5.6):

- Random component: Observations y_i come independently from an EDM such that $y_i \sim \text{EDM}(\mu_i, \phi/w_i)$ for $i = 1, 2, \ldots, n$, where the w_i are non-negative *prior weights*.
- Systematic component: A link function $g(\cdot)$ such that $g(\mu_i) = o_i + \beta_0 + \sum_{j=1}^{p} \beta_j x_{ji}$, where $g(\cdot)$ is a known, monotonic, differentiable *link function* and o_i is the offset.

The core structure of a GLM is denoted GLM(EDM; Link function) (Sect. 5.6).

The deviance function, a measure of total discrepancy between all the y_i and μ_i, is $D(y, \mu) = \sum_{i=1}^{n} w_i d(y_i, \mu_i)$. By the saddlepoint approximation, $D(y, \mu)/\phi \sim \chi^2_n$ as $\phi \to 0$ (Sect. 5.7). The unit deviance has a chi-square distribution for the normal and inverse Gaussian distributions (Sect. 5.4.5), and is approximately distributed as chi-square with the correct expected value when:

- Binomial distribution: $m\mu \geq 3$ and $m(1 - y\mu) \geq 3$.
- Poisson distribution: $\mu \geq 3$.
- Gamma distribution: $\phi \leq 1/3$.

Variance-stabilizing transformations $h(y)$ used with linear regression models are roughly equivalent to fitting a GLM with variance function $V(\mu) = 1/h'(\mu)^2$ and link function $g(\mu) = h(\mu)$ (Sect. 5.8).

Problems

Selected solutions begin on p. 536.

5.1. Determine which of the following distributions are EDMs by identifying (where possible) θ, $\kappa(\theta)$ and ϕ:

1. The beta distribution:

$$\mathcal{P}(y; a, b) = \frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)} y^{a-1}(1-y)^{b-1},$$

   for $0 < y < 1$, $a > 0$ and $b > 0$, where $\Gamma(\cdot)$ is the gamma function.
2. The geometric distribution:

$$\mathcal{P}(y; p) = p(1-p)^{y-1} \tag{5.27}$$

   for $y = 1, 2, \ldots$ and $0 < p < 1$.

3. The Cauchy distribution:

$$\mathcal{P}(y; c, s) = \frac{1}{\pi s \left\{1 + \left(\frac{y-c}{s}\right)^2\right\}} \tag{5.28}$$

for $-\infty < y < \infty$, $-\infty < c < \infty$, and $s > 0$.

4. The von Mises distribution, used for modelling angular data:

$$\mathcal{P}(y; \mu, \lambda) = \frac{1}{2\pi I_0(\lambda)} \exp\{\lambda \cos(y - \mu)\},$$

for $0 \leq y < 2\pi$, $0 \leq \mu < 2\pi$ and $\lambda > 0$, where $I_0(\cdot)$ is the modified Bessel function of order 0.

5. The strict arcsine distribution [5] used for modelling count data:

$$\mathcal{P}(y; p) = A(y; 1)\frac{p^y}{y!} \exp(-\arcsin p),$$

for $y = 0, 1, \ldots$ and $0 < p < 1$, where $A(y; 1)$ is a complicated normalising function.

5.2. Use the results $\mathrm{E}[y] = \kappa'(\theta)$ and $\mathrm{var}[y] = \phi\kappa''(\theta)$ to find the mean, variance and variance function for the distributions in Problem 5.1 that are EDMs.

5.3. Determine the canonical link function for the distributions in Problem 5.1 that are EDMs.

5.4. Use the definition of $K(t)$ and $M(t)$ to prove the following results.

1. Show that $\mathrm{d}K(t)/\mathrm{d}t$ evaluated at $t = 0$ is the mean of y.
2. Show that $\mathrm{d}^2K(t)/\mathrm{d}t^2$ evaluated at $t = 0$ is the variance of y.

5.5. Prove the result in (5.4), that $\kappa_r = \mathrm{d}^r\kappa(\theta)/\mathrm{d}\theta^r$ for EDMs.

5.6. Show that the mean and variance of a discrete EDM are given by $\mathrm{E}[y] = \kappa'(\theta)$ and $\mathrm{var}[y] = \phi\kappa'(\theta)$ respectively by following similar steps as shown in Sect. 5.3.5, but using summations rather than integrations.

5.7. For EDMs in the form of (5.1), show that the variance is $\mathrm{var}[y] = \phi\kappa''(\theta)$ by using the CGF (5.7).

5.8. Consider the gamma distribution, whose probability function is usually written as

$$\mathcal{P}(y; \alpha, \beta) = \frac{1}{\Gamma(\alpha)\beta^\alpha} y^{\alpha-1} \exp(-y/\beta)$$

for $y > 0$ with $\alpha > 0$ (the shape parameter) and $\beta > 0$ (the scale parameter), where $\Gamma(\cdot)$ is the gamma function.

1. Show that the gamma distribution is an EDM by identifying θ, $\kappa(\theta)$ and ϕ.
2. Show that the saddlepoint approximation applied to the gamma distribution is equivalent to using Stirling's formula (5.22).
3. Determine the canonical link function.
4. Deduce the unit deviance for the gamma distribution.
5. Write the probability function in dispersion model form (5.13).

5.9. Consider the inverse Gaussian distribution, which has the probability function

$$\mathcal{P}(y;\mu,\phi) = (2\pi y^3\phi)^{-1/2}\exp\left\{-\frac{1}{2\phi}\frac{(y-\mu)^2}{y\mu^2}\right\}$$

where $y > 0$, $\mu > 0$ and $\phi > 0$.

1. Show that the inverse Gaussian distribution is an EDM by identifying θ, $\kappa(\theta)$ and ϕ.
2. Show that the variance function is $V(\mu) = \mu^3$.
3. Determine the canonical link function.
4. Deduce the unit deviance and the deviance function.
5. Show that the saddlepoint approximation is exact for the inverse Gaussian distribution.

5.10. Prove the results in Table 5.2 (p. 233). For example, show that the variance-stabilizing transformation $1/\sqrt{y}$ used in a linear regression model is approximately equivalent to using an inverse Gaussian GLM with the link function $\eta = 1/\sqrt{\mu}$. (Use a Taylor series expanded about the mean μ, as in Sect. 5.8, p. 232.)

5.11. Consider the Conway–Maxwell–Poisson (CMP) distribution [8], which has the probability function

$$\mathcal{P}(y;\lambda;\nu) = \frac{\lambda^y}{Z(\lambda,\nu)(y!)^\nu},$$

where $y = 0, 1, 2, \ldots$, $\lambda > 0$, $\nu \geq 0$, and $Z(\lambda,\nu) = \sum_{k=0}^{\infty}\lambda^k/(k!)^\nu$. (When $\nu = 0$, the CMP distribution is undefined for $\lambda \geq 1$.)

1. Show that the CMP distribution is an EDM with $\phi = 1$ by identifying θ and $\kappa(\theta)$, provided ν is known.
2. When ν is known, show that

$$\mu = \mathrm{E}[y] = \frac{1}{Z(\lambda,\nu)}\sum_{k=0}^{\infty}\frac{k\lambda^k}{(k!)^\nu} \quad \text{and} \quad \mathrm{var}[y] = \frac{1}{Z(\lambda,\nu)}\sum_{k=0}^{\infty}\frac{k^2\lambda^k}{(k!)^\nu} - \mu^2.$$

3. Show that the CMP distribution allows for a non-linear decrease in successive probabilities:

$$\frac{\mathcal{P}(y-1;\lambda,\nu)}{\mathcal{P}(y;\lambda,\nu)} = \frac{y^\nu}{\lambda}.$$

4. Show that $\nu = 1$ corresponds to the Poisson distribution. (HINT: Use that $\sum_{i=0}^{\infty} x^i/i! = \exp x$.)
5. Show that $\nu = 0$ corresponds to the geometric distribution (5.27) when $\lambda < 1$ and the probability of success is $1 - \lambda$. (HINT: Use that $\sum_{i=0}^{\infty} x^i = 1/(1-x)$ provided $|x| < 1$.)
6. Show that $\nu \to \infty$ corresponds to the Bernoulli distribution (4.5) with mean proportion $\lambda/(1+\lambda)$.

5.12. As in Fig. 5.3, compute the relative error in using the saddlepoint and modified saddlepoint approximations for a Poisson distribution with $\mu = 2$. Then, repeat the calculations for another value of μ, say $\mu = 4$, and show that the relative error in the saddlepoint approximations are the same for both values of μ (to computer precision).

5.13. Using (5.23), show that the saddlepoint approximation is expected to hold for the Poisson distribution when $y \geq 3$.

5.14. Using (5.23), show that the saddlepoint approximation is expected to hold for the binomial distribution when $my \geq 3$ and $my(1-y) \geq 3$.

5.15. Using (5.23), show that the saddlepoint approximation is expected to hold for the gamma distribution when $\phi \leq 1/3$.

5.16. The probability function for a Poisson distribution is given in Example 5.2 (p. 213).

1. Show that the MGF for the Poisson distribution is $M(t) = \exp(-\mu + \mu e^t)$. (HINT: Use that $\exp x = \sum_{i=0}^{\infty} x^i/i!$.)
2. Hence compute the CGF for the Poisson distribution.
3. Confirm that the mean and the variance of the Poisson distribution are both μ by using the CGF.

5.17. Suppose $y_1, y_2, \ldots, y_n$ are independently and identically distributed as EDM(μ, ϕ). Show that $\bar{y}$ has the distribution EDM$(\mu, \phi/n)$ as follows.

1. Show that the CGF of $\bar{y}$ is $nK_Y(t/n)$, where $K_Y(t)$ is the CGF of y.
2. By substituting the CGF of y into the resulting expression, show that the CGF of $\bar{y}$ is $n\{\kappa(\theta + t\phi/n) - \kappa(\theta)\}/\phi$.
3. Show that this CGF is the CGF for an EDM$(\mu, \phi/n)$ distribution.

5.18. Consider the EDM with variance function $V(\mu) = 1+\mu^2$ (the generalized hyperbolic secant distribution [3]), which is defined for all real y and all real μ.

1. Find the canonical form (5.1) of the density function for this distribution. The normalizing constant $a(y, \phi)$ is difficult to determine in closed form but it is not necessary to do so.
2. Find the unit deviance for the EDM.
3. Write down the saddlepoint approximation to the probability function.

4. Use R to plot the saddlepoint approximation to the probability function for $\phi = 0.5$ and $\phi = 1$ when $\mu = -1$. Do you expect the saddlepoint approximation to be accurate? Explain.
5. Find the canonical link function.

5.19. Consider the EDM with variance function $V(\mu) = \mu^4$, which is defined for all real $y > 0$ and all real $\mu > 0$.

1. Find the canonical form (5.1) of the density function for this distribution. The normalizing constant $a(y, \phi)$ is difficult to determine in closed form but it is not necessary to do so.
2. Use that $\kappa(\theta) < \infty$ to show that $\theta \leq 0$.
3. Find the unit deviance for the EDM.
4. Write down the saddlepoint approximation to the probability function.
5. Use R to plot the saddlepoint approximation to the probability function for $\phi = 0.5$ and $\phi = 1$ when $\mu = 2$.
6. Find the canonical link function.

5.20. Prove that the canonical link function and the variance function are related by $V(\mu) = 1/g'(\mu) = \mathrm{d}\mu/\mathrm{d}\eta$, where $g(\mu)$ here is the *canonical* link function.

5.21. Consider the expressions for the deviance function of the normal and gamma distributions (Table 5.1, p. 221). Show that if each datum y_i is replaced by $100y_i$ (say a change of measurement units from metres to centimetres) that the numerical value of the gamma deviance function does not change, but the numerical value of the normal deviance function changes.

5.22. The probability function for a special case of the exponential distribution is $\mathcal{P}(y) = \exp(-y)$ for $y > 0$.

1. Show that the MGF for this distribution is $M(t) = (1 - t)^{-1}$ if $t < 1$.
2. Hence compute the CGF for this distribution.
3. Confirm that the mean and the variance of this distribution are both 1 by differentiating the CGF.

5.23. Consider a random variable y with the probability function $\mathcal{P}(y) = y \exp(-y)$ for $y > 0$.

1. Show that the MGF for the distribution is $M(t) = (1 - t)^{-2}$ if $t < 1$.
2. Hence compute the CGF for the distribution.
3. Confirm that the mean and the variance of this distribution are both 2 by differentiating the CGF.

5.24. Determine which of these functions are suitable link functions for a GLM. For those that are not suitable, explain why not.

1. $g(\mu) = -1/\mu^2$ when $\mu > 0$.
2. $g(\mu) = |\mu|$ when $-\infty < \mu < \infty$.

Table 5.3 The first six observations of the Nambeware products data (Problem 5.26)

Item	Diameter (in inches)	Grinding and polishing time (in min)	Price ($US)
Casserole dish	10.7	47.65	144.00
Casserole dish	14.0	63.13	215.00
Casserole dish	9.0	58.76	105.00
Bowl	8.0	34.88	69.00
Dish	10.0	55.53	134.00
Casserole dish	10.5	43.14	129.00
⋮	⋮	⋮	⋮

3. $g(\mu) = \log \mu$ when $\mu > 0$.
4. $g(\mu) = \mu^2$ when $-\infty < \mu < \infty$.
5. $g(\mu) = \mu^2$ when $0 < \mu < \infty$.

5.25. Children were asked to build towers as high as they could out of cubical and cylindrical blocks [2, 9]. The number of blocks used and the time taken were recorded (Table 2.12; data set: `blocks`). In this problem, only consider the number of blocks used y and the age of the child x.

1. Plot the number of blocks used against the age of the child.
2. From the plot and an understanding of the data, answer the two questions in Sect. 5.2 (p. 211) for these data, and hence propose a GLM for the data.

5.26. Nambe Mills, Santa Fe, New Mexico [1, 10], is a tableware manufacturer. After casting, items produced by Nambe Mills are shaped, ground, buffed, and polished. In 1989, as an aid to rationalizing production of its 100 products, the company recorded the total grinding and polishing times and the diameter of each item (Table 5.3; data set: `nambeware`). In this problem, only consider the item price y and the item diameter x.

1. Plot the price against diameter.
2. From the plot and an understanding of the data, argue that the answer to the two questions in Sect. 5.2 (p. 211) may suggest a gamma GLM.

References

[1] Data Desk: Data and story library (DASL) (2017). URL http://dasl.datadesk.com
[2] Johnson, B., Courtney, D.M.: Tower building. Child Development **2**(2), 161–162 (1931)
[3] Jørgensen, B.: The Theory of Dispersion Models. Monographs on Statistics and Applied Probability. Chapman and Hall, London (1997)

[4] Keller, D.K.: The Tao of Statistics: A Path to Understanding (with no Math). Sage Publications, Thousand Oaks, CA (2006)
[5] Kokonendji, C.C., Khoudar, M.: On strict arcsine distribution. Communications in Statistics—Theory and Methods **33**(5), 993–1006 (2004)
[6] Maron, M.: Threshold effect of eucalypt density on an aggressive avian competitor. Biological Conservation **136**, 100–107 (2007)
[7] Nelder, J.A., Pregibon, D.: An extended quasi-likelihood function. Biometrika **74**(2), 221–232 (1987)
[8] Shmueli, G., Minka, T.P., Kadane, J.B., Borle, S., Boatwright, P.: A useful distribution for fitting discrete data: Revival of the Conway–Maxwell–Poisson distribution. Journal of the Royal Statistical Society: Series C **54**(1), 27–142 (2005)
[9] Singer, J.D., Willett, J.B.: Improving the teaching of applied statistics: Putting the data back into data analysis. The American Statistician **44**(3), 223–230 (1990)
[10] Smyth, G.K.: Australasian data and story library (OzDASL) (2011). URL http://www.statsci.org/data
[11] Smyth, G.K., Verbyla, A.P.: Adjusted likelihood methods for modelling dispersion in generalized linear models. Environmetrics **10**, 695–709 (1999)
[12] Venables, W.N., Ripley, B.D.: Modern Applied Statistics with S, fourth edn. Springer-Verlag, New York (2002). URL http://www.stats.ox.ac.uk/pub/MASS4

Chapter 6
Generalized Linear Models: Estimation

The challenge for the model builder is to get the most out of the modelling process by choosing a model of the right form and complexity so as to describe those aspects of the system which are perceived as important.
Chatfield [1, p. 27]

6.1 Introduction and Overview

The previous chapter defined GLMs and studied the components of a GLM. This chapter discusses the estimation of the unknown parameters in the GLM: the regression parameters and possibly the dispersion parameter ϕ. Because GLMs assume a specific probability distribution for the responses from the EDM family, maximum likelihood estimation procedures (Sect. 4.4) are used for parameter estimation, and general formulae are developed for the GLM context. We first derive the score equations and information for the GLM context (Sect. 6.2), which are used to form algorithms for estimating the regression parameters for GLMs (Sect. 6.3). The residual deviance is then defined as a measure of the residual variability across n observations after fitting the model (Sect. 6.4). The standard errors of the regression parameters are developed in Sect. 6.5. In Sect. 6.6, matrix formulations are used to estimate the regression parameters. We then explore the important connection between the algorithms for fitting linear regression models and GLMs (Sect. 6.7). Techniques are then developed for estimating ϕ (Sect. 6.8). We conclude with a discussion of using R to fit GLMs (Sect. 6.9).

6.2 Likelihood Calculations for $\boldsymbol{\beta}$

6.2.1 Differentiating the Probability Function

We begin by considering a single observation $y \sim \text{EDM}(\mu, \phi/w)$, with probability function $\mathcal{P}(y; \mu, \phi/w)$. The probability function can be differentiated easily, using its canonical form (5.1), as

P. K. Dunn, G. K. Smyth, *Generalized Linear Models with Examples in R*,
Springer Texts in Statistics, https://doi.org/10.1007/978-1-4419-0118-7_6

$$\frac{\partial \log \mathcal{P}(y;\mu,\phi/w)}{\partial \theta} = \frac{w(y-\mu)}{\phi},$$

after substituting $\mu = \mathrm{d}\kappa(\theta)/\mathrm{d}\theta$. Therefore

$$\frac{\partial \log \mathcal{P}(y;\mu,\phi/w)}{\partial \mu} = \frac{\partial \log \mathcal{P}(y;\mu,\phi/w)}{\partial \theta}\frac{\mathrm{d}\theta}{\mathrm{d}\mu} \tag{6.1}$$

$$= \frac{w(y-\mu)}{\phi V(\mu)}, \tag{6.2}$$

because $\mathrm{d}\mu/\mathrm{d}\theta = \mathrm{d}^2\kappa(\theta)/\mathrm{d}\theta^2 = V(\mu)$. The simple form of this derivative underlies much of GLM theory.

Now suppose that

$$g(\mu) = \eta = o + \sum_{j=0}^{p} \beta_j x_j, \tag{6.3}$$

writing $x_0 = 1$ as the covariate for β_0, and where o is the offset. The derivatives of $\log \mathcal{P}(y;\mu,\phi/w)$ with respect to the β_j are

$$\frac{\partial \log \mathcal{P}(y;\mu,\phi/w)}{\partial \beta_j} = \frac{\partial \log \mathcal{P}(y;\mu,\phi/w)}{\partial \mu}\frac{\partial \mu}{\partial \beta_j} = (y-\mu)\frac{w x_j}{\phi V(\mu)\mathrm{d}\eta/\mathrm{d}\mu}. \tag{6.4}$$

To find the expected second derivatives, use the product rule to obtain

$$\frac{\partial^2 \log \mathcal{P}(y;\mu,\phi/w)}{\partial \beta_k \partial \beta_j} = \frac{\partial}{\partial \beta_k}(y-\mu)\frac{w}{\phi V(\mu)}\frac{x_j}{\mathrm{d}\eta/\mathrm{d}\mu} + (y-\mu)\frac{\partial}{\partial \beta_k}\left(\frac{w}{\phi V(\mu)}\frac{x_j}{\mathrm{d}\eta/\mathrm{d}\mu}\right).$$

The second term has expectation zero because of the factor $(y-\mu)$, so

$$\mathrm{E}\left[\frac{\partial^2 \log \mathcal{P}(y;\mu,\phi/w)}{\partial \beta_k \partial \beta_j}\right] = -\frac{w}{\phi V(\mu)}\frac{x_j x_k}{(\mathrm{d}\eta/\mathrm{d}\mu)^2}. \tag{6.5}$$

Again, this is a very simple expression.

6.2.2 Score Equations and Information for β

Now consider a GLM in which $y_i \sim \mathrm{EDM}(\mu_i, \phi/w_i)$ for observations $y_1, \ldots, y_n$, with the linear predictor in (6.3). The linear predictor contains p' unknown regression parameters β_j which need to be estimated from the data. Our approach is to estimate the β_j using *maximum likelihood*, using the techniques in Sect. 4.4. To this end, we need to find the first and second derivatives of the log-likelihood.

The *log-likelihood function* is

$$\ell(\beta_0, \ldots, \beta_p, \phi; y) = \sum_{i=1}^{n} \log \mathcal{P}(y_i; \mu_i, \phi/w_i).$$

From (6.4), the log-likelihood derivatives (score functions) are

$$U(\beta_j) = \frac{\partial \ell(\beta_0, \ldots, \beta_p, \phi; y)}{\partial \beta_j} = \frac{1}{\phi} \sum_{i=1}^{n} W_i \frac{\mathrm{d}\eta_i}{\mathrm{d}\mu_i}(y_i - \mu_i) x_{ji} \tag{6.6}$$

where, for later convenience,

$$W_i = \frac{w_i}{V(\mu_i)(\mathrm{d}\eta_i/\mathrm{d}\mu_i)^2}. \tag{6.7}$$

Equation (6.6) holds for $j = 0, \ldots, p$ if we define $x_{0i} = 1$ as the covariate for β_0. The W_i are called the *working weights*.

From (6.5), the Fisher information for the regression parameters has elements

$$\mathcal{I}_{jk}(\beta) = \frac{1}{\phi} \sum_{i=1}^{n} W_i x_{ji} x_{ki}. \tag{6.8}$$

Example 6.1. Consider a Poisson GLM using a logarithmic link function $\log \mu = \eta$, with all prior weights w set to one. For the Poisson distribution, $V(\mu) = \mu$ and $\phi = 1$, so $\mathrm{d}\eta/\mathrm{d}\mu = 1/\mu$ and $W = \mu$. Using (6.6) and (6.8), the score function and Fisher information are, respectively

$$U(\beta_j) = \sum_{i=1}^{n} (y_i - \mu_i) x_{ji} \qquad \text{and} \qquad \mathcal{I}_{jk}(\beta) = \sum_{i=1}^{n} \mu_i x_{ji} x_{ki}.$$

□

6.3 Computing Estimates of β

The *Fisher scoring* algorithm (Sect. 4.8, p. 186) provides a convenient and effective method for computing the MLEs of the β_j.

The MLEs of the β_j, denoted $\hat{\beta}_j$, are the simultaneous solutions of the p' score equations $U(\beta_j) = 0$ for $j = 0, \ldots, p$. The scoring algorithm computes the $\hat{\beta}_j$ by iteratively refining the working estimates until convergence. Each iteration consists of solving an equation involving the score function $U(\beta_j)$ and the information $\mathcal{I}_{jk}(\beta)$.

For convenience, define the *working responses* as

$$z_i = \eta_i + \frac{\mathrm{d}\eta_i}{\mathrm{d}\mu_i}(y_i - \mu_i). \tag{6.9}$$

It can be shown that each iteration of the scoring algorithm is equivalent to least squares regression of the working responses z_i on the covariates x_{ji} using the working weights W_i (6.7). That is, z_i is regressed onto x_{ji} using W_i as the weights.

At each iteration, the z_i and W_i are updated, and the regression is repeated to obtain new working coefficients $\hat{\beta}_j^{(r)}$ (the estimate of β_j at iteration r). The linear predictor η_i is updated from the working coefficients, these are used to update the fitted values $\mu_i = g^{-1}(\eta_i)$, then the iteration is repeated. Because the working weights change at each iteration, the algorithm is often called *iteratively reweighted least squares* (IRLS).

Importantly, ϕ doesn't appear in the scoring iteration for the β_j, so there is no need to know ϕ to estimate the β_j. Because of this, estimation of ϕ is deferred to Sect. 6.8.

Another important aspect of the scoring iteration is that the working responses z_i and working weights W_i depend on the working coefficient estimates $\hat{\beta}_j^{(r)}$ only through the fitted values μ_i. This allows the scoring algorithm to be initialized using the responses y_i. The aim of the modelling is to produce estimates $\hat{\mu}_i$ as close as possible to the observations y_i, so the algorithm is started by setting initial values $\hat{\mu}_i^{(0)} = y_i$. Sometimes a slight adjustment is needed to avoid taking logarithms or reciprocals of zero, so $\hat{\mu}_i^{(0)} = y_i + 0.1$ or similar is used when $\hat{\mu}_i^{(0)}$ would otherwise be zero. Binomial GLMs have problems when $\mu = 0$ or $\mu = 1$, so the algorithm starts using $(my + 0.5)/(m + 1)$. The algorithm usually converges quite rapidly from these starting values.

Example 6.2. In Example 5.9 (data set: `nminer`), a Poisson GLM is suggested for the noisy miner data [4] with systematic component $\log \mu = \beta_0 + \beta_1 x$, where x is number of eucalypts per 2 ha transect `Eucs`. Using the results from Example 6.1 (p. 245),

$$z = \log \hat{\mu} + \frac{y - \hat{\mu}}{\hat{\mu}}. \tag{6.10}$$

Solutions are found by regressing z on x using the weights W (using $W = \mu$ as defined in Example 6.1). The iterative solution is found by iterating (6.9).

We cannot start the algorithm by setting $\hat{\mu} = y$ because the data contain cases where $y = 0$. Setting $\hat{\mu} = y$ in those cases would result in computing the logarithms of zero and diving by zero in (6.10). For this reason, the algorithm starts by using $\hat{\mu} = y + 0.1$. The working weights W and working values z are computed and hence initial estimates of β_0 and β_1 are obtained. Initially, the algorithm starts with the values in Table 6.1. The estimates are

Table 6.1 Starting the iterations for fitting the Poisson GLM to the noisy miner data. Note that the algorithm starts with $\hat{\mu} = y + 0.1$ to avoid dividing by zero and taking logarithms of zero (Example 6.2)

Case	Observations	Fitted values	Working values	Working weights
i	y	$\hat{\mu}_0^{(0)}$	$z = \hat{\eta} + (y - \hat{\mu})/\hat{\mu}$	$W = \hat{\mu}$
1	0	0.10	−3.303	0.10
2	0	0.10	−3.303	0.10
3	3	3.10	1.099	3.10
4	2	2.10	0.6943	2.10
5	8	8.10	2.080	8.10
⋮	⋮	⋮	⋮	⋮

Table 6.2 Fitting the Poisson GLM to the noisy miner data; the iterations have converged to six decimal places (Example 6.2)

Iteration r	Constant $\hat{\beta}_0^{(r)}$	$\hat{\beta}_1^{(r)}$	$D(y, \mu^{(r)})$
1	0.122336	0.081071	82.146682
2	−0.589798	0.103745	64.495148
3	−0.851982	0.113123	63.326027
4	−0.876031	0.113975	63.317978
5	−0.876211	0.113981	63.317978
6	−0.876211	0.113981	63.317978

updated (Table 6.2), and converge quickly. The final fitted Poisson GLM has the systematic component

$$\log \hat{\mu} = -0.8762 + 0.1140x. \tag{6.11}$$

□

Naturally, explicitly using the iterative procedure just described is not necessary when using R. Instead, the function `glm()` is used, where the systematic component is specified in the same way as for normal linear regression models (Sect. 2.6). Specifying the EDM family distribution and the link function is also necessary. See Sect. 6.9 for more details about using R to fit GLMs.

Example 6.3. Fit the Poisson GLM suggested in Example 6.2 (data set: `nminer`) as follows:

```
> library(GLMsData); data(nminer)
> nm.m1 <- glm( Minerab ~ Eucs, data=nminer,
         family=poisson(link="log"),
         control=list(trace=TRUE) )   # Shows the deviance each iteration
Deviance = 82.14668 Iterations - 1
Deviance = 64.49515 Iterations - 2
Deviance = 63.32603 Iterations - 3
Deviance = 63.31798 Iterations - 4
Deviance = 63.31798 Iterations - 5
```

```
> nm.m1
Call:  glm(formula = Minerab ~ Eucs, family = poisson(link = "log"),
    data = nminer, control = list(trace = TRUE))

Coefficients:
(Intercept)         Eucs
    -0.8762       0.1140

Degrees of Freedom: 30 Total (i.e. Null);  29 Residual
Null Deviance:            150.5
Residual Deviance: 63.32         AIC: 121.5
```

The fitted object `nm.m1` contains a wealth of information about the fitted GLM, which is discussed in the sections that follow. □

6.4 The Residual Deviance

The unit deviance (Sect. 5.4.1) captures the part of an EDM probability function which depends on μ, as distinct from ϕ. For a GLM, the total deviance (Sect. 5.7) captures that part of the log-likelihood function which depends on the μ_i. So, for the purpose of estimating the β_j, maximizing the log-likelihood is equivalent to minimizing the total deviance.

The total deviance can be computed at each stage of the IRLS algorithm (Sect. 6.3) by comparing the responses y_i with the fitted values at each iteration of the IRLS algorithm $\hat{\mu}_i^{(r)}$. R uses the total deviance to declare convergence at iteration r when

$$\frac{|D(y, \hat{\mu}^{(r)}) - D(y, \hat{\mu}^{(r-1)})|}{|D(y, \hat{\mu}^{(r)})| + 0.1} < \epsilon,$$

where $\epsilon = 10^{-8}$ is the default value.

After computing the MLEs $\hat{\beta}_j$ and corresponding fitted values $\hat{\mu}$, the *residual deviance* is the minimized total deviance

$$D(y, \hat{\mu}) = \sum_{i=1}^{n} w_i d(y_i, \hat{\mu}_i). \tag{6.12}$$

The residual deviance is a measure of the residual variability across n observations after fitting the model, similar to the RSS (2.8) for linear regression models. In fact, as Example 6.4 shows, the residual deviance is precisely the RSS for normal linear regression models. The quantity $D^*(y, \hat{\mu}) = D(y, \hat{\mu})/\phi$ is called the *scaled residual deviance*. Computing the scaled residual deviance obviously requires knowledge of the value of ϕ.

Table 6.3 The unit deviance $d(y_i, \hat{\mu}_i)$ for each observation i and the residual deviance $D(y, \hat{\mu})$ for the noisy miner data, where $w_i = 1$ for all i (Example 6.5)

y	$\hat{\mu}$	$d(y, \hat{\mu})$	$wd(y, \hat{\mu})$
0	0.5230	1.0459	1.0459
0	1.3016	2.6032	2.6032
3	2.5792	0.0652	0.0652
2	4.0691	1.2971	1.2971
8	3.6307	3.9016	3.9016
⋮	⋮	⋮	⋮
Residual deviance:			63.3180

The residual deviance for a fitted GLM in R named `fit` is returned using `deviance(fit)`.

Example 6.4. Using the unit deviance from Example 5.1, the residual deviance for the normal distribution is

$$D(y, \hat{\mu}) = \sum_{i=1}^{n} w_i (y_i - \hat{\mu}_i)^2 = \text{RSS},$$

and the scaled deviance is

$$D^*(y, \hat{\mu}) = \frac{1}{\sigma^2} \sum_{i=1}^{n} w_i (y_i - \hat{\mu}_i)^2 = \sum_{i=1}^{n} w_i \left(\frac{y_i - \hat{\mu}_i}{\sigma} \right)^2,$$

provided the value of σ^2 is known. □

Example 6.5. Using the unit deviance for the Poisson distribution (Table 5.1, p. 221), the residual deviance for the Poisson distribution is

$$D(y, \hat{\mu}) = 2 \sum_{i=1}^{n} \left\{ y_i \log \frac{y_i}{\hat{\mu}_i} - (y_i - \hat{\mu}_i) \right\}.$$

Since $\phi = 1$ for the Poisson distribution, the scaled residual deviance is identical to the residual deviance. Consider Model (6.11) (p. 247) fitted to the noisy miner data in Example 6.2 (data set: `nminer`). Summing the unit deviances (Table 6.3), the residual deviance for the model is $D(y, \hat{\mu}) = 63.3180$, where $\hat{\mu} = \exp(-0.8762 + 0.1140x)$ from (6.11). Using R, the residual deviance is

```
> deviance(nm.m1)
[1] 63.31798
```

□

6.5 Standard Errors for $\hat{\boldsymbol{\beta}}$

After computing the MLEs $\hat{\beta}_j$, the standard errors for the estimates are computed from the information matrix $\mathcal{I}_{jk}(\beta)$ shown in (6.8). The standard errors are the square roots of the diagonal elements of the inverted information matrix. Specifically,

$$\text{se}(\hat{\beta}_j) = \sqrt{\phi}\, v_j \tag{6.13}$$

where the v_j are the square-root diagonal elements of the inverse of the working information matrix with (j,k)th element $\sum_{i=1}^{n} W_i x_{ij} x_{ik}$. If ϕ is not known, then some estimate of it is used.

Example 6.6. Consider Model (6.11) (p. 247) fitted to the noisy miner data in Example 6.2 (data set: `nminer`). The summary output for the GLM in R shows the MLEs for the two coefficients, and the corresponding standard errors:

```
> coef(summary(nm.m1))
             Estimate Std. Error   z value     Pr(>|z|)
(Intercept) -0.8762114 0.28279293 -3.098421 1.945551e-03
Eucs         0.1139813 0.01243104  9.169092 4.770189e-20
```

□

* 6.6 Estimation of β: Matrix Formulation

In matrix terms, the score vector $\mathbf{U} = [U_0, \ldots, U_p]^T$ for $\boldsymbol{\beta}$ is

$$\mathbf{U} = \frac{1}{\phi} \mathrm{X}^T \mathrm{W} \mathrm{M} (\mathbf{y} - \boldsymbol{\mu}), \tag{6.14}$$

where W is the diagonal matrix of working weights W_i (6.7) and M is the diagonal matrix of link derivatives $\mathrm{d}\eta_i/\mathrm{d}\mu_i$. This gives the vector of derivatives of the log-likelihood with respect to the coefficient vector $\boldsymbol{\beta} = [\beta_0, \ldots, \beta_p]$. The Fisher information matrix for $\boldsymbol{\beta}$, with elements $\mathcal{I}_{jk}(\beta)$ is

$$\mathcal{I} = \frac{1}{\phi} \mathrm{X}^T \mathrm{W} \mathrm{X}. \tag{6.15}$$

The Fisher scoring iteration (Sect. 4.8) to compute the MLE of $\boldsymbol{\beta}$ is

$$\hat{\boldsymbol{\beta}}^{(r+1)} = \hat{\boldsymbol{\beta}}^{(r)} + \mathcal{I}(\hat{\boldsymbol{\beta}}^{(r)})^{-1} U(\hat{\boldsymbol{\beta}}^{(r)}) \tag{6.16}$$

$$= \hat{\boldsymbol{\beta}}^{(r)} + (\mathrm{X}^T \mathrm{W} \mathrm{X})^{-1} \mathrm{X}^T \mathrm{W} \mathrm{M} (\mathbf{y} - \hat{\boldsymbol{\mu}}), \tag{6.17}$$

where the superscript (r) denotes the rth iterate, and all quantities on the right hand side (including $\hat{\boldsymbol{\mu}}$) are evaluated at $\hat{\boldsymbol{\beta}}^{(r)}$. Note that ϕ cancels out of the term $\mathcal{I}()^{-1}\mathbf{U}()$ on the right hand side.

The scoring iteration can be re-organized as iteratively weighted least squares as

$$\hat{\boldsymbol{\beta}}^{(r+1)} = (\mathrm{X}^T\mathrm{W}\mathrm{X})^{-1}\mathrm{X}^T\mathrm{W}\mathbf{z} \tag{6.18}$$

where $\mathbf{z}$ is the working response vector

$$\mathbf{z} = \hat{\boldsymbol{\eta}} + \mathrm{M}(\mathbf{y} - \hat{\boldsymbol{\mu}}), \tag{6.19}$$

where all quantities on the right hand side are evaluated at $\hat{\boldsymbol{\beta}}^{(r)}$. After each iteration, the linear predictor is updated as $\hat{\boldsymbol{\eta}}^{(r+1)} = \mathbf{o} + \mathrm{X}\hat{\boldsymbol{\beta}}^{(r+1)}$, where $\mathbf{o}$ is the vector of offsets, and the fitted values are updated as $\hat{\boldsymbol{\mu}}^{(r+1)} = g^{-1}(\hat{\boldsymbol{\eta}}^{(r+1)})$.

After the iterations have converged, the covariance matrix of the regression parameters is estimated from inverse information matrix

$$\mathrm{var}[\hat{\boldsymbol{\beta}}] = \mathcal{I}^{-1} = \phi(\mathrm{X}^T\mathrm{W}\mathrm{X})^{-1},$$

where some estimate of ϕ must be used if the value of ϕ is unknown. In particular, the standard errors are obtained from the diagonal elements

$$\mathrm{se}(\hat{\beta}_j) = \sqrt{\phi}\, v_j$$

where the v_j are the square-root diagonal elements of $(\mathrm{X}^T\mathrm{W}\mathrm{X})^{-1}$.

Example 6.7. The covariance matrix of the coefficients for the noisy miner data (`nminer`) is in the output variable `cov.scaled` that is contained in the model `summary()`:

```
> nm.m1 <- glm( Minerab ~ Eucs, data=nminer, family=poisson)
> cov.mat <- summary(nm.m1)$cov.scaled
> round( cov.mat, digits=5)
            (Intercept)     Eucs
(Intercept)     0.07997 -0.00324
Eucs           -0.00324  0.00015
```

The standard errors $\mathrm{se}(\hat{\beta}_j)$ are the square root of the diagonal elements:

```
> sqrt( diag( cov.mat ) )
(Intercept)        Eucs
 0.28279293  0.01243104
```

These agree with the standard errors computed by R within computer precision (Example 6.6, p. 250). □

The variance of $\hat{\boldsymbol{\mu}}$ is found by first considering $\hat{\boldsymbol{\eta}}$. Consider given values of the p' explanatory variables, given in the row vector $\mathbf{x}_g$. The best estimate of η is $\hat{\eta} = \mathbf{x}_g\hat{\boldsymbol{\beta}}$. The variance of $\hat{\eta}$ is

$$\mathrm{var}[\hat{\eta}] = \mathrm{var}[\mathbf{x}_g\hat{\boldsymbol{\beta}}] = \mathbf{x}_g(\mathrm{X}^T\mathrm{W}\mathrm{X})^{-1}\mathbf{x}_g^T\phi,$$

where some estimate of ϕ must be used if the value of ϕ is unknown. The variance of $\hat{\mu}$ is harder to compute directly. However, for inference involving μ (such as confidence intervals for μ), we work with $\hat{\eta}$ and then convert to $\hat{\mu}$ via the link function $\mu = g^{-1}(\eta)$.

6.7 Estimation of GLMs Is Locally Like Linear Regression

The formulation of the scoring algorithm for maximum likelihood estimation of GLMs as IRLS (Sects. 6.3 and 6.6) is much more than a computational convenience. It reveals an analogy between GLMs and linear regression which has many uses. To a first approximation, fitting a GLM is equivalent to least squares regression with responses z_i and weights W_i, with the working responses and working weights set to their final converged values. Conveniently, the *working residuals*

$$e_i = z_i - \hat{\eta}_i \tag{6.20}$$

and the working weights are stored as part of the standard output when GLMs are fitted in R (as `fit$residuals` and `fit$weights` respectively for a fitted model called `fit`). This means that all the methodology developed in Chaps. 2 and 3 can be applied to GLMs, simply by treating the working responses and working weights as fixed values. Quantities which may be computed in this way include the fitted values $\hat{\mu}$; the variance of $\hat{\beta}_j$; the leverages h; the value of the raw residuals; Cook's distance; DFFITS; DFBETAS. These connections are explored in later chapters.

6.8 Estimating ϕ

6.8.1 Introduction

Although knowledge of ϕ was not required for estimating the β_j, it will be required for hypothesis testing and confidence intervals (Chap. 7). So, unless ϕ is known *a priori*, it must be estimated. The most useful estimators of ϕ are described in this section.

The most common models for which ϕ is known are binomial and Poisson EDMs. Even then, estimation of ϕ can sometimes be useful when we wish to relax the usual assumptions, as we will see in Sects. 9.8 and 10.5.

6.8.2 The Maximum Likelihood Estimator of ϕ

In principle, we could apply MLE directly to the log-likelihood to estimate ϕ, just as we did for the β_j. However the MLE of ϕ is seriously biased, unless n is very large relative to p'.

Consider the case of normal linear regression models. Then the MLE of $\phi = \sigma^2$ is

$$\hat{\sigma}^2 = \frac{1}{n}\sum_{i=1}^{n} w_i(y_i - \hat{\mu}_i)^2, \tag{6.21}$$

which is never used because it is biased. Instead,

$$s^2 = \frac{1}{n-p'}\sum_{i=1}^{n} w_i(y_i - \hat{\mu}_i)^2, \tag{6.22}$$

is unbiased and is used in practice.

There are at least three ways to generalize the unbiased estimator s^2 to GLMs so that the normal linear regression model results remain special cases of the GLM results. We consider these in the next three subsections.

6.8.3 Modified Profile Log-Likelihood Estimator of ϕ

A more sophisticated strategy for estimating ϕ is based on the *profile log-likelihood*. The profile log-likelihood estimate for ϕ is found by first assuming ϕ is fixed and maximizing the log-likelihood with respect to β. Write the log-likelihood as $\ell(\hat{\beta}_0, \ldots, \hat{\beta}_p, \phi; y)$. Then, write the log-likelihood as a function of ϕ, treating each $\hat{\beta}_j$ as being fixed and maximize this log-likelihood with respect to ϕ. That is, the profile log-likelihood for ϕ is $\ell(\phi) = \ell(\hat{\beta}_0, \ldots, \hat{\beta}_p, \phi; y)$.

The *modified profile log-likelihood* (MPL) is, as the name suggests, a modification of the profile log-likelihood with better properties:

$$\ell^0(\phi) = \frac{p'}{2}\log\phi + \ell(\hat{\beta}_0, \ldots, \hat{\beta}_p, \phi; y).$$

The modified profile log-likelihood includes a penalty term which penalizes small values of ϕ. The value of ϕ maximizing $\ell^0(\phi)$ is called the modified profile log-likelihood estimator of ϕ, and is denoted $\hat{\phi}^0$. The MPL estimator is a consistent estimator and is approximately unbiased, even in quite small samples.

The main disadvantage of the MPL estimator is that, like the MLE, it is often inconvenient to compute. The estimator generally requires iterative estimation (as usual, the normal linear case is an exception). Even more

seriously, the derivatives of the log-likelihood with respect to ϕ involve the terms $\partial a(y, \phi/w)/\partial\phi$, which for some EDMs are difficult to obtain since $a()$ may not have a closed form.

Example 6.8. Consider the normal distribution. The profile log-likelihood is

$$\ell(\sigma^2) = -\frac{1}{2}\sum_{i=1}^{n} \log 2\pi\sigma^2/w_i - \frac{1}{2\sigma^2}\sum_{i=1}^{n} w_i(y_i - \hat{\mu}_i)^2.$$

Differentiating with respect to σ^2, setting to zero, and solving for σ^2 produces the profile log-likelihood estimate (identical to the MLE (6.21) for this case). The *modified* profile log-likelihood is

$$\ell^0(\sigma^2) = \frac{p'}{2}\log\sigma^2 - \frac{1}{2}\sum_{i=1}^{n} \log 2\pi\sigma^2/w_i - \frac{1}{2\sigma^2}\sum_{i=1}^{n} w_i(y_i - \hat{\mu}_i)^2.$$

Differentiating with respect to σ^2, setting to zero, and then solving, produces the modified profile likelihood estimator of σ^2

$$(\hat{\sigma}^2)^0 = \frac{1}{n-p'}\sum_{i=1}^{n} w_i(y_i - \hat{\mu}_i)^2,$$

identical to s^2 in (6.22). □

6.8.4 Mean Deviance Estimator of ϕ

It is easy to show (Problem 6.4) that, if the saddlepoint approximation for the EDM probability function (5.4.4) is exact, the maximum likelihood estimator of ϕ is the simple mean deviance $D(y, \hat{\mu})/n$. Like all MLEs, this estimator fails to take account of estimation of the β_j and the residual degrees of freedom. The linear regression case (6.22) motivates the *mean deviance estimator* of ϕ:

$$\tilde{\phi} = \frac{D(y, \hat{\mu})}{n-p'}.$$

Example 6.9. For normal GLMs, the residual deviance is equal to the RSS, so the mean deviance estimator of the dispersion parameter is simply $\tilde{\phi} = s^2$, the usual unbiased estimator of σ^2 (6.22). □

6.8.5 Pearson Estimator of ϕ

As pointed out in Sect. 6.7, GLMs can be treated to a first approximation like least squares models. Suppose we take this approach, and compute the RSS from the fitted model, treating the working responses and working weights as the actual responses and weights. This gives the working RSS

$$X^2 = \sum_{i=1}^{n} W_i(z_i - \hat{\eta}_i)^2 \tag{6.23}$$

$$= \sum_{i=1}^{n} \frac{w_i(y_i - \hat{\mu}_i)^2}{V(\hat{\mu}_i)}, \tag{6.24}$$

known as the *Pearson statistic*. Note that the unit Pearson statistic $\{w(y - \hat{\mu})^2\}/V(\hat{\mu})$ represents the contribution to the Pearson statistic of each observation, just as the unit deviance does for the deviance. The Pearson statistic makes intuitive sense as a measure of residual variability because the variance function $V(\hat{\mu})$ in the denominator of the unit statistic divides out the effect of non-constant variance from the squared residuals.

Continuing the analogy with least squares, the *Pearson estimator* of ϕ is defined by

$$\bar{\phi} = \frac{X^2}{n - p'}. \tag{6.25}$$

Example 6.10. For normal GLMs, $V(\mu) = 1$ (Table 5.1, p. 221) so the Pearson statistic reduces to the usual RSS, $X^2 = \text{RSS}$, and the Pearson estimator of the dispersion parameter is $\bar{\phi} = s^2$. The normal is the only distribution for which the the mean deviance and Pearson estimators of ϕ are the same. □

Example 6.11. The Poisson distribution has the variance function $V(\mu) = \mu$ (Table 5.1, p. 221), so the Pearson statistic is

$$X^2 = \sum_{i=1}^{n} \frac{w_i(y_i - \hat{\mu}_i)^2}{\hat{\mu}_i}.$$

□

6.8.6 Which Estimator of ϕ Is Best?

Given the different methods for estimating ϕ, which should be used? The MLE $\hat{\phi}$ is biased, unless p'/n is very small, so $\hat{\phi}$ is rarely used. On the other hand, the modified profile estimator $\hat{\phi}^0$ has excellent theoretical properties. It should be nearly efficient and nearly consistent. However it is often inconvenient to compute.

The mean deviance and Pearson estimators are very convenient, as they are readily available from the unit deviances and working residuals respectively. The mean deviance estimator should behave well when the saddlepoint approximation holds; that is, for normal or inverse Gaussian GLMs or when ϕ is relatively small. The Pearson estimator, however, is almost universally applicable, because $(y-\mu)^2/V(\mu)$ should always be unbiased for ϕ if μ is the correct mean and $V(\mu)$ is the correct variance function. In other words, the Pearson estimator is approximately unbiased given only first and second moment assumptions. This makes the Pearson estimator the most robust estimator, in the sense that it relies on fewest assumptions. For this reason, the `glm()` function in R uses the Pearson estimator for ϕ by default. In practice, the Pearson estimator tends to be more variable (less precise) but less biased than the mean deviance estimator.

As usual, it makes no difference for normal GLMs, because $\hat{\phi}^0$, $\tilde{\phi}$ and $\bar{\phi}$ are identical, and equal to the residual variance s^2 used in Chaps. 2 and 3.

For gamma GLMs, the mean deviance estimator can be sensitive to rounding errors as y approaches zero [5, p. 295, 296]. Indeed, the plot of the unit deviance (Fig. 5.2, bottom left panel, p. 222) shows how the value of $d(y,\mu)$ increases rapidly as $y \to 0$. A small change in y when y is small can result in a correspondingly large change in the value of $d(y,\mu)$ and hence in the value of $D(y,\hat{\mu})$. For this reason, the Pearson estimator may be preferred to the mean deviance estimator for gamma GLMs when rounding is an issue; that is, when small responses are not recorded to at least two or three significant figures. The same remark applies to other EDMs with support on the positive real line.

For binomial and Poisson GLMs, $\phi = 1$ and no estimation is necessary. However, the issue may arise for over-dispersed binomial or Poisson GLMs, which are considered in later chapters.

Example 6.12. In Example 3.14 (data set: `trees`), a gamma GLM is suggested for the cherry tree data, with systematic component $\log\mu = \beta_0 + \beta_1 \log d + \beta_2 \log h$. To fit this model in R, use:

```
> data(trees)
> cherry.m1 <- glm( Volume ~ log(Height) + log(Girth), data=trees,
                    family=Gamma(link="log"))
```

The regression parameters are

```
> coef( cherry.m1 )
(Intercept) log(Height)  log(Girth)
  -6.691109    1.132878    1.980412
```

Compute the Pearson estimator of ϕ defined by (6.23) explicitly in R using:

```
> w <- weights(cherry.m1, type="working")
> e <- residuals(cherry.m1, type="working")
> sum( w * e^2 ) / df.residual(cherry.m1);
[1] 0.006427286
```

Alternatively, since the Pearson estimator is used by default in R:

```
> summary(cherry.m1)$dispersion
[1] 0.006427286
```

The mean deviance estimator is

```
> deviance(cherry.m1) / df.residual(cherry.m1)
[1] 0.006554117
```

The two estimates are similar. □

6.9 Using R to Fit GLMs

In R, GLMs are fitted to data using the function `glm()`, and the inputs `formula`, `data`, `weights` and `subset` are used in the same way as for `lm()` (see Sect. 2.14, p. 79). The systematic component is given by the `formula` input, specified in the same way as for linear regression models using `lm()`. To use `glm()`, the distribution and link function also must be specified using the input `family`. As an example, a GLM(Poisson; log) model is specified using

```
glm( y ~ x1 + x2, family=poisson(link="log") )
```

Similarly, a GLM(binomial; logit) model is specified as

```
glm( y ~ x1 + x2, family=binomial(link="logit") )
```

If a link function is not explicitly given, the default link function used by R is the canonical link function (Table 6.4). As an example, the models above could be specified as

```
glm( y ~ x1 + x2, family=poisson )
glm( y ~ x1 + x2, family=binomial )
```

since the logarithmic link function is the canonical link function for a Poisson GLM, and the logistic link function is the canonical link function for the binomial GLM.

In R, valid GLM families are (noting the capitalization carefully):

- `gaussian()`: Specifying the Gaussian (normal) distribution;
- `binomial()`: Specifying a binomial EDM (Chap. 9);
- `poisson()`: Specifying a Poisson EDM (Chap. 10);
- `Gamma()`: Specifying a gamma EDM (Chap. 11);
- `inverse.gaussian()`: Specifying an inverse Gaussian EDM (Chap. 11).

More details are provided about each family in the indicated chapters. Three other families are discussed in Sect. 8.10, and are mentioned here for completeness: `quasi()`, `quasibinomial()` and `quasipoisson()`. Other families can also be used by writing a new `family` function. For example, the `tweedie()` family function (in package **statmod**) was written to enable the fitting of

Table 6.4 The link functions accepted by different `glm()` families in R are indicated using a tick ✓. The default (and canonical) links used by R are indicated with stars ★ (Sect. 6.9)

Link function	`gaussian`	`binomial` and `quasibinomial`	`poisson` and `quasipoisson`	`Gamma`	`inverse.gaussian`	`quasi`
`identity`	★		✓	✓	✓	★
`log`	✓		★	✓	✓	✓
`inverse`	✓			★	✓	✓
`sqrt`			✓			✓
`1/mu^2`					★	✓
`logit`		★				✓
`probit`		✓				✓
`cauchit`		✓				
`cloglog`		✓				✓
`power`						✓

Tweedie GLMs (Chap. 12). The different families accept different link functions, and have different defaults (Table 6.4). The `quasi()` family also accepts link functions defined using `power()`, which have the form $\eta = \mu^\lambda$ for $\lambda \geq 0$; the logarithmic link function is obtained when $\lambda = 0$.

Usually, the output from a fitted GLM is sent to an output object: `fit <- glm(y ~ x1 + x2, family=poisson)`, for example. The output object `fit` contains substantial information; see `?glm`. The most useful information is extracted from `fit` using extractor functions, which include:

- `coef(fit)`: Returns the coefficients $\hat{\beta}_j$ of the systematic component.
- `deviance(fit)`: Returns the residual deviance $D(y, \hat{\mu})$ for the fitted GLM.
- `summary(fit)`: Returns the summary of the fitted GLM (some parts of which are discussed in Chap. 7), with the corresponding standard errors, t- or z-statistics and two-tailed P-values for testing H_0: $\beta_j = 0$; the value of ϕ, or the Pearson estimate of ϕ if ϕ is unknown; the residual deviance $D(y, \hat{\mu})$ and corresponding residual degrees of freedom; and the AIC. The output of `summary()` (for example, `out <- summary(fit)`) contains substantial information (see `?summary.glm`). For example, `out$dispersion` displays the value of ϕ or its estimate, whichever is appropriate; `coef(out)` displays the parameter estimates and standard errors, plus the t- or z-values and two-tailed P-values for testing H_0: $\beta_j = 0$.
- `df.residual(fit)`: Extracts the residual degrees of freedom.
- `fitted(fit)`: Extracts the fitted values $\hat{\mu}$; `fitted.values(fit)` is equivalent.

The algorithm for fitting GLMs in R is usually stable and fast. However, sometimes the parameters controlling the fitting algorithm need to be adjusted using the input `glm.control()` when calling the `glm()` function. The following parameters can be adjusted:

- The convergence criterion (Sect. 6.4, p. 248), where `epsilon` is the value of ϵ. By default, `epsilon` $= 10^{-8}$. Setting `epsilon` to some other (usually smaller) value is occasionally useful.
- The maximum number of iterations, by changing the value of `maxit`. By default, the IRLS algorithm is permitted a maximum of 25 iterations. Occasionally the value of `maxit` needs to be increased to ensure the Fisher scoring algorithm converges.
- The information displayed. If the algorithm fails or produces unexpected results, viewing the details of each iteration in the IRLS algorithm can help diagnose the problem, by setting `trace=TRUE`.

As with `lm()`, models may be updated using `update()` rather than being completely specified (see Sect. 2.10.1, p. 61).

Example 6.13. The noisy miner data (data set: `nminer`) has been used in examples in this chapter. The following R commands fit Model (6.11) (p. 247):

```
> data(nminer)
> nm.m1 <- glm( Minerab ~ Eucs, data=nminer, family=poisson)
```

The R `summary()` for this model is shown in Fig. 6.1.

To demonstrate the use of `glm.control()`, we fit the model by changing the fitting parameters. We set the convergence criterion to $\epsilon = 10^{-15}$, permit a maximum of three iterations, and view the details of each iteration:

```
nm.m2 <- update( nm.m1, control=glm.control(
                          maxit=3,        # Max of 3 iterations
                          epsilon=1e-15,  # Stopping criterion
                          trace=TRUE) )   # Show details
Deviance = 82.14668 Iterations - 1
Deviance = 64.49515 Iterations - 2
Deviance = 63.32603 Iterations - 3
Warning message:
In glm.fit(x = X, y = Y, weights = weights, start = start,
etastart = etastart,  :  algorithm did not converge
```

The algorithm has not converged in three iterations to the requested level of accuracy $\epsilon = 10^{-15}$: the `trace` shows that the residual deviance is yet to converge. □

6.10 Summary

Chapter 6 discusses fitting GLMs to data. Fitting GLMs relies on the structure provided by EDMs. For example, for EDMs (Sect. 6.2) the derivative

$$\frac{\partial \log \mathcal{P}(y; \mu, \phi/w)}{\partial \mu} = \frac{w(y-\mu)}{\phi V(\mu)}$$

```
> data(nminer)
> nm.m1 <- glm( Minerab ~ Eucs, data=nminer, family=poisson)
> summary(nm.m1)

Call:
glm(formula = Minerab ~ Eucs, family = poisson, data = nminer)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.1454  -1.2530  -0.9673   0.5634   3.5603

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept) -0.87621    0.28279  -3.098  0.00195 **
Eucs         0.11398    0.01243   9.169  < 2e-16 ***
---
Signif. codes:  0 `***' 0.001 `**' 0.01 `*' 0.05 `.' 0.1 ` ' 1

(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 150.545  on 30  degrees of freedom
Residual deviance:  63.318  on 29  degrees of freedom
AIC: 121.47

Number of Fisher Scoring iterations: 5
```

Fig. 6.1 An example of the output of the `summary()` command after using `glm()` (Sect. 6.9)

has a simple form. The estimates $\hat{\beta}_j$ are found by Fisher scoring, using the iteratively reweighted least squares (IRLS) algorithm (Sect. 6.3). Importantly, the value of ϕ is not needed to find estimates of the β_j.

The matrix form of the score equations and the information matrix for $\boldsymbol{\beta}$ are $\mathbf{U} = \mathrm{X}^T\mathrm{WM}(\mathbf{y} - \boldsymbol{\mu})/\phi$ and $\mathcal{I} = \mathrm{X}^T\mathrm{WX}/\phi$, where W is the diagonal matrix of working weights W_i, and M is the diagonal matrix of link derivatives $\mathrm{d}\eta_i/\mathrm{d}\mu_i$ (Sect. 6.6).

The residual deviance $D(y, \hat{\mu}) = \sum_{i=1}^{n} w_i d(y_i, \hat{\mu}_i)$ is a measure of the total residual variability from a fitted model across n observations (Sect. 6.4). The scaled residual deviance is $D^*(y, \hat{\mu}) = D(y, \hat{\mu})/\phi$ (Sect. 6.4).

The standard errors of $\hat{\beta}_j$ are $\mathrm{se}(\hat{\beta}_j) = \sqrt{\phi}\, v_j$, where the v_j are the square-root diagonal elements of the inverse of the working information matrix. If ϕ is not known, then some estimate of it is used (Sect. 6.5).

Importantly, the estimation algorithm for fitting GLMs is locally the same as for fitting linear regression models, so various quantities used in regression can be computed from the final iteration of the IRLS algorithm for GLMs, such as the fitted values, the variance of $\hat{\beta}_j$, leverages, Cook's distance values, DFFITS, DFBETAS and the raw residuals (Sect. 6.7).

The dispersion parameter can be estimated using a modified profile log-likelihood estimator $\hat{\phi}^0$ (Sect. 6.8.3), the mean deviance estimator $\tilde{\phi}$ (Sect. 6.8.4) or the Pearson estimator $\bar{\phi}$ (Sect. 6.8.5). For all these estimators, the linear regression model results are special cases of the GLM results (Sect. 6.8). In R, the dispersion parameter ϕ is estimated using the Pearson estimate (Sect. 6.8).

The next chapter considers methods for inference concerning the fitted model.

Problems

Selected solutions begin on p. 537. Problems preceded by an asterisk * refer to the optional sections in the text, and may require matrix manipulations.

6.1. Consider a link function $\eta = g(\mu)$. Find the first two terms of the Taylor series expansion of $g(y)$ expanded about μ, and show that the result is equivalent to z, the working responses (6.9) (p. 246).

* **6.2.** Consider the linear regression model. Show that the iteration (6.18) (p. 251) reduces to the equation for finding the regression parameter estimates in the linear regression model case: $\hat{\boldsymbol{\beta}} = (\mathrm{X}^T\mathrm{W}\mathrm{X})^{-1}\mathrm{X}^T\mathrm{W}\mathbf{y}$.

6.3. If μ is known, show that the Pearson estimator of ϕ is unbiased (that is, $\mathrm{E}[\bar{\phi}] = \phi$).

6.4. Suppose the saddlepoint approximation (Sect. 5.4.2) $\tilde{\mathcal{P}}(y;\mu,\phi)$ is used to approximate the EDM probability function $\mathcal{P}(y;\mu,\phi)$. After writing $\tilde{\ell}(\mu,\phi;y) = \sum_{i=1}^{n} \log \tilde{\mathcal{P}}(y_i;\mu_i,\phi)$, show that the solution to $\partial\tilde{\ell}(\mu,\phi;y)/\partial\phi = 0$ is the simple mean deviance $D(y,\hat{\mu})/n$.

6.5. If the canonical link function is used in a GLM, then $V(\mu) = 1/g'(\mu) = \mathrm{d}\mu/\mathrm{d}\eta$ (Problem 5.20). Assuming a canonical link function, show that:

1. $U(\beta_j) = \sum_{i=1}^{n} w_i(y_i - \mu_i)x_{ji}/\phi$.
2. $\mathrm{d}U(\beta_j)/\mathrm{d}\mu = -\sum_{i=1}^{n} w_i x_{ji}/\phi$.

These results are used in some of the problems that follow.

6.6. Consider a binomial GLM using the canonical link function.

1. Determine the score function $U(\beta_j)$ and the Fisher information $\mathcal{I}_{jk}(\beta)$.
2. Determine the working responses z_i.

HINT: The results from Problem 6.5 will prove useful.

6.7. Consider a gamma GLM using the canonical link function.

1. Determine the score function $U(\beta_j)$ and the Fisher information $\mathcal{I}_{jk}(\beta)$.
2. Determine the working responses z_i.
3. Determine the Pearson estimator of ϕ.

HINT: The results from Problem 6.5 will prove useful.

6.8. Repeat Problem 6.7, but using the often-used logarithmic link function instead of the canonical link function.

6.9. Consider an inverse Gaussian GLM using a logarithmic link function, which is not the canonical link function.

1. Determine the score function $U(\beta_j)$ and the Fisher information $\mathcal{I}_{jk}(\beta)$.
2. Determine the working responses z_i.
3. Find the MLE of ϕ.
4. Find the mean deviance estimator of ϕ.
5. Find the Pearson estimator of ϕ.

6.10. Children were asked to build towers as high as they could out of cubical and cylindrical blocks [3, 6]. The number of blocks used and the time taken were recorded (data set: `blocks`). In this problem, only consider the number of blocks used y and the age of the child x. In Problem 5.25, a GLM was proposed for these data.

1. Fit this GLM using R, and write down the fitted model.
2. Determine the standard error for each regression parameter.
3. Compute the residual deviance.

6.11. Nambe Mills, Santa Fe, New Mexico [2, 7], is a tableware manufacturer. After casting, items produced by Nambe Mills are shaped, ground, buffed, and polished. In 1989, as an aid to rationalizing production of its 100 products, the company recorded the total grinding and polishing times and the diameter of each item (Table 5.3; data set: `nambeware`). In this problem, only consider the item price y and the item diameter x. In Problem 5.26, a GLM was proposed for these data.

1. Fit this GLM using R, and write down the fitted model.
2. Determine the standard error for each regression parameter.
3. Compute the residual deviance.
4. Compute the mean deviance estimate of ϕ.
5. Compute the Pearson estimate of ϕ.

References

[1] Chatfield, C.: Problem Solving: A Statistician's Guide, second edn. Texts in Statistical Science. Chapman and Hall/CRC, London (1995)
[2] Data Desk: Data and story library (DASL) (2017). URL http://dasl.datadesk.com
[3] Johnson, B., Courtney, D.M.: Tower building. Child Development **2**(2), 161–162 (1931)
[4] Maron, M.: Threshold effect of eucalypt density on an aggressive avian competitor. Biological Conservation **136**, 100–107 (2007)

[5] McCullagh, P., Nelder, J.A.: Generalized Linear Models, second edn. Monographs on Statistics and Applied Probability. Chapman and Hall, London (1989)
[6] Singer, J.D., Willett, J.B.: Improving the teaching of applied statistics: Putting the data back into data analysis. The American Statistician **44**(3), 223–230 (1990)
[7] Smyth, G.K.: Australasian data and story library (OzDASL) (2011). URL http://www.statsci.org/data

Chapter 7
Generalized Linear Models: Inference

There is no more pressing need in connection with the examination of experimental results than to test whether a given body of data is or is not in agreement with any suggested hypothesis.
Sir Ronald A. Fisher [2, p. 250]

7.1 Introduction and Overview

Section 4.10 discussed three types of inferential approaches based on likelihood theory: Wald, score and likelihood ratio. In Chap. 7, these approaches are applied in the context of GLMs. We first consider inference when ϕ is known (Sect. 7.2), then the large-sample asymptotic results (Sect. 7.3) that underlie all the distributional results for the test statistics in that section. Section 7.4 then introduces goodness-of-fit tests to determine whether the linear predictor sufficiently describes the systematic trends in the data. The distributional results for these goodness-of-fit tests rely on small dispersion asymptotic results (the large sample asymptotics do not apply), which are discussed in Sect. 7.5 where guidelines are presented for when these results hold. We then consider inference when ϕ is unknown (Sect. 7.6), and include a discussion of using the different estimates of ϕ. Wald, score and likelihood ratio tests are then compared (Sect. 7.7). Techniques for comparing non-nested GLMs (Sect. 7.8) are then discussed, followed by automated methods for selecting GLMs (Sect. 7.9).

7.2 Inference for Coefficients When ϕ Is Known

7.2.1 Wald Tests for Single Regression Coefficients

The simplest tests concerning regression coefficients are Wald tests, because they depend only on the estimated coefficients and standard errors. The regression coefficients $\hat{\beta}_j$ are approximately normally distributed when n is reasonably large, and this is the basis of Wald tests.

P. K. Dunn, G. K. Smyth, *Generalized Linear Models with Examples in R*,
Springer Texts in Statistics, https://doi.org/10.1007/978-1-4419-0118-7_7

Consider a GLM with p' regression parameters fitted to some data in a situation where ϕ is known. The Wald test of the null hypothesis H_0: $\beta_j = \beta_j^0$, where β_j^0 is some given value (typically zero), consists of comparing $\hat{\beta}_j - \beta_j^0$ to the standard error of $\hat{\beta}_j$ (Sect. 4.10.1). For a GLM with ϕ known, the Wald test statistic is

$$Z = \frac{\hat{\beta}_j - \beta_j^0}{\text{se}(\hat{\beta}_j)}$$

where the standard error $\text{se}(\hat{\beta}_j) = \sqrt{\phi}\, v_j$ is given by (6.13). If H_0 is true, Z follows approximately the standard normal distribution.

In R, using the `summary()` command shows the values of Z, $\text{se}(\hat{\beta}_j)$ and the two-tailed P-values for testing $\beta_j = 0$ for each fitted regression parameter.

Example 7.1. For the noisy miner data [4] (Example 1.5; data set: `nminer`), the Wald statistics for testing H_0: $\beta_j = 0$ for each parameter in the fitted model are shown as part of the output of the `summary()` command. More briefly, `coef(summary())` shows just the information about the coefficients:

```
> library(GLMsData); data(nminer)
> nm.m1 <- glm( Minerab ~ Eucs, data=nminer, family=poisson)
> printCoefmat( coef( summary(nm.m1) ) )
             Estimate Std. Error z value  Pr(>|z|)
(Intercept) -0.876211   0.282793 -3.0984  0.001946 **
Eucs         0.113981   0.012431  9.1691 < 2.2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The evidence suggests both coefficients in the model are non-zero. □

7.2.2 Confidence Intervals for Individual Coefficients

Confidence intervals may be computed using the Wald, score, or the likelihood-ratio test statistic as in Sect. 4.11 (p. 200). In practice, the Wald statistic is most commonly used, because the necessary quantities for computing the Wald standard errors are computed in the final iteration of the fitting algorithm so no further computations are necessary. Confidence intervals based on Wald statistics are symmetric on the η scale. The $100(1-\alpha)\%$ confidence interval for β_j when ϕ is known is

$$\hat{\beta}_j \pm z^*_{\alpha/2} \text{se}(\hat{\beta}_j)$$

where $z^*_{\alpha/2}$ is the value of z such that an area $\alpha/2$ is in each tail of the standard normal distribution. The R function `confint()` computes Wald confidence intervals from fitted `glm()` objects.

Example 7.2. For the noisy miner data (data set: `nminer`), the 95% confidence intervals for both coefficients are:

```
> confint(nm.m1)
                 2.5 %     97.5 %
(Intercept) -1.45700887 -0.3465538
Eucs         0.08985068  0.1386685
```

□

7.2.3 Confidence Intervals for μ

The fitted values $\hat{\mu}$ estimate the mean value for given values of the explanatory variables. Since $\hat{\eta} = g(\hat{\mu})$ is estimated from the $\hat{\beta}_j$, which are estimated with uncertainty, the estimates of $\hat{\mu}$ are also estimated with uncertainty. We initially work with $\hat{\eta}$, for which $\widehat{\text{var}}[\hat{\eta}]$ is easily found (Sect. 6.6). When ϕ is known, a $100(1-\alpha)\%$ Wald confidence interval for η is

$$\hat{\eta} \pm z^*_{\alpha/2}\text{se}(\hat{\eta}),$$

where $\text{se}(\hat{\eta}) = \sqrt{\text{var}[\hat{\eta}]}$, and where $z^*_{\alpha/2}$ is the value such that an area $\alpha/2$ is in each tail of the standard normal distribution. The confidence interval for μ is found by applying the inverse link function (that is, $\mu = g^{-1}(\eta)$) to the lower and upper limit of the interval found for $\hat{\eta}$. Note that the confidence interval is necessarily symmetric on the η scale.

Rather than explicitly returning a confidence interval, R optionally returns the standard errors when making predictions using `predict()`, by using the input `se.fit=TRUE`. This information can be used to form confidence intervals. Note that `predict()` returns the value of $\hat{\eta}$ by default, and the fitted values $\hat{\mu}$ (and corresponding standard errors if `se.fit=TRUE`) are returned by specifying `type="response"`.

Example 7.3. For the noisy miner data `nminer`, suppose we wish to estimate the mean number of noisy miners for a transect with ten eucalyptus trees per 2 ha transect. First, we compute the predictions and standard errors on the scale of the linear predictor:

```
>    # By default, this computes statistics on the linear predictor scale:
> out <- predict( nm.m1,                        # The model used to predict
                  newdata=data.frame(Eucs=10), # New data for predicting
                  se.fit=TRUE)                 # Return the std errors
> out2 <- predict( nm.m1, newdata=data.frame(Eucs=10), se.fit=TRUE,
                  type="response") # Return predictions on mu scale
> c( exp( out$fit ), out2$fit )     # Both methods give the same answer
      1       1
1.30161 1.30161
```

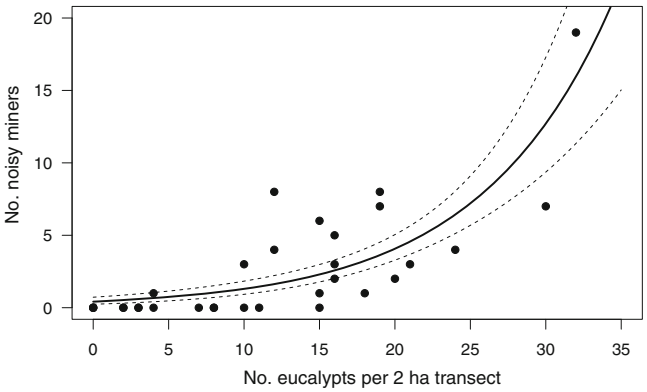

Fig. 7.1 The predicted relationship between the mean number of noisy miners and the number of eucalyptus trees (solid), with the 95% confidence intervals shown (dashed lines) (Example 7.3)

Then we form the confidence interval for μ by using the inverse of the logarithmic link function:

```
> zstar <- qnorm(p=0.975) # For 95% CI
> ci.lo <- exp( out$fit - zstar*out$se.fit)
> ci.hi <- exp( out$fit + zstar*out$se.fit)
> c( Lower=ci.lo, Estimate=exp(out$fit), Upper=ci.hi)
   Lower.1 Estimate.1    Upper.1
  0.924013   1.301610   1.833512
```

We see that $\hat{\mu} = 1.302$, and that the 95% interval is from 0.9240 to 1.834. Notice that this confidence interval is not symmetric:

```
> c( ci.lo-exp(out$fit), ci.hi-exp(out$fit))
         1          1
-0.3775972  0.5319019
```

This idea can be extended to show the confidence intervals for all transects with varying numbers of eucalyptus trees (Fig. 7.1):

```
> newEucs <- seq(0, 35, length=100)
> newMab <- predict( nm.m1, se.fit=TRUE, newdata=data.frame(Eucs=newEucs))
> ci.lo <- exp(newMab$fit-zstar*newMab$se.fit)
> ci.hi <- exp(newMab$fit+zstar*newMab$se.fit)
> plot( Minerab~Eucs, data=nminer,
     xlim=c(0, 35), ylim=c(0, 20), las=1, pch=19,
     xlab="No. eucalypts per 2 ha transect", ylab="No. noisy miners")
> lines(exp(newMab$fit) ~ newEucs, lwd=2)
> lines(ci.lo ~ newEucs, lty=2); lines(ci.hi ~ newEucs, lty=2)
```

The intervals are wider as $\hat{\mu}$ gets larger, since $V(\mu) = \mu$ for the Poisson distribution. □

7.2.4 Likelihood Ratio Tests to Compare Nested Models: χ^2 Tests

Consider comparing two nested GLMs, based on the same EDM but with different fitted systematic components:

$$\begin{aligned} \text{Model A:} \quad & g(\hat{\mu}_A) = \hat{\beta}_0 + \hat{\beta}_1 x_1 + \cdots + \hat{\beta}_{p_A} x_{p_A} \\ \text{Model B:} \quad & g(\hat{\mu}_B) = \hat{\beta}_0 + \hat{\beta}_1 x_1 + \cdots + \hat{\beta}_{p_A} x_{p_A} + \cdots + \hat{\beta}_{p_B} x_{p_B}. \end{aligned}$$

Notice that Model A is a special case of Model B, with $p_B > p_A$. We say that Model A is *nested* in Model B. To determine if the simpler Model A is adequate for modelling the data, the hypothesis H_0: $\beta_{p_A+1} = \cdots = \beta_{p_B} = 0$ is to be tested.

Under H_0 (that is, Model A is sufficient for the data), denote the fitted values as $\hat{\mu}_A$, producing the log-likelihood $\ell_A = \ell_A(\hat{\mu}_1, \ldots, \hat{\mu}_n, \phi; y)$ and residual deviance $D(y, \hat{\mu}_A)$. For Model B, denoted the fitted values as $\hat{\mu}_B$, producing the log-likelihood $\ell_B = \ell_B(\hat{\mu}_1, \ldots, \hat{\mu}_n, \phi; y)$ and residual deviance of $D(y, \hat{\mu}_B)$.

We have previously observed that the total deviance function captures that part of the log-likelihood which depends on μ_i. So, if ϕ is known, the likelihood ratio test statistic for comparing Models A and B is

$$L = 2\{\ell_B - \ell_A\} = \frac{D(y, \hat{\mu}_A) - D(y, \hat{\mu}_B)}{\phi}. \tag{7.1}$$

The dispersion model form of the EDM (5.13) has been used here, and the terms $b(y, \phi/w_i)$ not involving μ_i cancel out. Standard asymptotic likelihood theory asserts that $L \sim \chi^2_{p'_B - p'_A}$ approximately under the null hypothesis if n is large relative to p'.

Likelihood ratio tests are traditionally used to test two-tailed alternative hypotheses. However, if Model B and Model A differ by only one coefficient, then we can define a signed likelihood ratio statistic to test a one-tailed alternative hypothesis about the true coefficient. Suppose that $p'_B - p'_A = 1$. We can define a z-statistic from the signed square-root of L as

$$Z = \text{sign}(\hat{\beta}_{p_B}) L^{1/2}.$$

Standard asymptotic likelihood theory asserts that $Z \sim N(0, 1)$ under the null hypothesis H_0: $\beta_{p_B} = 0$. The signed likelihood ratio test statistic can be used similarly to Wald test statistics.

Example 7.4. For the noisy miner data `nminer`, we can fit the model with just a constant term in the model, then the model with both a constant term and the number of eucalypts in the model:

```
> nm.m0 <- glm( Minerab ~ 1,    data=nminer, family=poisson)
> nm.m1 <- glm( Minerab ~ Eucs, data=nminer, family=poisson)
```

Then compute the residual deviance and residual degrees of freedom for each model:

```
> c( "Dev(m0)"= deviance( nm.m0 ),    "Dev(m1)" = deviance( nm.m1 ) )
  Dev(m0)   Dev(m1)
150.54532  63.31798
> c( "df(m0)" = df.residual( nm.m0 ), "df(m1)"  = df.residual( nm.m1 ) )
df(m0) df(m1)
    30     29
```

Since $\phi = 1$ for the Poisson distribution, use (7.1) to compare the two models:

```
> L <- deviance( nm.m0 ) - deviance( nm.m1 ); L
[1] 87.22735
> pchisq(L, df.residual(nm.m0) - df.residual(nm.m1), lower.tail=FALSE )
[1] 9.673697e-21
```

The P-value is very small, indicating that the addition of `Eucs` is significant. □

7.2.5 Analysis of Deviance Tables to Compare Nested Models

Often a *series* of nested models is compared. The initial model might contain no explanatory variables, then each explanatory variable might be added in turn. If successive pairs of models are compared using likelihood ratio tests, this amounts to computing differences in residual deviances for successive models. The computations can be organized into an *analysis of deviance* table (Table 7.1), which is a direct generalization of ANOVA tables for linear models (Sect. 2.10).

In R, the analysis of deviance table is produced using the `anova()` function. The argument `test="Chisq"` must be specified to obtain P-values for the deviances relative to χ^2 distributions on the appropriate degrees of freedom. If ϕ is not equal to the default value of one, the value of ϕ can be provided using the `dispersion` argument in the `anova()` call.

Example 7.5. For the noisy miner data `nminer`, and the models fitted in Example 7.4, produce the analysis of deviance table in R using:

Table 7.1 The analysis of deviance table for model `nm.m1` fitted to the noisy miner data (Sect. 7.2.5)

Source	Deviance	df	L	P-value
Due to `Eucs`	87.23	1	87.23	< 0.001
Residual	63.32	29		
Total	150.5	30		

```
> anova(nm.m1, test="Chisq")
     Df Deviance Resid. Df Resid. Dev  Pr(>Chi)
NULL                    30    150.545
Eucs  1   87.227         29     63.318 < 2.2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The residual deviances, and the difference between them, are the same as reported in Example 7.4. Notice that R also reports the residual deviance and residual degrees of freedom for each model in addition to the analysis of deviance information. □

7.2.6 Score Tests

Score tests may also be used to test hypotheses about single parameters or about sets of parameters. Whereas Wald and likelihood ratio tests are used to test hypotheses about explanatory variables in the current fitted model, score tests enable testing of hypotheses about explanatory variables not (yet) in the current model, but which might be added. Score tests play a strong role in GLM theory and practice because of their relationship to Pearson statistics.

Suppose we want to add a new predictor x_{p+1} to an existing GLM. Write $e(y)_i$ for the ith working residual (6.20) from the GLM. Similarly write $e(x_{p+1})_i$ for the ith residual from the least squares regression of x_{p+1} on the existing predictors with weights W_i. The score statistic for testing the null hypothesis H_0: $\beta_{p+1} = 0$ is

$$Z = \frac{\sum_{i=1}^{n} e(x_{p+1})_i e(y)_i}{\{\sum_{i=1}^{n} e(x_{p+1})_i^2\}^{1/2}}.$$

If H_0 is true, then $Z \sim N(0,1)$ approximately. In R, score test statistics for individual predictors are computed using the function `glm.scoretest()` in package **statmod**.

Example 7.6. For the noisy miner data `nminer`, we conduct a score test to determine if `Eucs` should be added to the null model using `glm.scoretest()`:

```
> library(statmod)  # Provides glm.scoretest
> nm.m0 <- glm( Minerab ~ 1, data=nminer, family=poisson)
> z.stat <- glm.scoretest(nm.m0, nminer$Eucs)
> p.val <- 2 * pnorm( abs(z.stat), lower.tail=FALSE)
> round( c(score.stat=z.stat, P=p.val), 4)
score.stat          P
    9.7565     0.0000
```

The evidence strongly suggests that `Eucs` should be added to the model. □

Example 7.7. The well-known Pearson chi-square test of independence in a contingency table is an example of a score test. To illustrate this, we can construct a small example:

```
> Y <- matrix(c(10,20,20,10),2,2)
> rownames(Y) <- c("A1","A2")
> colnames(Y) <- c("B1","B2")
> Y
   B1 B2
A1 10 20
A2 20 10
```

The Pearson test P-value is:

```
> chisq.test(Y, correct=FALSE)$p.value
[1] 0.009823275
```

The same P-value can be obtained from a Poisson log-linear regression and a score test for interaction:

```
> y <- as.vector(Y)
> A <- factor(c(1,2,1,2))
> B <- factor(c(1,1,2,2))
> fit <- glm(y~A+B, family=poisson)
> z.stat <- glm.scoretest(fit, x2=c(0,0,0,1))
> 2 * pnorm( -abs(z.stat) )
[1] 0.009823231
```

□

* 7.2.7 Score Tests Using Matrices

Suppose we wish to consider adding a set of k new explanatory variables to the current GLM. Write X_2 for the matrix with the new explanatory variables as columns, and write E_2 for the matrix of residuals after least squares regression of the columns of X_2 on the predictors already in the GLM; that is,

$$\mathrm{E}_2 = \mathrm{X}_2 - \mathrm{X}\left(\mathrm{X}^T\mathrm{W}\mathrm{X}\right)^{-1}\mathrm{X}^T\mathrm{W}\mathrm{X}_2$$

where X is the model matrix and W is the diagonal matrix of working weights from the current fitted model. Although this might seem an elaborate expression, E_2 can be computed very quickly and easily using the information stored in the `glm()` fit object in R. If X_2 is a single column, then the Z score test statistic is

$$Z = \frac{\mathrm{E}_2^T\mathrm{W}\mathbf{e}}{\left(\mathrm{E}_2^T\mathrm{W}\mathrm{E}_2\right)^{1/2}}$$

where $\mathbf{e}$ is the vector of working residuals from the current fitted model. If E_2 is a matrix, then the chi-square score test statistic is

$$X^2 = \mathbf{e}^T \mathrm{W}\mathrm{E}_2 \left(\mathrm{E}_2^T \mathrm{W}\mathrm{E}_2\right)^{-1} \mathrm{E}_2 \mathrm{W}\mathbf{e}.$$

Under the null hypothesis, that none of the new covariates are useful explanatory variables, $X^2 \sim \chi^2_k$ approximately.

In R, score test statistics for a set of predictors are computed using the function `glm.scoretest()` in package **statmod**.

7.3 Large Sample Asymptotics

All the distributional results for the test statistics given in this chapter so far are standard asymptotic results from likelihood theory (Sect. 4.10). The distributions should be good approximations when the number of observations n is reasonably large. We call these results *large sample asymptotics*.

It is hard to give a guideline for how large n needs to be before we should be confident that the asymptotics hold, but, on the whole, the results tend to hold well for score tests and likelihood ratio tests even for moderate sized samples. Wald tests, especially for binomial EDMs with small m, tend to need larger samples to be reliable. For Wald tests, the asymptotic results tend to be conservative, in that small samples generally result in large standard errors and non-significant Wald test statistics. When the sample size is large enough for the standard errors $\mathrm{se}(\hat{\beta}_j)$ to be small, then the asymptotics should be reasonably accurate.

As usual, everything is exact for normal linear GLMs.

Example 7.8. Consider a small regression with binary data:

```
> y <- c(0, 0, 0, 1, 0, 1, 1, 1, 1)
> x <- 1:9
> fit <- glm(y~x, family=binomial)
```

An exact permutation P-value can be obtained for this data using a Mann-Whitney (or Wilcoxon) rank-sum test, without using any asymptotic assumptions. This shows there is good evidence for a trend in the data:

```
> wilcox.test(x ~ y)$p.value
[1] 0.03174603
```

The Wald z-test proves to be conservative, failing to detect the trend:

```
> coef(summary(fit))
             Estimate Std. Error   z value  Pr(>|z|)
(Intercept) -5.811289  4.0019503 -1.452114 0.1464699
x            1.292257  0.8497008  1.520838 0.1283006
```

The likelihood ratio test possibly over-states the statistical significance:

```
> as.data.frame(anova(fit, test="Chisq")[2,])
  Df Deviance Resid. Df Resid. Dev    Pr(>Chi)
x  1 7.353132         7   5.012176 0.006694603
```

The score test seems about right:

```
> fit <- glm(y~1, family=binomial)
> 2 * pnorm(-abs(glm.scoretest(fit, x)))
[1] 0.01937237
```

□

7.4 Goodness-of-Fit Tests with ϕ Known

7.4.1 The Idea of Goodness-of-Fit

This chapter has so far examined tests of whether particular explanatory variables should be retained or added to the current model. One would often like to ask: how many explanatory variables are sufficient? When can we stop testing for new explanatory variables? Goodness-of-fit tests determine whether the current linear predictor already includes enough explanatory variables to fully describe the systematic trends in the data. In that case, no more explanatory variables are useful or necessary. This sort of test is only possible when ϕ is known, because it requires a known distribution for the residual variability.

A goodness-of-fit test compares the current model (Model A say) with an alternative model (Model B) of a particular type. In this case, Model B is the largest possible model which can, in principle, be fitted to the data. This model has as many explanatory variables as data points, so that $p' = n$, and is known as the *saturated model*. Under the saturated model, the fitted values are all equal to the data values: $\hat{\mu}_i = y_i$. This is generally true, regardless of the specific explanatory variables in the saturated model, as long at there are p' linearly independent predictors, so we talk of *the* saturated model rather than *a* saturated model. The test is on $n - p'$ degrees of freedom, because the saturated model has n parameters compared to the current model with p'.

If the goodness-of-fit test is rejected, then this is evidence that the current model is not adequate. By "not adequate" we mean that the systematic component does not explain everything that can be explained, so there must be other important explanatory variables which are missing from our model.

7.4.2 Deviance Goodness-of-Fit Test

The residual deviance for the saturated model is zero, so the likelihood ratio test statistic of the current model versus the saturated model turns out to be simply the residual deviance $D(y, \hat{\mu})$ of the current model.

Following the usual results for likelihood ratio tests, it is tempting to treat the residual deviance as chi-square on $n - p'$ degrees of freedom. However, the usual large-sample asymptotics do not hold here, because the number of parameters in the saturated model increases with the number of observations. Instead, appealing to the saddlepoint approximation is necessary, which we do in Sect. 7.5.

Example 7.9. The well-known *G-test* for independence in a two-way contingency table is a deviance goodness-of-fit statistic. □

7.4.3 Pearson Goodness-of-Fit Test

The (chi-square) score test statistic of the current model versus the saturated model turns out to be the Pearson statistic X^2. Following the usual results for score tests, it is tempting to treat the Pearson statistic as chi-square on $n - p'$ degrees of freedom, but the usual large-sample asymptotics do not hold, for the same reason as for the residual deviance. Instead appealing to the Central Limit Theorem is necessary, which we do in Sect. 7.5.

Example 7.10. The well-known *Pearson chi-square test* for independence in a two-way contingency table is a Pearson goodness-of-fit statistic. □

Example 7.11. In modern molecular genetics research, it is common to study transgenic mice which have mutations in a specified gene but which are otherwise identical to normal mice. In a study at the Walter and Eliza Hall Institute of Medical Research (Melbourne), a number of heterozygote mice (having one normal allele A and one mutant allele a for the gene of interest) were mated together. Simple Mendelian inheritance would imply that the AA (normal), Aa (heterozygote mutant) and aa (homozygote mutant) genotypes should occur in the offspring in the proportions 1/4, 1/2 and 1/4 respectively. A particular experiment gave rise to the numbers of offspring given in Table 7.2.

Are these numbers compatible with Mendelian inheritance? We answer this question by fitting a Poisson GLM for which the fitted values are in the Mendelian proportions:

```
> y <- c(15, 26, 4); x <- c(1/4, 1/2, 1/4)
> fit <- glm( y ~ 0+x, family=poisson)
```

Table 7.2 The number of offspring mice of each genotype from matings between *Aa* heterozygote parents (Example 7.11)

AA	*Aa*	*aa*
15	26	4

Note the `0` to omit the intercept from the linear predictor. Then compute goodness-of-fit tests:

```
> pearson.gof <- sum(fit$weights * fit$residuals^2)
> tab <- data.frame(GoF.Statistic=c(fit$deviance, pearson.gof))
> tab$DF <- rep(fit$df.residual, 2)
> tab$P.Value <- pchisq(tab$GoF, df=tab$DF, lower.tail=FALSE)
> row.names(tab) <- c("Deviance", "Pearson"); print(tab, digits=3)
         GoF.Statistic DF P.Value
Deviance          12.2  2 0.00227
Pearson           17.5  2 0.00016
```

Both the deviance and Pearson goodness-of-fit tests reject the null hypothesis that the model is adequate. The proportion of *aa* mutants appears to be too low. One explanation is that the mutation is harmful so that homozygote mutants tend to die before birth. □

7.5 Small Dispersion Asymptotics

The large sample asymptotics considered earlier are not sufficient for goodness-of-fit tests to be valid. For goodness-of-fit tests, we require distributional results to hold reasonably well for individual observations. Therefore, here we consider results which hold when the precision of individual observations becomes large. We call these results *small dispersion asymptotics*.

The work-horses of small dispersion asymptotics are the saddlepoint approximation (for results about the deviance statistics), and the Central Limit Theorem (for results about Pearson statistics).

The accuracy of the saddlepoint approximation has been previously discussed (Sect. 5.4.4). We noted that the accuracy of the saddlepoint approximation to a probability function depended only on y, not μ, for a given EDM. The criterion $\tau \leq 1/3$ (see Sect. 5.23, p. 225) was given to ensure a good approximation (where $\tau = \phi V(y)/(y - \text{boundary})^2$). We noted in Sect. 5.4.5 that limits did need to placed on μ for the chi-square distributional approximation to hold well for the unit deviance. For a fitted GLM, we can cover both of these conditions by requiring that the criterion $\tau \leq 1/3$ is satisfied for all y_i, $i = 1, \ldots, n$ [9]. As a guideline, this generally ensures that both the responses y_i and the fitted values $\hat{\mu}_i$ are in the required range for the approximation to hold.

The Central Limit Theorem has a slower convergence rate than the saddlepoint approximation ($O(\phi^{1/2})$ instead of $O(\phi)$), so we apply a slightly stricter criterion, that $\tau \leq 1/5$ for all observations.

The Pearson statistic (Sect. 6.8.5, p. 255) has approximately a chi-square distribution

$$\frac{X^2}{\phi} \sim \chi^2_{n-p'},$$

when the Central Limit Theorem holds for individual observations. However, the Pearson estimator of ϕ should remain approximately unbiased even for smaller τ, at least in large sample situations.

The residual deviance has approximately a chi-square distribution

$$\frac{D(y, \hat{\mu})}{\phi} \sim \chi^2_{n-p'},$$

when the saddlepoint approximation holds. This criterion ensures that the mean-deviance estimator of ϕ is approximately unbiased. The distributional approximation is likely to be better for the deviance than for the Pearson statistic for moderate values of ϕ. For very small values of ϕ, the deviance and Pearson statistics are almost identical.

The guidelines translate into the following rules for common EDMs. The saddlepoint approximation is sufficiently accurate when

- Binomial: $\min\{m_i y_i\} \geq 3$ and $\min\{m_i(1 - y_i)\} \geq 3$;
- Poisson: $\min\{y_i\} \geq 3$;
- Gamma: $\phi \leq 1/3$.

Recall that saddlepoint approximation is exact for normal and inverse Gaussian GLMs.

The Central Limit Theorem is sufficiently accurate for individual observations when

- Binomial: $\min\{m_i y_i\} \geq 5$ and $\min\{m_i(1 - y_i)\} \geq 5$;
- Poisson: $\min\{y_i\} \geq 5$;
- Gamma: $\phi \leq 1/5$.

Of course, residual deviance and Pearson statistic have exact chi-square distributions for normal linear regression models.

These conditions should be sufficient to ensure that the chi-square distribution approximations for the residual deviance or Pearson statistics are sufficiently accurate for routine use. The chi-square approximations might continue to be good enough for practical use when the criteria are not satisfied, depending on the number of observations for which the criteria fail. Examination of the specifics of each data situation is recommended in these cases.

Example 7.12. In Example 7.11, the mouse offspring counts are Poisson with $\min\{y_i\} = 4$. The saddlepoint approximation guideline is satisfied, but that

for the Central Limit Theorem is not quite, so the deviance goodness-of-fit test is more reliable than the Pearson test in this case. □

Example 7.13. The noisy miner data (Example 6.5, p. 249) contains several zero counts, so small dispersion asymptotics do not apply for a Poisson EDM. Neither the deviance nor Pearson goodness-of-fit tests are reliable for these data. □

7.6 Inference for Coefficients When ϕ Is Unknown

7.6.1 Wald Tests for Single Regression Coefficients

When ϕ is unknown, Wald tests are similar to the case with ϕ known (Sect. 7.2.1) except that an estimator of ϕ must be used to compute the standard errors. The Wald statistic to test the null hypothesis H_0: $\beta_j = \beta_j^0$ becomes

$$T = \frac{\hat{\beta}_j - \beta_j^0}{\text{se}(\hat{\beta}_j)},$$

where now the standard error $\text{se}(\hat{\beta}_j) = sv_j$ involves a suitable estimator s^2 of ϕ (6.13). The Pearson estimator $s^2 = \bar{\phi}$ is used by R.

If a consistent estimator of ϕ is used, and the sample size is very large, the estimate of ϕ will be close to the true value and T will be roughly standard normal under the null hypothesis. In small or moderate sized samples, a better approximation is to treat T as following a t-distribution with $n-p'$ degrees of freedom. The result for normal linear regression, in which T-statistics follow t-distributions exactly, is a special case.

In R, using the `summary()` command shows that the values of Z (or T if ϕ is unknown), $\text{se}(\hat{\beta}_j)$ and the two-tailed P-values for testing H_0: $\beta_j = 0$ for each fitted regression coefficient. If ϕ is known, the Wald statistic is labelled `z` and the P-values are computed by referring to a $N(0,1)$ distribution. If ϕ is estimated (by $\bar{\phi}$), the Wald statistic is labelled `t` and the two-tailed P-values are computed by referring to a $t_{n-p'}$ distribution. Other estimators of ϕ may be used, as shown in Example 7.14, but beware that the dispersion will then be treated as known.

Example 7.14. Consider the cherry tree data from Example 3.14 (data set: `trees`) for modelling the volume y in cubic feet of $n = 31$ cherry trees. The model fitted in that example can be summarized using:

```
> data(trees)
> tr.m2 <- glm( Volume  ~ log(Girth) + log(Height),
               family=Gamma(link="log"), data=trees  )
> printCoefmat(coef(summary(tr.m2)))
```

```
            Estimate Std. Error t value  Pr(>|t|)
(Intercept) -6.69111    0.78784 -8.4929 3.108e-09 ***
log(Girth)   1.98041    0.07389 26.8021 < 2.2e-16 ***
log(Height)  1.13288    0.20138  5.6255 5.037e-06 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The `summary()` shows that the regression coefficients for `log(Girth)` and `log(Height)` are non-zero, in the presence of each other. Since the dispersion ϕ is very small, the Pearson and mean deviance estimators of ϕ are very similar:

```
> phi.meandev <- deviance(tr.m2) / df.residual(tr.m2)
> phi.pearson <- summary(tr.m2)$dispersion
> c(Mean.deviance=phi.meandev, Pearson=phi.pearson)
Mean.deviance       Pearson
  0.006554117   0.006427286
```

R uses the Pearson estimator. To use the mean deviance estimator of ϕ to compute the Wald statistics, use:

```
> printCoefmat(coef(summary(tr.m2, dispersion=phi.meandev)))
             Estimate Std. Error z value  Pr(>|z|)
(Intercept) -6.691109   0.795578 -8.4104 < 2.2e-16 ***
log(Girth)   1.980412   0.074616 26.5415 < 2.2e-16 ***
log(Height)  1.132878   0.203361  5.5708 2.536e-08 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Note though that R has now conducted z-tests using a normal distribution instead of t-tests, treating the dispersion as known, meaning that the significance of the tests is now slightly over-stated.

The R output above tests $\beta_j = 0$. However, different hypotheses may be more interesting for these data. For example, the theoretical models developed in Example 3.14 are based on approximating the shape of the cherry trees as cones or cylinders. Hypotheses of interest may be H_0: $\beta_0 = \log(\pi/1728)$ (suggesting a conical shape) and H_0: $\beta_0 = \log(\pi/576)$ (suggesting a cylindrical shape). While these tests are not performed automatically by R, the Wald test computations are easily completed:

```
> beta0.hat <- coef(summary(tr.m2))[1,"Estimate"]
> beta0.se <-  coef(summary(tr.m2))[1,"Std. Error"]
>    #
>    # Test beta_0 = log(pi/1728) (for a cone)
> beta0.cone <- log( pi/1728 )
> t1 <- ( beta0.hat - beta0.cone ) / beta0.se
>    # Test beta_0 =  log(pi/576) (for a cylinder)
> beta0.cylinder <- log( pi/576 )
> t2 <- ( beta0.hat - beta0.cylinder ) / beta0.se
>    #
>    # Compute P-values
> p1 <- 2 * pt( -abs(t1), df=df.residual(tr.m2) )
```

```
> p2 <- 2 * pt( -abs(t2), df=df.residual(tr.m2) )
> tab <- array( c(t1, t2, p1, p2), dim=c(2, 2))
> rownames(tab) <- c("Cone:","Cylinder:")
> colnames(tab) <- c("t-scores","P-values"); tab
           t-scores   P-values
Cone:     -0.483750 0.63232520
Cylinder: -1.878206 0.07080348
```

No strong evidence exists to reject either hypothesis, though the fit of the cylindrical model is less good than that of the conic. □

7.6.2 Confidence Intervals for Individual Coefficients

When ϕ is unknown, Wald confidence intervals are similar to the case with ϕ known (Sect. 7.2.2) except that an estimator of ϕ must be used to compute the standard errors. The $100(1-\alpha)\%$ Wald confidence interval for β_j is

$$\hat{\beta}_j \pm t^*_{\alpha/2,n-p'} \text{se}(\hat{\beta}_j),$$

where $t^*_{\alpha/2,n-p'}$ is the value of t such that an area $\alpha/2$ is in each tail of the t-distribution with $n-p'$ degrees of freedom. The results apply in the large-sample case, and when the saddlepoint approximation is satisfactory. The R function `confint()` computes Wald confidence intervals from fitted `glm()` objects. Again, the result for ϕ unknown is based on t-statistics (using the Pearson estimate of ϕ) so that the results for the special case of the normal linear regression models are exact. Other estimates of ϕ can be use by setting the `dispersion` input in the `confint()` call.

Example 7.15. For the cherry tree data `trees` (Example 7.14, p. 278), the Wald confidence intervals for the regression coefficients are found as follows:

```
> confint(tr.m2)
                 2.5 %    97.5 %
(Intercept) -8.2358004 -5.139294
log(Girth)   1.8359439  2.124974
log(Height)  0.7364235  1.528266
```

The theoretical development in Example 3.14 (p. 125) suggest $\beta_1 \approx 2$ and $\beta_2 \approx 1$. The confidence intervals show that the estimate for β_1 is reasonably precise, and contains the value $\beta_1 = 2$; the confidence interval for β_2 is less precise, but contains the value $\beta_2 = 1$. Furthermore, from Example 3.14, the values $\beta_0 = \log(\pi/1728) = -6.310$ (for a cone) and $\beta_0 = \log(\pi/576) = -5.211$ (for a cylinder) both lie within the 95% confidence interval for β_0. □

* 7.6.3 Confidence Intervals for μ

When ϕ is unknown, confidence intervals for the fitted values $\hat{\mu}$ are similar to the case with ϕ known (Sect. 7.2.3) except that an estimator of ϕ must be used to compute the standard errors. We initially work with $\hat{\eta} = g(\hat{\mu})$, for which $\widehat{\text{var}}[\hat{\eta}]$ is easily found (Sect. 6.6). Then, when ϕ is unknown and an estimate is used, a $100(1-\alpha)\%$ Wald confidence interval for η is

$$\hat{\eta} \pm t^*_{\alpha/2,n-p'} \text{se}(\hat{\eta}),$$

where $\text{se}(\hat{\eta}) = \sqrt{\widehat{\text{var}}[\hat{\eta}]}$, and where $t^*_{\alpha/2,n-p'}$ is the value such that an area $\alpha/2$ is in each tail of the t-distribution with $n-p'$ degrees of freedom. The confidence interval for μ is found applying the inverse link function (that is, $\mu = g^{-1}(\eta)$) to the lower and upper limit of the interval found for $\hat{\eta}$. Rather than explicitly returning the confidence interval, R optionally returns the standard errors when making prediction using `predict()` with the input `se.fit=TRUE`. This information can be used to form confidence intervals. Note that `predict()` returns the value of $\hat{\eta}$ by default. The fitted values (and standard errors) are returned by specifying `type="response"`. The confidence interval is necessarily symmetric on the η scale.

Example 7.16. For the trees data `trees`, suppose we wish to estimate the mean volume of trees with height 70 ft and girth 15 in. First, we compute the predictions and standard errors on the scale of the linear predictor:

```
> out <- predict( tr.m2, newdata=data.frame(Height=70, Girth=15),
                  se.fit=TRUE)
```

Then we form the confidence interval for μ by using the inverse of the logarithmic link function:

```
> tstar <- qt(p=0.975, df=df.residual(tr.m2)) # For 95% CI
> ci.lo <- exp(out$fit - tstar*out$se.fit)
> ci.hi <- exp(out$fit + tstar*out$se.fit)
> c( Lower=ci.lo, Estimate=exp(out$fit), Upper=ci.hi)
   Lower.1 Estimate.1    Upper.1
  30.81902   32.62157   34.52955
```

We see that $\hat{\mu} = 32.62$, and that the 95% confidence interval is from 30.82 to 34.53.

This idea can be extended to compute the confidence intervals for the mean volume of all trees with varying height and girth 15 in:

```
> newHt <- seq(min(trees$Height), max(trees$Height), by=4)
> newVol <- predict( tr.m2, se.fit=TRUE,
                     newdata=data.frame(Height=newHt, Girth=15))
> ci.lo <- exp(newVol$fit-tstar*newVol$se.fit)
> ci.hi <- exp(newVol$fit+tstar*newVol$se.fit)
> cbind( newHt, ci.lo, Vol=exp(newVol$fit), ci.hi, width=ci.hi - ci.lo)
```

```
  newHt    ci.lo      Vol    ci.hi    width
1    63 26.33168 28.95124 31.83141 5.499733
2    67 28.88896 31.04230 33.35614 4.467187
3    71 31.45834 33.15002 34.93267 3.474330
4    75 33.93192 35.27358 36.66829 2.736366
5    79 36.10127 37.41225 38.77084 2.669571
6    83 37.87594 39.56537 41.33016 3.454225
7    87 39.40973 41.73232 44.19180 4.782065
```

□

7.6.4 Likelihood Ratio Tests to Compare Nested Models: F-Tests

In Sect. 7.2.4 (p. 269), likelihood ratio tests were developed for comparing nested models when ϕ is known. If ϕ is unknown, an estimate of ϕ must be used. With ϕ unknown, the appropriate statistic for comparing Model A (with fitted values $\hat{\mu}_A$) which is nested in Model B (with fitted values $\hat{\mu}_B$) is

$$F = \frac{\{D(y, \hat{\mu}_A) - D(y, \hat{\mu}_B)\} / (p'_B - p'_A)}{s^2}, \tag{7.2}$$

where the models have p'_A and p'_B parameters respectively, and s^2 is some suitable estimate of ϕ based on Model B. This is analogous to the linear regression model case (2.30) (p. 63). Estimators of ϕ considered in Sect. 6.8 include the modified profile likelihood estimator $\hat{\phi}^0$, the Pearson estimator $\bar{\phi}$, and the mean deviance estimator $\tilde{\phi}$. The corresponding F-statistics based on using the three estimators of ϕ may be written

$$\hat{F}^0 = \frac{\{D(y, \hat{\mu}_A) - D(y, \hat{\mu}_B)\} / (p'_B - p'_A)}{\hat{\phi}^0_B} \tag{7.3}$$

$$\bar{F} = \frac{\{D(y, \hat{\mu}_A) - D(y, \hat{\mu}_B)\} / (p'_B - p'_A)}{\bar{\phi}_B} \tag{7.4}$$

$$\tilde{F} = \frac{\{D(y, \hat{\mu}_A) - D(y, \hat{\mu}_B)\} / (p'_B - p'_A)}{\tilde{\phi}_B}, \tag{7.5}$$

where all estimates of ϕ are based on Model B.

As usual, all three F-statistics are identical for linear regression models and, in that case, the statistic follows exactly an F-distribution with $(p'_B - p'_A, n - p'_B)$ degrees of freedom under the null hypothesis that the two models A and B are equal. For other GLMs, the F-statistics are approximately F-distributed under the null hypothesis. The approximation is likely to be good whenever the denominator of the F-statistic follows a scaled chi-square distribution, and the conditions for this are discussed in Sect. 7.5. Empirically, however, the F-distribution approximation for the F-statistic is often

more accurate than the chi-square approximation to the denominator. For this reason, the F-test based on the F-statistics tends to be serviceable in a wide variety of situations.

The choice between the three F-statistics mirrors the choice between the three estimators discussed in Sect. 6.8.6. $\hat{F}^0$ can be expected to have the best properties but is inconvenient to compute. $\bar{F}$ will follow an F-distribution accurately under the null hypothesis when the saddlepoint approximation applies (small dispersion asymptotics). In other situations, $\tilde{F}$ is likely to be the less biased than $\bar{F}$ and is therefore the default statistic used by the GLM functions in R.

Although F-tests are usually used for two-tailed tests, if Model B and Model A differ by only one coefficient, then we can define a signed statistic to test a one-tailed alternative hypothesis about the value of the true coefficient. Suppose that $p'_B - p'_A = 1$. We can define a t-statistic from the signed square-root of F as

$$t = \text{sign}(\hat{\beta}_{p_B}) F^{1/2}.$$

Then $t \sim t_{n-p'_B}$ approximately under the null hypothesis H_0: $\beta_{p_B} = 0$.

Example 7.17. For a normal GLM, the residual deviance is the RSS (Sect. 6.4, p. 248). The F-statistic for comparing two nested models is

$$F = \frac{(\text{RSS}_A - \text{RSS}_B)/(p'_B - p'_A)}{s^2},$$

which is the usual F-statistic familiar from ANOVA in the linear regression model case (2.30). □

Example 7.18. Consider the cherry tree data `trees` and model `tr.m2` fitted in Example 7.14. Fit the two explanatory variables `log(Girth)` and `log(Height)` sequentially, and record the residual deviance and residual degrees of freedom for each model:

```
> data(trees)
> tr.m0 <- glm( Volume ~ 1, family=Gamma(link="log"), data=trees)
> tr.m1 <- update(tr.m0, . ~ . + log(Girth) )
> tr.m2 <- update(tr.m1, . ~ . + log(Height) )
> c( deviance(tr.m0), deviance(tr.m1), deviance(tr.m2) )
[1] 8.3172012 0.3840839 0.1835153
> c( df.residual(tr.m0), df.residual(tr.m1), df.residual(tr.m2) )
[1] 30 29 28
```

Then compute the deviances between the models by computing the corresponding changes in the residual deviance (and also compute the residual degrees of freedom):

```
> dev1 <- deviance(tr.m0) - deviance(tr.m1)
> dev2 <- deviance(tr.m1) - deviance(tr.m2)
> df1 <- df.residual(tr.m0) - df.residual(tr.m1)
> df2 <- df.residual(tr.m1) - df.residual(tr.m2)
> c( dev1, dev2)
```

```
[1] 7.9331173 0.2005686
> c( df1, df2)
[1] 1 1
```

To compute the F-test statistics as shown in (7.3)–(7.5), first an estimate of ϕ is needed:

```
> phi.meandev <- deviance(tr.m2) / df.residual(tr.m2)   # Mean dev.
> phi.Pearson <- summary(tr.m2)$dispersion              # Pearson
> c("Mean deviance" = phi.meandev, "Pearson" = phi.Pearson )
Mean deviance       Pearson
  0.006554117   0.006427286
```

The Pearson and mean deviance estimates are very similar. Likewise, the F-statistics and corresponding P-values computed using these two estimates are similar:

```
> F.Pearson <- c( dev1/df1, dev2/df2 ) / phi.Pearson
> F.meandev <- c( dev1/df1, dev2/df2 ) / phi.meandev
> P.Pearson <- pf( F.Pearson, df1, df.residual(tr.m2), lower.tail=FALSE )
> P.meandev <- pf( F.meandev, df2, df.residual(tr.m2), lower.tail=FALSE )
> tab <- data.frame(F.Pearson, P.Pearson, F.meandev, P.meandev)
> rownames(tab) <- c("Girth","Height")
> print(tab, digits=3)
       F.Pearson P.Pearson F.meandev P.meandev
Girth     1234.3  1.05e-24    1210.4  1.38e-24
Height      31.2  5.60e-06      30.6  6.50e-06
```

These results show that `log(Girth)` is significant in the model, and that `log(Height)` is significant in the model after adjusting for `log(Girth)`. □

7.6.5 Analysis of Deviance Tables to Compare Nested Models

When a series of GLMs is to be compared, the computations discussed in Sect. 7.6.4 are often arranged in an analysis of deviance table (similar to the case when ϕ is known; Sect. 7.2.5). A series of nested models is fitted to the data, and the residual deviance and residual degrees of freedom for each model recorded. The *changes* in the residual deviance and residual degrees of freedom are then compiled into the *analysis of deviance table*. In R, the analysis of deviance table is produced by the `anova()` function. The argument `test="F"` must be specified to obtain P-values for deviance differences relative to F distributions on the appropriate degrees of freedom. In R, the F-statistics are computed using the Pearson estimator $\bar{\phi}$ by default when computing the ANOVA table (the reasons for this choice in R are given in Sect. 6.8.6). Other estimates of ϕ can be provided using the `dispersion` argument in the `anova()` call.

Table 7.3 The analysis of deviance table for model `tr.m2` fitted to the cherry tree data, writing x_1 for `log(Girth)` and x_2 for `log(Height)` for brevity (Example 7.18)

Source	Deviance	Change in df	Mean deviance	F	P-value
Due to x_1	7.933	1	7.933	1234	< 0.001
Due to x_2, adjusted for x_1	0.2006	1	0.2006	31.21	< 0.001
Residual	0.1835	28			
Total	8.317	30			

Example 7.19. For the `trees` data, the information computed in Example 7.18 is usually compiled into an analysis of deviance table (Table 7.3).

Observe that the mean deviance estimator of ϕ is easy to compute from the analysis of deviance table ($\tilde{\phi} = 0.1835/28 = 0.006554$), but the Pearson estimator is used by R. The analysis of deviance table produced by R is:

```
> anova(tr.m2, test="F")
           Df Deviance Resid. Df Resid. Dev        F    Pr(>F)
NULL                         30     8.3172
log(Girth)   1   7.9331        29     0.3841 1234.287 < 2.2e-16 ***
log(Height)  1   0.2006        28     0.1835   31.206 5.604e-06 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Notice that R also reports the residual deviance and residual degrees of freedom for each model in addition to the analysis of deviance information. To base the test on the mean deviance estimator, use the `dispersion` argument:

```
> phi.meandev <- deviance( tr.m2) / df.residual(tr.m2)
> anova(tr.m2, test="F", dispersion=phi.meandev)
           Df Deviance Resid. Df Resid. Dev        F    Pr(>F)
NULL                         30     8.3172
log(Girth)   1   7.9331        29     0.3841 1210.402 < 2.2e-16 ***
log(Height)  1   0.2006        28     0.1835   30.602 3.168e-08 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The results are very similar for either estimate of ϕ. □

The *order* of fitting terms into a model is important when interpreting the results from the analysis of deviance tables. The order in which terms are added to the model may affect whether or not they are statistically significant. This means that the actual effect of any one variable can only be stated conditionally on other variables in the model, which impacts on the interpretation of the effects.

Example 7.20. Consider fitting `log(Girth)` and `log(Height)` in reverse order to that of `tr.m2`:

```
> tr.rev <- glm( Volume ~ log(Height) + log(Girth),
                   family=Gamma(link="log"), data=trees)
> anova(tr.rev, test="F")
            Df Deviance Resid. Df Resid. Dev      F    Pr(>F)
NULL                           30     8.3172
log(Height)  1   3.5345        29     4.7827 549.92 < 2.2e-16 ***
log(Girth)   1   4.5992        28     0.1835 715.57 < 2.2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Here, the conclusions are the same when compared to model `tr.m2` (the evidence strongly suggests both regression coefficients are non-zero) but the F-statistics and the corresponding P-values are not the same. □

7.6.6 Score Tests

Strictly speaking, score tests assume that ϕ is known, but they can be used in an approximate sense when ϕ is unknown simply by substituting an estimate for ϕ. By default, the `glm.scoretest()` function (in package **statmod**) uses the Pearson estimator for ϕ. Other estimates of ϕ can be used by using the `dispersion` argument in the call to `glm.scoretest()`. As with Wald tests, we treat the score test statistics as approximately t-distributed instead of normal when ϕ is unknown. The score statistic is approximately $t_{n-p'}$ distributed under the null hypothesis when an estimator ϕ is used.

Example 7.21. (Data set: `trees`) Consider the cherry tree data again. The score test can be used to test if `log(Girth)` and `log(Height)` are useful in the model, using the function `glm.scoretest()` in R package **statmod**. First consider `log(Height)`, conditional on `log(Girth)` appearing in the model:

```
> library(statmod)
> mA <- glm( Volume ~ log(Girth), family=Gamma(link="log"), data=trees )
> t.Ht <- glm.scoretest( mA, log(trees$Height) )
> p.Ht <- 2 * pt( -abs(t.Ht), df=df.residual(mA) ) # Two-tailed P-value
> tab <- data.frame(Score.stat = t.Ht, P.Value=p.Ht )
> print(tab, digits=3)
  Score.stat P.Value
1       3.83 0.00063
```

Then consider `log(Girth)`, conditional on `log(Height)` appearing in the model:

```
> mB <- glm( Volume ~ log(Height), family=Gamma(link="log"), data=trees)
> t.Girth<- glm.scoretest( mB, log(trees$Girth) )
> p.Girth <- 2 * pt( -abs(t.Girth), df=df.residual(mB) )
> tab <- data.frame(Score.stat = t.Girth, P.Value=p.Girth )
> print(tab, digits=3)
  Score.stat  P.Value
1       5.22 1.36e-05
```

The test statistics and two-tailed P-values are somewhat more conservative than the corresponding Wald test results shown previously (Example 7.14, p. 278). The conservatism can be partly attributed to fact that the score tests use dispersion estimates from the null models with one explanatory variable instead of from the full model with both explanatory variables. Nevertheless, the conclusions are the same. The score tests strongly support adding `log(Girth)` to the model in the presence of `log(Height)`, and also support adding `log(Height)` to the model in the presence of `log(Girth)`. We conclude that both explanatory variables are needed. □

7.7 Comparing Wald, Score and Likelihood Ratio Tests

The most common tests used in practice with GLMs are Wald tests for individual coefficients and the likelihood ratio tests for comparing nested models. Wald tests are easily understood because they simply relate the coefficient estimates to their standard errors and, for this reason, they are routinely presented as part of the summary output for a GLM fit in R. Likelihood ratio tests correspond to deviance differences and can be computed using the `anova()` function in R. Score tests are much less often used, except in their incarnation as Pearson goodness-of-fit statistics. Score tests deserve perhaps to be more used than they are—they are a good choice when testing whether new explanatory variables should be added to the current model.

For normal linear regression models, Wald, score and likelihood ratio statistics all enjoy exact null distributions regardless of sample size. For GLMs, the test statistics have *approximate* distributions, as discussed in the previous sections. In general, the distributional approximations for likelihood ratio tests and score tests tend to be somewhat better than those for Wald tests. This is particularly true for binomial or Poisson GLMs when fitted values occur on or near the boundary of the range of possible values (for example an exact zero fitted mean for a Poisson GLM or fitted proportions exactly zero or one for a binomial GLM). Wald tests are unsuitable in this situation because some or all of the estimated coefficients become infinitely large (as will be discussed in Sect. 9.9), yet likelihood ratio tests remain reasonably accurate.

Wald tests and score tests can be used to test either one-tailed or two-tailed tests for single regression coefficients. Likelihood ratio tests are traditionally used only for two-sided hypotheses. Nevetheless they too can be used to test one-tailed hypotheses for single coefficients via signed likelihood ratio statistics.

7.8 Choosing Between Non-nested GLMs: AIC and BIC

The hypothesis tests discussed in Sects. 7.2.4 and 7.6.4 only apply when the models being compared are nested models. However, sometimes a researcher wishes to compare non-nested models. As with linear regression, the AIC and BIC may be used to compare non-nested models, though using the AIC or BIC does not constitute a formal testing procedure.

Using definitions (4.34) and (4.35) (p. 202), the AIC and BIC for a GLM with n observations, p' regression parameters and known ϕ are

$$\begin{aligned} \text{AIC} &= -2 \times \ell(\hat{\beta}_0, \ldots, \hat{\beta}_p, \phi; y) + 2p' \\ \text{BIC} &= -2 \times \ell(\hat{\beta}_0, \ldots, \hat{\beta}_p, \phi; y) + (\log n)p', \end{aligned}$$

where ℓ is the log-likelihood. Using this definition, smaller values of the AIC (closer to $-\infty$) represent better models. When ϕ is unknown,

$$\begin{aligned} \text{AIC} &= -2 \times \ell(\hat{\beta}_0, \ldots, \hat{\beta}_p, \hat{\phi}; y) + 2(p' + 1) \\ \text{BIC} &= -2 \times \ell(\hat{\beta}_0, \ldots, \hat{\beta}_p, \hat{\phi}; y) + (\log n)(p' + 1), \end{aligned}$$

where $\hat{\phi}$ is the MLE of ϕ. In fact, R inserts the simple mean deviance estimate $D(y, \hat{\mu})/n$ for ϕ. This is the MLE for normal and inverse Gaussian GLMs. For gamma GLMs, this is approximately the MLE when the saddlepoint approximation is accurate.

The definitions of the AIC and BIC given above are computed in R using `AIC()` and `BIC()` respectively. The function `extractAIC()` also computes the AIC and BIC using these definitions for GLMs, but omits all constant terms when computing the AIC and BIC for linear regression models (and so uses the forms presented in Sect. 2.11). In other words, the results from using `AIC()` and `BIC()` allow comparisons between linear regression models and GLMs, but `extractAIC()` does not. Note that the BIC is found using `extractAIC()` by specifying the penalty `k=log(nobs(y))` where `y` is the response variable. (For more information, see Sect. 4.12.)

Example 7.22. For the cherry tree data `trees`, suppose we wish to compare the models

$$\begin{aligned} \text{Model 1:} \quad & \log \mu = \beta_0 + \ 2x_1 + \beta_2 x_2 \\ \text{Model 2:} \quad & \log \mu = \beta_0 + \beta_1 x_1 + \ \ x_2, \end{aligned}$$

writing x_1 for `log(Girth)` and x_2 for `log(Height)`. Note that these models are not nested. The coefficients for `log(Girth)` and `log(Height)` are treated in turn as an offset (Sect. 5.5.2) by using their theoretical values. First we fit both models:

```
> tr.aic1 <- glm( Volume ~ offset(2*log(Girth)) + log(Height),
            family=Gamma(link="log"), data=trees)
> tr.aic2 <- glm( Volume ~ log(Girth) + offset(log(Height)),
            family=Gamma(link="log"), data=trees)
```

We can compute the corresponding AICs using either `extractAIC()` or `AIC()`, which produce the same answers for GLMs:

```
> c(extractAIC(tr.aic1), extractAIC(tr.aic2))
[1]   2.0000 137.9780   2.0000 138.3677
> c(       AIC(tr.aic1),        AIC(tr.aic2))
[1] 137.9780 138.3677
```

The AIC suggests that the first model is preferred for prediction, so prefer the model which sets the coefficient for `log(Girth)` to two, and estimating the coefficient for `log(Height)`. □

7.9 Automated Methods for Model Selection

The same automatic procedures used for normal linear regression (Sect. 2.12.2, p. 73) can also be used for GLMs: `drop1()`, `add1()` and `step()`, and in the same manner. R bases the decisions about model selection on the value of the AIC by default. The same objections remain to automated variable selection in the GLM context as in the linear regression context (Sect. 2.12.3).

Care is needed when applying the automated methods with GLMs when ϕ is estimated, since the estimate of ϕ is different for each model being compared, and the estimate is not the MLE (the simple mean deviance estimate is used). In other words, the computed AIC is only approximate (Sect. 7.8).

Example 7.23. To use an automated procedure for fitting a model to the cherry tree data (data set: `trees`), use `step()` as follows. (This is shown for illustration only, as such a process is not necessary in this situation.)

```
> min.model <- glm( Volume~1, data=trees, family=Gamma(link="log"))
> max.model <- glm( Volume~log(Girth) + log(Height),
                    data=trees, family=Gamma(link="log"))
> m.f <- step( min.model, scope=list(lower=min.model, upper=max.model),
              direction="backward")
```

The backward elimination and stepwise regression procedures are used in the following way:

```
> m.b <- step( max.model, scope=list(lower=min.model, upper=max.model),
              direction="backward")
> m.s <- step( min.model, scope=list(lower=min.model, upper=max.model),
              direction="both")
```

In this case, all methods suggest the same model, which is the model suggested from a theoretical basis:

```
> coef(m.s)
(Intercept)  log(Girth) log(Height)
  -6.691109    1.980412    1.132878
```

□

7.10 Using R to Perform Tests

Various R functions are used to conduct inference on a fitted model named, say, `fit` produced from a call to `glm()`.

`summary(fit)`: The `summary()` of the model `fit` prints the following (see Fig. 6.1): the parameter estimates, with the corresponding standard errors (or estimated standard errors); the Wald statistic for testing H_0: $\beta_j = 0$, and the corresponding P-values; the value of ϕ if ϕ is fixed, or the Pearson estimate of ϕ if ϕ is unknown; the null deviance (the residual deviance after fitting just the constant term as an explanatory variable) and the corresponding degrees of freedom; the residual deviance after fitting the given model, and the corresponding degrees of freedom; the AIC for the model; and the number of Fisher scoring iterations necessary for convergence of the IRLS algorithm.

The output of `summary()` (for example, `out <- summary(fit)`) contains substantial information. `out$family` displays the EDM and the link function used to fit the model, and `out$dispersion` displays the value of the Pearson estimate of ϕ. `coef(out)` displays the parameter estimates and standard errors, plus the z- or t-values (for ϕ known and unknown respectively) and two-tailed P-values for testing H_0: $\beta_j = 0$. See `?summary.glm` for further information.

`summary()` uses the Pearson estimator of ϕ by default; other estimates can be used by specifying the estimate using `dispersion` input in the call to `summary()`. `deviance()` returns the deviance of a model, and `df.residual()` returns the residual degrees of freedom for the model.

`glm.scoretest(fit, x2)`: The function `glm.scoretest()` (available in the package **statmod**) is used to conduct score tests to determine if the explanatory variables in `x2` should be added to the model `fit`. The Pearson

estimator of ϕ is used when ϕ is unknown, but other estimates can be used by specifying the estimate using `dispersion` input in the call to `glm.scoretest()`.

`anova()`: The `anova()` function reports the results of comparing nested models. `anova()` can be used in two forms:

1. `anova(fit)`: When a single GLM model is given as input, an ANOVA table is produced that sequentially tests the significance of each term as it is added to the model.
2. `anova(fit1, fit2, ...)`: Compare any set of nested GLMs by providing all the models to `anova()`. The models are then tested against one another in the specified order, where models earlier in the list of models are nested in later models.

`anova( ..., test="F")` produces P-values by explicitly referring to an F-distribution when ϕ is estimated (Sect. 7.6.4). `anova( ..., test="Chisq")` produces P-values by explicitly referring to a χ^2 distribution when ϕ is known (Sect. 7.2.4).

`anova()` uses the Pearson estimator of ϕ, but other estimates can be used by specifying the estimate using `dispersion` input in the call to `anova()`.

`confint()`: Returns the 95% Wald confidence interval for all the estimated coefficients $\hat{\beta}_j$ in the systematic component. For different confidence levels, use `confint(fit, level=0.99)`, for example, which creates 99% confidence intervals. The Pearson estimate of ϕ is used by default, but other estimates can be supplied using the `dispersion` input.

`AIC(fit)` and `BIC(fit)`: Returns the AIC and BIC for the given model respectively. The function `extractAIC(fit)` also returns the AIC (as the second value returned); the BIC is computed using `extractAIC(fit, k=log(nobs(y)))`.

`drop1()` and `add1()`: Drops or adds explanatory variables one at a time from the given model. Decisions are based on the AIC by default; F-test results are displayed by using `test="F"` and χ^2-test results are displayed by using `test="Chisq"`. To use `add1()`, the second input shows the scope of the models to be considered.

`step()`: Uses automated methods for selecting a GLM based on the AIC. Common usage is `step(object, scope, direction)`, where `direction` is one of `"forward"` for forward regression, `"backward"` for backward elimination, or `"both"` for stepwise regression. `object` is an initial GLM, and `scope` defines extent of the models to be considered. Sect. 2.12.2 (p. 73) demonstrates the use of `step()` for the three types of automated methods.

7.11 Summary

Chapter 7 considers various inference methods for GLMs.

Wald tests can be used to test for the statistical significance of individual regression coefficients, using a one- or two-tailed alternative (Sect. 7.2.1 when ϕ is known; Sect. 7.6.1 when ϕ is unknown). Confidence intervals for individual regression coefficients are conveniently computed using the Wald statistic (Sect. 7.2.2 when ϕ is known, Sect. 7.6.2 when ϕ is unknown).

Confidence intervals for $\hat{\mu}$ are found by first computing confidence intervals for $\hat{\eta}$, and then applying the inverse link function (that is, $\mu = g^{-1}(\eta)$) to the lower and upper limit of the interval found for $\hat{\eta}$ (Sect. 7.2.3 when ϕ is known; Sect. 7.6.3 when ϕ is unknown).

Two nested GLMs, say Model A nested in Model B, can be compared using a likelihood ratio test. When ϕ is known, the likelihood ratio statistic is approximately distributed as $\chi^2_{p'_B - p'_A}$ if n is relatively large compared to p' (Sect. 7.2.4). When ϕ is unknown, the likelihood ratio statistic is approximately distributed as an F-distribution with $(p'_B - p'_A, n - p'_B)$ degrees of freedom, provided the appropriate estimator of ϕ is used. The Pearson estimator or the modified profile likelihood estimator of ϕ are used in the large sample case, and the mean deviance estimator of ϕ is used in the small dispersion case (Sect. 7.6.4).

Commonly, a series of nested models is compared using likelihood ratio tests. The information from these tests are organized into analysis of deviance tables (Sects. 7.2.5 if ϕ is known, and 7.6.5 if ϕ is unknown).

The score test statistic can be used to test the null hypothesis (against one- or two-tailed alternatives) that a set of covariates are useful predictors (Sect. 7.2.7 when ϕ is known; Sect. 7.6.6 when ϕ is unknown).

The Wald, likelihood ratio and score tests are based on large-sample asymptotic results, which apply when n is reasonably large (Sect. 7.3).

When ϕ is known, goodness-of-fit tests can be used to determine if the linear predictor already includes enough explanatory variables to fully describe the systematic trends in the data (Sect. 7.4). The saturated model is the largest possible model which can, in principle, be fitted to the data (Sect. 7.4.1). The saturated model has as many explanatory variables as observations ($p' = n$) and the fitted values are all equal to the responses ($\hat{\mu} = y$).

The deviance goodness-of-fit test statistic is the residual deviance $D(y, \hat{\mu})$ (Sect. 7.4.2). The Pearson goodness-of-fit test statistic is the Pearson statistic X^2 (Sect. 7.4.3). The distributional assumptions of goodness-of-fit test statistics rely on small dispersion asymptotic results (the saddlepoint approximation and the Central Limit Theorem), not large sample asymptotic results (Sect. 7.5).

The Pearson statistic has an approximate chi-square distribution when the Central Limit Theorem holds for individual observations (Sect. 7.5, where guidelines are provided). The residual deviance has an approximate chi-square

distribution when the saddlepoint approximation holds for individual observations (Sect. 7.5, where guidelines are provided).

In practice, Wald tests are commonly used for tests about individual coefficients, and likelihood ratio tests for comparing nested models (Sect. 7.7). The likelihood ratio and score tests are recommended over Wald tests for determining if a variable should be included in the model, as the distributional assumptions of Wald tests are often quite inaccurate. Likelihood ratio tests are traditionally used to test two-tailed alternative hypotheses (Sect. 7.7).

The AIC and BIC can be used to compare non-nested GLMs (Sect. 7.8). Automated procedures for choosing between models include forward regression, backward elimination and step-wise regression (Sect. 7.9).

Problems

Selected solutions begin on p. 537.

7.1. A study examined the relationships between weather conditions during the first 21 days posthatch of scaled quail broods and their survival to 21 days of age [5]. A binomial GLM was fitted, using the systematic component $\log\{\mu/(1-\mu)\} = \eta$, where $0 < \mu < 1$ is the fitted probability that the chicks survived 21 days. A total of 54 broods were used in the study (Table 7.4).

1. Suggest a model based on the likelihood ratio statistics.
2. Use Wald tests to determine which explanatory variables are significant.
3. Interpret the final model.
4. Find the 95% confidence interval for the regression coefficient for maximum temperature.

7.2. To model the number of species ('species abundance') of freshwater mussels in a sample of 44 rivers in parts of the USA [6, 10], a Poisson GLM (with a logarithmic link function) was used with these potential explanatory variables: the log of the drainage basin area (LA); stepping-stone distance from the Alabama–Coosa River (AC); stepping-stone distance from the Apalachicola river (AP); stepping-stone distance from the Savannah River

Table 7.4 The parameter estimates and standard errors for a binomial GLM, and the likelihood ratio test statistic L when the indicated variable was excluded from the full model containing all three explanatory variables (Problem 7.1)

Explanatory variable	$\hat{\beta}_j$	$\text{se}(\hat{\beta}_j)$	L
Minimum temperature during first 12 days	0.143	0.19	0.602
Maximum temperature during first 7 days	1.247	0.45	14.83
Number days with precipitation during first 7 days	−0.706	0.45	2.83

Table 7.5 The analysis of deviance table (left) for the species abundance of freshwater mussels where $D^*(y, \mu)$ is the residual scaled deviance, and the fitted regression parameters (right) for the main-effects model containing all explanatory variables (Problem 7.2)

Model	Residual deviance $D^*(y, \mu)$	Residual df		Parameters in full model $\hat{\beta}_j$	se($\hat{\beta}_j$)
Full main-effects model	35.77	36			
− SL	35.90	37	SL	−0.0118	0.0326
− AC	35.91	38	AC	−0.0212	0.0654
− SV	38.44	39	SV	0.0473	0.0473
− N	39.60	40	N	0.0110	0.0112
− H	50.97	41	H	−0.0334	0.0115
− SR	60.26	42	SR	−0.0024	0.0007
− AP	77.82	43	AP	−0.0222	0.0053
(Note: LA not removed)			LA	0.2821	0.0566

(SV); stepping-stone distance from the St Lawrence River (SL); nitrate content of river water (N); solid residue in river water (SR); and hydronium ion concentration of river water (H).

1. Suggest a model based on the changes in residual deviance.
2. What method of selecting a model (forward, backward, or step-wise) is implied by Table 7.5?
3. Use the AIC to recommend a model. (HINT: Using (5.26) may prove useful.)
4. Use Wald tests to determine which explanatory variables are significant.
5. Give possible reasons to explain why the explanatory variables suggested for the two models may be different for the Wald and likelihood ratio tests.
6. The *final* Poisson GLM chosen in the source is

$$\log \hat{\mu} = 0.7219 - 0.0264\text{AP} - 0.0022\text{SR} - 0.0336\text{H} + 0.2773\text{LA}, \quad (7.6)$$

   where the standard errors for each coefficient are, respectively, 0.46, 0.005, 0.0006, 0.011 and 0.05. Compute the Wald statistic for each parameter in this final model.
7. Why are the parameter estimates in (7.6) different than those in Table 7.5?
8. Interpret the final model.

7.3. A study [11] compared the number of days each week that 82 junior British and Irish legislators spent in their constituency, by using a Poisson GLM. The dummy variable Nation is coded as 0 for British and 1 for Irish legislators. The mean number of days spent in their constituency is 1.8 in Britain, and 2.5 in Ireland.

1. Explain why a Poisson GLM may not be appropriate for these data, but why a Poisson GLM is probably reasonably useful anyway.

Table 7.6 The parameter estimates and standard errors from a study of the number of days per week junior legislators spend in their constituency (Problem 7.3)

	Constant	Safeness of seat	Expectation of punishment	Present role	Future role?	Geographic proximity	Nation
$\hat{\beta}_j$	0.23	0.04	0.06	0.01	0.09	0.05	0.30
$\text{se}(\hat{\beta}_j)$	0.13	0.04	0.05	0.03	0.06	0.02	0.07

2. Using the reported results (Table 7.6), determine if there is a difference between the number of days spent in the constituency by British and Irish legislators.
3. Interpret the regression coefficient for Nation.
4. Form a 90% confidence interval for the regression coefficient for Nation.
5. Which terms are statistically significant?
6. Write down the full fitted model.

7.4. Children were asked to build towers as high as they could out of cubical and cylindrical blocks [3, 7]. The number of blocks used and the time taken were recorded (data set: `blocks`). In this problem, only consider the number of blocks used y and the age of the child x. In Problem 6.10, a GLM was fitted for these data.

1. Use a Wald test to determine if age seems necessary in the model.
2. Use a score test to determine if age seems necessary in the model.
3. Use a likelihood ratio test to determine if age seems necessary in the model.
4. Compare the results from the Wald, score and likelihood ratio tests. Comment.
5. Is the saddlepoint approximation expected to be accurate? Explain.
6. Is the Central Limit Theorem expected to be accurate? Explain.
7. Find the 95% Wald confidence intervals for the regression coefficients.
8. Plot the number of blocks used against age, and show the relationship described by the fitted model. Also plot the lines indicating the lower and upper 95% confidence intervals for these fitted values.

7.5. Nambe Mills, Santa Fe, New Mexico [1, 8], is a tableware manufacturer. After casting, items produced by Nambe Mills are shaped, ground, buffed, and polished. In 1989, as an aid to rationalizing production of its 100 products, the company recorded the total grinding and polishing times and the diameter of each item (Table 5.3; data set: `nambeware`). In this problem, only consider the item price y and item diameter x. In Problem 6.11, a GLM was fitted to these data.

1. Use a Wald test to determine if diameter is significant.
2. Use a score test to determine if diameter is significant.
3. Use a likelihood ratio test to determine if diameter is significant.

4. Compare the results from the Wald, score and likelihood ratio tests. Comment.
5. Is the saddlepoint approximation expected to be accurate? Explain.
6. Is the Central Limit Theorem expected to be accurate? Explain.
7. Find the 95% Wald confidence intervals for the regression coefficients.
8. Plot the price against diameter, and show the relationship described by the fitted model. Also plot the lines indicating the lower and upper 95% confidence intervals for these fitted values.

References

[1] Data Desk: Data and story library (DASL) (2017). URL http://dasl.datadesk.com

[2] Fisher, S.R.A.: Statistical Methods for Research Workers. Hafner Press, New York (1970)

[3] Johnson, B., Courtney, D.M.: Tower building. Child Development **2**(2), 161–162 (1931)

[4] Maron, M.: Threshold effect of eucalypt density on an aggressive avian competitor. Biological Conservation **136**, 100–107 (2007)

[5] Pleasant, G.D., Dabbert, C.B., Mitchell, R.B.: Nesting ecology and survival of scaled quail in the southern high plains of Texas. The Journal of Wildlife Managament **70**(3), 632–640 (2006)

[6] Sepkoski, J.J., Rex, M.A.: Distribution of freshwater mussels: Coastal rivers as biogeographic islands. Systematic Zoology **23**, 165–188 (1974)

[7] Singer, J.D., Willett, J.B.: Improving the teaching of applied statistics: Putting the data back into data analysis. The American Statistician **44**(3), 223–230 (1990)

[8] Smyth, G.K.: Australasian data and story library (OzDASL) (2011). URL http://www.statsci.org/data

[9] Smyth, G.K., Verbyla, A.P.: Adjusted likelihood methods for modelling dispersion in generalized linear models. Environmetrics **10**, 695–709 (1999)

[10] Vincent, P.J., Hayworth, J.M.: Poisson regression models of species abundance. Journal of Biogeography **10**, 153–160 (1983)

[11] Wood, D.M., Young, G.: Comparing constitutional activity by junior legislators in Great Britain and Ireland. Legislative Studies Quarterly **22**(2), 217–232 (1997)

Chapter 8
Generalized Linear Models: Diagnostics

Since all models are wrong the scientist must be alert to what is importantly wrong. It is inappropriate to be concerned about mice when there are tigers abroad.
Box [1, p. 792]

8.1 Introduction and Overview

This chapter introduces some of the necessary tools for detecting violations of the assumptions in a GLM, and then discusses possible solutions. The assumptions of the GLM are first reviewed (Sect. 8.2), then the three basic types of residuals (Pearson, deviance and quantile) are defined (Sect. 8.3). The leverages are then given in the GLM context (Sect. 8.4) leading to the development of standardized residuals (Sect. 8.5). The various diagnostic tools for checking the model assumptions are introduced (Sect. 8.7) followed by techniques for identifying unusual and influential observations (Sect. 8.8). Comments about using each type of residual and the nomenclature of residuals are given in Sect. 8.6. We then discuss techniques to remedy or ameliorate any weaknesses in the models (Sect. 8.9), including the introduction of quasi-likelihood (Sect. 8.10). Finally, collinearity is discussed (Sect. 8.11).

8.2 Assumptions of GLMs

The assumptions made when fitting GLMs concern:

- Lack of outliers: All responses were generated from the same process, so that the same model is appropriate for all the observations.
- Link function: The correct link function $g()$ is used.
- Linearity: All important explanatory variables are included, and each explanatory variable is included in the linear predictor on the correct scale.
- Variance function: The correct variance function $V(\mu)$ is used.
- Dispersion parameter: The dispersion parameter ϕ is constant.
- Independence: The responses y_i are independent of each other.

P. K. Dunn, G. K. Smyth, *Generalized Linear Models with Examples in R*,
Springer Texts in Statistics, https://doi.org/10.1007/978-1-4419-0118-7_8

- Distribution: The responses y_i come from the specified EDM.

The first assumption concerns the suitability of the model overall. The other assumptions are ordered here from those that affect the first moment of the responses (the mean), to the second moment (variances) to third and higher moments (complete distribution of y_i). Generally speaking, assumptions that affect the lower moments of y_i are the most basic. Compare these to the assumptions for the (normal) linear regression model (Sect. 3.2). This chapter discusses methods for assessing the validity of these assumptions.

Importantly, the assumptions are never *exactly* true. Instead, it is important to be aware of the sensitivity of the conclusions to deviations from the model assumptions. The model assumptions should always be checked after fitting a model to identify potential problems, and this information used to improve the model where possible.

8.3 Residuals for GLMs

8.3.1 Response Residuals Are Insufficient for GLMs

The distances $y_i - \hat{\mu}_i$ are called the *response residuals*, and are the basis for residuals in linear regression. The response residuals are inadequate for assessing a fitted GLM, because GLMs are based on EDMs where (in general) the variance depends on the mean. As an example, consider the cherry tree data (Example 3.14, p. 125), and the theory-based model fitted to the data:

```
> data(trees)
> cherry.m1 <- glm( Volume ~ log(Girth) + log(Height),
                    family=Gamma(link=log), data=trees)
> coef( cherry.m1 )
(Intercept)  log(Girth) log(Height)
  -6.691109    1.980412    1.132878
```

Consider two volumes y_1 and y_2 marked on Fig. 8.1. Also shown are the modelled distributions of the observations for the corresponding fitted values $\hat{\mu}_i$ (based on the gamma distribution). Note that both observations are $y_i - \hat{\mu}_i = 7$ greater than the respective predicted means. However, observation y_1 is in the extreme tail of the fitted distribution, but observation y_2 is not in the extreme tail of the distribution, even though the response residuals $y_i - \hat{\mu}_i$ are the same for each case. A new definition of residuals is necessary.

Ideally, residuals for GLMs should behave similarly to residuals for linear regression models, because residuals in that case are familiar and easily interpreted. That is, ideally residuals for GLMs should be approximately normally distributed with mean zero and constant variance. Response residuals do not necessarily have constant variance or a normal distribution.

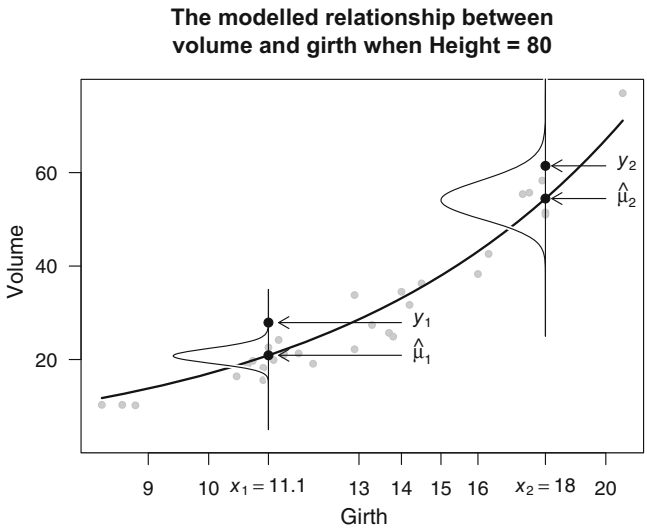

Fig. 8.1 The cherry tree data. The solid line shows the modelled relationship between `Volume` and `log(Girth)` when `Ht=80`. Two observations from the gamma GLM as fitted to the cherry tree data are also shown. Observation y_1 is extreme, but observation y_2 is not extreme, yet the difference $y_i - \hat{\mu}_i = 7$ is the same in both cases. Note that log-scale is used on the horizontal axis since the covariate is `log(Girth)` (Sect. 8.3.1)

8.3.2 Pearson Residuals

The most direct way to handle the non-constant variance in EDMs is to divide out the effect of non-constant variance. In this spirit, define *Pearson residuals* as

$$r_P = \frac{y - \hat{\mu}}{\sqrt{V(\hat{\mu})/w}},$$

where $V()$ is the variance function. Notice that r_P is the square root of the unit Pearson statistic (Sect. 6.8.5). For a fitted GLM in R, say `fit`, the Pearson residuals are found using `resid(fit, type="pearson")`. The Pearson residuals are actually the ordinary residuals when the GLM is treated as a least-squares regression model using the working responses and weights (Sect. 6.7).

The Pearson statistic has an approximate chi-square distribution when the Central Limit Theorem applies, under the conditions given in Sect. 7.5 (p. 276). Under these same conditions, the Pearson residuals have an approximate normal distribution.

Example 8.1. For the normal distribution, $V(\mu) = 1$ (Table 5.1), and so the Pearson residuals are $r_P = (y - \hat{\mu})\sqrt{w}$. □

Example 8.2. For the Poisson distribution, $V(\mu) = \mu$ (Table 5.1), and so the Pearson residuals are $r_P = (y - \hat{\mu})/\sqrt{\hat{\mu}/w}$. □

8.3.3 Deviance Residuals

The Pearson residuals are the square root of the unit Pearson statistic. Similarly, define the deviance residuals r_D as the signed square root of the unit deviance (Sect. 5.4):

$$r_D = \text{sign}(y - \hat{\mu})\sqrt{wd(y, \hat{\mu})}. \tag{8.1}$$

(The function sign(x) equals 1 if $x > 0$; -1 if $x < 0$; and 0 if $x = 0$.) For a fitted model in R, say `fit`, the deviance residuals are found using `resid(fit)`. In other words, the deviance residuals are computed by default by `resid()`. A summary of the deviance residuals is given in the `summary()` of the output object produced by `glm()` (as seen in Fig. 6.1).

The deviance statistic has an approximate chi-square distribution when the saddlepoint approximation applies, under the conditions given in Sect. 7.5 (p. 276). Under these same conditions, the deviance residuals have an approximate normal distribution.

Example 8.3. Using the unit deviance for the normal distribution (Table 5.1), the deviance residuals are $r_D = (y - \hat{\mu})\sqrt{w}$. The deviance residuals are the same as the Pearson residuals for the normal distribution, and only for the normal distribution. □

Example 8.4. Using the unit deviance for the Poisson distribution (Table 5.1), the deviance residuals are

$$r_D = \text{sign}(y - \hat{\mu})\sqrt{2w\left\{y \log\left(\frac{y}{\hat{\mu}}\right) - (y - \hat{\mu})\right\}}.$$

□

8.3.4 Quantile Residuals

The Pearson and deviance residuals have approximate normal distributions as explained above, with the deviance residuals more likely to be more normally distributed than the Pearson residuals [12]. When the guidelines in Sect. 7.5 (p. 276) are not met, the Pearson and deviance residuals can be clearly non-normal, especially for discrete distributions.

An alternative to Pearson and deviance residuals are the quantile residuals [5], which are *exactly* normally distributed apart from the sampling variability in estimating μ and ϕ, assuming that the correct EDM is used. The quantile residual r_Q for an observation has the same cumulative probability on a standard normal distribution as y does for the fitted EDM. A simple modification involving randomization is needed for discrete EDMs. For a fitted model in R, say `fit`, the quantile residuals are found using `qresid(fit)`, using the function `qresid()` from package **statmod**.

8.3.4.1 Quantile Residuals: Continuous Response

Quantile residuals are best described in the context of an example. Consider an exponential EDM (4.37) (which is a gamma EDM with $\phi = 1$) fitted to data where one observation is $y = 1.2$ with $\hat{\mu} = 3$. First, determine the cumulative probability that an observation is less than or equal to y on this fitted exponential distribution using `pexp()` (Fig. 8.2, left panel):

```
> y <- 1.2; mu <- 3
> cum.prob <- pexp(y, rate=1/mu); cum.prob
[1] 0.32968
```

Then find the value of the standard normal variate with the same cumulative probability using `qnorm()`; this is the quantile residual (Fig. 8.2, right panel):

```
> rq <- qnorm(cum.prob); rq
[1] -0.4407971
```

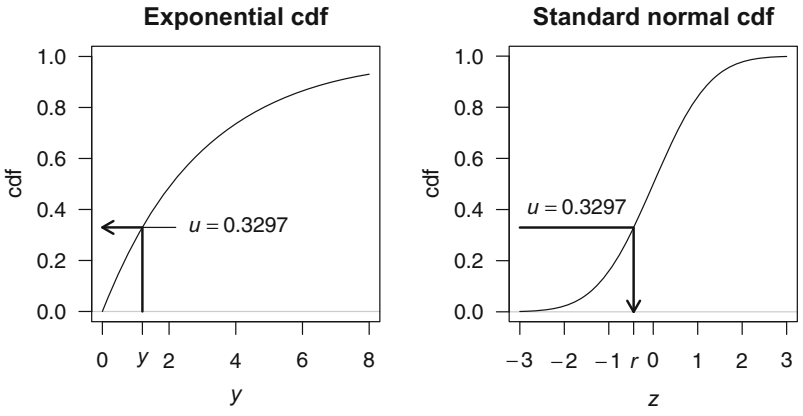

Fig. 8.2 Computing the quantile residuals for an exponential EDM for an observation $y = 1.2$, when $\hat{\mu} = 3$ (Sect. 8.3.4.2)

More formally, let $\mathcal{F}(y;\mu,\phi)$ be the cumulative distribution function (CDF) of a random variable y (it need not belong to the EDM family). The quantile residuals are

$$r_Q = \Phi^{-1}\{\mathcal{F}(y;\hat{\mu},\phi)\},$$

where $\Phi(\cdot)$ is the CDF of the standard normal distribution. (For example, $\Phi^{-1}(0.975) = 1.96$ and $\Phi^{-1}(0.025) = -1.96$.) If ϕ is unknown, use the Pearson estimator of ϕ.

Example 8.5. For the exponential distribution, the probability function is given in (4.37). The CDF is

$$\mathcal{F}(y) = 1 - \frac{1}{\mu}\exp\left(-\frac{y}{\mu}\right)$$

for $y > 0$. The quantile residual is

$$r_Q = \Phi^{-1}\left\{1 - \frac{1}{\hat{\mu}}\exp\left(-\frac{y}{\hat{\mu}}\right)\right\}.$$

□

Example 8.6. For the normal distribution, $\mathcal{F}$ is the CDF of a normal distribution with mean μ and variance σ^2/w. Since $\Phi^{-1}(\cdot)$ is the *inverse* of the standard normal CDF, the quantile residuals are

$$r_Q = \frac{(y-\hat{\mu})\sqrt{w}}{s},$$

where s is the estimate of σ. For the normal distribution, $r_Q = r_P/s = r_D/s$.
□

8.3.4.2 Quantile Residuals: Discrete Response

For discrete EDMs, a simple modification is necessary to define the quantile residuals. Consider a Poisson EDM for the observation $y = 1$ when $\hat{\mu} = 2.6$.

Locate the observation $y = 1$ on the Poisson CDF (Fig. 8.3, left panel). Since the CDF is discrete at $y = 1$, the CDF makes a discrete jump between $a = 0.074$ and $b = 0.267$:

```
> y <- 1; mu <- 2.6
> a <- ppois(y-1, mu); b <- ppois(y, mu)
> c(a, b)
[1] 0.07427358 0.26738488
```

Choose a point at random from the shaded area of the plot between a and b:

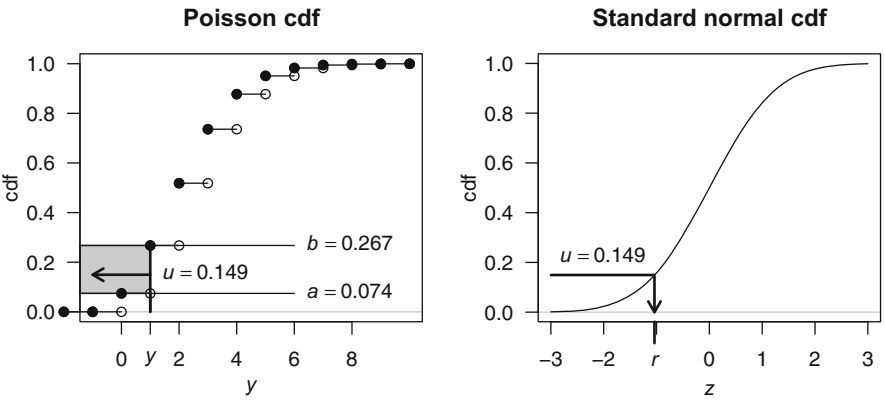

Fig. 8.3 Computing the quantile residuals for a situation where the observed value is $y = 1$ when $\hat{\mu} = 2.6$ for a Poisson distribution. The filled circles indicate the value is included, while a hollow circle indicates the value is excluded (Sect. 8.3.4.2)

```
> u <- runif(1, a, b); u
[1] 0.1494077
```

In this example, the chosen random number is $u = 0.149$. Then find the value of a standard normal variate with the same cumulative probability, as in the continuous EDM case (Fig. 8.3, right panel). This standard normal variate is the quantile residual for that observation:

```
> rq <- qnorm( u ); rq
[1] -1.038977
```

In this example, the quantile residual is $r_Q = \Phi^{-1}(0.149) = -1.039$. (Using the extremities of the interval for u_i, the quantile residual will be between approximately -0.621 and -1.445.)

This randomization is an advantage: the quantile residuals are continuous even for discrete distributions, unlike deviance and Pearson residuals (Example 8.8; Problem 8.4). As for the continuous case, the quantile residuals have an exact standard normal distribution.

Symbolically, let the lower and upper limits of the region in the CDF be $a = \lim_{\epsilon \uparrow 0} \mathcal{F}(y + \epsilon; \hat{\mu}, \phi)$ and $b = \mathcal{F}(y; \hat{\mu}, \phi)$ respectively. (The notation $\lim_{\epsilon \uparrow 0}$ means the limit as ϵ approaches 0 from *below*, so that ϵ is always negative.) Then, define randomized quantile residuals as

$$r_Q = \Phi^{-1}(u),$$

where u is a uniform random variable on the interval $(a, b]$. For the Poisson example above, $b = \mathcal{F}(y = 1; \hat{\mu} = 2.6)$, where $\mathcal{F}$ is the CDF for the Poisson distribution. The value of a is the value of the CDF as y approaches but is *less* than $y = 1$. Thus, $a = \lim_{\epsilon \uparrow 0} \mathcal{F}(y + \epsilon; \hat{\mu} = 2.6) = \mathcal{F}(y = 0.2, \hat{\mu} = 2.6)$.

Four replications of the quantile residuals are recommended [5] when used with discrete distributions because quantile residuals for a discrete response

have a random component. Any features not preserved across all four sets of residuals are considered artifacts of the randomization. In the discrete case, quantile residuals are sometimes called *randomized* quantile residuals, for obvious reasons.

Quantile residuals are best used in residual plots where trends and patterns are of interest, because $y - \hat{\mu} < 0$ does not necessarily imply $r_Q < 0$ (Problem 8.7). Quantile residuals are strongly encouraged for discrete EDMs (Example 8.8).

8.4 The Leverages in GLMs

8.4.1 Working Leverages

As previously explained in Sect. 6.7, a GLM can be treated locally as a linear regression model with working responses z_i and working weights W_i. The working responses and weights are functions of the fitted values $\hat{\mu}_i$, but, if we treat them as fixed, we can compute leverages (or hat values) for each observation exactly as for linear regression (Sect. 3.4.2).

The ith leverage h_i is the weight that observation z_i receives when computing the corresponding value of the linear predictor $\hat{\eta}_i$. If the leverage is small, this is evidence that many observations, not just one, are contributing to the estimation of the fitted value. In the extreme case that $h_i = 1$, the ith fitted value will be entirely determined by the ith observation, so that $\hat{\eta}_i = z_i$ and $\hat{\mu}_i = y$.

The variance of the working residuals $e_i = z_i - \hat{\eta}_i$ can be approximated by (see Sect.6.7)

$$\text{var}[e_i] \approx \phi V(\hat{\mu}_i)(1 - h_i).$$

If ϕ is unknown, a suitable estimate is used to give $\widehat{\text{var}}[e_i]$. As in linear regression, the leverages are computed using `hatvalues()` in R.

* *8.4.2 The Hat Matrix*

In the context of GLMs, the *hat matrix* is

$$\text{H} = \text{W}^{1/2}\text{X}(\text{X}^T\text{WX})^{-1}\text{X}^T\text{W}^{1/2}, \tag{8.2}$$

where W is the diagonal matrix of weights from the final iteration of the fitting algorithm (Sect. 6.3). The form is exactly the same as used in linear regression (Sect. 3.4.2), except in the GLM case W depends on the fitted values

$\hat{\boldsymbol{\mu}}$. The leverages (or hat diagonals) h_i are the diagonal elements of H, and are found in R using `hatvalues()`.

8.5 Leverage Standardized Residuals for GLMs

The Pearson, deviance and quantile residuals discussed in Sect. 8.3 are the basic types of residuals (called *raw residuals*). As with linear regression, standardized residuals have approximately constant variance, and are defined analogously:

$$\begin{aligned} r'_P &= \frac{r_P}{\sqrt{\phi(1-h)}} = \frac{y-\hat{\mu}}{\sqrt{\phi V(\hat{\mu})(1-h)/w}} \\ r'_D &= \frac{r_D}{\sqrt{\phi(1-h)}} = \frac{\text{sign}(y-\hat{\mu})\sqrt{wd(y,\hat{\mu})}}{\sqrt{\phi(1-h)}} \\ r'_Q &= \frac{r_Q}{\sqrt{1-h}}, \end{aligned} \tag{8.3}$$

where h are the leverages. If ϕ is unknown, use an estimate of ϕ (R uses the Pearson estimate $\bar{\phi}$). The standardized deviance residuals are found directly using `rstandard()`; the standardized Pearson and quantile residuals must be computed in R using the formulae above.

The standardized deviance residuals have a useful interpretation. The square of the standardized deviance residuals is approximately the reduction in the residual deviance when Observation i is omitted from the data, scaled by ϕ (Problem 8.6).

Observe that division by ϕ (or its estimate) is not needed for the quantile residuals as the quantile residuals are transformed to the standard normal distribution with variance one.

Example 8.7. For the model `cherry.m1` fitted to the cherry tree data (Sect. 8.3; data set: `trees`), compute the three types of raw residuals in R as follows:

```
> library(statmod)          # Provides qresid()
> rP <-  resid( cherry.m1, type="pearson" )
> rD <-  resid( cherry.m1 ) # Deviance resids are the default
> rQ <- qresid( cherry.m1 )
```

Then compute the standardized residuals also:

```
> phi.est <- summary( cherry.m1 )$dispersion  # Pearson estimate
> rP.std <- rP / sqrt( phi.est*(1 - hatvalues(cherry.m1)) )
> rD.std <- rstandard(cherry.m1)
> rQ.std <- rQ / sqrt( 1 - hatvalues(cherry.m1) )
> all.res <- cbind( rP, rP.std, rD, rD.std, rQ, rQ.std )
> head( all.res )  # Show the first six values only
```

```
           rP     rP.std          rD     rD.std         rQ     rQ.std
1  0.01935248  0.2620392  0.01922903  0.2603676  0.2665369  0.2893348
2  0.03334904  0.4558288  0.03298537  0.4508579  0.4380951  0.4800656
3  0.01300934  0.1811459  0.01295335  0.1803663  0.1882715  0.2101705
4 -0.01315583 -0.1691519 -0.01321397 -0.1698994 -0.1380666 -0.1423184
5 -0.04635977 -0.6169148 -0.04709620 -0.6267146 -0.5606192 -0.5980889
6 -0.04568564 -0.6188416 -0.04640051 -0.6285250 -0.5519432 -0.5993880
> apply( all.res, 2, var ) # Find the variance of each column
         rP      rP.std          rD      rD.std          rQ      rQ.std
0.005998800 1.013173741 0.006113175 1.032103295 0.950789672 1.031780512
```

The variance of the quantile residuals is near one since they are mapped to a standard normal distribution. The standardized residuals are all similar for this example. □

8.6 When to Use Which Type of Residual

Quantile, deviance and Pearson residuals all have exact normal distributions when the responses come from a normal distribution, apart from variability in $\hat{\mu}$ and $\hat{\phi}$. The deviance residuals are also exactly normal for inverse Gaussian GLMs. However, in many cases neither the Pearson nor deviance residuals can be guaranteed to have distributions close to normal, especially for discrete EDMs. The simple rules in Sect. 7.5 (p. 276) can be used to determine when the normality can be expected to be sufficiently accurate.

Quantile residuals are especially encouraged for discrete EDMs, since plots using deviance and Pearson residuals may contain distracting patterns (Example 8.8). Furthermore, standardizing or Studentizing the residuals is encouraged, as these residuals have more constant variance. For some specific diagnostic plots, special types of residuals are used, such as partial residuals and working residuals (Sect. 8.7.3).

8.7 Checking the Model Assumptions

8.7.1 Introduction

As with linear regression models, plots involving the residuals are used for assessing the validity of the model assumptions for GLMs. These plots are discussed in this section. Remedies for any identified problems follow in Sect. 8.9.

A strategy similar to that used for linear regression is adopted for assessing assumptions with GLMs. First, check independence when possible (Sect. 8.7.2). Then, use plots of the residuals against $\hat{\mu}$ and residuals against each explanatory variable to identify structural problems in the model. In

all these situations, the ideal plots contain no patterns or trends. Finally, plotting residuals in a Q–Q plot (Sect. 8.8) is convenient for detecting large residuals.

8.7.2 Independence: Plot Residuals Against Lagged Residuals

Independence of the responses is the most important assumption. Independence of the responses is usually a result of how the data are collected, so is often impossible to detect using residuals. As for linear regression, independence is, in most cases, best assessed from understanding the process by which the data were collected. However, if the data are collected over time, independence can be checked by plotting residuals against the previous residual in time. Ideally, the plots show no pattern under independence. If the data are spatial, independence can be checked by plotting the residuals against spatial explanatory variables (such as latitude and longitude). Again, the ideal plots show no pattern under independence.

The discussion for linear regression is still relevant (Sect. 3.5.5, p. 106), including the typical plots in Fig. 3.8.

8.7.3 Plots to Check the Systematic Component

Plots of the residuals against the fitted values $\hat{\mu}$ and the residuals against x_j are the main tools for diagnostic analysis. Using either the standardized deviance or quantile residuals is preferred in these plots because they have approximately constant variance. Quantile residuals are especially encouraged for discrete EDMs to avoid distracting patterns in the residuals (Example 8.8).

Two features of the plots are important:

- Trends: Any trends appearing in these plots indicate that the systematic component can be improved. This could mean changing the link function, adding extra explanatory variables, or transforming the explanatory variables.
- Constant variation: If the random component is correct (that is, the correct EDM is used), the variance of the points is approximately constant.

The plots can be constructed in R using `plot()`, or using `scatter.smooth()` which also adds a smoothing curve to the plots which may help detect trends. Detecting trends in the plots is often easier if the fitted values $\hat{\mu}$ are spread out more evenly horizontally. This is achieved by using the appropriate variance-stabilizing transformation of $\hat{\mu}$ (Table 5.2), often called the constant-information scale in this context (Table 8.1).

Table 8.1 The constant-information scale transformations of $\hat{\mu}$ for common EDMs for use in residual plots (Sect. 8.7.3)

EDM Scale	EDM Scale
Binomial: $\sin^{-1}\sqrt{\hat{\mu}}$	Inverse Gaussian: $1/\sqrt{\hat{\mu}}$
Poisson: $\sqrt{\hat{\mu}}$	Tweedie ($V(\mu) = \mu^{\xi}$): $\hat{\mu}^{(2-\xi)/2}$
Gamma: $\log \hat{\mu}$	

If the evidence shows problems with the systematic component, then the cause may be an incorrect link function, or an incorrect linear predictor (for example, important explanatory variables are missing, or covariates should be transformed), or both. To further examine the link function, an informal check is to plot the *working responses* (6.9)

$$z_i = \hat{\eta}_i + \frac{\mathrm{d}\eta_i}{\mathrm{d}\mu_i}(y_i - \hat{\mu}_i)$$

against $\hat{\eta}_i$. If the link function is appropriate, then the plot should be roughly linear [10, §12.6.3]. If a noticeable curvature is apparent in the plot, then another choice of link function should be considered. The working responses z_i are found in R using that $z_i = e_i + \hat{\eta}_i$, where e_i are the working residuals (Sect. 6.7), found in R using `resid(fit, type="working")`. Other methods also exist for evaluating the choice of link function [2, 13].

To determine if covariate x_j is included on the incorrect scale, use *partial residuals*

$$u_j = e_i + \hat{\beta}_j x_j, \tag{8.4}$$

found in R using `resid(fit, type="partial")`. This command produces an $n \times p$ array holding the partial residuals for each explanatory variable x_j in the p columns. A plot of u_j against x_j (called a *component-plus-residual plot* or *partial residual plot*) is linear if x_j is included on the correct scale. The R function `termplot()` can also be used to produce partial residual plots, as in linear regression. If many explanatory variables are included on the incorrect scale, the process of examining the partial residual plots for each explanatory variables is iterative: one covariate at a time is fixed, and the partial residual plots re-examined.

Example 8.8. A binomial GLM with a logit link function was used to model 60 observations each with a sample size of 3 (that is, $m = 3$). The systematic component of the fitted model assumed $\eta = \log\{\mu/(1-\mu)\} = \beta_0 + \beta_1 x$ for the covariate x. After fitting the model, the plot of quantile residuals against x shows a curved trend (Fig. 8.4, top left panel), indicating that the model is inadequate. Interpreting the deviance residuals is difficult (Fig. 8.4, top right panel), as the data lie on parallel curves, corresponding to the four possible values of y.

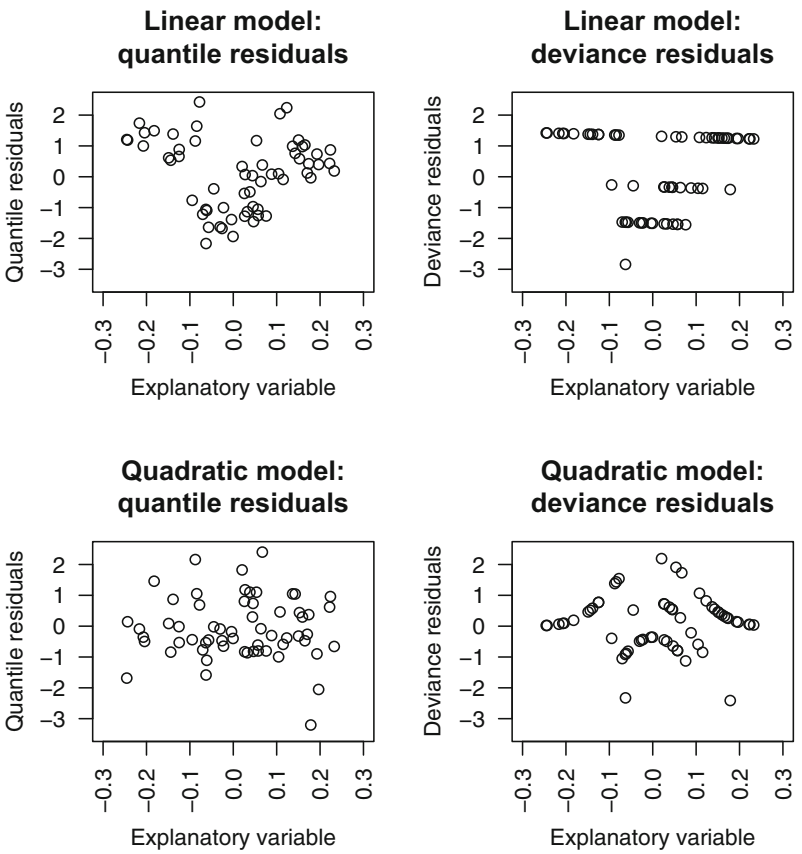

Fig. 8.4 The residuals from a fitted binomial GLM. Top panels: the binomial GLM with a linear systematic component plotted against the explanatory variable; bottom panels: the binomial GLM with a quadratic systematic component plotted against the explanatory variable; left panels: the quantile residuals; right panel: the deviance residuals (Example 8.8)

After fitting the systematic component $\eta = \log\{\mu/(1-\mu)\} = \beta_0 + \beta_1 x + \beta_2 x^2$, the plot of quantile residuals against x (Fig. 8.4, bottom left panel) shows no trend and indicates the model now fits well. The deviance residuals still contain distracting parallel curves (Fig. 8.4, bottom right panel) that make any interpretation difficult. The data actually are randomly generated from a binomial distribution so that η truly depends quadratically on x. (This example is based on [5].) □

Example 8.9. Consider the model `cherry.m1` fitted to the cherry tree data (Example 3.14; data set: `trees`). We now examine the plots of r'_D against $\hat{\mu}$, against `log(Girth)` and against `log(Height)` (Fig. 8.5, top panels):

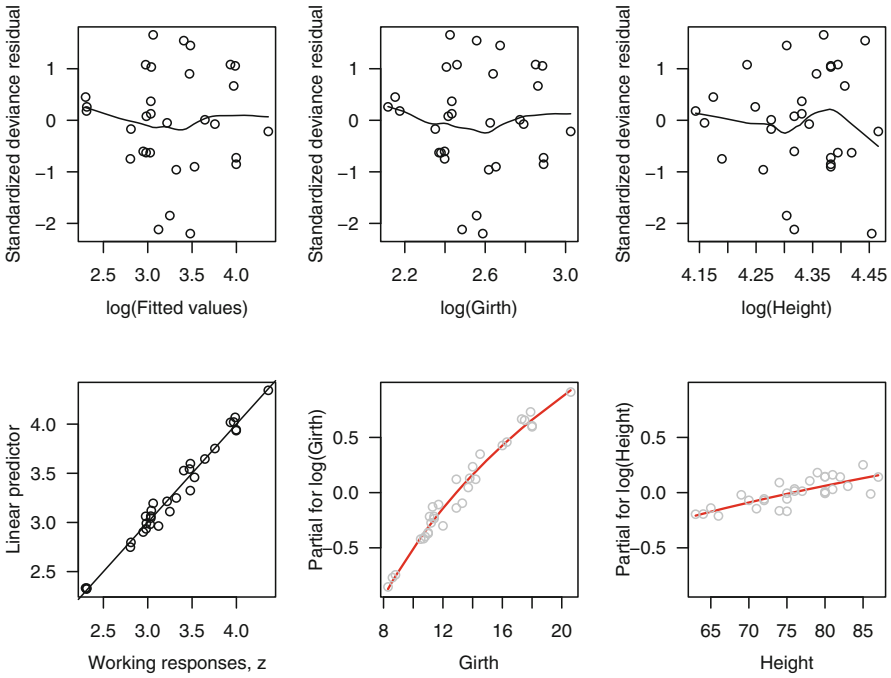

Fig. 8.5 Diagnostic plots for Model `cherry.m1` fitted to the cherry tree data. Top left panel: r'_D against $\log \hat{\mu}_i$; top centre panel: r'_D against `log(Girth)`; top right panel: r'_D against `log(Height)`; bottom left panel: $\hat{\eta}$ against z; bottom centre panel: the partial residual plot for girth; bottom right panel: the partial residual plot for height (Example 8.9)

```
> scatter.smooth( rstandard(cherry.m1) ~ log(fitted(cherry.m1)), las=1,
    ylab="Standardized deviance residual", xlab="log(Fitted values)" )
> scatter.smooth( rstandard(cherry.m1) ~ log(trees$Girth), las=1,
    ylab="Standardized deviance residual", xlab="log(Girth)" )
> scatter.smooth( rstandard(cherry.m1) ~ log(trees$Height), las=1,
    ylab="Standardized deviance residual", xlab="log(Height)" )
```

(The constant-information scale (Table 8.1) is the logarithmic scale for the gamma distribution, as used in the top left panel.) The plots appear approximately linear, but the variance of the residuals for smaller values of $\hat{\mu}$ may be less than for larger values of $\hat{\mu}$. The plot of z_i against $\hat{\eta}_i$ is also approximately linear (Fig. 8.5, bottom left panel) suggesting a suitable link function:

```
> z <- resid(cherry.m1, type="working") + cherry.m1$linear.predictor
> plot( z ~ cherry.m1$linear.predictor, las=1,
    xlab="Working responses, z", ylab="Linear predictor")
> abline(0, 1)                    # Adds line of equality
```

The plot of the partial residual (Fig. 8.5, bottom centre and right panels) suggest `Girth` and `Height` are included on the appropriate scale:

```
> termplot(cherry.m1, partial.resid=TRUE, las=1)
```

The line shown on each `termplot()` represents is the ideal relationship, so in both cases the plots suggest the model is adequate. □

8.7.4 Plots to Check the Random Component

The choice of random component for a GLM is usually based on an understanding of the data type: proportions of cases are modelled using binomial GLMs, and counts by a Poisson GLM, for example. However, Q–Q plots may be used to determine if the choice of distribution is appropriate [5]. Quantile residuals are used for these plots, since quantile residuals have an exact normal distribution (apart from sampling variability in estimating μ and ϕ) if the correct EDM has been chosen.

Example 8.10. Consider the model `cherry.m1` (Sect. 8.3) fitted to the cherry tree data (Example 3.14; data set: `trees`). A Q–Q plot of the quantile residuals (Fig. 8.6) shows that using a gamma GLM seems reasonable.

```
> qr.cherry <- qresid( cherry.m1 )
> qqnorm( qr.cherry, las=1 ); qqline( qr.cherry)
```

□

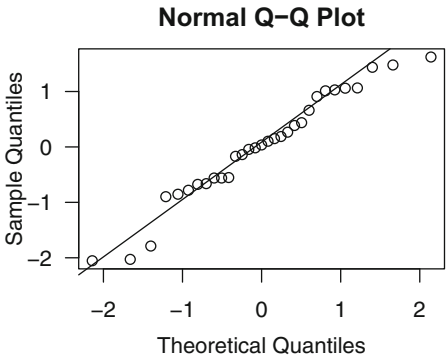

Fig. 8.6 The Q–Q plot of quantile residuals for Model `cherry.m1` fitted to the cherry tree data (Example 8.10)

8.8 Outliers and Influential Observations

8.8.1 Introduction

As for linear regression models, outliers are observations inconsistent with the rest of the data, and influential observations are outliers that substantially change the fitted model when removed from the data set. The tools used to identify outliers (Sect. 3.6.2) and influential observations (Sect. 3.6.3) in linear regression models are also used for GLMs, using results from the final step of the IRLS algorithm (Sect. 6.3), as discussed next.

8.8.2 Outliers and Studentized Residuals

For GLMs, as with linear regression models, outliers are identified as observations with unusually large residuals (positive or negative); the Q–Q plot is often convenient for doing this. Standardized deviance residuals are commonly used, though the use of quantile residuals are strongly encouraged for discrete data.

As for linear regression, *Studentizing* the residuals may also be useful (Sect. 3.6.2). For GLMs, computing Studentized deviance residuals requires refitting the original model n further times, when each observation is omitted one at a time. For each model without Observation i, the reduction in the deviance is computed. Fitting $n + 1$ models is necessary to do this, which is computationally expensive, and is avoided by approximating the Studentized residuals [18] by using

$$r_i'' = \text{sign}(y_i - \hat{\mu}_i)\sqrt{\frac{1}{\phi}\left(r_{D,i}^2 + \frac{h_i}{1-h_i}r_{P,i}^2\right)}.$$

If ϕ is unknown, estimate ϕ using

$$\bar{\phi}_{(i)} = \frac{D(y,\hat{\mu}) - r_{D,i}^2/(1-h_i)}{n - p' - 1},$$

which approximates the mean deviance estimate of ϕ in the model without Observation i (written $\bar{\phi}_{(i)}$). The approximate Studentized deviance residuals can be found in R using `rstudent()`, as used for linear regression models.

Example 8.11. Consider the cherry tree data and the model `cherry.m1` fitted in Sect. 8.3 (data set: `trees`). Compute the raw quantile residuals, raw deviance residuals, standardized deviance residuals, and Studentized residuals:

```
> library( statmod )   # To compute quantile residuals
> rs <- cbind( rD=resid(cherry.m1), "r'D"=rstandard(cherry.m1),
               "r''"=rstudent(cherry.m1), rQ=qresid(cherry.m1))
> head(rs)
           rD        r'D        r''         rQ
1  0.01922903  0.2603676  0.2537382  0.2665369
2  0.03298537  0.4508579  0.4408129  0.4380951
3  0.01295335  0.1803663  0.1756442  0.1882715
4 -0.01321397 -0.1698994 -0.1652566 -0.1380666
5 -0.04709620 -0.6267146 -0.6125166 -0.5606192
6 -0.04640051 -0.6285250 -0.6140386 -0.5519432
> apply( abs(rs), 2, max)   # The maximum absolute for each residual
      rD      r'D      r''       rQ
0.166763 2.197761 2.329122 2.053011
```

Since ϕ is small in this case, the saddlepoint approximation is suitable (Sect. 5.4.4), and the quantile, standardized and Studentized residuals are very similar. No large residuals exist. □

8.8.3 Influential Observations

Influential observations are outliers with high leverage. The measures of influence used for linear regression models, such as Cook's distance D, DFFITS, DFBETAS and the covariance ratio, are approximated for GLMs by using results from the final iteration of the IRLS algorithm (Sect. 6.7).

An approximation to Cook's distance for GLMs is

$$D \approx \left(\frac{r_P}{1-h}\right)^2 \frac{h}{\phi p'} = \frac{(r'_P)^2}{p'} \frac{h}{1-h} \tag{8.5}$$

as computed by the function `cooks.distance()` in R, where the Pearson estimator $\bar{\phi}$ of ϕ is used if it is unknown. Thus, Cook's distance is a combination of the size of the residual (measured by r'_P) and the leverage (measured by a monotonic function of h). Applying (8.5) for a linear regression model produces the same formula for Cook's distance given in (3.6) (p. 110).

DFBETAS, DFFITS, and the covariance ratio CR are computed using the same formulae as those used in linear regression (Sect. 3.6.3, p. 110), using the deviance residuals and using $\bar{\phi}_{(i)}$ in place of $s^2_{(i)}$. As for linear regression models, these statistics can be computed in R using `dffits()` (for DFFITS), `dfbetas()` (for DFBETAS), and `covratio()` (for CR). The function `influence.measures()` returns DFBETAS, DFFITS, CR, D, and the leverages h, flagging which are deemed influential (or high leverage in the case of h) according to the criteria in Sect. 3.6.3.

Example 8.12. For the model `cherry.m1` fitted to the cherry tree data (Sect. 8.3; data set: `trees`), influential observations are identified using `influence.measures()`:

```
> im <- influence.measures(cherry.m1); names(im)
[1] "infmat" "is.inf" "call"
> im$infmat <- round(im$infmat, 3 ); head( im$infmat )
  dfb.1_ dfb.l(G) dfb.l(H)  dffit cov.r cook.d   hat
1  0.015   -0.083    0.005  0.107 1.305  0.004 0.151
2  0.120   -0.082   -0.090  0.197 1.311  0.014 0.167
3  0.065   -0.021   -0.054  0.087 1.385  0.003 0.198
4 -0.011    0.021    0.004 -0.041 1.181  0.001 0.059
5  0.145    0.171   -0.170 -0.228 1.218  0.018 0.121
6  0.186    0.191   -0.212 -0.261 1.261  0.023 0.152
> colSums( im$is.inf )
  dfb.1_ dfb.l(G) dfb.l(H)    dffit    cov.r   cook.d      hat
       0        0        0        0        3        0        0
```

Three observations are identified as influential, but only by CR. Since none of the other measures identify these observations as influential, we should not be too concerned. Sometimes, plots of the influence statistics are useful (Fig. 8.7):

```
> cherry.cd <- cooks.distance( cherry.m1)
> plot( cherry.cd, type="h", ylab="Cook's distance", las=1)
> plot( dffits(cherry.m1), type="h", las=1, ylab="DFFITS")
> infl <- which.max(cherry.cd)  # The Observation number of largest D
> infl                          # Which observation?
18
18
> cherry.cd[infl]               # The value of D for that observation
       18
0.2067211
```

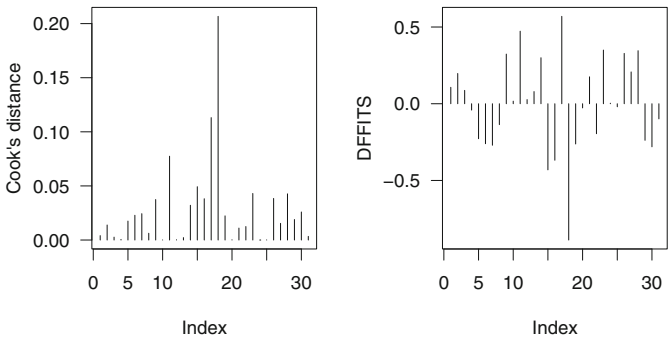

Fig. 8.7 Identifying influential observations for model `cherry.m1` fitted to the cherry tree data. Left panel: Cook's distance; right panel: DFFITS (Example 8.12)

The value of Cook's distance for Observation 18 is much larger than any others, but the observation is not identified as significantly influential. To demonstrate, we fit the model without Observation 18, then compare the estimated coefficients:

```
> cherry.infl <- update(cherry.m1, subset=(-infl) )
> coef(cherry.m1)
(Intercept)  log(Girth) log(Height)
  -6.691109    1.980412    1.132878
> coef(cherry.infl)
(Intercept)  log(Girth) log(Height)
  -7.209148    1.957366    1.267528
```

(The negative sign in `subset=(-infl)` *omits* Observation `infl` from the data set for this fit only.) The changes are not substantial, apart perhaps from the intercept. Contrast to the changes in the coefficients when another observation with a smaller value of D is omitted:

```
> cherry.omit1 <- update(cherry.m1, subset=(-1) ) # Omit Obs. 1
> coef(cherry.m1)
(Intercept)  log(Girth) log(Height)
  -6.691109    1.980412    1.132878
> coef(cherry.omit1)
(Intercept)  log(Girth) log(Height)
  -6.703461    1.986711    1.131840
```

The coefficients are very similar to those from model `cherry.m1` when Observation 1 is omitted: Observation 1 is clearly not influential. □

8.9 Remedies: Fixing Identified Problems

The techniques of Sects. 8.7 and 8.8 identify weaknesses in the fitted model. This section discusses possible remedies for these weaknesses. The following strategy can be adopted:

- If the responses are not independent (Sect. 8.7.2), use other methods, such as generalized estimating equations [7], generalized linear mixed models [2, 11] or spatial GLMs [4, 6]. These are beyond the scope of this book.
- Ensure the correct EDM is used (Sect. 8.7.3); that is, ensure the random component is adequate. For GLMs, the response data usually suggest the EDM:
  - Proportions of totals may be modelled using a binomial EDM (Chap. 9).
  - Count data may be modelled using a Poisson or negative binomial EDM (Chap. 10).

  - Positive continuous data may be modelled using a gamma or inverse Gaussian EDM (Chap. 11). In some cases, a Tweedie EDM may be necessary (Sect. 12.2.3).
  - Positive continuous data with exact zeros may be modelled using a Tweedie EDM (Sect. 12.2.4).

  Occasionally, a mean–variance relationship may be suggested that does not correspond to an EDM. In these cases, quasi-likelihood may be used (Sect. 8.10), or a different model may be necessary.
- Ensure the systematic component is correct (Sect. 8.7.3):
  - The link function may need to change. Changing the link function may be undesirable, because this changes the relationship between y and every explanatory variable, and because only a small number of link functions are useful for interpretability.
  - Important explanatory variables may be missing.
  - The covariates may need to be transformed. Partial residual plots may be used to determine if the covariates are included on the correct scale (and can be produced using `termplot()`). Simple transformations, polynomials in covariates (Sect. 3.10) or data-driven systematic components based on regression splines (Sect. 3.12) may be necessary in the model. R functions such as `poly()`, `bs()` and `ns()` are used for GLMs in the same way as for linear regression models.

Outliers and influential observations also may be remedied by making structural changes to the model. Sometimes, other strategies are needed to accommodate outliers and influential observations, including (under appropriate circumstances) omitting these observations; see Sect. 3.13.

Example 8.13. A suitable model for the cherry tree data was found in Sect. 8.3 (data set: `trees`). However, as an example we now consider residual plots from fitting a naive gamma GLM using the default (reciprocal) link function (Fig. 8.8):

```
> m.naive <- glm( Volume ~ Girth + Height, data=trees, family=Gamma)
> scatter.smooth( rstandard(m.naive) ~ log(fitted(m.naive)), las=1,
    xlab="Fitted values", ylab="Standardized residuals")
> scatter.smooth( rstandard(m.naive) ~ trees$Girth, las=1,
    xlab="Girth", ylab="Standardized residuals")
> scatter.smooth( rstandard(m.naive) ~ trees$Height, las=1,
    xlab="Height", ylab="Standardized residuals")
> eta <- m.naive$linear.predictor
> z <- resid(m.naive, type="working") + eta
> plot( z ~ eta, las=1,
    xlab="Linear predictor, eta", ylab="Working responses, z")
> abline(0, 1, col="grey")
> termplot(m.naive, partial.resid=TRUE, las=1)
```

(The constant-information scale (Table 8.1) is the logarithmic scale for the gamma distribution, as used in the top left panel.) The plots of r'_D against

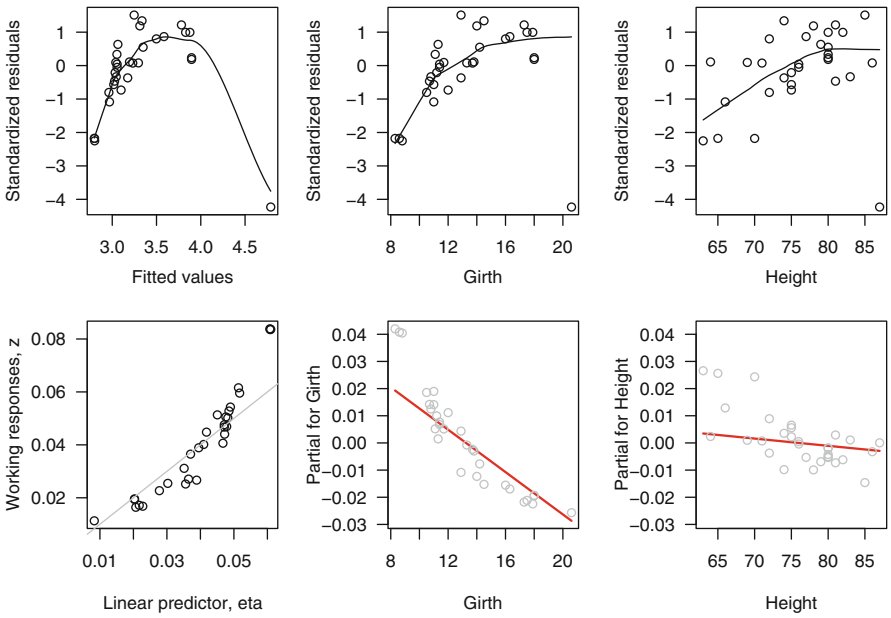

Fig. 8.8 Diagnostic plots for Model `m.naive` fitted to the cherry tree data. Top left panel: r'_D against $\log \hat{\mu}_i$; top centre panel: r'_D against `Girth`; top right panel: r'_D against `Height`; bottom left panel: z against $\hat{\eta}$; bottom centre panel: the partial residual plot for girth; bottom right panel: the partial residual plot for height (Example 8.13)

$\log \hat{\mu}$ (Fig. 8.8, top left panel) and r'_D against the covariates (top centre and top right panels) show an inadequate systematic component as shown by the trends and patterns. The plot of z_i against $\hat{\eta}_i$ (bottom left panel) suggests an incorrect link function. The partial residual plots (bottom centre and bottom right panels) suggest the covariates are included in the model incorrectly. In response to these diagnostic plots, consider the same model but with the more usual logarithmic link function (Fig. 8.9):

```
> m.better <- update(m.naive, family=Gamma(link="log"))
> scatter.smooth( rstandard(m.better) ~ log(fitted(m.better)), las=1,
    xlab="log(Fitted values)", ylab="Standardized residuals")
> scatter.smooth( rstandard(m.better) ~ trees$Girth, las=1,
    xlab="Girth", ylab="Standardized residuals")
> scatter.smooth( rstandard(m.better) ~ trees$Height, las=1,
    xlab="Height", ylab="Standardized residuals")
> eta <- m.better$linear.predictor
> z <- resid(m.better, type="working") + eta
> plot( z ~ eta, las=1, las=1,
    xlab="Linear predictor, eta", ylab="Working residuals, z")
> abline(0, 1, col="grey")
> termplot(m.better, partial.resid=TRUE, las=1)
```

The partial residual plots are much improved (Fig. 8.9, bottom centre and bottom right panels), and the plot of z_i against $\hat{\eta}$ (bottom left panel) suggests

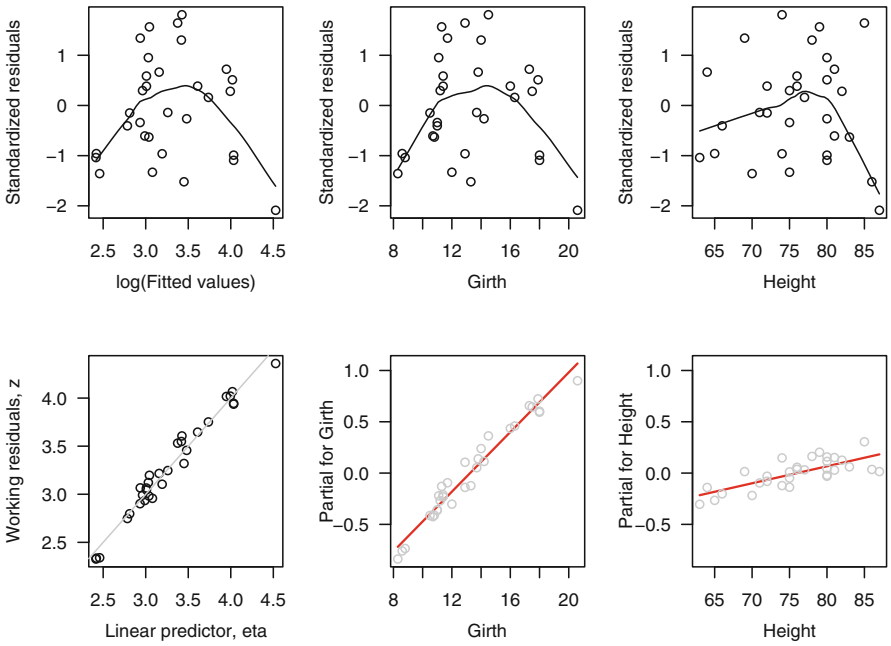

Fig. 8.9 Diagnostic plots for Model `m.better` fitted to the cherry tree data. Top left panel: r'_D against $\log \hat{\mu}_i$; top centre panel: r'_D against `Girth`; top right panel: r'_D against `Height`; bottom left panel: z against $\hat{\eta}$; bottom centre panel: the partial residual plot for girth; bottom right panel: the partial residual plot for height (Example 8.13)

the correct link function is used. However, the plots of r'_D against $\log \hat{\mu}$ (top left panel) and r'_D against the covariates (top centre and top right panels) still suggest a structural problem with the model.

In response to these diagnostic plots, model `cherry.m1` could be adopted. The residual plots from model `cherry.m1` then show an adequate model (Fig. 8.5, p. 310). In any case, `cherry.m1` has sound theoretical grounds, and should be preferred anyway. □

8.10 Quasi-Likelihood and Extended Quasi-Likelihood

In rare cases, sometimes the mean–variance relationship for a data set suggests a distribution that is not an EDM. However, the theory developed for GLMs is all based on distributions in the EDM family. However, note that for EDMs, the log-probability function has the neat derivative (Sect. 6.2)

$$\frac{\partial \log \mathcal{P}(\mu, \phi; y)}{\partial \mu} = \frac{y - \mu}{\phi V(\mu)}. \tag{8.6}$$

This relationship is used in fitting GLMs to find the estimates $\hat{\beta}_j$ (Sect. 6.2); the estimates of β_j and the standard errors $\text{se}(\hat{\beta}_j)$ are consistent given only the mean and variance information.

Motivated by these results, consider a situation where only the form of the mean and the variance are known, but no distribution is specified. Since no distribution is specified, no log-likelihood exists. However, analogous to (8.6), some *quasi-probability function* $\bar{\mathcal{P}}$ exists which satisfies

$$\frac{\partial \log \bar{\mathcal{P}}(y; \mu, \phi)}{\partial \mu} = \frac{y - \mu}{\phi V(\mu)}, \tag{8.7}$$

when only the variance function $V(\cdot)$ is known. On integrating,

$$\log \bar{\mathcal{P}}(y; \mu, \phi) = \int^{\mu} \frac{y - u}{\phi V(u)}\, du.$$

Suppose we have a series of observations y_i, for which we assume $\text{E}[y_i] = \mu_i$, and $\text{var}[y_i] = \phi V(\mu_i)/w_i$. Suppose a link-linear predictor for μ_i in terms of regression coefficients β_j, as for a GLM. Then the *quasi-likelihood function* (more correctly, the quasi-log-likelihood) is defined by

$$\mathcal{Q}(y; \mu) = \sum_{i=1}^{n} \log \bar{\mathcal{P}}(y_i; \mu_i, \phi/w_i).$$

The quasi-likelihood $\mathcal{Q}$ behaves like a log-likelihood function, but does not correspond to any probability function. As a result, the AIC and related statistics (Sect. 7.8) are not defined for quasi-models. In addition, quantile residuals (Sect. 8.3.4) are not defined for quasi-likelihood models since the quantile residuals require the CDF to be defined.

The unit deviance can be defined for quasi-likelihoods. First, notice that the unit deviance in (5.12) can be written as

$$\begin{aligned} d(y, \mu) &= 2\left\{t(y, y) - t(y, \mu)\right\} \\ &= 2\frac{\phi}{w}\left\{\log \mathcal{P}(y; y, \phi/w) - \log \mathcal{P}(y; \mu, \phi/w)\right\}. \end{aligned}$$

Using the quasi-likelihood in place of the log-likelihood,

$$\begin{aligned} d(y, \mu) &= 2\frac{\phi}{w}\left\{\log \bar{\mathcal{P}}(y; y, \phi/w) - \log \bar{\mathcal{P}}(y; \mu, \phi/w)\right\} \\ &= 2 \times \frac{\phi}{w} \int_{\mu}^{y} \frac{y - u}{\phi V(u)/w}\, du \\ &= 2 \int_{\mu}^{y} \frac{y - u}{V(u)}\, du. \end{aligned} \tag{8.8}$$

In this definition, the unit deviance depends only on the mean and variance. The total deviance is the (weighted) sum of the unit deviances as usual:

$$D(y, \mu) = \sum_{i=1}^{n} w_i d(y_i, \mu_i).$$

If there exists a genuine EDM for which $V(\mu)$ is the variance function, then the unit deviance and all other quasi-likelihood calculations derived from $V(\mu)$ reduce to the usual likelihood calculations for that EDM. This has the interesting implication that estimation and inference for GLMs depends only on the mean μ and the variance function $V(\mu)$. Since quasi-likelihood estimation is consistent, it follows that estimation for GLMs is robust against mis-specification of the probability distribution, because consistency of the estimates and tests is guaranteed as long as the first and second moment assumptions (means and variances) are correct.

Quasi-likelihood gives us a way to conduct inference when there is no EDM for a given mean–variance relationship. To specify a quasi-type model structure, write quasi-GLM($V(\mu)$; Link function), where $V(\mu)$ is the identifying variance function.

The most commonly-used quasi-models are for overdispersed Poisson-like or overdispersed binomial-like counts. These models vary the usual variance functions in some way, often by assuming a value for the dispersion ϕ greater than one, something which is not possible with the family of EDMs.

We discuss models for overdispersed Poisson-like counts, called quasi-Poisson models, at some length in Sect. 10.5.3. Quasi-Poisson models are specified in R using `glm()` with `family=quasipoisson()`. We discuss models for overdispersed binomial-like counts, called quasi-binomial models, at some length in Sect. 9.8. Quasi-binomial models are specified in R using `glm()` with `family=quasibinomial()`. Other quasi-models are specified in R using `family=quasi()`. For more details, see Sect. 8.13.

Inference for these quasi-models uses the same functions as for GLMs: `summary()` shows the results of the Wald tests, and `glm.scoretest()` in package **statmod** performs a score test. `anova()` performs the equivalent of likelihood ratio tests for comparing nested models by comparing the quasi-likelihood, which essentially compares changes in deviance. Analysis of deviance tests are based on the F-tests since ϕ is estimated for the quasi-models.

Example 8.14. For a Poisson distribution, $\text{var}[y] = \mu$ so that $V(\mu) = \mu$. However, in practice, often the variation in the data exceeds μ. This is called *overdispersion* (Sect. 10.5). One solution is to propose the variance structure $\text{var}[y] = \phi\mu$, but this variance structure does not correspond to any discrete EDM. Using quasi-likelihood,

$$\log \bar{\mathcal{P}}(y; \mu, \phi) = \int^{\mu} \frac{y - u}{\phi u}\, du = \frac{y \log \mu - \mu}{\phi}.$$

The same algorithms for fitting GLMs can be used to fit the model based on this quasi-likelihood. The unit deviance is

$$d(y,\mu) = 2\int_{\mu}^{y} \frac{y-u}{u}\,du = 2\left\{y\log\frac{y}{\mu} - (y-\mu)\right\},$$

identical to the unit deviance for the Poisson distribution (Table 5.1, p. 221). □

In defining the quasi-likelihood, we considered the derivative of $\log\bar{\mathcal{P}}$ with respect to μ but not ϕ. Hence the quasi-probability function is defined only up to terms not including μ. To deduce a complete quasi-probability function, the saddlepoint approximation can be used. This gives

$$\log\tilde{\mathcal{P}}(y;\mu,\phi) = -\frac{1}{2}\log\{2\pi\phi V(y)\} - \frac{d(y,\mu)}{2\phi},$$

which we call the *extended quasi-log-probability function*. Then

$$\mathcal{Q}^{+}(y;\mu,\phi/w) = \sum_{i=1}^{n}\log\tilde{\mathcal{P}}(y_i;\mu_i,\phi/w_i)$$

defines the *extended quasi-likelihood*. Solving $\mathrm{d}\mathcal{Q}^{+}(y;\mu,\phi/w)/\mathrm{d}\mu = 0$ shows that the solutions regarding μ are the same as for the quasi-likelihood and hence the log-likelihood. However, the extended quasi-likelihood has the advantage that solving $\mathrm{d}\mathcal{Q}^{+}(y;\mu,\phi/w)/\mathrm{d}\phi = 0$ produces the mean deviance estimate of ϕ.

The key use of extended quasi-likelihood is to facilitate the estimation of extended models which contains unknown parameters in the variance function $V()$, or which model some structure for the dispersion ϕ in terms of covariates.

8.11 Collinearity

As in linear regression (Sect. 3.14), *collinearity* occurs when at least some of the covariates are highly correlated with each other, implying they measure almost the same information.

As discussed in Sect. 3.14, collinearity causes no problems in prediction, but the parameter estimates $\hat{\beta}_j$ are hard to estimate with precision. Several equations may be found from which to compute the predictions, all of which may be effective but which produce different interpretations.

Collinearity is most easily identified by examining the correlations between the covariates. Any correlations greater than some (arbitrary) value, perhaps 0.7, are of concern. Other methods also exist for identifying collinearity. The same remedies apply as for linear regression (Sect. 3.14):

- Omitting some explanatory variables from the analysis.
- Combine explanatory variables in the model provided the combination makes sense.
- Collect more data, if there are observations that can be made that better distinguish the correlated covariates.
- Use special methods, such as ridge regression [17, §11.2], which are beyond the scope of this book.

Example 8.15. For the cherry tree data (Example 3.14; data set: `trees`), the two explanatory variables are correlated:

```
> cor( trees$Girth, trees$Height)
[1] 0.5192801
> cor( log(trees$Girth), log(trees$Height) )
[1] 0.5301949
```

Although correlated (that is, taller trees tend to have larger girths), collinearity is not severe enough to be a concern. □

8.12 Case Study

The noisy miner data [9] have been used frequently in this book (Example 1.5; `nminer`). The GLM fitted to model the number of noisy miners `Minerab` from the number of eucalypt trees `Eucs` is:

```
> library(GLMsData); data(nminer)
> nm.m1 <- glm( Minerab ~ Eucs, data=nminer, family=poisson)
> printCoefmat(coef(summary(nm.m1)))
             Estimate Std. Error z value  Pr(>|z|)
(Intercept) -0.876211   0.282793 -3.0984  0.001946 **
Eucs         0.113981   0.012431  9.1691 < 2.2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The diagnostic plots (Fig. 8.10) are informative:

```
> library(statmod)             # To find randomized quantile residuals
> qr <- qresid( nm.m1 )
> qqnorm(qr, las=1); qqline(qr)
> plot( qr ~ sqrt(fitted(nm.m1)), las=1 )
> plot( cooks.distance(nm.m1), type="h", las=1 )
> plot( hatvalues(nm.m1), type="h", las=1 )
```

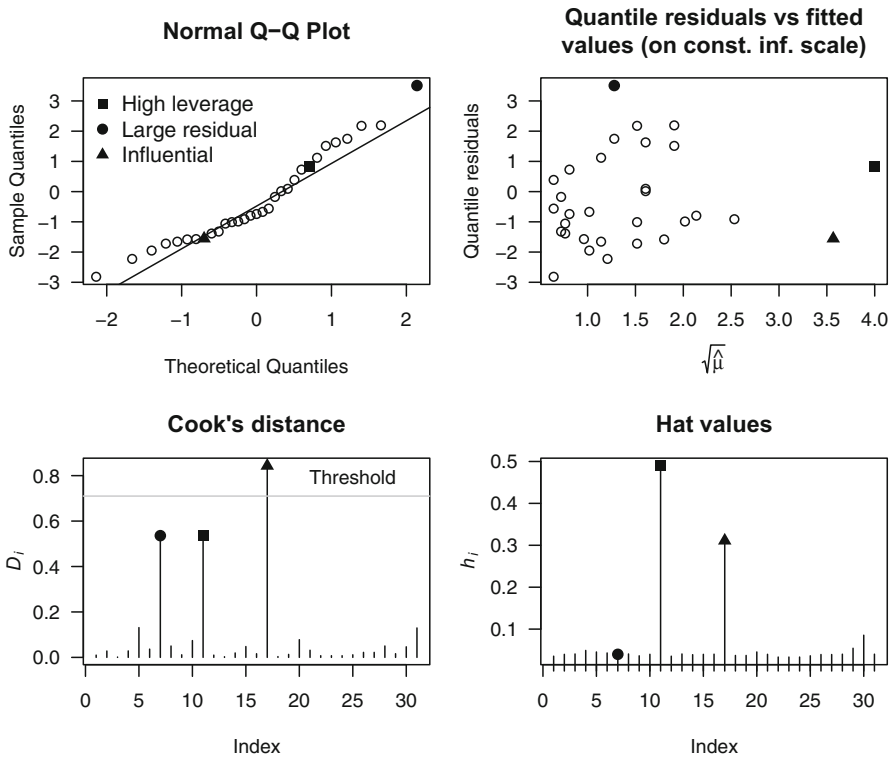

Fig. 8.10 Diagnostic plots for the GLM fitted to the noisy miner data. Top left: Q–Q plot of quantile residuals; top right: quantile residuals against $\sqrt{\hat{\mu}}$ (using the constant-information scale for the Poisson distribution); bottom left: Cook's distance, with the threshold for significance shown; bottom right: the leverages (Sect. 8.12)

We now locate the observations with the largest leverage, the largest absolute quantile residual, and the most influential observation:

```
> maxhat <- which.max( hatvalues(nm.m1) )         # Largest leverage
> maxqr <- which.max( abs(qr) )                   # Largest abs. residual
> maxinfl <- which.max( cooks.distance(nm.m1))    # Most influential
> c( MaxLeverage=maxhat, MaxResid=maxqr, MaxInfluence=maxinfl)
 MaxLeverage.11        MaxResid MaxInfluence.17
             11               7              17
```

Only Observation 17 is influential according to R's criterion (Sect. 3.6.3):

```
> which(influence.measures(nm.m1)$is.inf[,"cook.d"] )
17
17
```

In summary, Observation 11 (plotted with a filled square) has high leverage, but the residual is small and so it is not influential; Observation 7 (plotted with filled circle) has a large residual, but the leverage is small and so it is not

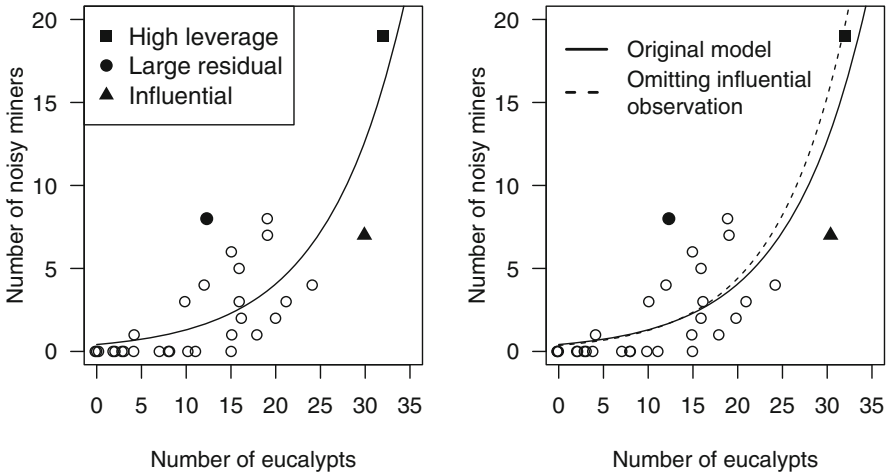

Fig. 8.11 Plots of the noisy miner data: left: the data plotted showing the location of three important observations; right: the data plotted with the fitted models, with and without the influential observation, Observation 17 (Sect. 8.12)

influential; Observation 17 (plotted with a filled triangle) has a reasonably large residual and leverage, and so it is influential.

Observe the changes in the regression coefficients after omitting Observation 17:

```
> nm.m2 <- glm( Minerab ~ Eucs, family=poisson, data=nminer,
           subset=(-maxinfl)) #  A negative index removes the obs.
> c( "Original model"=coef(nm.m1), "Without Infl"=coef(nm.m2))
Original model.(Intercept)        Original model.Eucs
                -0.8762114                  0.1139813
  Without Infl.(Intercept)          Without Infl.Eucs
                -1.0112791                  0.1247156
```

The two fitted models appear slightly different for transects with larger numbers of eucalypts (near Observation 17; Fig. 8.11, right panel):

```
> plot( Minerab ~ jitter(Eucs), data=nminer,
    xlab="Number of eucalypts", ylab="Number of noisy miners")
> newE <- seq( 0, 35, length=100)
> newM1 <- predict( nm.m1, newdata=data.frame(Eucs=newE), type="response")
> newM2 <- predict( nm.m2, newdata=data.frame(Eucs=newE), type="response")
> lines( newM1 ~ newE, lty=1); lines( newM2 ~ newE, lty=2)
```

These results suggest that the two transects with the largest number of eucalypts are important for understanding the data. Overdispersion may be an issue for these data, which we explore in Problem 10.10:

```
> c( deviance(nm.m1), df.residual(nm.m1) )
[1] 63.31798 29.00000
```

8.13 Using R for Diagnostic Analysis of GLMs

Residuals are computed in R for a fitted GLM, say `fit`, using:

- Pearson residuals r_P: `resid(fit, type="pearson")`.
- Deviance residuals r_D: `resid(fit)`, since deviance residuals are the default.
- Quantile residuals r_Q: `qresid(fit)` after loading package **statmod**.
- Partial residuals u_j: `resid(fit, type="partial")`.
- Working residuals e: `resid(fit, type="working")`.
- Response residuals $y - \hat{\mu}$: `resid(fit, type="response")`.
- Standardized deviance residuals r'_D: `rstandard(fit)`.
- Studentized deviance residuals r''_D: approximated using `rstudent(fit)`.

The longer form `residuals(fit)` is equivalent to `resid(fit)`. Each type of residual apart from `type="partial"` returns n values, one for each observation. Using `type="partial"` returns an array with n rows and a column corresponding to each β_j (apart from β_0).

Other useful R commands for diagnostics analysis, used in the same way as for linear regression models, are: `fitted(fit)` for producing fitted values; `hatvalues(fit)` for producing the leverages; `qqnorm()` for producing Q–Q plots of residuals; and `qqline()` for adding reference lines to Q–Q plots.

Measures of influence are computed for GLMs using the same R functions as for linear regression models:

- Cook's distance D: use `cooks.distance(fit)`.
- DFBETAS: use `dfbetas(fit)`.
- DFFITS: use `dffits(fit)`.
- Covariance ratio CR: use `covratio(fit)`.

All these measures of influence, together with the leverages h, are returned using `influence.measures(fit)`. Observations are flagged according to the criteria explained in Sect. 3.6.3 (p. 110).

Fitted GLMs can also `plot()`-ed (Sect. 3.16, p. 146). These commands produce four residual plots by default; see `?plot.lm`.

For remedying problems, the function `poly()` is used to create orthogonal polynomials of covariates, and `bs()` and `ns()` (both in the R package **splines**) for using regression splines in the systematic component.

Fit quasi-GLMs in R using the `glm()` function, but using specific `family` functions:

- `quasibinomial()` is used to fit quasi-binomial models. The default link function is the `"logit"` link function as for binomial GLMs. `"probit"`, `"cloglog"` (complementary log-log), `"cauchit"` and `"log"` links are also permitted, as for binomial GLMs (Sect. 9.8).
- `quasipoisson()` is used to fit quasi-Poisson models. The default link function is the `"log"` link function as for Poisson GLMs. `"identity"` and `"sqrt"` links are also permitted, as for Poisson GLMs (Sect. 10.5.3).

- `quasi()` is used to fit quasi-models more generally. Because this function is very general, any of the link functions provided by R are permitted (but may not all be sensible): `"identity"` (the default), `"logit"`, `"probit"`, `"cloglog"`, `"cauchit"`, `"log"`, `"identity"`, `"sqrt"` and `"1/mu^2"` are all permitted. Additional link functions can be defined using the `power()` function; for example, `link=power(lambda=1/3)` uses a link function of the form $\mu^{1/3} = \eta$. Using `lambda=0` is equivalent to using the logarithmic link function.
  To fit the quasi-models, the variance structure must also be defined, using for example, `family = quasi(link="log", variance="mu")`, which uses the variance function $V(\mu) = \mu$. The possible variance structures permitted for the `variance` are:
  - `"constant"`, the default, for which $V(\mu)$ is constant;
  - `"mu(1-mu)"` for which $V(\mu) = \mu(1 - \mu)$;
  - `"mu"` for which $V(\mu) = \mu$;
  - `"mu^2"` for which $V(\mu) = \mu^2$;
  - `"mu^3"` for which $V(\mu) = \mu^3$.

  Other variance functions can also be specified by writing appropriate R functions, but are rarely required and require extra effort and so are not discussed further.

The AIC is not shown in the model `summary()` for quasi-models, since the AIC is not defined for quasi-models. `summary()`, `anova()` and `glm.scoretest()` work as usual for quasi-models.

8.14 Summary

Chapter 8 discusses methods for identifying possible violations of assumptions in GLMs, and then remedying or ameliorating these problems.

The assumptions for GLMs are, in order of importance (Sect. 8.2):

- Lack of outliers: The model is appropriate for all observations.
- Link function: The correct link function $g()$ is used.
- Linearity: All important explanatory variables are included, and each explanatory variable is included in the linear predictor on the correct scale.
- Variance function: The correct variance function $V(\mu)$ is used.
- Dispersion: The dispersion parameter ϕ is constant.
- Independence: The responses y_i are independent of each other.
- Distribution: The responses y_i come from the specified EDM.

The main tool for diagnostic analysis is residuals. Pearson, deviance and quantile residuals can be used for GLMs (Sect. 8.3). Quantile residuals are highly recommended for discrete EDMs. Standardized or Studentized residuals are preferred as they have approximately constant variance (Sect. 8.6).

For GLMs, the leverages are the diagonal elements of the hat matrix $\mathrm{H} = \mathrm{W}^{1/2}\mathrm{X}(\mathrm{X}^T\mathrm{W}\mathrm{X})^{-1}\mathrm{X}^T\mathrm{W}^{1/2}$ (Sect. 8.4.2).

A strategy for diagnostic analysis of GLMs is (Sects. 8.7 and 8.9):

- Check for independence of the responses (Sect. 8.7.2). If the residuals show non-independence, use other methods.
- Plot residuals against $\hat{\mu}$ and residuals against each x_j (Sect. 8.7.3). If the variation is not constant, an incorrect EDM may have been used.
  If a trend exists, the systematic component may need changing: change the link function, add extra explanatory variables, or transform a covariates.
- To further examine the link function, plot z against $\hat{\eta}$ (Sect. 8.7.3).
- To determine if the source of the non-linearity is that covariate x_j is included on the incorrect scale, plot u_j against x_j (called a *component plus residual plot* or a *partial residual plot*) (Sect. 8.7.3).
- The choice of distribution can be checked using a Q–Q plot of quantile residuals (Sect. 8.7.4).

Outliers can be identified using Studentized residuals (Sect. 8.8). Outliers and influential observations also may be remedied by changes made to the model (Sect. 8.8). Influential observations can be identified using Cook's distance, DFFITS, DFBETAS or CR (Sect. 8.8).

Quasi-likelihood may be used when a suitable EDM cannot be identified, but information about the mean and variance is available (Sect. 8.10).

Collinearity occurs when at least some of the covariates are highly correlated with each other, implying they measure almost the same information (Sect. 8.11).

Problems

Selected solutions begin on p. 539.

8.1. Consider the Poisson distribution.

1. For $y = 0$, show that the smallest possible value of r_P is $-\sqrt{w\hat{\mu}}$.
2. For $y = 0$, show that the smallest possible value of r_D is $-\sqrt{2w\hat{\mu}}$.
3. For $y = 0$, what is the smallest value r_Q can take? Explain.
4. Comment on the normality of the residuals in light of the above results.

8.2. Show that the Pearson residuals for a gamma EDM cannot be less than $r_P = -1/\sqrt{w}$, but have no theoretical upper limit. Use these results to comment on the approximate normality of Pearson residuals for gamma EDMs. What range of values can be taken by deviance and quantile residuals?

8.3. Consider the binomial distribution.

1. Determine the deviance residuals for the binomial distribution.
2. In the extreme case $m = 1$, show that r_D will either take the value $\sqrt{2\log(1-\hat{\mu})}$ or $-\sqrt{2\log\hat{\mu}}$.

8.4. Use the R function `rpois()` to generate 1000 random numbers, say `y`, from a Poisson distribution with mean 1. Fit a Poisson GLM using the systematic component `y~1`. Then, plot the Q–Q plot of the residuals from this model using the Pearson, deviance and quantile residuals, and comment on the Q–Q plots produced using the different types of residuals. (Remember to generate more than one set of quantile residuals due to the added randomness.)

8.5. Consider the situation where the observations y come from distributions with known mean μ and known ϕ. Show that the Pearson residuals have mean zero and variance ϕ for any EDM.

8.6. The standardized deviance residual $r'_{D,i}$ is approximately the reduction in the residual deviance when Observation i is omitted from the data. Demonstrate this in R using the `trees` data as follows.

- Fit the model `cherry.m1` (Sect. 8.3.1). Compute the residual deviance, the Pearson estimate of ϕ, and the standardized deviance residuals from this model.
- Omit Observation 1 from `trees`, and refit the model. Call this model `cherry.omit1`.
- Compute the difference between the residual deviance for the full model `cherry.m1` and for model `cherry.omit1`. Show that this differences divided by the Pearson estimate of ϕ is approximately the standardized deviance residuals squared.

Repeat the above process for every observation i. At each iteration, call this model `cherry.omiti`. Then, compute the difference between the deviance for the full model `cherry.lm` and for model `cherry.omiti`. Show that these differences divided by ϕ are approximately the standardized residuals squared.

8.7. Consider the exponential distribution (4.37) defined for $y > 0$.

1. When $\mu = 3.5$ and $y = 1.5$, compute the Pearson, deviance and quantile residuals when the weights are all one.
2. When $\mu = 3.5$ and $y = 3.5$, compute the Pearson, deviance and quantile residuals when the weights are all one.
3. Comment on what the above shows.

8.8. Consider a transformation $A(y)$ of a response variable y.

1. Expand $A(y)$ about μ using the first two terms of the Taylor series to show that $A(y) - A(\mu) \approx A'(\mu)(y - \mu)$.

2. Using the previous result, compute the variance of both sides to show that
$$r_A = \frac{A(y) - A(\mu)}{A'(\mu)\sqrt{V(\mu)}},$$
called the Anscombe residual [10, 12], has a variance of ϕ approximately.
3. For GLMs, $A(t) = \int V(t)^{-1/3}(t)\,\mathrm{d}t$, where $V(\mu)$ is the variance function. Hence show that the Anscombe residuals for the Poisson distribution are
$$r_A = \frac{3(y^{2/3} - \mu^{2/3})}{2\mu^{1/6}}.$$
4. Compute the Anscombe residuals for the gamma and inverse Gaussian distributions.

8.9. Suppose a situation implies a variance function of the form $V(\mu) = \mu^2(1-\mu)^2$, where $0 < \mu < 1$ (for example, see [10, §9.2.4]). This variance function does not correspond to any known EDM.

1. Deduce the quasi-likelihood.
2. Deduce the unit deviance.

8.10. A study [16] counted the number of birds from four different species of seabirds in ten different quadrats in the Anadyr Strait (off the Alaskan coast) during summer, 1998 (Table 8.2; data set: `seabirds`). Because the responses are counts, a Poisson GLM may be appropriate.

1. Fit the Poisson GLM with a logarithmic link function, using the systematic component `Count ~ Species + factor(Quadrat)`.
2. Using the guidelines in Sect. 7.5 to determine when the Pearson and deviance residuals are expected to be adequate or poor.
3. Using this model, plot the deviance residuals against the fitted values, and also against the fitted values transformed to the constant-information scale. Using the plots, determine if the model is adequate.
4. Using the same model, plot the quantile residuals against the fitted values, and also against the fitted values transformed to the constant-information scale. Using the plots, determine if the model is adequate.
5. Comparing the plots based on the deviance and quantile residuals, which type of residual is easier to interpret?

8.11. Children were asked to build towers as high as they could out of cubical and cylindrical blocks [8, 14]. The number of blocks used and the time taken were recorded (data set: `blocks`). In this problem, only consider the number of blocks used y and the age of the child x.

In Problem 6.10, a GLM was fitted for these data. Perform a diagnostic analysis, and determine if the model is suitable.

Table 8.2 The number of each species of seabird counted in ten quadrats in the Anadyr Strait during summer, 1998 (Problem 8.10)

	Quadrat									
Species	1	2	3	4	5	6	7	8	9	10
Murre	0	0	0	1	1	0	0	1	1	3
Crested auklet	0	0	0	2	3	1	5	0	1	5
Least auklet	1	2	0	0	0	0	1	3	2	3
Puffin	1	0	1	1	0	0	3	1	1	0

8.12. Nambe Mills, Santa Fe, New Mexico [3, 15], is a tableware manufacturer. After casting, items produced by Nambe Mills are shaped, ground, buffed, and polished. In 1989, as an aid to rationalizing production of its 100 products, the company recorded the total grinding and polishing times and the diameter of each item (Table 5.3; data set: `nambeware`).

In Problem 6.11, a GLM was fitted to these data. Perform a diagnostic analysis, and determine if the model is suitable.

8.13. In Problem 3.24 (p. 157), a linear regression model was fitted to artificial data (data set: `triangle`), generated so that $\mu = \sqrt{x_1^2 + x_2^2}$; that is, x_1 and x_2 are the lengths of the sides of a right-angled triangle, and $\mathrm{E}[y] = \mu$ is the length of the hypotenuse (where some randomness has been added).

1. Based on the true relationships between the variables, write down the corresponding systematic component for fitting a GLM for modelling the hypotenuse. What link function is necessary?
2. Fit an appropriate GLM to the data, using the normal and gamma distributions to model the randomness. Which GLM is preferred?

References

[1] Box, G.E.P.: Science and statistics. Journal of the American Statistical Association **71**, 791–799 (1976)
[2] Breslow, N.E., Clayton, D.G.: Approximate inference in generalized linear mixed models. Journal of the American Statistical Association **88**(421), 9–25 (1993)
[3] Data Desk: Data and story library (DASL) (2017). URL http://dasl.datadesk.com
[4] Diggle, P.J., Tawn, J.A., Moyeed, R.A.: Model-based geostatistics. Applied Statistics **47**(3), 299–350 (1998)
[5] Dunn, P.K., Smyth, G.K.: Randomized quantile residuals. Journal of Computational and Graphical Statistics **5**(3), 236–244 (1996)

[6] Gotway, C.A., Stroup, W.W.: A generalized linear model approach to spatial data analysis and prediction. Journal of Agricultural, Biological and Environmental Statistics **2**(2), 157–178 (1997)
[7] Hardin, J.W., Hilbe, J.M.: Generalized Estimating Equations. Chapman and Hall/CRC, Boca Raton (2012)
[8] Johnson, B., Courtney, D.M.: Tower building. Child Development **2**(2), 161–162 (1931)
[9] Maron, M.: Threshold effect of eucalypt density on an aggressive avian competitor. Biological Conservation **136**, 100–107 (2007)
[10] McCullagh, P., Nelder, J.A.: Generalized Linear Models, second edn. Chapman and Hall, London (1989)
[11] McCulloch, C.E.: Generalized linear mixed models. Institute of Mathematical Statistics (2003)
[12] Pierce, D.A., Shafer, D.W.: Residuals in generalized linear models. Journal of the American Statistical Association **81**, 977–986 (1986)
[13] Pregibon, D.: Goodness of link tests for generalized linear models. Applied Statistics **29**(1), 15–24 (1980)
[14] Singer, J.D., Willett, J.B.: Improving the teaching of applied statistics: Putting the data back into data analysis. The American Statistician **44**(3), 223–230 (1990)
[15] Smyth, G.K.: Australasian data and story library (OzDASL) (2011). URL http://www.statsci.org/data
[16] Solow, A.R., Smith, W.: Detecting in a heterogenous community sampled by quadrats. Biometrics **47**(1), 311–317 (1991)
[17] Weisberg, S.: Applied Linear Regression. John Wiley and Sons, New York (1985)
[18] Williams, D.A.: Generalized linear models diagnostics using the deviance and single-case deletions. Applied Statistics **36**(2), 181–191 (1987)

Chapter 9
Models for Proportions: Binomial GLMs

We believe no statistical model is ever *final; it is simply a placeholder until a better model is found.*
Singer and Willett [22, p. 105]

9.1 Introduction and Overview

Chapters 5–8 develop the theory of GLMs in general. This chapter focuses on one specific GLM: the binomial GLM. The binomial GLM is the most commonly used of all GLMs. It is used to model proportions, where the proportions are obtained as the number of 'positive' cases out of a total number of independent cases. We first compile important information about the binomial distribution (Sect. 9.2), then discuss the common link functions used for binomial GLMs (Sect. 9.3), and the threshold interpretation of the link function (Sect. 9.4). We then discuss model interpretation in terms of odds (Sect. 9.5), and how binomial GLMs can be used to estimate the median effective dose ED50 (Sect. 9.6). The issue of overdispersion is then discussed (Sect. 9.8), followed by a warning about a potential problem with parameter estimation in binomial GLMs (Sect. 9.9). Finally, we explain why goodness-of-fit tests are not appropriate for binary data (Sect. 9.10).

9.2 Modelling Proportions

The outcome of many studies is a proportion y of a total number m: the proportion of individuals having a disease; the proportion of voters who vote in favour of a particular election candidate; the proportion of insects that die after being exposed to different doses of a poison. For all these examples, a binomial distribution may be an appropriate response distribution. In each case, the m individuals in each group are assumed to be independent, and each individual can be classified into one of two possible outcomes.

The binomial distribution has already been established as an EDM (Example 5.3), and binomial GLMs used in examples in previous chapters to

P. K. Dunn, G. K. Smyth, *Generalized Linear Models with Examples in R*,
Springer Texts in Statistics, https://doi.org/10.1007/978-1-4419-0118-7_9

develop the theory of GLMs. Useful information about the binomial distribution appears in Table 5.1 (p. 221). The probability function for a binomial EDM is

$$\mathcal{P}(y;\mu,m) = \binom{m}{my}\mu^{my}(1-\mu)^{m(1-y)} \tag{9.1}$$

where m is known and $\phi = 1$, and where $y = 0, 1/m, 2/m, \ldots 1$, and the expected proportion is $0 < \mu < 1$. To use the binomial distribution in a GLM, the prior weights w are set to the group totals m. The unit deviance for the binomial distribution is

$$d(y,\mu) = 2\left\{y\log\frac{y}{\mu} + (1-y)\log\frac{1-y}{1-\mu}\right\}.$$

When $y = 0$ or $y = 1$, the limit form of the unit deviance (5.14) is used. The residual deviance is $D(y,\hat{\mu}) = \sum_{i=1}^{n} m_i d(y_i,\hat{\mu}_i)$. By the saddlepoint approximation, $D(y,\hat{\mu}) \sim \chi^2_{n-p'}$ for a model with p' parameters in the linear predictor. The saddlepoint approximation is adequate if $\min\{m_i y_i\} \geq 3$ and $\min\{m_i(1-y_i)\} \geq 3$ (Sect. 7.5).

A binomial GLM is denoted GLM(binomial; link), and is specified in R using `family=binomial()` in the `glm()` call. Binomial responses may be specified in the `glm()` formula in one of three ways:

1. The response can be supplied as the observed proportions y_i, when the sample sizes m_i are supplied as the `weights` in the call to `glm()`.
2. The response can be given as a two-column array, the columns giving the numbers of successes and failures respectively in each group of size m_i. The prior weights `weights` do not need to be supplied (R computes the weights m as the sum of the number of successes and failures for each row).
3. The response can be given as a factor (when the first factor level corresponds to failures, and all others levels to successes) or as a logicals (either `TRUE`, which is treated as the success, or `FALSE`). The prior weights `weights` do not need to be supplied in this specification (and are set to one by default). This specification is useful if the data have one row for each observation (see Example 9.1). In this form, the responses are binary and the model is a Bernoulli GLM (see Example 4.6). While many of the model statistics are the same (Problem 9.14), there are some limitations with using this form (Sect. 9.10).

For binomial GLMs, the use of quantile residuals [5] is strongly recommended for diagnostic analysis (Sect. 8.3.4.2).

Example 9.1. An experiment running turbines for various lengths of time [19, 20] recorded the proportion of turbine wheels y_i out of a total of m_i turbines developing fissures (narrow cracks) (Table 9.1; Fig. 9.1; data set: `turbines`). A suitable model for the data may be a binomial GLM.

Table 9.1 The number of turbine wheels developing fissures and the number of hours they are run (Example 9.1)

Case i	Hours x_i	Turbines m_i	Prop. of fissures y_i	No. of fissures $m_i y_i$	Case i	Hours x_i	Turbines m_i	Prop. of fissures y_i	No. of fissures $m_i y_i$
1	400	39	0.0000	0	7	3000	42	0.2143	9
2	1000	53	0.0755	4	8	3400	13	0.4615	6
3	1400	33	0.0606	2	9	3800	34	0.6471	22
4	1800	73	0.0959	7	10	4200	40	0.5250	21
5	2200	30	0.1667	5	11	4600	36	0.5833	21
6	2600	39	0.2308	9					

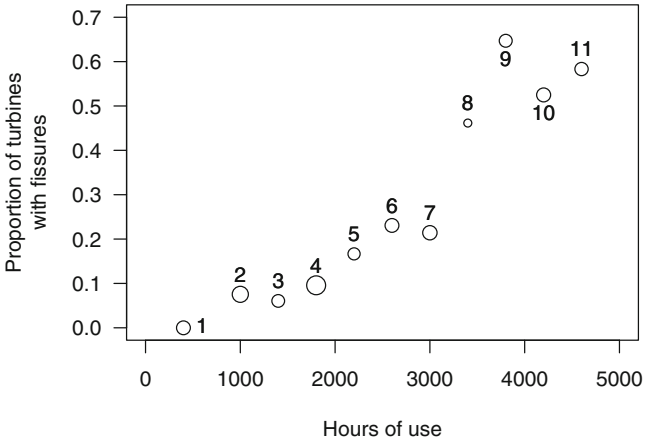

Fig. 9.1 The proportion of turbine wheels developing fissures plotted against the number of hours of use. Larger plotting symbols indicate proportions based on larger sample sizes. The numbers beside the points refer to the case number (Example 9.1)

For these data, the first and second forms of specifying the response are appropriate and equivalent:

```
> library(GLMsData); data(turbines)
> tur.m1 <- glm( Fissures/Turbines ~ Hours, family=binomial,
                 weights=Turbines, data=turbines)
> tur.m2 <- glm( cbind(Fissures, Turbines-Fissures) ~ Hours,
                 family=binomial, data=turbines)
> coef(tur.m1); coef(tur.m2)
  (Intercept)         Hours
-3.9235965551  0.0009992372
  (Intercept)         Hours
-3.9235965551  0.0009992372
```

To use the third form of data entry, the data would need to be rearranged so that each individual turbine was represented in its own line, hence having $\sum_{i=1}^{n} m_i = 432$ rows. □

9.3 Link Functions

Specific link functions are required for binomial GLMs to ensure that $0 < \mu < 1$. Numerous suitable choices are available. Three link functions are commonly used with the binomial distribution:

1. The *logit* (or logistic) link function, which is the canonical link function for the binomial distribution and the default link function in R:

$$\eta = \log \frac{\mu}{1-\mu} = \text{logit}(\mu). \tag{9.2}$$

   (R uses natural logarithms.) This link function is specified in R using `link="logit"`. A binomial GLM with a logit link function is often called a *logistic regression model*.
2. The *probit link function*: $\eta = \Phi^{-1}(\mu) = \text{probit}(\mu)$, where $\Phi(\cdot)$ is the CDF for the normal distribution. This link function is specified in R as `link="probit"`.
3. The *complementary log-log link function*: $\eta = \log\{-\log(1-\mu)\}$. This link function is specified in R as `link="cloglog"`.

In practice, the logit and probit link functions are very similar (Fig. 9.2). In addition, both are symmetric about $\mu = 0.5$, whereas the complementary log-log link function is not.

Two other less common link functions permitted in R for binomial GLMs are the `"cauchit"` and `"log"` links. The `"cauchit"` link function is based on the Cauchy distribution (see Sect. 9.4), but is rarely used in practice. The

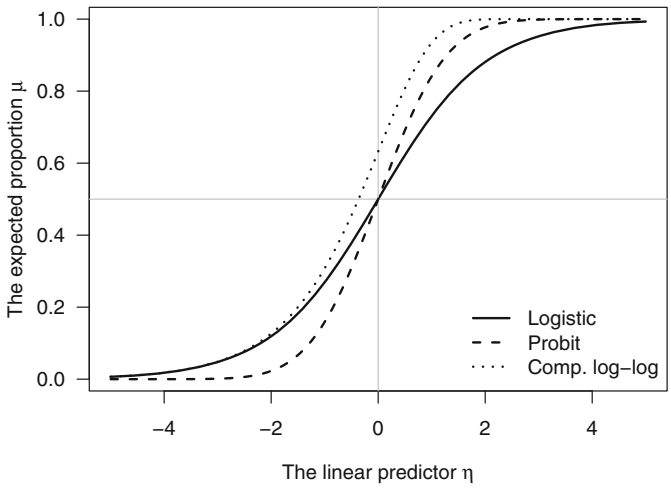

Fig. 9.2 Common link functions used with the binomial distribution: the logit, probit, and complementary log-log link functions (Sect. 9.3)

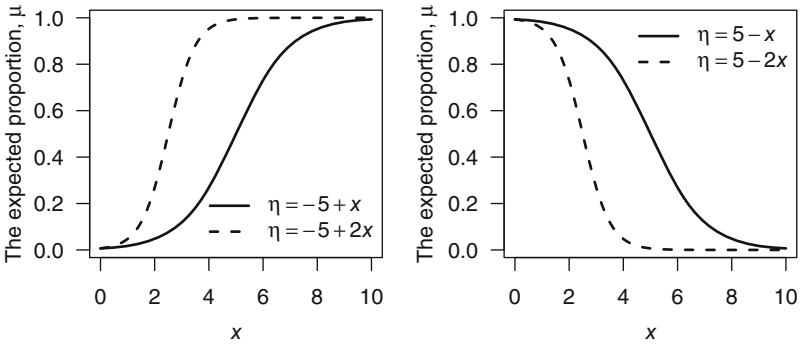

Fig. 9.3 The relationships between x and the predicted proportions μ for various linear predictors η using the logit link function, where $\text{logit}(\mu) = \eta$ (Sect. 9.3)

`"log"` link function is sometimes used for modelling risk ratios or relative risks. It is an approximation to the logit link when μ is small [16].

To understand the relationship between the explanatory variables and μ, consider the case of one explanatory variable where $\eta = \beta_0 + \beta_1 x$. Figure 9.3 shows the corresponding relationships between x and μ using the logit link function.

Example 9.2. For the turbine data (data set: `turbines`), we can fit binomial GLMs using the three common link functions, using the hours run-time as the explanatory variable:

```
> tr.logit  <- glm( Fissures/Turbines ~ Hours, data=turbines,
                    family=binomial, weights=Turbines)
> tr.probit <- update( tr.logit, family=binomial(link="probit") )
> tr.cll    <- update( tr.logit, family=binomial(link="cloglog") )
> tr.array <- rbind( coef(tr.logit), coef(tr.probit), coef(tr.cll))
> tr.array <- cbind( tr.array, c(deviance(tr.logit),
                     deviance(tr.probit), deviance(tr.cll)) )
> colnames(tr.array) <- c("Intercept", "Hours","Residual dev.")
> rownames(tr.array) <- c("Logit","Probit","Comp log-log")
> tr.array
             Intercept        Hours Residual dev.
Logit        -3.923597 0.0009992372     10.331466
Probit       -2.275807 0.0005783211      9.814837
Comp log-log -3.603280 0.0008104936     12.227914
```

The residual deviances are similar for the logit and probit GLMs, and slightly larger for the complementary log-log link function. The coefficients from the three models are reasonably different. However, the models themselves are very similar, as we can see by plotting the models. To do so, first set up a vector of values for the run-time:

```
> newHrs <- seq( 0, 5000, length=100)
```

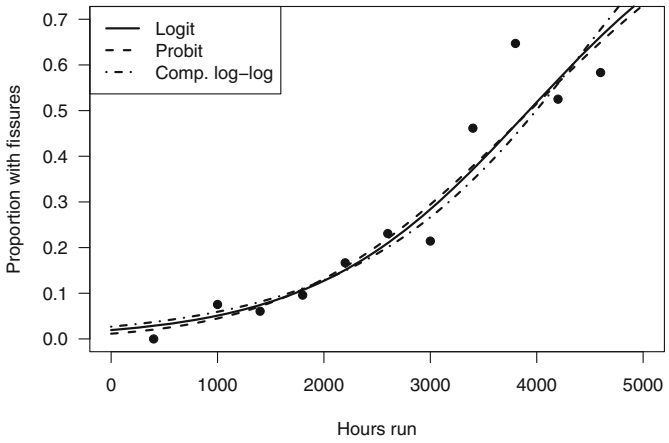

Fig. 9.4 The turbines data, showing the fitted binomial GLMs, using logistic, probit and complementary log-log link functions (Example 9.2)

Now, make predictions from these values using each model:

```
> newdf <- data.frame(Hours=newHrs)
> newP.logit  <- predict( tr.logit,  newdata=newdf, type="response")
> newP.probit <- predict( tr.probit, newdata=newdf, type="response")
> newP.cll    <- predict( tr.cll,    newdata=newdf, type="response")
```

The type of prediction is set as `"response"` because, by default, `predict()` returns the predictions on the linear predictor scale (that is, $\hat{\eta}$ is returned rather than $\hat{\mu}$). Now, plot these predictions using `lines()`, then add a legend (Fig. 9.4):

```
> plot( Fissures/Turbines ~ Hours, data=turbines, pch=19, las=1,
        xlim=c(0, 5000), ylim=c(0, 0.7),
        xlab="Hours run", ylab="Proportion with fissures")
> lines( newP.logit  ~ newHrs, lty=1, lwd=2)
> lines( newP.probit ~ newHrs, lty=2, lwd=2)
> lines( newP.cll    ~ newHrs, lty=4, lwd=2)
> legend("topleft", lwd=2, lty=c(1, 2, 4),
         legend=c("Logit","Probit","Comp. log-log"))
```

All three models produce similar predictions, which is not unusual. □

9.4 Tolerance Distributions and the Probit Link

The link functions can be understood using a threshold interpretation. In what follows, we show how the threshold interpretation applies for the probit link function, using the `turbines` data as the example.

Assume each individual turbine has a different tolerance beyond which it will develop fissures. As part of the natural variation in turbines, this

tolerance varies from turbine to turbine (but is fixed for any one turbine). Denote this tolerance level as t_i for turbine i; note that t_i is a continuous variable. Assume that t_i follows a normal distribution with mean tolerance τ_i, so that

$$\begin{cases} t_i \sim N(\tau_i, \sigma^2) \\ \tau_i = \beta_0' + \beta_1' x_i, \end{cases} \tag{9.3}$$

where x_i is the number of hours that turbine i is run. In this context, the normal distribution in (9.3) is called the *tolerance distribution*.

The variable of interest is whether or not the turbines develop fissures. Assume that turbines develop fissures if the tolerance level t_i of turbine i is less than some fixed tolerance threshold T. In other words, define

$$y_i = \begin{cases} 1 \text{ if } t_i \leq T, \text{ and the turbine develops fissures} \\ 0 \text{ if } t_i > T, \text{ and the turbine does not develop fissures.} \end{cases}$$

Then, the probability that turbine i develops fissures is

$$\mu_i = \Pr(y_i = 1) = \Pr(t_i \leq T) = \Phi\left(\frac{T - \tau_i}{\sigma}\right), \tag{9.4}$$

where $\Phi(\cdot)$ is the CDF of the standard normal distribution. We can write

$$\frac{T - \tau_i}{\sigma} = \frac{T - \beta_0' - \beta_1' x_i}{\sigma} = \beta_0 + \beta_1 x_i$$

with $\beta_0 = (T - \beta_0')/\sigma$ and $\beta_1 = -\beta_1'/\sigma$. Then (9.4) becomes

$$g(\mu_i) = \beta_0 + \beta_1 x_i$$

where $g()$ is the probit link function.

Other choices of the tolerance distribution lead to other link functions by a similar process (Table 9.2). The logit link function emerges as the link function when the logistic distribution is used as the tolerance distribution (Problem 9.4). The complementary log-log link function emerges as the link function when the extreme value (or Gumbel) distribution is used as the tolerance distribution. The cauchit link function assumes the threshold distribution is the Cauchy distribution. The logistic and normal tolerance distributions are both symmetric, and usually give similar results except for probabilities near zero or one. In contrast, the extreme value distribution is not symmetric, so the complementary log-log link function often gives somewhat different results than using the logit and probit link functions (Fig. 9.2). In principle, the CDF for any continuous distribution can be used as a basis for the link function.

Table 9.2 Tolerance distributions leading to link functions for binomial GLMs (Sect. 9.3)

Link function	Tolerance distribution	Distribution function
Logit	Logistic	$\mathcal{F}(y) = \exp(y)/\{1+\exp(y)\}$
Probit	Normal	$\mathcal{F}(y) = \Phi(y)$
Complementary log-log	Extreme value	$\mathcal{F}(y) = 1 - \exp\{-\exp(y)\}$
Cauchit	Cauchy	$\mathcal{F}(y) = \{\arctan(y) + 0.5\}/\pi$

9.5 Odds, Odds Ratios and the Logit Link

Using the logit link function with the binomial distribution produces a useful interpretation. To understand this interpretation, the concept of *odds* first must be understood. If event A has probability μ of occurring, then the *odds* of event A occurring is the ratio of the probability that A occurs to the probability that A does not occur: $\mu/(1-\mu)$. For example, if the probability that a turbine develops fissures is 0.6, the *odds* that a turbine develops fissures is $0.6/(1-0.6) = 1.5$. This means that the probability of observing fissures is 1.5 times greater than the probability of *not* observing a fissure (that is, $1.5 \times 0.4 = 0.6$). Clearly, using the logit link function in a binomial GLM is equivalent to modelling the logarithm of the odds (or the 'log-odds').

The binomial GLM using the logit function can be written as

$$\log(\text{odds}) = \beta_0 + \beta_1 x$$
$$\text{or equivalently} \quad \text{odds} = \exp(\beta_0)\{\exp(\beta_1)\}^x.$$

As x increases by one unit, the log-odds increase by linearly by an amount β_1. Alternatively, if x increases by one unit, the odds increase by a *factor* of $\exp(\beta_1)$. These interpretations in terms of the odds have intuitive appeal, and for this reason the logit link function is often preferred for the link function.

Example 9.3. For the turbines data (data set: `turbines`), the fitted logistic regression model (Example 9.1) has coefficients:

```
> coef(tr.logit)
  (Intercept)         Hours
-3.9235965551  0.0009992372
```

In this model, increasing `Hours` by one increases the odds of a turbine developing fissures by $\exp(0.0009992) = 1.001$. In this case, the interpretation is more useful if we consider increasing `Hours` by 1000 h. This increases the odds of a turbine developing fissures by $\exp(1000 \times 0.0009992) = 2.716$ times. Using the logistic regression model `tr.logit` assumes that the relationship between the run-time and the log-odds is approximately linear (Fig. 9.5):

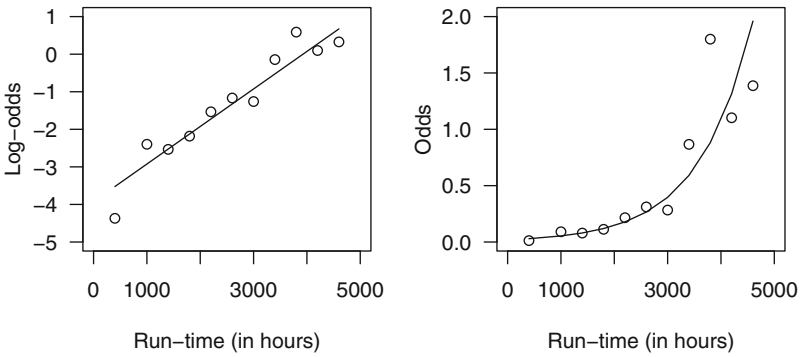

Fig. 9.5 The log-odds plotted against the run-time (left panel) and the odds plotted against the run-time (right panel) for the binomial logistic GLM fitted to the turbine data (Example 9.3)

```
> LogOdds <- predict( tr.logit ); odds <- exp( LogOdds )
> plot( LogOdds ~ turbines$Hours, type="l", las=1,
        xlim=c(0, 5000), ylim=c(-5, 1),
        ylab="Log-odds", xlab="Run-time (in hours)" )
> my <- turbines$Fissures; m <- turbines$Turbines
> EmpiricalOdds <- (my + 0.5)/(m - my + 0.5)   # To avoid log of zeros
> points( log(EmpiricalOdds) ~ turbines$Hours)
> #
> plot( odds ~ turbines$Hours, las=1, xlim=c(0, 5000), ylim=c(0, 2),
        type="l", ylab="Odds", xlab="Run-time (in hours)")
> points( EmpiricalOdds ~ turbines$Hours)
```

Note the use of empirical log-odds, adding 0.5 to both the numerator and denominator of the odds, so that the log-odds can be computed even when $y = 0$. □

Logistic regression models are often fitted to data sets that include factors as explanatory variables. In these situations, the concept of the *odds ratio* is useful to define. Consider the binomial GLM with systematic component

$$\log \frac{\mu}{1-\mu} = \text{log-odds} = \beta_0 + \beta_1 x,$$

where x is a dummy variable taking the values 0 or 1. From this equation, we see that the odds of observing a success when $x = 0$ is $\exp(\beta_0)$, and the odds of observing a success when $x = 1$ is $\exp(\beta_0 + \beta_1) = \exp(\beta_0)\exp(\beta_1)$. The ratio of these two odds is $\exp(\beta_1)$. This means that the odds of a success occurring when $x = 1$ is $\exp(\beta_1)$ times greater than when $x = 0$. This ratio is called the *odds ratio*, often written OR. When a number of factors are fitted as explanatory variables, each of the corresponding regression parameters β_j can be interpreted as odds ratios in a similar manner.

Table 9.3 The germination of two types of seeds for two root extracts. The number of seeds germinating *my* from *m* seeds planted is shown (Table 9.4)

O. aegyptiaco 75 seeds				*O. aegyptiaco* 73 seeds			
Bean extracts		Cucumber extracts		Bean extracts		Cucumber extracts	
my	*m*	*my*	*m*	*my*	*m*	*my*	*m*
10	39	5	6	8	16	3	12
23	62	53	74	10	30	22	41
23	81	55	72	8	28	15	30
26	51	32	51	23	45	32	51
17	39	46	79	0	4	3	7
		10	13				

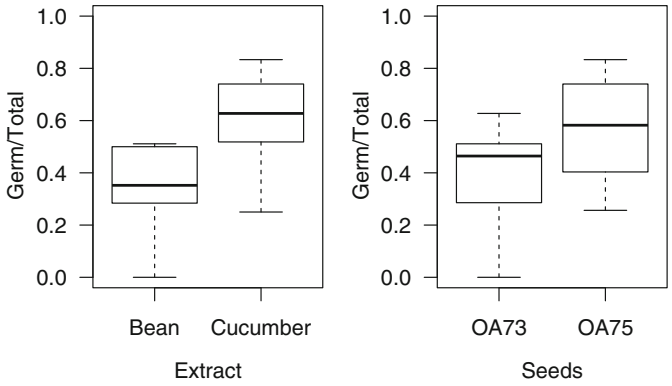

Fig. 9.6 The germination data: germination proportions plotted against extract type (left panel) and seed type (right panel) (Example 9.4)

Example 9.4. A study [3] of seed germination used two types of seeds and two types of root stocks (Table 9.3; data set: `germ`). A plot of the data (Fig. 9.6) shows possible relationships between the proportions of seeds germinating and both factors:

```
> data(germ); str(germ)
'data.frame':        21 obs. of  4 variables:
 $ Germ   : int  10 23 23 26 17 5 53 55 32 46 ...
 $ Total  : int  39 62 81 51 39 6 74 72 51 79 ...
 $ Extract: Factor w/ 2 levels "Bean","Cucumber": 1 1 1 1 1 2 2 2 2 2 ...
 $ Seeds  : Factor w/ 2 levels "OA73","OA75": 2 2 2 2 2 2 2 2 2 2 ...
> plot( Germ/Total ~ Extract, data=germ, las=1, ylim=c(0, 1) )
> plot( Germ/Total ~ Seeds,   data=germ, las=1, ylim=c(0, 1) )
```

The model with both factors as explanatory variables can be fitted:

```
> gm.m1 <- glm(Germ/Total ~ Seeds + Extract, family=binomial,
               data=germ, weights=Total)
> printCoefmat(coef(summary(gm.m1)))
```

```
                Estimate Std. Error z value  Pr(>|z|)
(Intercept)     -0.70048    0.15072 -4.6475 3.359e-06 ***
SeedsOA75        0.27045    0.15471  1.7482   0.08044 .
ExtractCucumber  1.06475    0.14421  7.3831 1.546e-13 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Recall the R output means that the R variable `Seeds` takes the value one for `OA75` and is zero for `OA73`. Likewise the R variable `Extract` takes the value one for `Cucumber` and is zero for `Bean`.

Note that

```
> exp( coef(gm.m1) )
    (Intercept)       SeedsOA75 ExtractCucumber
      0.4963454       1.3105554       2.9001133
```

This means that the odds of seed germination occurring using cucumber extracts is 2.900 times the odds of seed germination occurring using bean extracts. Similarly, the odds of seed germination occurring using *O. aegyptiaco* 75 seeds are 1.311 times the odds of seed germination occurring using *O. aegyptiaco* 73 seeds.

These data are explored later also (Example 9.8), where the interaction term is considered. □

9.6 Median Effective Dose, ED50

Binomial GLMs are commonly used to examine the relationship between the dose d of a drug or poison and the proportion y of insects (or plants, or animals) that survive. These models are called *dose–response models*. Associated with these experiments is the concept of the median effective dose, ED50: the dose of poison affecting 50% of the insects. Different fields use different names for similar concepts, such as median lethal dose LD50 or median lethal concentration LC50. Here, for simplicity, we use ED50 to refer to any of these quantities. The ED50 concept can be applied to other contexts also. By definition, $\mu = 0.5$ at the ED50.

For a binomial GLM using a logit link function, $\eta = \text{logit}(\mu) = 0$ when $\mu = 0.5$. Writing the linear predictor as $\eta = \beta_0 + \beta_1 d$ where d is the dose, then solving for the dose d shows that $\text{ED50} = -\hat{\beta}_0/\hat{\beta}_1$. More generally, the dose effective on any proportion ρ of the population, denoted $\text{ED}(\rho)$, is estimated by

$$\text{ED}(\rho) = \frac{g(\rho) - \beta_0}{\beta_1},$$

where $g()$ refers to the link function used in fitting the model. In Problem 9.2, formulae are developed for computing ED50 for the probit and complementary log-log link functions.

The function `dose.p()` in the R package **MASS** (which comes with R distributions) conveniently returns $\widehat{\text{ED}}(\rho)$ and the corresponding estimated standard error. The first input to `dose.p()` is the `glm()` object, and the second input identifies the two coefficients of importance: the coefficient for the intercept and for the dose (in that order). By default, these are assumed to be the first and second coefficients. The third input is ρ; by default $\rho = 0.5$, and so $\widehat{\text{ED}}50$ is returned by default.

Example 9.5. Consider the turbine data again (data set: `turbines`). The ED50 corresponds to the run time for which 50% of turbines would be expected to experience fissures:

```
> library(MASS)    # MASS  comes with R
> ED50s <- cbind("Logit"     = dose.p(tr.logit),
                 "Probit"    = dose.p(tr.probit),
                 "C-log-log" = dose.p(tr.cll))
> ED50s
            Logit   Probit C-log-log
p = 0.5: 3926.592 3935.197  3993.575
```

Running the turbines for approximately 3927 h would produce fissures in about 50% of the turbines (using the logistic link function model). All three link functions produce similar estimates of ED50, which seems reasonable based on Fig. 9.4 (p. 338). □

9.7 The Complementary Log-Log Link in Assay Analysis

A common problem in biology is to determine the proportion of cells or organisms of interest amongst a much larger population. For example, does a sample of tissue contain infective bacteria, and how many? Or what is the frequency of adult stem cells in a sample of tissue?

Suppose the presence of active particles can be detected by undertaking an assay. For example, the presence of bacteria might be detected by incubating the sample on an agar plate, and observing whether a bacterial culture grows. Or the presence of stem cells might be detected by transplanting cells into a host animal, and observing whether a new growth occurs. However, the same response is observed, more or less, regardless of the number of active particles in the original sample. A single stem cell would result in a new growth. When a growth is observed, we cannot determine directly whether there was one stem cell or many to start with.

Dilution assays are an experimental technique to estimate the frequency of active cells. The idea is to dilute the sample down to the point where some assays yield a positive result (so at least one active particles is present) and some yield a negative result (so no active particles are present).

The fundamental property of limiting dilution assays is that each assay results in a positive or negative result. Write μ_i for the probability of a

positive result given that the expected number of cells in the culture is d_i. If m_i independent cultures are conducted at dose d_i, then the number of positive results follows a binomial distribution.

Write λ for the proportion of active cells in the cell population, so that the expected number of active cells in the culture is λd_i. If the cells behave independently (that is, if there are no community effects amongst the cells), and if the cell dose is controlled simply by dilution, then the actual number of cells in each culture will vary according to a Poisson distribution. A culture will give a negative result only if there are no active cells in the assay. The Poisson probability formula tells us that this occurs with probability

$$1 - \mu_i = \exp(-\lambda d_i).$$

This formula can be linearized by taking logarithms of both sides, as

$$\log(1 - \mu_i) = -\lambda d_i \tag{9.5}$$

or, taking logarithms again,

$$\log\{-\log(1 - \mu_i)\} = \log \lambda + \log d_i. \tag{9.6}$$

This last formula is the famous complementary log-log transformation from Mather [18].

The proportion of active cells can be estimated by fitting a binomial GLM with a complementary log-log link:

$$g(\mu_i) = \beta_0 + \log d_i \tag{9.7}$$

where $\log d_i$ is an offset and $g()$ is the complementary log-log link function. The estimated proportion of active cells is then $\hat{\lambda} = \exp(\hat{\beta}_0)$.

In principle, a GLM could also have be fitted using (9.5) as a link-linear predictor, in this case with a log-link. However (9.6) is superior, because it leads to a GLM (9.7) without any constraints on the coefficient β_0.

As usual, a confidence interval is given by

$$\hat{\beta}_0 \pm z_{\alpha/2}\text{se}(\hat{\beta}_0)$$

where $\text{se}(\hat{\beta}_0)$ is the standard error of the estimate and $z_{\alpha/2}$ is the critical value of the normal distribution, e.g., $z = 1.96$ for a 95% confidence interval. To get back to the active cell frequency simply exponentiate and invert the estimate and the confidence interval: $1/\hat{\lambda} = \exp(-\hat{\beta}_0)$. Confidence intervals can be computed for $1/\lambda$, representing the number of cells required on average to obtain one responding cell.

The dilution assay model assumes that a single active cell is sufficient to achieve a positive result, so it is sometimes called the *single-hit* model (though other assumptions are possible [25]). One way to check this model is

Table 9.4 The average number of cells in each assay in which cells were transplanted in host mice, the number of assays at that cell number, and the number of assays giving a positive outcome, a milk gland outgrowth (Example 9.6)

Number of cells per assay	Number of assays	Number of outgrowths
15	38	3
40	6	6
60	17	13
90	8	6
125	12	9

to fit a slightly larger model in which the offset coefficient is not set to one:

$$g(\mu_i) = \beta_0 + \beta_1 \log d_i.$$

The correctness of the single-hit model can then be checked [10] by testing the null hypothesis H_0: $\beta_1 = 1$.

Example 9.6. Shackleton et al. [21] demonstrated the existence of adult mammary stem cells. They showed, for the first time, that a complete mammary milk producing gland could be produced in mice from a single cell. After a series of steps, they were able to purify a population of cells that was highly enriched for mammary stem cells, although stem cells were still a minority of the total.

The data (Table 9.4; data set: `mammary`) relate to a number of assays in which cells were transplanted into host mice. A positive outcome here consists of seeing a milk gland outgrowth, evidence that the sample of cells included as least one stem cell. The data give the average number of cells in each assay, the number of assays at that cell number, and the number of assays giving a positive outcome.

```
> data(mammary); mammary
  N.Cells N.Assays N.Outgrowths
1      15       38            3
2      40        6            6
3      60       17           13
4      90        8            6
5     125       12            9
> y <- mammary$N.Outgrowths / mammary$N.Assays
> fit <- glm(y~offset(log(N.Cells)), family=binomial(link="cloglog"),
             weights=N.Assays, data=mammary)
> coef(summary(fit))
             Estimate Std. Error   z value      Pr(>|z|)
(Intercept) -4.163625  0.1744346 -23.86925 6.391454e-126
> frequency <- 1/exp(coef(fit)); frequency
(Intercept)
   64.30418
```

The mammary stem cell frequency is estimated to be about 1 in 64 cells. A 95% confidence interval is computed as follows:

```
> s <- summary(fit)
> Estimate <- s$coef[, "Estimate"]
> SE <- s$coef[, "Std. Error"]
> z <- qnorm(0.05/2, lower.tail=FALSE)
> CI <- c(Lower=Estimate+z*SE, Estimate=Estimate, Upper=Estimate-z*SE)
> CI <- 1/exp(CI); round(CI, digits=1)
   Lower Estimate    Upper
    45.7     64.3     90.5
```

The frequency of stem cells is between 1/46 and 1/91. There is no evidence of any deviation from the single-hit model:

```
> fit1 <- glm(y~log(N.Cells), family=binomial(link="cloglog"),
              weights=N.Assays, data=mammary)
> anova(fit, fit1, test="Chisq")
Analysis of Deviance Table

Model 1: y ~ offset(log(N.Cells))
Model 2: y ~ log(N.Cells)
  Resid. Df Resid. Dev Df Deviance Pr(>Chi)
1         4     16.852
2         3     16.205  1   0.6468   0.4213
```

□

9.8 Overdispersion

For a binomial distribution, $\text{var}[y] = \mu(1-\mu)$. However, in practice the amount of variation in the data can exceed $\mu(1-\mu)$, even for ostensibly binomial-like data. This is called *overdispersion*. Underdispersion also occurs, but is less common.

Overdispersion has serious consequences for the GLM. It means that standard errors returned by the GLM are underestimated, and tests on the explanatory variables will generally appear to be more significant that warranted by the data, leading to overly complex models.

Overdispersion is detected by conducting a goodness-of-fit test, as described in Sect. 7.4. If the residual deviance and Pearson statistics are much greater than the residual degrees of freedom, then there is evidence of lack of fit. Lack of fit may be caused by an inadequate model, for example because important explanatory variables are missing from the model. However, if all relevant or possible explanatory variables are already included in the model, and the data has been checked for outliers that might inflate the residuals, but lack of fit remains, then overdispersion is the alternative interpretation.

Overdispersion means that the binomial model is incorrect in some respect. Overdispersion can arise from two major causes. The probabilities μ_i

are not constant between observations, even when all the explanatory variables are unchanged. Alternatively the m_i cases, of which observation y_i is a proportion, are not independent.

The first type of overdispersion can be modelled by a hierarchical model. Suppose that $m_i y_i$ follows a binomial distribution with m_i cases and success probability p_i. Suppose that the p_i is itself a random variable, with mean μ_i. Then

$$\mathrm{E}[y_i] = \mu_i$$

but

$$\mathrm{var}[y_i] > \mu_i(1-\mu_i)/m_i.$$

The greater the variability of p_i the greater the degree of overdispersion. A commonly-used model is to assume that p_i follows a beta distribution [3]. This leads to a beta-binomial model for y_i in which

$$\mathrm{var}[y_i] = \phi_i \mu_i(1-\mu_i)/m_i, \tag{9.8}$$

where ϕ_i depends on m_i and the parameters of the beta distribution.

More generally, overdispersion arises when the m_i Bernoulli cases, that make up observation y_i, are positively correlated. For example, positive cases may arrive in clusters rather than as individual cases. Writing ρ for the correlation between the Bernoulli trials leads to the same variance as the beta-binomial model (9.8) with $\phi_i = 1 + (m_i - 1)\rho$. If the m_i are approximately equal, or if ρ is inversely proportional to $m_i - 1$, then the ϕ_i will be approximately equal. In this case, both overdispersion models lead to variances

$$\mathrm{var}[y_i] = \phi \mu_i(1-\mu_i)/m_i,$$

which are larger but proportional to the variances under the binomial model. Note that overdispersion cannot arise for binary data with $m_i = 1$.

This reasoning leads to the idea of quasi-binomial models (Sect. 8.10). Quasi-binomial models keep the same variance function $V(\mu) = \mu(1-\mu)$ as binomial GLMs, but allow a general positive dispersion ϕ instead of assuming $\phi = 1$. The dispersion parameter is usually estimated by the Pearson estimator (Sect. 6.8.5). Quasi-binomial models do not correspond to any EDM, but the quasi-likelihood theory of Sect. 8.10 provides reassurance that the model will still yield consistent estimators provided that the variance function represents the correct mean–variance relationship. In particular, quasi-binomial models will give consistent estimators of the model coefficients under the beta-binomial or correlation models described above when the m_i are roughly equal. Even when the m_i are not equal, a quasi-binomial model is likely still preferable to assuming $\phi = 1$ when overdisperion is present.

The parameter estimates for binomial and quasi-binomial GLMs are identical (since the estimates $\hat{\beta}_j$ do not depend on ϕ), but the standard errors are different. The effect of using the quasi-binomial model is to inflate the standard error of the parameter estimates by $\sqrt{\phi}$, so confidence intervals and statistics for testing hypotheses tests will change.

A quasi-binomial model is fitted in R using `glm()` by using `family=quasibinomial()`. As for `family=binomial()`, the default link function for the `quasibinomial()` family is the `"logit"` link, while `"probit"`, `"cloglog"`, `"cauchit"`, and `"log"` are also permitted. Since the quasi-binomial model is not based on a probability model, the AIC is undefined.

Example 9.7. Machine turbines operate more or less independently, so it seems reasonable to suppose that independence between Bernoulli trials might hold for the turbines data (data set: `turbines`). Indeed neither the residual deviance nor the Pearson statistics show any evidence of overdispersion (using model `tr.logit` fitted in Example 9.1):

```
> c(Df = df.residual( tr.logit ),
    Resid.Dev = deviance( tr.logit ),
    Pearson.X2 = sum( resid(tr.logit, type="pearson")^2 ))
       Df  Resid.Dev Pearson.X2
 9.000000  10.331466   9.250839
```

Neither goodness-of-fit statistic is appreciably larger than the residual degrees of freedom. This data set does contain two small values of $m_i y_i$, but these are too few to change the conclusion even if the residuals for these observations were underestimated. □

Example 9.8. Example 9.4 (p. 341) discussed the seed germination for two types of seeds and two types of root stocks (data set: `germ`). Since seeds are usually planted together in common plots, it is highly possible that they might interact or be affected by common causes; in other words we might well expect seeds to be positively correlated, leading to overdispersion. We start by fitting a binomial GLM with `Extract` and `Seed` and their interaction as explanatory variables:

```
> gm.m1 <- glm( Germ/Total ~ Extract * Seeds, family=binomial,
                weights=Total, data=germ )
> anova(gm.m1, test="Chisq")
             Df Deviance Resid. Df Resid. Dev  Pr(>Chi)
NULL                             20     98.719
Extract       1   55.969        19     42.751 7.364e-14 ***
Seeds         1    3.065        18     39.686   0.08000 .
Extract:Seeds 1    6.408        17     33.278   0.01136 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
> df.residual(gm.m1)
[1] 17
```

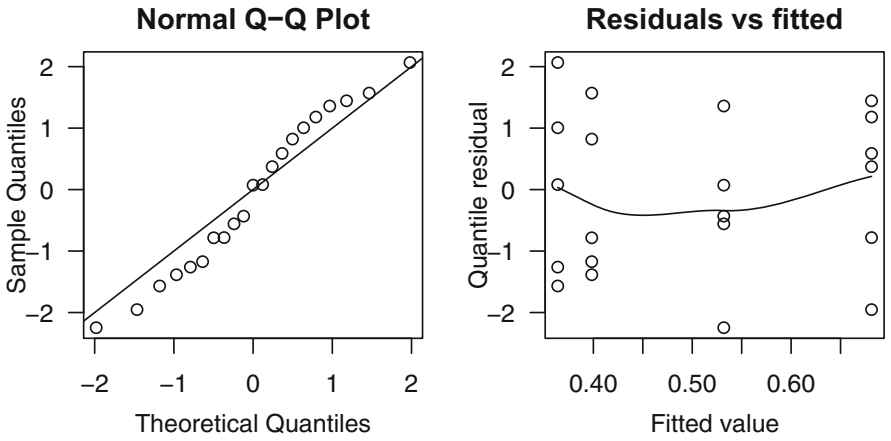

Fig. 9.7 Diagnostic plots after fitting a binomial GLM to the seed germination data (Example 9.8)

Despite the fact that the maximal possible explanatory model has been fitted, overdispersion is clearly present; the residual deviance is much larger than the residual degrees of freedom:

```
> c( deviance(gm.m1), df.residual(gm.m1) )
[1] 33.27779 17.00000
```

The Pearson statistic tells the same story:

```
> sum( resid(gm.m1, type="pearson")^2 ) # Pearson.X2
[1] 31.65114
```

There are no large residuals present that would suggest outliers (Fig. 9.7):

```
> library(statmod)
> qres <- qresid(gm.m1); qqnorm(qres, las=1); abline(0, 1)
> scatter.smooth( qres~fitted(gm.m1), las=1, main="Residuals vs fitted",
      xlab="Fitted value", ylab="Quantile residual")
```

The chi-square approximation to the goodness-of-fit statistics seems good enough. The data includes one observation (number 16) with $my = 0$ and other with $m - my = 1$ (number 6), but neither has a large enough residual to be responsible for the apparent overdispersion:

```
> qres[c(6, 16)]
[1]  1.180272 -1.172095
```

Finally, this a designed experiment, with nearly equal numbers of observations in each combination of the experimental factors `Extract` and `Seeds`, so influential observations cannot be an issue.

Having ruled out all alternative explanations, we accept that overdispersion is present and fit a quasi-binomial model:

```
> gm.od <- update(gm.m1, family=quasibinomial)
> anova(gm.od, test="F")
             Df Deviance Resid. Df Resid. Dev       F    Pr(>F)
NULL                            20     98.719
Extract       1   55.969        19     42.751 30.0610 4.043e-05 ***
Seeds         1    3.065        18     39.686  1.6462   0.21669
Extract:Seeds 1    6.408        17     33.278  3.4418   0.08099 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Note that F-tests are used for comparisons between quasi-binomial models. This follows because the dispersion ϕ is estimated (using the Pearson estimator by default). The quasi-binomial analysis of deviance suggests that only `Extract` is significant in the model, so germination frequency differs by root stock but not by seed type, unlike the binomial GLM which showed a significant `Extract` by `Seeds` interaction.

The binomial and quasi-binomial GLMs give identical coefficient estimates, but the standard errors from the quasi-binomial GLM are $\sqrt{\phi}$ times those from the binomial model:

```
> sqrt(summary(gm.od)$dispersion)
[1] 1.36449
> beta <- coef(summary(gm.m1))[,"Estimate"]
> m1.se <- coef(summary(gm.m1))[,"Std. Error"]
> od.se <- coef(summary(gm.od))[,"Std. Error"]
> data.frame(Estimate=beta, Binom.SE=m1.se,
             Quasi.SE=od.se, Ratio=od.se/m1.se)
                            Estimate  Binom.SE  Quasi.SE   Ratio
(Intercept)                -0.4122448 0.1841784 0.2513095 1.36449
ExtractCucumber             0.5400782 0.2498130 0.3408672 1.36449
SeedsOA75                  -0.1459269 0.2231659 0.3045076 1.36449
ExtractCucumber:SeedsOA75   0.7781037 0.3064332 0.4181249 1.36449
```

□

9.9 When Wald Tests Fail

Standard errors and Wald tests experience special difficulties when the fitted values from binomial GLMs are very close to zero or one. When the linear predictor includes factors, sometimes in practice there is a factor level for which the y_i are either all zero or all one. In this situation, the fitted values estimated by the model will also be zero or one for this level of the factor. This situation inevitably causes problems for standard errors and Wald tests, because at least one of the coefficients in the linear predictor must tend to infinity as the fitted model converges.

Suppose for example that the logit link function is used, so the fitted values are related to the linear predictor by

$$\hat{\mu} = \frac{\exp(\hat{\eta})}{1 + \exp(\hat{\eta})}. \tag{9.9}$$

Suppose also that the model includes just one explanatory variable x, so $\eta = \beta_0 + \beta_1 x$. The only way for $\hat{\mu}$ to be zero or one is for $\hat{\eta}$ to be $\pm\infty$. If $\hat{\mu} \to 0$, then $\hat{\eta} \to -\infty$, which implies $\hat{\beta}_0 \to -\infty$ and/or $\hat{\beta}_1 x \to -\infty$. In other words, one or both of the parameters must approach $\pm\infty$. If $\hat{\mu} \to 1$, then $\hat{\eta} \to \infty$ and a similar situation exists. The phenomenon is the same for other link functions.

When parameter estimates approach $\pm\infty$, the standard errors for those parameters must also approach $\pm\infty$, and Wald test statistics, which are ratios of coefficients to standard errors (Sect. 7.2.1), become very unreliable [23, p. 197]. In particular, the standard errors often tend to infinity faster than the coefficients themselves, meaning that the Wald statistic tends to zero, regardless of the true significance of the variable. This is called the *Hauck–Donner effect* [7].

Despite the problems with Wald tests, the likelihood ratio and score test usually remain quite serviceable in these situations, even when fitted values are zero or one. This is because the problem of infinite parameters is removable, in principle, by re-parametrising the model, and likelihood ratio and score tests are invariant to reparameterization. Wald tests are very susceptible to infinite parameters in the model because they are dependent on the particular parameterization used.

Example 9.9. A study [17] of the habitats of the noisy miner (a small but aggressive native Australian bird) recorded whether noisy miners were detected in various two hectare transects in buloke woodland patches (data set: `nminer`). Part of this data frame was discussed in Example 1.5 (p. 14), where models were fitted for the *number* of noisy miners.

Here we consider fitting a binomial GLM to model the presence of noisy miners in each buloke woodland patch (`Miners`). More specifically, we study whether the presence of noisy miners is impacted by whether or not the number of eucalypts exceeds 15 or not:

```
> data(nminer); Eucs15 <- nminer$Eucs>15
> m1 <- glm(Miners ~ Eucs15, data=nminer, family=binomial)
> printCoefmat(coef(summary(m1)))
             Estimate Std. Error z value Pr(>|z|)
(Intercept)   -0.84730    0.48795 -1.7364  0.08249 .
Eucs15TRUE    20.41337 3242.45694  0.0063  0.99498
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The Wald test results indicate that the explanatory variable is not significant: $P = 0.995$. Note the large standard error for `Eucs15`. Compare to the likelihood ratio test results:

```
> anova(m1, test="Chisq")
      Df Deviance Resid. Df Resid. Dev  Pr(>Chi)
NULL                     30     42.684
Eucs15  1    18.25        29     24.435 1.937e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The likelihood ratio test results indicate that the explanatory variable is highly significant: $P \approx 0$. Similarly, the score test results indicate that `Miners` is highly significant also:

```
> m0 <- glm(Miners ~ 1, data=nminer, family=binomial)
> z.score <- glm.scoretest(m0, Eucs15)
> P.score <- 2*(1-pnorm(abs(z.score))); c(z.score, P.score)
[1] 3.7471727820 0.0001788389
```

Despite the Wald test results, a plot of `Miners` against `Eucs15` (Fig. 9.8) shows an obvious relationship: in woodland patches with more than 15 eucalypts, noisy miners were *always* observed:

```
> plot( factor(Miners, labels=c("No","Yes")) ~ factor(Eucs15), las=1,
    ylab="Noisy miners present?", xlab="Eucalypts > 15", data=nminer)
> plot( Miners ~ Eucs, pch=ifelse(Eucs15, 1, 19), data=nminer, las=1)
> abline(v=15.5, col="gray")
```

The situation is exactly as described in the text, and an example of the Hauck–Donner effect. This means that the Wald test results are not trustworthy. When the number of eucalypts exceeds 15, all woodland patches in the sample have noisy miners, so $\hat{\mu} \to 1$. This is achieved as $\hat{\beta}_1 \to \infty$. The fitted probability when `Eucs15` is `TRUE` is one to computer precision:

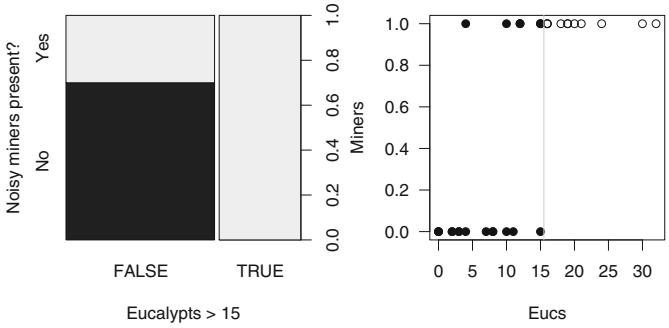

Fig. 9.8 The presence of noisy miners. Left panel: the presence of noisy miners as a function of whether 15 eucalypts are observed or not; right panel: the presence of noisy miners as a function of the number of eucalypts, showing the division at 15 eucalypts (Example 9.9)

```
> tapply(fitted(m1), Eucs15, mean)
FALSE  TRUE
  0.3   1.0
```

In this situation, the score or likelihood ratio tests must be used instead of the Wald test. □

9.10 No Goodness-of-Fit for Binary Responses

When $m_i = 1$ for all i, the binomial responses y_i are all 0 or 1; that is, the data are binary. In this case the residual deviance and Pearson goodness-of-fit statistics are determined entirely by the fitted values. This means that there is no concept of residual variability, and goodness-of-fit tests are not meaningful. For binary data, likelihood ratio tests and score tests should be used, making sure that p' is much smaller than n.

Example 9.10. In the `nminer` example in the previous section, the residual deviance is less than the residual degrees of freedom. This might be thought to suggest underdispersion, but it has no meaning. The size of the residual deviance is determined only by the sizes of the fitted values, and how far they are from zero and one. □

9.11 Case Study

An experiment [8, 13] exposed batches of insects to various deposits (in mg) of insecticides (Table 9.5; data set: `deposit`). The proportion of insects y killed after six days of exposure in each batch of size m is potentially a function of the dose of insecticide and the type of insecticide. The data are available in the R package **GLMsData**:

Table 9.5 The number of insects killed $z_i = y_i m_i$ out of a total of m_i insects, after three days exposure to different deposits of insecticides (Sect. 9.11)

	Amount of deposit (in mg)											
	2.00		2.64		3.48		4.59		6.06		8.00	
Insecticide	z_i	m_i	z_i	m_i	z_i	m_i	z_i	m_i	z_i	m_i	z_i	m_i
A	3	50	5	49	19	47	19	38	24	29	35	50
B	2	50	14	49	20	50	27	50	41	50	40	50
C	28	50	37	50	46	50	48	50	48	50	50	50

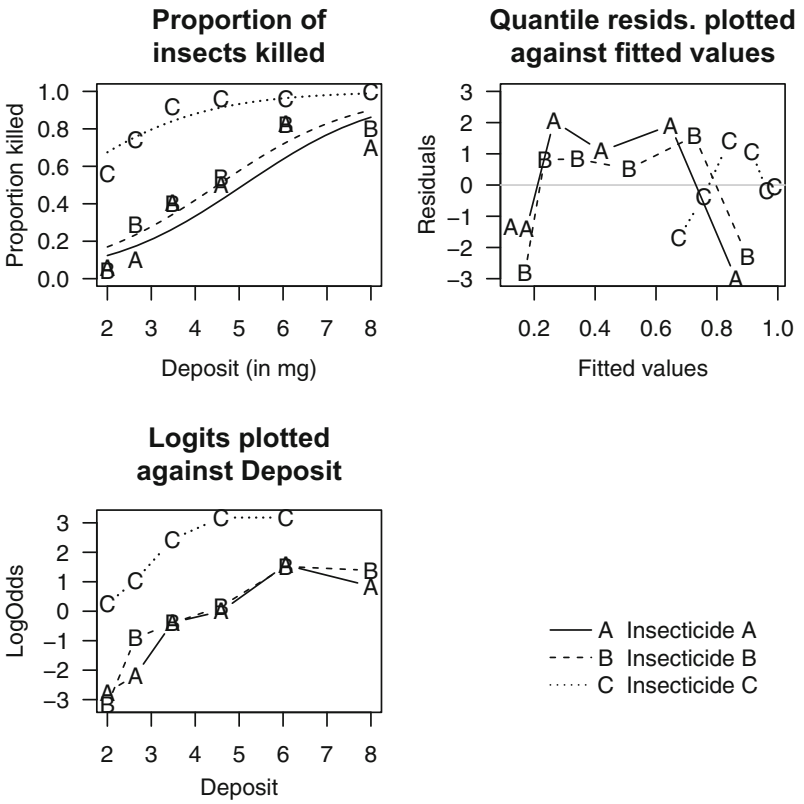

Fig. 9.9 The insecticide data. Top left panel: the data, showing the fitted model `ins.m1`; top right panel: a plot of the quantile residuals against the fitted values; bottom panel: the log-odds plotted against the deposit (Sect. 9.11)

```
> data(deposit); str(deposit)
'data.frame':        18 obs. of  4 variables:
 $ Killed     : int  3 5 19 19 24 35 2 14 20 27 ...
 $ Number     : int  50 49 47 38 29 50 50 49 50 50 ...
 $ Insecticide: Factor w/ 3 levels "A","B","C": 1 1 1 1 1 1 2 2 2 2 ...
 $ Deposit    : num  2 2.64 3.48 4.59 6.06 8 2 2.64 3.48 4.59 ...
```

A plot of the data (Fig. 9.9, p. 355, top left panel) shows insecticides A and B appear to have similar effects, while insecticide C appears different from A and B. The amount of deposit clearly is significant:

```
> deposit$Prop <- deposit$Killed / deposit$Number
> plot( Prop ~ Deposit, type="n", las=1, ylim=c(0, 1),
     data=deposit, main="Proportion of\ninsects killed",
     xlab="Deposit (in mg)", ylab="Proportion killed")
> points( Prop ~ Deposit, pch="A", subset=(Insecticide=="A"), data=deposit)
> points( Prop ~ Deposit, pch="B", subset=(Insecticide=="B"), data=deposit)
> points( Prop ~ Deposit, pch="C", subset=(Insecticide=="C"), data=deposit)
```

A model using the deposit amount and the type of insecticide as explanatory variables seems sensible:

```
> ins.m1 <- glm(Killed/Number ~ Deposit + Insecticide,
       family = binomial, weights = Number, data = deposit)
> coef(ins.m1)
 (Intercept)      Deposit InsecticideB InsecticideC
  -3.2213638    0.6316762    0.3695267    2.6880162
```

The fitted lines are shown in the top left panel of Fig. 9.9:

```
> newD <- seq( min(deposit$Deposit), max(deposit$Deposit), length=100)
> newProp.logA <- predict(ins.m1, type="response",
     newdata=data.frame(Deposit=newD, Insecticide="A") )
> newProp.logB <- predict(ins.m1, type="response",
     newdata=data.frame(Deposit=newD, Insecticide="B") )
> newProp.logC <- predict(ins.m1, type="response",
     newdata=data.frame(Deposit=newD, Insecticide="C") )
> lines( newProp.logA ~ newD, lty=1); lines( newProp.logB ~ newD, lty=2)
> lines( newProp.logC ~ newD, lty=3)
```

Before evaluating this model, we pause to demonstrate the estimation of ED50. The function `dose.p()` requires the name of the model, and the location of the coefficients that refer to the intercept and the slope. For insecticide A:

```
> dose.p(ins.m1, c(1, 2))
             Dose        SE
p = 0.5: 5.099708 0.2468085
```

For other insecticides, the intercept term is not contained in a single parameter. However, consider fitting an equivalent model:

```
> ins.m1A <- update( ins.m1, .~. - 1)   # Do not fit a constant term
> coef( ins.m1A )
     Deposit InsecticideA InsecticideB InsecticideC
   0.6316762   -3.2213638   -2.8518371   -0.5333477
```

Fitting the model without β_0 forces R to fit a model with separate intercept terms for each insecticide. Then, being careful to give the location of the intercept term first:

```
> ED50s <- cbind( dose.p(ins.m1A, c(2, 1)),    dose.p(ins.m1A, c(3, 1)),
                  dose.p(ins.m1A, c(4, 1)) )
> colnames(ED50s) <- c("Insect. A","Insect. B", "Insect. C"); ED50s
         Insect. A Insect. B Insect. C
p = 0.5:  5.099708  4.514714 0.8443372
```

Returning now to the diagnostic analysis of the model, close inspection of the top left panel in Fig. 9.9 shows model `ins.m1` is inadequate. The pattern in the residuals is easier to see in the top right panel:

```
> library(statmod)     # For  qresid()
> plot( qresid(ins.m1) ~ fitted(ins.m1), type="n", las=1, ylim=c(-3, 3),
     main="Quantile resids. plotted\nagainst fitted values",
     xlab="Fitted values", ylab="Residuals")
> abline(h = 0, col="grey")
> points( qresid(ins.m1) ~ fitted(ins.m1), pch="A", type="b", lty=1,
           subset=(deposit$Insecticide=="A") )
> points( qresid(ins.m1) ~ fitted(ins.m1), pch="B", type="b", lty=2,
           subset=(deposit$Insecticide=="B") )
> points( qresid(ins.m1) ~ fitted(ins.m1), pch="C", type="b", lty=3,
           subset=(deposit$Insecticide=="C"))
```

For each insecticide, the proportions are under-estimated at the lower and higher values of deposit. Plotting the log-odds against the deposit shows the relationship is not linear on the log-odds scale (Fig. 9.9, bottom panel):

```
> LogOdds <- with(deposit, log(Prop/(1-Prop)) )
> plot( LogOdds ~ Deposit, type="n", xlab="Deposit", data=deposit,
     main="Logits plotted\nagainst Deposit", las=1)
> points( LogOdds ~ Deposit, pch="A", type="b", lty=1,
           data=deposit, subset=(Insecticide=="A") )
> points( LogOdds ~ Deposit, pch="B", type="b", lty=2,
           data=deposit, subset=(Insecticide=="B") )
> points( LogOdds ~ Deposit, pch="C", type="b", lty=3,
           data=deposit, subset=(Insecticide=="C") )
```

As suggested earlier (Sect. 9.2), the *logarithm* of the dose is commonly used in dose–response models, so we try such a model (Fig. 9.10, top left panel):

```
> deposit$logDep <- log( deposit$Deposit )
> ins.m2 <- glm(Killed/Number ~ logDep + Insecticide - 1,
       family = binomial, weights = Number, data = deposit)
```

The ED50 estimates are on the log-scale for this model:

```
> ED50s <- cbind( dose.p(ins.m2, c(2, 1)),    dose.p(ins.m2, c(3, 1)),
                  dose.p(ins.m2, c(4, 1)) )
> colnames(ED50s) <- c("Insect. A","Insect. B", "Insect. C"); exp(ED50s)
         Insect. A Insect. B Insect. C
p = 0.5:  4.688232  4.154625  1.753202
```

The ED50 estimates are quite different from those computed using model `ins.m1A`.

While model `ins.m2` is an improvement over model `ins.m1`, proportions are still under-estimated for all types at the lower and higher values of deposit (Fig. 9.10, top right panel).

Plotting the log-odds against the logarithm of `Deposit` indicates that the log-odds are not constant, but are perhaps quadratic (Fig. 9.10, bottom panel; code not shown). Because of this, we try this model:

```
> ins.m3 <- glm(Killed/Number ~ poly(logDep, 2) + Insecticide,
       family = binomial, weights = Number, data = deposit)
```

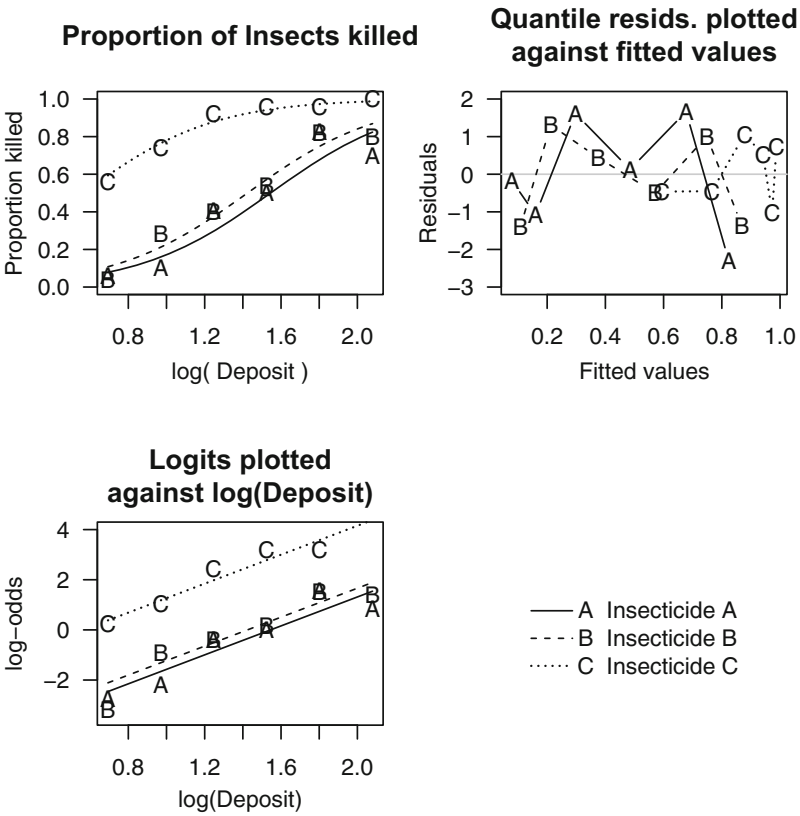

Fig. 9.10 The binomial GLMs for the insecticide data using the *logarithm of deposit* as an explanatory variable in model `ins.m2`. Top left panel: the log-odds against the logarithm of deposit showing the fitted models; top right panel: the quantile residuals plotted against the fitted values; bottom panel: the log-odds plotted against the logarithm of deposit (Sect. 9.11)

Now compare the two models involving `logDep`:

```
> anova( ins.m2, ins.m3, test="Chisq")
Analysis of Deviance Table

Model 1: Killed/Number ~ logDep + Insecticide - 1
Model 2: Killed/Number ~ poly(logDep, 2) + Insecticide
  Resid. Df Resid. Dev Df Deviance Pr(>Chi)
1        14     23.385
2        13     15.090  1   8.2949 0.003976 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

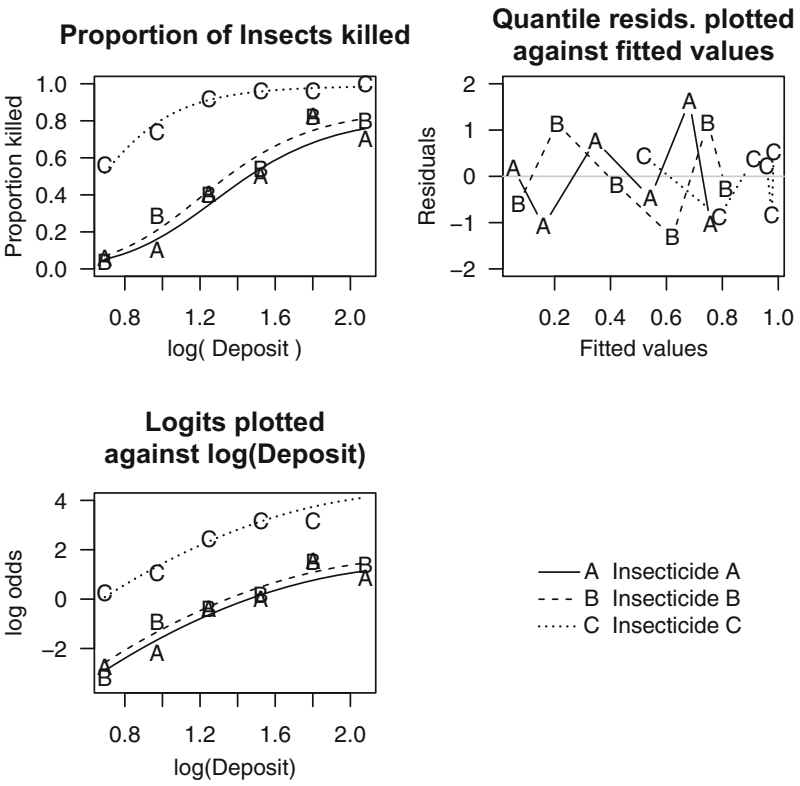

Fig. 9.11 The binomial GLMs for the insecticide data using the *square of the logarithm of deposit* as an explanatory variable in model `ins.m3`. Top left panel: the log-odds against the logarithm of deposit showing the fitted models; top right panel: the quantile residuals plotted against the fitted values; bottom panel: the log-odds plotted against the logarithm of deposit (Sect. 9.11)

This quadratic model is a statistically significantly improvement; the plotted lines appear much better (Fig. 9.11):

```
> newD <- seq( min(deposit$logDep), max(deposit$logDep), length=200)
> newProp4.logA <- predict(ins.m3, type="response",
     newdata=data.frame(logDep=newD, Insecticide="A") )
> newProp4.logB <- predict(ins.m3, type="response",
     newdata=data.frame(logDep=newD, Insecticide="B") )
> newProp4.logC <- predict(ins.m3, type="response",
     newdata=data.frame(logDep=newD, Insecticide="C") )
> lines( newProp4.logA ~ newD, lty=1); lines( newProp4.logB ~ newD, lty=2)
> lines( newProp4.logC ~ newD, lty=3)
```

The ED50 for this quadratic model cannot be computed using `dose.p` (because of the quadratic term in `logDep`), but can be found using simple algebra (Problem 9.3).

The structural changes to the model show that the model now is adequate (diagnostic plots not shown). No evidence exists to support overdispersion:

```
> c( deviance( ins.m3 ), df.residual( ins.m3 ) )
[1] 15.09036 13.00000
```

However, the saddlepoint approximation is probably not satisfactory and so this conclusion may not be entirely trustworthy:

```
> c( min( deposit$Killed ), min( deposit$Number - deposit$Killed) )
[1] 2 0
```

9.12 Using R to Fit GLMs to Proportion Data

Binomial GLMs are fitted in R using `glm()` with `family=binomial()`. The link functions `"logit"` (the default), `"probit"`, `"cloglog"` (the complementary log-log), `"log"` and `"cauchit"` are permitted. The response for a binomial GLM can be supplied in one of three ways:

- `glm( y ~ x, weights=m, family=binomial)`, where `y` are the observed proportions of successes in `m` trials.
- `glm( cbind(success, fail) ~ x, family=binomial)`, where `success` is a column of the number of successes, and `fail` is a column of the corresponding number of failures.
- `glm( fac ~ x, family=binomial)`, where `fac` is a factor. The first level denotes failure and all other levels denote successes, or where `fac` consists of logicals (either `TRUE`, which is treated as the success, or `FALSE`). Each individual in the study is represented by one row. This fits a Bernoulli GLM.

9.13 Summary

Chapter 9 considers fitting binomial GLMs. Proportions may be modelled using the binomial distribution (Sect. 9.2) where μ is the expected proportion where $0 < \mu < 1$, and $y = 0, 1/m, 2/m, \ldots, 1$. The prior weights are $w = m$. The residual deviance is suitably described by a $\chi^2_{n-p'}$ distribution if $\min\{m_i\mu_i\} \geq 3$ and $\min\{m_i(1-\mu_i)\} \geq 3$.

Commonly-used link functions are the logit (the canonical link function), probit and complementary log-log link functions (Sects. 9.3 and 9.4). Using the logistic link function enables an interpretation in terms of odds $\mu/(1-\mu)$ and odds ratios (OR) (Sect. 9.5).

The median effective dose (ED50) is the value of the covariates when the expected proportion is $\mu = 0.5$ (Sect. 9.6).

Overdispersion is observed when the variation in the data is greater than expected under the binomial model (Sect. 9.8). If overdispersion is observed, a quasi-binomial model may be fitted, which assumes $V(\mu) = \phi\mu(1-\mu)$. Overdispersion causes the estimates of the standard error to be underestimated and confidence intervals for parameters to be too narrow (Sect. 9.8).

For binomial GLMs, the Wald tests can fail in circumstances where one or more of the regression parameters tend to $\pm\infty$ (Sect. 9.9).

Problems

Selected solutions begin on p. 539.

9.1. Suppose the proportion y has the binomial distribution so that $z \sim \text{Bin}(\mu, m)$ where $z = my$ is the number of successes. Show that the transformation $y^* = \sin^{-1}\sqrt{y}$ produces approximately constant variance, by first expanding the transformation about μ using a Taylor series. (HINT: Follow the steps outlined in Sect. 5.8.)

9.2. Suppose that a given dose–response experiment records the dose of poison d and proportion y of insects out of m that are killed at each dose, such that the model has the systematic component $g(\eta) = \beta_0 + \beta_1 d$.

1. Show that the ED50 for such a model using a probit link function is $\text{ED50} = -\beta_0/\beta_1$.
2. Show that the ED50 for such a model using the complementary log-log link function is $\text{ED50} = \{\log(\log 2) - \beta_0\}/\beta_1$.
3. Show that the ED50 for such a model using the logarithmic link function is $\text{ED50} = (\log 0.5 - \beta_0)/\beta_1$.

9.3. Consider a binomial GLM using a logistic link function with systematic component $\text{logit}(\mu) = \beta_0 + \beta_1 \log x + \beta_2 (\log x)^2$.

1. For this model, deduce a formula for estimating the ED50.
2. Use this result to estimate the ED50 for the three insecticides using model `ins.m3` fitted in Sect. 9.11.

9.4. In Sect. 9.3 (p. 336), the probit binomial GLM was developed as a threshold model. Here consider using the *logistic distribution* with mean μ and variance σ^2 as the tolerance distribution. The logistic distribution has the probability function

$$\mathcal{P}(y; \mu, \sigma^2) = \frac{\pi \exp\{-(y-\mu)\pi/(\sigma\sqrt{3})\}}{\sigma\sqrt{3}\left[1 + \exp\{-(y-\mu)\pi/(\sigma\sqrt{3})\}\right]^2}$$

for $-\infty < y < \infty$, $-\infty < \mu < \infty$ and $\sigma > 0$.

Table 9.6 The logistic regression model fitted to data relating hypertension to sleep apnoea-hypopnoea (Problem 9.5)

Variable	$\hat{\beta}_j$	$\text{se}(\hat{\beta}_j)$
Intercept	−6.949	0.377
Age	0.805	0.0444
Sex	0.161	0.113
Body mass index	0.332	0.0393
Apnoea-hypopnoea index	0.116	0.0204

1. Show that the logistic distribution is not an EDM.
2. Determine the CDF for the logistic distribution.
3. Plot the density function and CDF for the logistic distribution with mean 0 and variance 1. Also plot the same graphs for the normal distribution with mean 0 and variance 1. Comment on the similarities and differences between the two probability functions.
4. Using the logistic distribution as the tolerance distribution, show that the threshold model in Sect. 9.4 corresponds to a binomial GLM with a logistic link function.

9.5. In a study [14] of the relationship between hypertension and sleep apnoea-hypopnoea (breathing difficulties while sleeping), a logistic regression model was fitted. The dependent variable was the presence of hypertension. The independent variables were dichotomized as follows: Age: 0 for 10 years or under, and 1 otherwise; sex: 0 for females, and 1 for males; body mass index: 0 if under $5\,\text{kg/m}^2$, and 1 otherwise; apnoea-hypopnoea index: 0 if fewer than ten events per hour of sleep, and 1 otherwise. Age, sex and body mass index are extraneous variables. The fitted model is summarized in Table 9.6.

1. Write down the fitted model.
2. Use a Wald test to test if $\beta_j = 0$ for each independent variable. Which variables seems important in the model?
3. Find 95% confidence intervals for each regression parameter.
4. Compute and interpret the odds ratios for each independent variable.
5. Predict the mean probability of observing hypertension in 30 year-old males with a BMI of $6\,\text{kg/m}^2$ who have an apnoea-hypopnoea index value of 5.

9.6. A study of stress and aggression in youth [15] measured the 'role stress' (an additive index from survey responses) and adolescent aggression levels (1 if the subject had engaged in at least one aggressive act as a youth, and 0 otherwise) in non-Hispanic whites. The response variable was aggression as an adult (1 if the subject had engaged in at least one aggressive act, and 0 otherwise). The fitted model is summarized in Table 9.7. (A number of other extraneous variables are also fitted, such as marital status and illicit drug use, but are not displayed in the table.)

Table 9.7 Two binomial GLMs fitted to the aggression data (Problem 9.6)

	Males		Females	
Variable	$\hat{\beta}_j$	se($\hat{\beta}_j$)	$\hat{\beta}_j$	se($\hat{\beta}_j$)
Intercept	0.45	0.40	−0.22	0.53
Role stress, *RS*	0.04	0.08	0.26	0.06
Adolescent aggression, *AA*	0.25	0.15	0.82	0.19
Interaction, *RS.AA*	0.23	0.17	−0.22	0.11
Residual deviance	57.40		121.67	
p'	13		13	
n	1323		1427	

1. Write down the two fitted models (one for males, one for females).
2. Use a Wald statistic to test if $\beta_j = 0$ for the interaction terms for both the male and female models. Comment.
3. The residual deviances for the fitted logistic regression models without the interaction term are 53.40 (males) and 117.82 (females). Use a likelihood ratio test to determine if the interaction terms are necessary in the models. Compare with the results of the Wald test.
4. Find 95% confidence intervals for both interaction terms.
5. Compute and interpret the odds ratios for *AA*.
6. Is overdispersion likely to be a problem for the models shown in the table?
7. Suppose a logistic GLM was fitted to the data with role stress, adolescent aggression, gender (G) and all the extraneous variables fitted to the model. Do you think the regression parameter for the three-way interaction *RS.AA.G* would be different from zero? Explain.

9.7. After the explosion of the space shuttle *Challenger* on January 28, 1986, a study was conducted [1, 4] to determine if previously-collected data about the ambient air temperature at the time of launch could have been used to foresee potential problems with the launch (Table 4.1; data set: `shuttles`). In Example 4.2, a model was proposed for these data.

1. Plot the data.
2. Fit and interpret the proposed model.
3. Perform a diagnostic analysis.
4. On the day of the *Challenger* launch, the forecast temperature was 31°F. What is the predicted probability of an O-ring failure?
5. What would the ED50 mean in this context? What would be a more sensible ED for this context?

9.8. An experiment [11] studied the survival of mice after receiving a test dose of culture with five different doses of antipneumococcus serum (in cc) (Table 9.8; data set: `serum`).

Table 9.8 The number of mice surviving exposure to pneumococcus after receiving a dose of antipneumococcus (Problem 9.8)

Dose (in cc)	Total number of mice	Number of survivors
0.000625	40	7
0.00125	40	18
0.0025	40	32
0.005	40	35
0.01	40	38

Table 9.9 The number of tobacco budworm moths (*Heliothis virescens*) out of 20 that were killed when exposed for three days to pyrethroid *trans*-cypermethrin (Problem 9.9)

	Pyrethroid dose (in µg)					
Gender	1	2	4	8	16	32
Male	1	4	9	13	18	20
Female	0	2	6	10	12	16

1. Fit and interpret a logistic regression model to the data with systematic component `Survivors/Number ~ 1 + log(Dose)`.
2. Examine the diagnostics from the above model.
3. Plot the data with the fitted lines, and the corresponding 95% confidence intervals.
4. Estimate the ED50.
5. Interpret your fitted model using the threshold interpretation for the link function.

9.9. The responses of the tobacco budworm *Heliothis virescens* to doses of pyrethroid *trans*-cypermethrin were recorded (Table 9.9; data set: `budworm`) [2, 23] from a small experiment. Twenty male and twenty female moths were exposed at each of six doses of the pyrethroid, and the number killed was recorded.

1. Plot survival proportions against dose, distinguishing male and female moths. Explain why using the logarithms of dose as a covariate is sensible given the values used for the pyrethroid dose.
2. Fit a binomial GLM to the data, ensuring a diagnostic analysis. Begin by fitting a model with a systematic component of the form `1 + log2(Dose) * Gender`, and show that the interaction term is not significant. Hence refit the model with systematic component `1 + log2(Dose) + Gender`.
3. Plot the fitted lines on the plot of the data (distinguishing between males and females) and comment on the suitability of the model.
4. Determine the odds ratio for comparing the odds of a male moth dying to the odds to a female moth dying.

Table 9.10 The gender of candidates in the 1992 British general election; M means males and F means females (Problem 9.10)

	Cons		Labour		Lib-Dem		Greens		Other	
Region	M	F	M	F	M	F	M	F	M	F
South East	101	8	84	25	81	28	42	15	86	27
South West	45	3	36	12	35	13	21	6	61	11
Great London	76	8	57	27	63	19	37	13	93	21
East Anglia	19	1	16	4	16	4	6	4	23	8
East Midlands	39	3	35	7	36	6	8	3	19	7
Wales	36	2	34	4	30	8	7	0	44	10
Scotland	63	9	67	5	51	21	14	6	87	17
West Midlands	50	8	43	15	49	9	11	4	30	5
Yorks and Humbers	51	3	45	9	42	12	22	3	22	6
North West	65	8	57	16	61	12	17	5	75	20
North	32	4	34	2	32	4	7	1	6	3

5. Determine if there is any evidence of a difference in the mortality rates between the male and female moths.
6. Determine estimates of the ED50 for both genders.
7. Determine the 90% confidence interval for the gender effect.

9.10. The *Independent* newspaper tabulated the gender of all candidates running for election in the 1992 British general election (Table 9.10; data set: `belection`) [6].

1. Plot the proportion of female candidates against the Party, and comment.
2. Plot the proportion of female candidates against the Region, and comment.
3. Find a suitable binomial GLM, ensuring a diagnostic analysis.
4. Is overdispersion evident?
5. Interpret the fitted model.
6. Estimate and interpret the odds of a female candidate running for the Conservative and Labour parties. Then compute the odds ratio of the Conservative party fielding a female candidate to the odds of the Labour party fielding a female candidate.
7. Determine if the saddlepoint approximation is likely to be suitable for these data.

9.11. A study [9, 12] of patients treated for nonmetastatic sarcoma obtained data on the gender of the patients, the presence of lymphocytic infiltration and any asteoid pathology. The treatment was considered a success if patients were disease-free for 3 years (Table 9.11). Here, consider the effect of lymphocytic infiltration on the proportion of success.

1. Plot the proportion of successes against gender. Then plot the proportion of successes against the presence or absence of lymphocytic infiltration. Comment on the relationships.

Table 9.11 The nonmetastatic sarcoma data (Problem 9.11)

Lymphotic infiltration	Gender	Osteoid pathology	Group size m	Number of successes my
Absent	Female	Absent	3	3
Absent	Female	Present	2	2
Absent	Male	Absent	4	4
Absent	Male	Present	1	1
Present	Female	Absent	5	5
Present	Female	Present	5	3
Present	Male	Absent	9	5
Present	Male	Present	17	6

2. Fit the binomial GLM using the gender and presence or absence of lymphocytic infiltration as explanatory variables. Show that the Wald test results indicate that the effect of lymphocytic infiltration is not significant.
3. Show that the likelihood ratio test indicates that the effect of lymphocytic infiltration is significant.
4. Show that the score test also indicates that the effect of lymphocytic infiltration is significant.
5. Explain the results from the three tests.

9.12. Chromosome aberration assays are used to determine whether or not a substance induces structural changes in chromosomes. One study [24] compared the results of two substances at various doses (Table 9.12). A large number of cells were sampled at each dose to see how many were aberrant.

1. Fit a binomial GLM to determine if there is evidence of a difference between the two substances.
2. Use the dose and the logarithm of dose as an explanatory variable in separate GLMs, and compare. Which is better, and why?
3. Compute the 95% confidence interval for the dose regression parameter, and interpret.
4. Why would estimation of the ED50 be inappropriate?

9.13. A study [17] of the habitats of the noisy miner (a small but aggressive native Australian bird; data set: `nminer`) recorded whether noisy miners were present in various two hectare transects in buloke woodland patches (`Miners`), and considered the following potential explanatory variables: the number of eucalypt trees (`Eucs`); the number of buloke trees (`Bulokes`); the area of contiguous remnant patch vegetation in which each site was located (`Area`); whether the area was grazed (`Grazed`: `1` means yes); whether shrubs were present in the transect (`Shrubs`: `1` means yes); and the number of pieces of fallen timber (`Timber`). Part of this data frame was discussed in Example 1.5 (p. 14), where models were fitted for the *number* of noisy miners.

Table 9.12 The number of aberrant cells for different doses of two substances (Problem 9.12)

Substance	Dose (in mg/ml)	No. cell samples	No. cells aberrant	Substance	Dose (in mg/ml)	No. cell samples	No. cells aberrant
A	0	400	3	B	0.0	400	5
A	20	200	5	B	62.5	200	2
A	100	200	14	B	125.0	200	2
A	200	200	4	B	250.0	200	4
				B	500.0	200	7

Fit a suitable logistic regression model for predicting the *presence* of noisy miners in two hectare transects in buloke woodland patches, ensuring an appropriate diagnostic analysis. Also estimate the number of eucalypt trees in which there is a greater than 90% chance of finding noisy miners.

9.14. In Example 9.4, data [3] were introduced regarding the germination of seeds, using two types of seeds and two types of root stocks (Table 9.3). An alternative way of entering the data is to record whether or not each individual seed germinates or not (data set: `germBin`).

1. Fit the equivalent model to that fitted in Example 9.4, but using data prepared as in the data file `germBin`. This model is based on using a Bernoulli distribution.
2. Show that both the Bernoulli and binomial GLMs produce the same values for the parameter estimates and standard errors.
3. Show that the two models produce different values for the residual deviance, but the same values for the deviance.
4. Show that the two models produce similar results from the sequential likelihood-ratio tests.
5. Compare the log-likelihoods for the binomial and Bernoulli distributions. Comment.
6. Explain why overdispersion cannot be detected in the Bernoulli model.

References

[1] Chatterjee, S., Handcock, M.S., Simonoff, J.S.: A Casebook for a First Course in Statistics and Data Analysis. John Wiley and Sons, New York (1995)
[2] Collett, D.: Modelling Binary Data. Chapman and Hall, London (1991)
[3] Crowder, M.J.: Beta-binomial anova for proportions. Applied Statistics **27**(1), 34–37 (1978)

[4] Dala, S.R., Fowlkes, E.B., Hoadley, B.: Risk analysis of the space shuttle: pre-Challenger prediction of failure. Journal of the American Statistical Association **84**(408), 945–957 (1989)

[5] Dunn, P.K., Smyth, G.K.: Randomized quantile residuals. Journal of Computational and Graphical Statistics **5**(3), 236–244 (1996)

[6] Hand, D.J., Daly, F., Lunn, A.D., McConway, K.Y., Ostrowski, E.: A Handbook of Small Data Sets. Chapman and Hall, London (1996)

[7] Hauck Jr., W.W., Donner, A.: Wald's test as applied to hypotheses in logit analysis. Journal of the American Statistical Association **72**, 851–853 (1977)

[8] Hewlett, P.S., Plackett, T.J.: Statistical aspects of the independent joint action of poisons, particularly insecticides. II Examination of data for agreement with hypothesis. Annals of Applied Biology **37**, 527–552 (1950)

[9] Hirji, K.F., Mehta, C.R., Patel, N.R.: Computing distributions for exact logistic regression. Journal of the American Statistical Association **82**(400), 1110–1117 (1987)

[10] Hu, Y., Smyth, G.K.: ELDA: Extreme limiting dilution analysis for comparing depleted and enriched populations in stem cell and other assays. Journal of Immunological Methods **347**, 70–78 (2009)

[11] Irwin, J.O., Cheeseman, E.A.: On the maximum-likelihood method of determining dosage-response curves and approximations to the median-effective dose, in cases of a quantal response. Supplement to the Journal of the Royal Statistical Society **6**(2), 174–185 (1939)

[12] Kolassa, J.E., Tanner, M.A.: Small-sample confidence regions in exponential families. Biometrics **55**(4), 1291–1294 (1999)

[13] Krzanowski, W.J.: An Introduction to Statistical Modelling. Arnold, London (1998)

[14] Lavie, P., Herer, P., Hoffstein, V.: Obstructive sleep apnoea syndrome as a risk factor for hypertension: Population study. British Medical Journal **320**(7233), 479–482 (2000)

[15] Liu, R.X., Kaplan, H.B.: Role stress and aggression among young adults: The moderating influences of gender and adolescent aggression. Social Psychology Quarterly **67**(1), 88–102 (2004)

[16] Lumley, T., Kronmal, R., Ma, S.: Relative risk regression in medical research: Models, contrasts, estimators, and algorithms. UW Biostatistics Working Paper Series 293, University of Washington (2006)

[17] Maron, M.: Threshold effect of eucalypt density on an aggressive avian competitor. Biological Conservation **136**, 100–107 (2007)

[18] Mather, K.: The analysis of extinction time data in bioassay. Biometrics **5**(2), 127–143 (1949)

[19] Myers, R.H., Montgomery, D.C., Vining, G.G.: Generalized Linear Models with Applications in Engineering and the Sciences. Wiley Series in Probability and Statistics. Wiley, Chichester (2002)

[20] Nelson, W.: Applied Life Data Analysis. John Wiley and Sons, New York (1982)
[21] Shackleton, M., Vaillant, F., Simpson, K.J., Sting, J., Smyth, G.K., Asselin-Labat, M.L., Wu, L., Lindeman, G.J., Visvader, J.E.: Generation of a functional mammary gland from a single stem cell. Nature **439**, 84–88 (2006)
[22] Singer, J.D., Willett, J.B.: Applied Longitudinal Data Analysis: Modeling Change and Event Occurrence. Oxford University Press, New York (2003)
[23] Venables, W.N., Ripley, B.D.: Modern Applied Statistics with S, fourth edn. Springer-Verlag, New York (2002). URL http://www.stats.ox.ac.uk/pub/MASS4
[24] Williams, D.A.: Tests for differences between several small proportions. Applied Statistics **37**(3), 421–434 (1988)
[25] Xie, G., Roiko, A., Stratton, H., Lemckert, C., Dunn, P., Mengersen, K.: Guidelines for use of the approximate beta-Poisson dose-response models. Risk Analysis **37**, 1388–1402 (2017)

Chapter 10
Models for Counts: Poisson and Negative Binomial GLMs

Poor data and good reasoning give poor results.
Good data and poor reasoning give poor results.
Poor data and poor reasoning give rotten results.
E. C. Berkeley [4, p. 20]

10.1 Introduction and Overview

The need to count things is ubiquitous, so data in the form of counts arise often in practice. Examples include: the number of alpha particles emitted from a source of radiation in a given time; the number of cases of leukemia reported per year in a certain jurisdiction; the number of flaws per metre of electrical cable. This chapter is concerned with counts when the individual events being counted are independent, or nearly so, and where there is no clear upper limit for the number of events that can occur, or where the upper limit is very much greater than any of the actual counts. We first compile important information about the Poisson distribution (Sect. 10.2), the distribution most often used with count data. Poisson regression, or models for count data described by covariates, has already been covered in Sect. 8.12 and elsewhere. In this chapter, we then focus on describing the models for three types of count data: models for count data described by covariates, models for rates (Sect. 10.3) and models for counts organized in tables (Sect. 10.4). Overdispersion is discussed in Sect. 10.5, including a discussion of negative binomial GLMs and quasi-Poisson models as alternative models.

10.2 Summary of Poisson GLMs

The distribution most often used for modelling counts is the Poisson distribution, which has the probability function

$$\mathcal{P}(y;\mu) = \frac{\exp(-\mu)\mu^y}{y!}$$

P. K. Dunn, G. K. Smyth, *Generalized Linear Models with Examples in R*,
Springer Texts in Statistics, https://doi.org/10.1007/978-1-4419-0118-7_10

for $y = 0, 1, 2, \ldots$, with expected counts $\mu > 0$. The Poisson distribution has already been established as an EDM (Example 5.2), and a Poisson GLM proposed for the noisy miner data in Example 1.5. Useful information about the Poisson distribution appears in Table 5.1. The unit deviance for the Poisson distribution is

$$d(y, \mu) = 2\left\{ y \log \frac{y}{\mu} - (y - \mu) \right\},$$

when the residual deviance is $D(y, \hat{\mu}) = \sum_{i=1}^{n} w_i d(y_i, \hat{\mu}_i)$, where w_i are the prior weights. When $y = 0$, the limit form of the unit deviance (5.14) is used. By the saddlepoint approximation, $D(y, \hat{\mu}) \sim \chi^2_{n-p'}$ where p' is the number of coefficients in the linear predictor. The approximation is adequate if $y_i \geq 3$ for all i (Sect. 7.5, p. 276).

The most common link function used for Poisson GLMs is the logarithmic link function (which is also the canonical link function), which ensures $\mu > 0$ and enables the regression parameters to be interpreted as having multiplicative effects. Using the logarithmic link function (`"log"` in R), the general form of a Poisson GLM is

$$\begin{cases} y \sim \text{Pois}(\mu) \\ \log \mu = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_p x_p. \end{cases} \tag{10.1}$$

The systematic component of (10.1) can be written as

$$\begin{aligned} \mu &= \exp(\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_p x_p) \\ &= \exp \beta_0 \times (\exp \beta_1)^{x_1} \times (\exp \beta_2)^{x_2} \times \cdots \times (\exp \beta_p)^{x_p}. \end{aligned}$$

This shows that the impact of each explanatory variable is multiplicative. Increasing x_j by one increases μ by *factor* of $\exp(\beta_j)$. If $\beta_j = 0$ then $\exp(\beta_j) = 1$ and μ is not related to x_j. If $\beta_j > 0$ then μ increases if x_j increases; if $\beta_j < 0$ then μ decreases if x_j increases.

Sometimes, the link functions `"identity"` ($\eta = \mu$) or `"sqrt"` ($\eta = \sqrt{\mu}$) are used with Poisson GLMs. A Poisson GLM is denoted GLM(Pois; link), and is specified in R using `family=poisson()` in the `glm()` call.

When the explanatory variables are all qualitative (that is, factors), the data can be summarized as a contingency table and the model is often called a *log-linear model* (Sect. 10.4). If any of the explanatory variables are quantitative (that is, covariates), the model is often called a *Poisson regression model*. Since Poisson regression has been discussed earlier (Sect. 8.12), we do not consider Poisson regression models further (but see Sect. 10.6 for a Case Study).

When the linear predictor includes an intercept term (as is almost always the case), and the log-link function is used, the residual deviance can be simplified to

$$D(y, \hat{\mu}) = 2 \sum_{i=1}^{n} w_i y_i \log(y_i / \hat{\mu}_i);$$

that is, the second term in the unit deviance can be dropped as it sums to zero (Problem 10.2). This identity will be used later to clarify the analysis of contingency tables.

For Poisson GLMs, the use of quantile residuals [12] is strongly recommended (Sect. 8.3.4.2).

10.3 Modelling Rates

The first context we consider is when the maximum number of events is known but large; that is, there is an upper bound for each count response, but the upper bound is very large. For such applications, the maximum number of events is usually representative of some population, and the response can be usefully viewed as a *rate* rather than just as a count. The size of each population needs to be specified to make comparisons meaningful. For example, consider comparing the number of people with a certain disease in various cities. The number of cases in each city may be useful information for planning purposes. However, quoting just the number of people with the disease in each city is an unfair comparison, as some cities have a far larger population than others. Comparing the number of people with the disease per unit of population (for example, per thousand people) is a fairer comparison. That is, the disease *rate* is often more suitable for modelling than the actual number of people with the disease.

In principle, rates can treated as proportions, and analysed using binomial GLMs, but Poisson GLMs are more convenient when the populations are large and the rates are relatively small, less than 1% say.

Example 10.1. As a numerical example, consider the number of incidents of lung cancer from 1968 to 1971 in four Danish cities (Table 10.1; data set: `danishlc`), recorded by age group [2, 26]. The *number* of cases of lung cancer in each age group is remarkably similar for Fredericia. However, using the number of cases does not accurately reflect the information in the data because five times as many people are in the 40–54 age group than in the over-75 age group. Understanding the data is enhanced by considering the *rate* of lung cancer, such as the number of lung cancer cases per unit of population. A plot of the cancer rates against city and age (Fig. 10.1) suggests the lung cancer rate may change with age:

```
> data(danishlc)
> danishlc$Rate <- danishlc$Cases / danishlc$Pop * 1000  # Rate per 1000
> danishlc$Age <- ordered(danishlc$Age,  # Ensure age-order is preserved
    levels=c("40-54", "55-59", "60-64", "65-69", "70-74", ">74") )
```

Table 10.1 Incidence of lung cancer in four Danish cities from 1968 to 1971 inclusive (Example 10.1)

	Fredericia		Horsens		Kolding		Vejle	
Age	Cases	Population	Cases	Population	Cases	Population	Cases	Population
40–54	11	3059	13	2879	4	3142	5	2520
55–59	11	800	6	1083	8	1050	7	878
60–64	11	710	15	923	7	895	10	839
65–69	10	581	10	834	11	702	14	631
70–74	11	509	12	634	9	535	8	539
Over 74	10	605	2	782	12	659	7	619

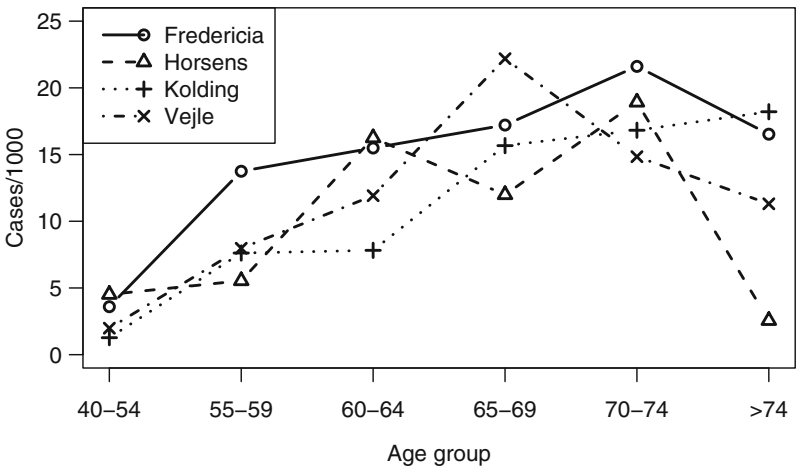

Fig. 10.1 The Danish lung cancer rates for various age groups in different cities (Example 10.1)

```
> danishlc$City <- abbreviate(danishlc$City, 1)   # Abbreviate city names
> matplot( xtabs( Rate ~ Age+City, data=danishlc), pch=1:4, lty=1:4,
           type="b", lwd=2, col="black", axes=FALSE, ylim=c(0, 25),
          xlab="Age group", ylab="Cases/1000")
> axis(side=1, at=1:6, labels=levels(danishlc$Age))
> axis(side=2, las=1); box()
> legend("topleft", col="black", pch=1:4, lwd=2, lty=1:4, merge=FALSE,
        legend=c("Fredericia", "Horsens", "Kolding", "Vejle") )
```

The R function `ordered()` informs R that the levels of factor `Age` have a particular order; without declaring `Age` as an ordered factor, `Age` is plotted with `">74"` as the first level. The plots show no clear pattern by city, but the lung cancer rate appears to grow steadily for older age groups for each city, then falls away for the `>74` age group. The lung cancer rate for Horsens in the `>74` age group seems very low.

An unfortunate side-effect of declaring `Age` as an ordered factor is that R uses polynomial contrasts for coding, which are not appropriate here (the

ordered categories are not equally spaced) and are hard to interpret anyway. To instruct R to use the familiar treatment coding for ordered factors, use:

```
> options(contrasts= c("contr.treatment", "contr.treatment"))
```

The first input tells R to use treatment coding for unordered factors (which is the default), and the second to use treatment coding for ordered factors (rather than the default `"contr.poly"`).

Define y_i as the observed number of lung cancers in group i where the corresponding population is T_i. The lung cancer *rate* per unit of population is y_i/T_i, and the expected rate is $\mathrm{E}[y_i/T_i] = \mu_i/T_i$, where μ_i possibly depends on the explanatory variables, and T_i is known. Using a logarithmic link function, the suggested systematic component is $\log(\mu_i/T_i) = \eta_i$. Dropping the subscript i, the model suggested for cancer rates is

$$\begin{cases} y \sim \mathrm{Pois}(\mu) \\ \log \mu = \log T + \beta_0 + \sum_{j=1}^{p} \beta_j x_j, \end{cases}$$

where the explanatory variables x_j are the necessary dummy variables required for the cities and age groups. The parameters β_j must be estimated, but no parameters need to be estimated for $\log T$. In other words, the term $\log T$ is an *offset* (Sect. 5.5.2).

Fit the model in R as follows, starting with the interaction model:

```
> dlc.m1 <- glm( Cases ~ offset( log(Pop) ) + City * Age,
                 family=poisson, data=danishlc)
> anova(dlc.m1, test="Chisq")
         Df Deviance Resid. Df Resid. Dev Pr(>Chi)
NULL                        23    129.908
City      3    3.393        20    126.515  0.33495
Age       5  103.068        15     23.447  < 2e-16 ***
City:Age 15   23.447         0      0.000  0.07509 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

We decide to retain only `Age` in the model.

```
> dlc.m2 <- update(dlc.m1, . ~ offset(log(Pop)) + Age )
```

An alternative model might consider `Age` as quantitative (since the categories are not equally spaced), using the lower class boundary of each class. (The *lower* boundary are preferred since the final class only has a lower boundary; the class midpoint or upper boundary becomes subjective for the final class.)

```
> danishlc$AgeNum <- rep( c(40, 55, 60, 65, 70, 75), 4)
> dlc.m3 <- update(dlc.m1, . ~ offset( log(Pop) ) +  AgeNum)
```

Figure 10.1 may suggest a possible quadratic relationship, but note the lower class boundaries are not equally spaced:

```
> dlc.m4 <- update( dlc.m3, . ~ offset( log(Pop) ) + poly(AgeNum, 2) )
```

The quadratic model is an improvement over the model linear in `AgeNum`:

```
> anova( dlc.m3, dlc.m4, test="Chisq")
Analysis of Deviance Table

Model 1: Cases ~ AgeNum + offset(log(Pop))
Model 2: Cases ~ poly(AgeNum, 2) + offset(log(Pop))
  Resid. Df Resid. Dev Df Deviance  Pr(>Chi)
1        22     48.968
2        21     32.500  1   16.468 4.948e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Since the models are not nested, we compare the four models using the AIC:

```
> c( "With interaction"=AIC(dlc.m1), "Without interaction"=AIC(dlc.m2),
    "Age (numerical)"=AIC(dlc.m3), "Age (numerical; quadratic)"=AIC(dlc.m4) )
          With interaction        Without interaction
                  144.3880                  136.6946
           Age (numerical) Age (numerical; quadratic)
                  149.3556                  134.8876
```

The AIC suggests the quadratic model `dlc.m4` produces the best predictions, but the AIC for models `dlc.m2` and `dlc.m4` are very similar.

The saddlepoint approximation is suitable for Poisson distributions when $y_i > 3$ for all observations. For these data:

```
> sort( danishlc$Cases )
 [1]  2  4  5  6  7  7  7  8  8  9 10 10 10 10 11 11 11 11 11 12 12 13 14
[24] 15
```

which shows that the saddlepoint approximation may be suspect. However, only one observation fails to meet this criterion, and only just, so we use the goodness-of-fit tests remembering to be cautious:

```
> D.m2 <- deviance(dlc.m2); df.m2 <- df.residual( dlc.m2 )
> c( Dev=D.m2, df=df.m2, P = pchisq( D.m2, df.m2, lower = FALSE) )
        Dev          df           P
28.30652745 18.00000000  0.05754114
> D.m4 <- deviance(dlc.m4); df.m4 <- df.residual( dlc.m4 )
> c( Dev=D.m4, df=df.m4, P=pchisq( D.m4, df.m4, lower = FALSE) )
        Dev          df           P
32.49959158 21.00000000  0.05206888
```

Both models are reasonably adequate. Consider the diagnostic plots (Fig. 10.2), where the constant-information scale is from Table 8.1:

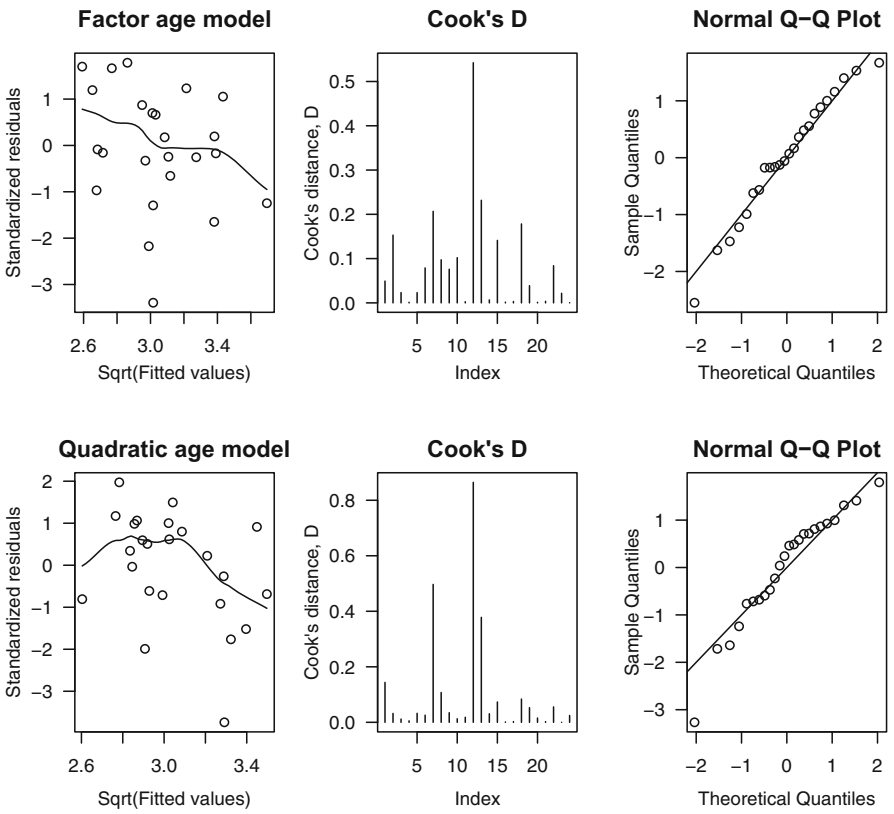

Fig. 10.2 Diagnostic plots for two models fitted model to the Danish lung cancer data. Top panels: treating age as a factor (model `dlc.m2`); bottom panels: fitting a quadratic in age (model `dlc.m4`). The Q–Q plots use quantile residuals (Example 10.1)

```
> library(statmod)  # For quantile residuals
> scatter.smooth( rstandard(dlc.m2) ~ sqrt(fitted(dlc.m2)),
         ylab="Standardized residuals", xlab="Sqrt(Fitted values)",
         main="Factor age model", las=1 )
> plot( cooks.distance(dlc.m2), type="h", las=1, main="Cook's D",
        ylab="Cook's distance, D")
> qqnorm( qr<-qresid(dlc.m2), las=1 ); abline(0, 1)
> scatter.smooth( rstandard(dlc.m4) ~ sqrt(fitted(dlc.m4)),
         ylab="Standardized residuals", xlab="Sqrt(Fitted values)",
         main="Quadratic age model", las=1 )
> plot( cooks.distance(dlc.m4), type="h", las=1, main="Cook's D",
        ylab="Cook's distance, D")
> qqnorm( qr<-qresid(dlc.m4), las=1 ); abline(0, 1)
```

The diagnostics suggest that both models are reasonable models, though we prefer model `dlc.m2`, since model `dlc.m4` appears to show three observations with high influence relative to the other observations, and is a simpler model.

□

10.4 Contingency Tables: Log-Linear Models

10.4.1 Introduction

Count data commonly appear in tables, called *contingency tables*, where the observations are cross-classified according to the levels of the classifying factors. To discuss the issues relevant to contingency tables, we begin with two cross-classifying factors (two-dimensional tables; Sect. 10.4.2 and 10.4.3) then extend to three cross-classifying factors (three-dimensional tables; Sect. 10.4.4) and then extend to higher-order tables (Sect. 10.4.7).

10.4.2 Two Dimensional Tables: Systematic Component

The simplest contingency table is a two-way (or two-dimensional) table, with factors A and B. If factor A has I levels and factor B has J levels, the contingency table has size $I \times J$. In general, the entries in an $I \times J$ table are defined as shown in Table 10.2, where y_{ij} refers to the observed count in row i and column j for $i = 1, 2, \ldots I$ and $j = 1, 2, \ldots J$.

Write μ_{ij} for to the expected *count* in cell (i, j). For convenience, also define π_{ij} as the expected *probability* that an observation is in cell (i, j), where $\mu_{ij} = m\pi_{ij}$, and m is the total number of observations. We write $m_{i\bullet}$ to mean the sum of counts in row i over all columns, and $m_{\bullet j}$ to mean the sum of counts in column j over all rows. The use of the dot $\bullet$ in this context means to sum over all the elements of the index that the dot replaces.

If factors A and B are independent, then $\pi_{ij} = \pi_{i\bullet}\pi_{\bullet j}$ is true. Writing $\mu_{ij} = m\pi_{i\bullet}\pi_{\bullet j}$, take logarithms to obtain

$$\log \mu_{ij} = \log m + \log \pi_{i\bullet} + \log \pi_{\bullet j} \tag{10.2}$$

Table 10.2 The general $I \times J$ contingency table. The cell count y_{ij} corresponds to level i of A and level j of B (Sect. 10.4.2)

		Factor B				
		Column 1	Column 2	$\cdots$	Column J	Total
Factor A	Row 1	y_{11}	y_{12}	$\cdots$	y_{1J}	$m_{1\bullet}$
	Row 2	y_{21}	y_{22}	$\cdots$	y_{2J}	$m_{2\bullet}$
	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$
	Row I	y_{I1}	y_{I2}	$\cdots$	y_{IJ}	$m_{I\bullet}$
	Total	$m_{\bullet 1}$	$m_{\bullet 2}$	$\cdots$	$m_{\bullet J}$	m

Table 10.3 The attitude of Australians to genetically modified foods (factor A) according to income (factor B) (Example 10.2)

	High income ($x_2 = 0$)	Low income ($x_2 = 1$)	Total
For GM foods ($x_1 = 0$)	263	258	521
Against GM foods ($x_1 = 1$)	151	222	373
Total	414	480	894

for the systematic component. This systematic component may be re-expressed using dummy variables, since the probabilities $\pi_{i\bullet}$ depend on which unique row the observation is in, and the probabilities $\pi_{\bullet j}$ depends on which unique column the observation is in.

Example 10.2. To demonstrate and fix ideas, first consider the smallest possible table of counts: a 2×2 table. The data in Table 10.3 were collected between December 1996 and January 1997, and comprise a two-dimensional (or two-way) table of counts collating the attitude of Australians to genetically modified (GM) foods (factor A) according to their income (factor B) [28, 31].

To analyse the data in R, first define the variables:

```
> Counts <- c(263, 258, 151, 222)
> Att <- gl(2, 2, 4, labels=c("For", "Against") )
> Inc <- gl(2, 1, 4, labels=c("High", "Low") )
> data.frame( Counts, Att, Inc)
  Counts     Att  Inc
1    263     For High
2    258     For  Low
3    151 Against High
4    222 Against  Low
```

The function `gl()` is used to generate factors by specifying the pattern in the factor levels. The first input indicates the number of levels, the second input the number of times each level is repeated as a run according to how the counts are defined, and the third input is the length of the factor. The `labels` input is optional, and defines the names for each level of the factor. The variable `Inc`, for example, has two levels repeated one at a time (given the order of the counts supplied in `Counts`), and has a length of four. As a check, the contingency table in Table 10.3 can be created using

```
> gm.table <- xtabs( Counts ~ Att + Inc ); gm.table
          Inc
Att        High Low
  For       263 258
  Against   151 222
```

To test whether attitude is independent of income, a probabilistic model for the counts is needed. A complete model for the data in Table 10.3 depends on

how the sample of individuals was collected. We will see in the next section that a number of different possible sampling scenarios lead us back to the same basic statistical analysis. □

10.4.3 Two-Dimensional Tables: Random Components

10.4.3.1 Introduction

We now consider how the sample of individuals, tabulated in the contingency table, was collected. In particular, we consider whether any or all of the margins of the table were preset by the sampling scheme. A table of counts may arise from several possible sampling schemes, each suggesting a different probability model. Three possible scenarios are:

- The m observations are allocated to factors A and B as the observations randomly arrive; neither row nor column totals are fixed.
- A fixed total number of m observations are cross-classified by the factors A and B.
- The row totals are fixed, and observations allocated to factor B within each level of A. (Alternatively, the column total are fixed, and observations allocated to factor A within each level of B.)

10.4.3.2 No Marginal Totals Are Fixed

Firstly, assume no marginal totals are fixed, as would be the case if, for example, the data in Table 10.3 are collated from survey forms completed by customers randomly arriving at a large shopping centre over 1 week. In this scenario, no marginal total is fixed; no limits exists on how large the counts can be (apart from the city population, which is much larger than the counts in the table).

If the total number of individuals observed (the grand total in the table) can be viewed as Poisson distributed, and if the individuals give responses independently of one another, then each of the counts in the table must follow a Poisson distribution. The log-likelihood function for the 2×2 table is

$$\ell(\mu; y) = \sum_{i=1}^{2} \sum_{j=1}^{2} \left(-\mu_{ij} + y_{ij} \log \mu_{ij}\right), \tag{10.3}$$

ignoring the terms not involving the parameters μ_{ij}. The residual deviance is

$$D(y, \hat{\mu}) = \sum_{i=1}^{2} \sum_{j=1}^{2} y_{ij} \log \frac{y_{ij}}{\hat{\mu}_{ij}}, \tag{10.4}$$

omitting the term $y_{ij} - \hat{\mu}_{ij}$, which always sums to zero if the log-linear predictor contains the constant term (Sect. 10.2).

Example 10.3. A Poisson model can be fitted to the GM foods data (Example 10.2) in R as follows:

```
> gm.1 <- glm( Counts ~ Att + Inc, family=poisson)
> anova( gm.1, test="Chisq")
     Df Deviance Resid. Df Resid. Dev  Pr(>Chi)
NULL                     3     38.260
Att   1  24.6143         2     13.646 7.003e-07 ***
Inc   1   4.8769         1      8.769   0.02722 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

(Recall the logarithmic link function is the default in R for the Poisson distribution.) This model fits a log-linear model equivalent to (10.2), and hence assumes that attitude and income are independent. Both `Att` and `Inc` are statistically significant in the order they are fitted. The Poisson GLM has the coefficients

```
> coef( gm.1 )
(Intercept)  AttAgainst      IncLow
  5.4859102  -0.3341716   0.1479201
```

Thus the model has the systematic component

$$\log \hat{\mu}_{ij} = 5.486 - 0.3342x_1 + 0.1479x_2, \tag{10.5}$$

where $x_1 = 1$ for row $i = 2$ (against GM foods) and is zero otherwise, and $x_2 = 1$ for column $j = 2$ (low income) and is zero otherwise. (The R notation means, for example, that `AttAgainst` $= 1$ when the variable `Att` has the value `Against` and is zero otherwise.) The systematic component in the form of (10.5) is the usual regression model representation of the systematic component, where dummy variables are explicitly used for the rows and columns. Since each cell of the table belongs to just one row and one column, the dummy variables are often zero for any given cell.

Log-linear models are often easier to interpret when converted back to the scale of the fitted values. In particular, $\exp(\hat{\beta}_0)$ gives the fitted expected count for the first cell in the table, while similar expressions for the other parameters give the relative increase in counts for one level of a factor over the first. By unlogging, the systematic component (10.5) becomes

$$\begin{aligned}\hat{\mu}_{ij} &= \exp(5.486) \times \exp(-0.3342x_1) \times \exp(0.1479x_2) \\ &= 241.3 \times 0.7159^{x_1} \times 1.159^{x_2}.\end{aligned}$$

Compare the values of $\hat{\mu}_{ij}$ when $x_2 = 1$ to the values when $x_2 = 0$:

$$\begin{aligned} \text{When } x_2 = 0: &\quad \hat{\mu}_{i1} = 241.3 \times 0.7159^{x_1} \\ \text{When } x_2 = 1: &\quad \hat{\mu}_{i2} = 241.3 \times 0.7159^{x_1} \times 1.159. \end{aligned} \tag{10.6}$$

Under this model, the fitted values for $\hat{\mu}_{i2}$ are always 1.159 times the fitted values for $\hat{\mu}_{i1}$, for either value of x_1. From Table 10.3, the ratio of the corresponding column marginal totals is

```
> sum(Counts[Inc=="Low"]) /  sum(Counts[Inc=="High"])
[1] 1.15942
```

This value is exactly the factor in (10.6), which is no coincidence. This demonstrates an important feature of the main effects terms in log-linear models: the main effect terms in the model simply model the marginal totals. These marginal totals are usually not of interest. The purpose of the GM study, for example, is to determine the *relationship* between income and attitudes towards GM foods, not to estimate the proportion of Australians with high incomes. That is, the real interest lies with the *interaction* term in the model:

```
> gm.int <- glm( Counts ~ Att * Inc, family=poisson)
> anova( gm.int, test="Chisq")
        Df Deviance Resid. Df Resid. Dev  Pr(>Chi)
NULL                        3     38.260
Att      1  24.6143         2     13.646 7.003e-07 ***
Inc      1   4.8769         1      8.769  0.027218 *
Att:Inc  1   8.7686         0      0.000  0.003065 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The analysis of deviance table shows the interaction term is necessary in the model. Notice that after fitting the interaction term, no residual deviance remains and no residual degrees of freedom remain, so the fit is perfect. This indicates that the number of coefficients in the model is the same as the number of entries in the table:

```
> length(coef(gm.int))
[1] 4
```

This means that the 2×2 table cannot be summarized by a smaller set of model coefficients. Since the interaction term is significant, the data suggest an association between income levels and attitude towards GM foods. We can examine the percentage of low and high income respondents who are `For` and `Against` GM foods by income level using `prop.table()`:

```
> round(prop.table(gm.table, margin=2)*100, 1)  # margin=2  means columns
         Inc
Att       High  Low
  For     63.5 53.8
  Against 36.5 46.2
```

This table shows that high income Australians are more likely to be in favour of GM foods than low income Australians.

Observe that the main result of the model fitting is that the interaction is significant (and hence that income and attitude to GM food are associated), rather than the individual estimates of the regression parameters. □

10.4.3.3 The Grand Total Is Fixed

Another scenario that may have produced the data in Table 10.3 assumes a fixed number of 894 people were sampled. For example, the researchers may have decided to survey 894 people in total, and then classify each respondent as `Low` or `High` income, and also classify each respondent as `For` or `Against` GM foods. While the counts are free to vary within the table, the counts have the restriction that their sum is capped at 894. However, the Poisson distribution has no upper limits on y by definition. Instead, the *multinomial distribution* is appropriate. For a 2×2 table, the probability function for the multinomial distribution is

$$\mathcal{P}(y_{11}, y_{12}, y_{21}, y_{22}; \mu_{11}, \mu_{12}, \mu_{21}, \mu_{22}) = \frac{m!}{y_{11}!y_{12}!y_{21}!y_{22}!} \left(\frac{\mu_{11}}{m}\right)^{y_{11}} \left(\frac{\mu_{12}}{m}\right)^{y_{12}} \left(\frac{\mu_{21}}{m}\right)^{y_{21}} \left(\frac{\mu_{22}}{m}\right)^{y_{22}}.$$

Ignoring terms not involving μ_{ij}, the log-likelihood function is

$$\ell(\mu; y) = \sum_{i=1}^{2} \sum_{j=1}^{2} y_{ij} \log \mu_{ij}, \tag{10.7}$$

and the residual deviance is

$$D(y, \hat{\mu}) = \sum_{i=1}^{2} \sum_{j=1}^{2} y_{ij} \log \frac{y_{ij}}{\hat{\mu}_{ij}}, \tag{10.8}$$

after ignoring terms not involving $\hat{\mu}_{ij}$. Estimating μ_{ij} by maximizing the log-likelihood for the multinomial distribution requires the extra condition $\sum_i \sum_j \mu_{ij} = m$ to ensure that the grand total is fixed at $\sum_i \sum_j y_{ij} = m$ as required by the sampling scheme.

Notice the similarity between the log-likelihood for the Poisson (10.3) and multinomial (10.7) distributions: the first term in (10.3) is the extra condition to ensure the grand total is fixed, and the second term is identical to (10.7). The residual deviance is exactly the same for the Poisson (10.4) and multinomial (10.7) distributions, after ignoring terms not involving μ_{ij}. These similarities for the multinomial and Poisson distributions have one fortunate implication: even though the multinomial distribution is the appropriate

probability model, a Poisson GLM can be used to model the data under appropriate conditions. When the grand total is fixed, the appropriate condition is that the constant term β_0 must appear in the linear predictor, because this ensures $\sum_{i=1}^{2}\sum_{j=1}^{2}\hat{\mu}_{ij} = m$ (Problem 10.2). The effect of including the constant term in the model is that all inferences are conditional on the grand total. The Poisson model, conditioning on the grand total, is equivalent to a multinomial model. Thus, *a Poisson model is still an appropriate model for the randomness, provided the constant term is in the model.*

10.4.3.4 The Column (or Row) Totals Are Fixed

A third scenario that may have produced the data in Table 10.3 assumes that the column (or row) totals are fixed. For example, the researchers may have decided to survey 480 low income people and 414 high income people, then record their attitudes towards GM foods. In this case, the totals in each column are fixed and the counts again have restrictions. For example, the number of high income earners *against* GM foods is known once the number of high income earners *in favour* of GM foods is known.

A multinomial distribution applies separately within each column of the table, because the numbers in each column are fixed and not random. Assuming the counts in each column are independent, the probability function is

$$\begin{aligned}&\mathcal{P}(y_{11}, y_{12}, y_{21}, y_{22}; \mu_{11}, \mu_{12}, \mu_{21}, \mu_{22})\\ &= \overbrace{\frac{m_{\bullet 1}!}{y_{11}!y_{21}!}\left(\frac{\mu_{11}}{m_{\bullet 1}}\right)^{y_{11}}\left(\frac{\mu_{21}}{m_{\bullet 1}}\right)^{y_{21}}}^{\text{For column 1}}\\ &\quad \underbrace{\times \frac{m_{\bullet 2}!}{y_{12}!y_{22}!}\left(\frac{\mu_{12}}{m_{\bullet 2}}\right)^{y_{12}}\left(\frac{\mu_{22}}{m_{\bullet 2}}\right)^{y_{22}}}_{\text{For column 2}}\end{aligned} \tag{10.9}$$

where $m_{\bullet j}$ is the total of column j. The log-likelihood function is

$$\ell(\mu; y) = \sum_{i=1}^{2}\sum_{j=1}^{2} y_{ij}\log\mu_{ij}, \tag{10.10}$$

when terms not involving the parameters μ_{ij} are ignored. To solve for the parameters μ_{ij}, the extra constraints $\sum_{i=1}^{2}\mu_{i1} = m_{\bullet 1}$ and $\sum_{i=1}^{2}\mu_{i2} = m_{\bullet 2}$ must also be added to ensure both column totals are fixed.

Again, notice the similarity between the log-likelihood (10.10) and the log-likelihood for the Poisson (10.3). The residual deviances are exactly the same, after ignoring terms not involving μ_{ij}. This means the Poisson distribution

can be used to model the data, provided the coefficients corresponding to the row totals appear in the linear predictor, since this ensures

$$m_{\bullet 2} = \sum_{i=1}^{2} y_{i2} = \sum_{i=1}^{2} \hat{\mu}_{i2}.$$

Requiring β_0 in the model also ensures that $\sum y_{i1} = \sum_{i=1}^{2} \hat{\mu}_{i1}$ also, and so the row totals are fixed.

Similarly, if the column totals are fixed, a Poisson GLM is appropriate if the coefficients corresponding to the column totals are in the model. If both the row and column totals are fixed, a Poisson GLM is appropriate if the coefficients corresponding to the row and column totals are in the linear predictor.

These general ideas can be extended to larger tables. In general, a Poisson GLM can be fitted to contingency table data provided the coefficients in the linear predictor corresponding to fixed margins are included in the linear predictor.

10.4.4 Three-Dimensional Tables

10.4.4.1 Introduction

Three-dimensional tables cross-classify subjects according to three factors, say A, B and C. If the factors have I, J and K levels respectively, the table is an $I \times J \times K$ table. As an example, the entries in a $3 \times 2 \times 2$ table are defined as shown in Table 10.2, where y_{ijk} refers to the observed count in row i ($i = 1, 2, \ldots I$) and column j ($j = 1, 2, \ldots J$) for group k ($k = 1, 2, \ldots K$); μ_{ijk} refers to the expected count in cell (i, j, k); and $\pi_{ijk} = \mu_{ijk}/m$ refers to the expected probability that an observation is in cell (i, j, k). In other words, Factor A has I levels, Factor B has J levels, and Factor C has K levels (Table 10.4).

Table 10.4 The $3 \times 2 \times 2$ contingency table. The cell count y_{ijk} corresponds to level i of A, level j of B and level k of C (Sect. 10.4.4)

	C_1			C_2					
	B_1	B_2	Total	B_1	B_2	Total	Total B_1	Total B_2	Total
A_1	y_{111}	y_{121}	$m_{1\bullet 1}$	y_{112}	y_{122}	$m_{1\bullet 2}$	$m_{11\bullet}$	$m_{12\bullet}$	$m_{1\bullet\bullet}$
A_2	y_{211}	y_{221}	$m_{2\bullet 1}$	y_{212}	y_{222}	$m_{2\bullet 2}$	$m_{21\bullet}$	$m_{22\bullet}$	$m_{2\bullet\bullet}$
A_3	y_{311}	y_{321}	$m_{3\bullet 1}$	y_{312}	y_{322}	$m_{3\bullet 2}$	$m_{31\bullet}$	$m_{32\bullet}$	$m_{3\bullet\bullet}$
Total	$m_{\bullet 11}$	$m_{\bullet 21}$	$m_{\bullet\bullet 1}$	$m_{\bullet 12}$	$m_{\bullet 22}$	$m_{\bullet\bullet 2}$	$m_{\bullet 1\bullet}$	$m_{\bullet 2\bullet}$	m

Table 10.5 The kidney stone data. The success rates of two methods are given by size; S means a success, and F means a Failure (Example 10.4)

	Small stones			Large stones					
	S	F	Total	S	F	Total	Total S	Total F	Total
Method A	81	6	87	192	71	263	273	77	350
Method B	234	36	270	55	25	80	289	61	350
Total	315	42	357	247	96	343	562	138	700

The meaning of the main effect terms in a Poisson GLM has been discussed in the two-dimensional context: the main effect terms model the marginal totals. Scientific interest focuses on the interactions between the factors. The model with main-effects only acts as the base model for contingency tables against which interaction models are compared. In a three-dimensional table, three two-factor interactions are possible, as well as an interaction term with all three factors. Different interpretations exist depending on which interaction terms appear in the final model. These interpretations are considered in this section. We now introduce the example data to be used.

Example 10.4. The example data in this section (Table 10.5; data set: `kstones`) comes from a study of treatments for kidney stones [8, 24], comparing the success rates of various methods for small and large kidney stones.

```
> data(kstones); str(kstones)
'data.frame':        8 obs. of  4 variables:
 $ Counts : int  81 6 234 36 192 71 55 25
 $ Size   : Factor w/ 2 levels "Large","Small": 2 2 2 2 1 1 1 1
 $ Method : Factor w/ 2 levels "A","B": 1 1 2 2 1 1 2 2
 $ Outcome: Factor w/ 2 levels "Failure","Success": 2 1 2 1 2 1 2 1
```

We treat the method as factor A, the kidney stone size as factor B, and the outcome (success or failure) as factor C.

Note that 350 patients were selected for use with each method. Since this marginal total is fixed, the corresponding main effect term `Method` must appear in the Poisson GLM. The Poisson GLM with all three main effect terms ensures all the marginal totals from the original table are retained, but the parameters themselves are of little interest. □

10.4.4.2 Mutual Independence

If A, B and C are independent, then $\pi_{ijk} = \pi_{i\bullet\bullet} \times \pi_{\bullet j\bullet} \times \pi_{\bullet\bullet k}$ so that, on a log-scale,

$$\log \mu_{ijk} = \log m + \log \pi_{i\bullet\bullet} + \log \pi_{\bullet j\bullet} + \log \pi_{\bullet\bullet k},$$

using that $\mu_{ijk} = m\pi_{ijk}$. This is called *mutual independence*. As seen for the two-dimensional tables, including the main effect terms effectively ensures the marginal totals are preserved. If the mutual independence model is appropriate, then the table may be understood from just the marginal totals.

For the kidney stone data, the mutual independence model states that the success or failure is independent of the method used, and independent of the size of the kidney stones, and that the method used is also independent of the size of the kidney stone. Adopting this model assumes the data can be understood for each variable separately. In other words, equal proportions of patients are in each method; $138/700 = 19.7\%$ of all treatments fail; and $343/700 = 49.0\%$ of patients have large kidney stones. Fit the model using:

```
> ks.mutind <- glm( Counts ~ Size + Method + Outcome,
                    family=poisson, data=kstones)
```

In this section, we will fit the models then comment and compare the models after all the models are fitted.

10.4.4.3 Partial Independence

Suppose A and B are not independent, but both are independent of C; then $\pi_{ijk} = \pi_{ij\bullet} \times \pi_{\bullet\bullet k}$, or $\log \mu_{ijk} = \log m + \log \pi_{ij\bullet} + \log \pi_{\bullet\bullet k}$ on a log-scale. Since A and B are not independent, $\pi_{ij\bullet} \neq \pi_{i\bullet\bullet} \times \pi_{\bullet j\bullet}$. To ensure that the marginal totals are preserved, the main effects are also included in the model (along the lines of the marginality principle; Sect. 2.10.4). This means that the model

$$\log \mu_{ijk} = \log m + \log \pi_{i\bullet\bullet} + \log \pi_{\bullet j\bullet} + \log \pi_{\bullet\bullet k} + \log \pi_{ij\bullet}$$

is suggested. This systematic component has one two-factor interaction $A.B$. This is called *partial independence* (or *joint independence*). If a partial independence model is appropriate, then the two-way tables for each level of C are multiples of each other, apart from randomness. The data can be understood by combining the tables over C.

For the kidney stone data, we can fit all three models that have one of the two-factor interactions:

```
> ks.SM <- glm( Counts ~ Size * Method + Outcome,
                family=poisson, data=kstones )
> ks.SO <- update(ks.SM, . ~ Size * Outcome + Method)
> ks.OM <- update(ks.SM, . ~ Outcome * Method + Size)
```

10.4.4.4 Conditional Independence

Suppose that A and B are independent of each other when considered separately for each level of C. Then the probabilities π_{ijk} are independent

conditional on the level of k, when $\pi_{ij|k} = \pi_{i\bullet|k} \times \pi_{\bullet j|k}$. Each conditional probability can be written in terms of marginal totals:

$$\pi_{ij|k} = \frac{\pi_{ijk}}{\pi_{\bullet\bullet k}}; \quad \pi_{i\bullet|k} = \frac{\pi_{i\bullet k}}{\pi_{\bullet\bullet k}}; \quad \pi_{\bullet j|k} = \frac{\pi_{\bullet jk}}{\pi_{\bullet\bullet k}},$$

so that $\pi_{ijk} = (\pi_{i\bullet|k} \times \pi_{\bullet j|k})\pi_{\bullet\bullet k} = \pi_{i\bullet k}\pi_{\bullet jk}/\pi_{\bullet\bullet k}$ hold. In other words, $\log \mu_{ijk} = \log m + \log \pi_{i\bullet k} + \log \pi_{\bullet jk} - \log \pi_{\bullet\bullet k}$ on a log-scale. To ensure the marginal totals are preserved, use the model

$$\log \mu_{ijk} = \log m + \log \pi_{i\bullet\bullet} + \log \pi_{\bullet j\bullet} + \log \pi_{\bullet\bullet k} + \log \pi_{i\bullet k} + \log \pi_{\bullet jk}$$

which includes the main effects. The systematic component has the two two-factor interactions $A.C$ and $B.C$. This is called *conditional independence*.

If a conditional independence model is appropriate, then each two-way table for each level of C considered separately shows independence between A and B. The data can be understood by creating separate tables involving factors A and B, one for each level of C.

The three models with two of the two-factor interactions are:

```
> ks.noMO <- glm( Counts ~ Size * (Method + Outcome),
                  family=poisson, data=kstones )
> ks.noOS <- update(ks.noMO, . ~ Method * (Outcome + Size) )
> ks.noMS <- update(ks.noMO, . ~ Outcome * (Method + Size) )
```

10.4.4.5 Uniform Association

Consider the case where all three two-factor interactions are present but the three-factor interaction $A.B.C$ only is absent. This means that each two-factor interaction is unaffected by the level of the third factor. No interpretation in terms of independence or through the marginal totals is possible. The model is

$$\log \mu_{ijk} = \log m + \log \pi_{i\bullet\bullet} + \log \pi_{\bullet j\bullet} + \log \pi_{\bullet\bullet k} + \log \pi_{i\bullet k} + \log \pi_{\bullet jk} + \log \pi_{ij\bullet}$$

which contains all two-way interactions. This is called *uniform association*. If the uniform association model is appropriate, then the data can be understood by examining all three individual two-way tables. For the kidney stone data the model with all of the two-factor interactions is:

```
> ks.no3 <- glm( Counts ~ Size*Method*Outcome - Size:Method:Outcome,
                 family=poisson, data=kstones )
```

Uniform association is simple enough to define from a mathematical point of view, but is often difficult to interpret from a scientific point of view.

10.4.4.6 The Saturated Model

If all interaction terms are necessary in the linear predictor, the model is the saturated model

$$\log \mu_{ijk} = \log m + \log \pi_{i\bullet\bullet} + \log \pi_{\bullet j\bullet} + \log \pi_{\bullet\bullet k} + \log \pi_{i\bullet k} + \log \pi_{\bullet jk} + \log \pi_{ij\bullet} + \log \pi_{ijk}$$

which includes all interactions. The model has zero residual deviance (in computer arithmetic) and zero residual degrees of freedom. In other words, the model produces a perfect fit:

```
> ks.all <- glm( Counts ~ Size * Method * Outcome,
                 family=poisson, data=kstones )
> c( deviance( ks.all ), df.residual(ks.all) )
[1] -2.930989e-14  0.000000e+00
```

This means that there are as many parameter estimates as there are cells in the table, and so the data cannot be summarized using a smaller set of coefficients. If the saturated model is appropriate, then the data cannot be presented in a simpler form than giving the original $I \times J \times K$ table.

10.4.4.7 Comparison of Models

For the kidney stone data the saddlepoint approximation is sufficiently accurate since $\min\{y_i\} \geq 3$. This means that goodness-of-fit tests can be used to examine and compare the models (Table 10.6). The mutual independence model and partial independence models are not appropriate, as the residual deviance far exceeds the residual degrees of freedom. Model `ks.noMO` appears the simplest suitable model. This implies that the data are best understood by creating separate tables for large and small kidney stones, but small and large kidney stones data should not be combined.

10.4.5 Simpson's Paradox

Understanding which interaction terms are necessary in a log-linear model has important implications for condensing the tabular data. If a table is collapsed over a factor incorrectly, incorrect and misleading conclusions may be reached. An extreme example of this is *Simpson's paradox*. To explain, consider the kidney stones data (Table 10.5). The most suitable model appears to be model `ks.noMO` (Table 10.6). This model has two two-factor interactions, indicating conditional independence between `Outcome` and `Method`, depending on the `Size` of the kidney stones. The dependence on `Size` means that the data must be stratified by kidney stone size for the correct relationship between `Method` and `Outcome` to be seen. Combining the data over `Size`s, and

Table 10.6 The fitted values for all Poisson GLMs fitted to the kidney stone data. Model `ks.noMO` is the selected model and is flagged * (Sect. 10.4.4)

	Mutual independence	Partial independence			Conditional independence			Uniform association	Saturated model
Count	`ks.mutind`	`ks.SM`	`ks.SO`	`ks.OM`	`ks.noMO` *	`ks.noOS`	`ks.noMS`	`ks.no3`	`ks.all`
81	143.3	69.8	157.5	139.2	76.8	67.9	153.0	79.0	81
6	35.2	17.2	21.0	39.3	10.2	19.1	23.4	8.0	6
234	143.3	216.8	157.5	147.4	238.2	222.9	162.0	236.0	234
36	35.2	53.2	21.0	31.1	31.8	47.1	18.6	34.0	36
192	137.7	211.2	123.5	133.8	189.4	205.1	120.0	194.0	192
71	33.8	51.8	48.0	37.7	73.6	57.9	53.6	69.0	71
55	137.7	64.2	123.5	141.6	57.6	66.1	127.0	53.0	55
25	33.8	15.8	48.0	29.9	22.4	13.9	42.4	27.0	25
Res. dev.:	234.4	33.1	204.8	232.1	3.5	30.8	202.4	1.0	0
Res. df:	4	3	3	3	2	2	2	1	0
G-o-F P:	0.00	0.00	0.00	0.00	0.18	0.00	0.00	0.32	1.00

hence considering a single combined two-way table of `Method` and `Outcome` (and hence ignoring `Size`), is an incorrect summary. To demonstrate, consider *incorrectly* collapsing the contingency table over `Size`. First, use `xtabs()` to create a suitable three-dimensional table of counts:

```
> ks.tab <- xtabs(Counts ~ Method + Outcome + Size, data=kstones)
> ks.tab
, , Size = Large

      Outcome
Method Failure Success
     A      71     192
     B      25      55

, , Size = Small

      Outcome
Method Failure Success
     A       6      81
     B      36     234
```

Then sum over `Size`, which is the third dimension:

```
> MO.tab <- apply( ks.tab, c(1, 2), sum) # Sums over the 3rd dimension
> MO.tab    # An *incorrect* collapsing of the data
      Outcome
Method Failure Success
     A      77     273
     B      61     289
```

The table suggests that Method B has a higher success rate than Method A:

```
> prop.table(MO.tab, 1) # Compute proportions in each row (dimension 1)
      Outcome
Method   Failure   Success
     A 0.2200000 0.7800000
     B 0.1742857 0.8257143
```

The overall success rate for Method A is about 78%, and for Method B the success rate is about 83%, so we would prefer Method B. However, recall that the table `MO.tab` is *incorrectly* collapsed over `Size`: the conditional independence suggest the relationship between `Method` and `Outcome` should be examined *separately* for each level of `Size`.

Consequently, now examine the two-way table for large and small kidney stones separately:

```
> MO.tab.SizeLarge <- ks.tab[, , "Large"] # Select Large stones
> prop.table(MO.tab.SizeLarge, 1)  # Compute proportions in each row
      Outcome
Method  Failure  Success
     A 0.269962 0.730038
     B 0.312500 0.687500
```

For large kidney stones, the success rate for Method A is about 73%, and for Method B the success rate is about 69% so we would prefer Method A.

```
> MO.tab.SizeSmall <- ks.tab[, , "Small"] # Select Small stones
> prop.table(MO.tab.SizeSmall, 1)  # Compute proportions in each row
      Outcome
Method    Failure    Success
     A 0.06896552 0.93103448
     B 0.13333333 0.86666667
```

For small kidney stones, the success rate for Method A is about 93%, and for Method B the success rate is about 87%, so we would prefer Method A.

In this example, incorrectly collapsing the table over `Size` has completely changed the conclusion. Ignoring `Size`, Method B has a higher overall success rate, but Method A actually has a higher success rate for both small and large kidney stones. This is called *Simpson's paradox*, which is a result of incorrectly collapsing a table.

To explain the apparent paradox, first notice that the large kidney stone group reported a far lower success rate for both methods compared to the small kidney stone group. Since Method A was used on a larger proportion of patients with large kidney stones, Method A reports a high number of total failures when the two groups are combined. In contrast, Method B was used on a larger proportion of patients with small kidney stones, where the success rate for both methods is better, and so Method B reports a smaller number of total failures.

10.4.6 Equivalence of Binomial and Poisson GLMs

In many contingency table contexts, interest focuses on explaining one of the factors in terms of the others. When the response factor of interest takes two levels, interest focuses on explaining the proportion of responses that are allocated to each of the two levels. In this case, there is a binomial GLM with the logistic link that is equivalent to the Poisson log-linear model. The reason is that for large m and small proportions, the binomial distribution approaches the Poisson distribution. To see this, write the probability of a success in the binomial distribution as π. Then, the variance function for the *number* of successes using the binomial model is $V(\pi) = m\pi(1-\pi)$. When π is small and m is large, $V(\pi) = m\pi(1-\pi) \to m\pi$. This is equivalent to the variance of the Poisson distribution. This means that the binomial distribution approaches the Poisson distribution for large m and small π.

For example, consider the data of Table 10.3 (p. 379) relating GM attitude to income. Here interest focuses on whether income level affects GM attitude, so the data could be equally well analysed in R by treating `Att` as the response variable:

```
> y <- ifelse(Att == "Against", 1, 0)
> gm.bin <- glm(y~Inc, family=binomial, weights=Counts)
> anova(gm.bin, test="Chisq")
     Df Deviance Resid. Df Resid. Dev Pr(>Chi)
NULL                     3     1214.7
Inc   1   8.7686         2     1206.0 0.003065 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The deviance goodness-of-fit test for `Inc` is identical to the test for `Att:Inc` interaction given in Sect. 10.4.3.2, with the same P-value and the same interpretation. The odds of being against GM foods are nearly 50% greater for low-income respondents:

```
> coef(summary(gm.bin))
             Estimate Std. Error   z value     Pr(>|z|)
(Intercept) -0.5548742  0.1021018 -5.434518 5.494476e-08
IncLow       0.4045920  0.1371323  2.950378 3.173854e-03
> exp(coef(gm.bin)["IncLow"])
  IncLow
1.498691
```

Example 10.5. For the kidney stones data (Table 10.5; data set: `kstones`), interest may focus on comparing the success rates of the two methods. From this point of view, the data may be analysed via a binomial GLM:

```
> y <- ifelse(kstones$Outcome=="Success", 1, 0)
> ks.bin <- glm(y~Size*Method, family=binomial,
              weights=Counts, data=kstones)
> anova(ks.bin, test="Chisq")
            Df Deviance Resid. Df Resid. Dev  Pr(>Chi)
NULL                           7     694.98
Size         1  29.6736         6     665.31 5.113e-08 ***
Method       1   2.4421         5     662.87    0.1181
Size:Method  1   1.0082         4     661.86    0.3153
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The analysis of deviance shows that success depends strongly on the size of the kidney stones (better success for small stones), but there is no evidence for any difference between the two methods, either overall or separately for small or large stones. This conclusion agrees with the contingency table analysis, which concluded that `Outcome` was conditionally independent of `Method` given `Size`. The contingency table model `ks.noMO` contains the additional information that `Method` is associated with `Size`. Indeed it is clear from Table 10.5 that Method A is predominately used for large stones and Method B for small stones. Whether the ability to test for associations between explanatory factors, provided by the contingency table analysis, is of interest depends on the scientific context. For these data, the choice of method is likely made based on established hospital protocols, and hence would be known before the data were collected. □

10.4.7 Higher-Order Tables

Extending these ideas to situations with more than three factors is easy in practice using R, though interpreting the final models is often difficult.

Example 10.6. A study of seriously emotionally disturbed (SED) and learning disabled (LD) adolescents [19, 29] reported their depression levels (Table 10.7; data set: `dyouth`). The data are counts classified by four factors: `Age` (using `12-14` as the reference group), `Group` (either `LD` or `SED`), `Gender` and level of `Depression` (either low `L` or high `H`). Since none of the totals were fixed beforehand and are free to vary randomly, no variables *need* to be included in the model. With four factors, $\binom{4}{2} = 6$ two-factor interactions, $\binom{4}{3} = 4$ three-factor interactions and one four-factor interaction are potentially in the model. As usual, the main-effect terms are included in the model to ensure the marginal totals are preserved.

Table 10.7 Depression levels in youth (Example 10.6)

		Depression low L		Depression high H	
Age	Group	Males	Females	Males	Females
12-14	LD	79	34	18	14
	SED	14	5	5	8
15-16	LD	63	26	10	11
	SED	32	15	3	7
17-18	LD	36	16	13	1
	SED	36	12	5	2

The most suitable model for the data [11] (Problem 10.8) appears to be:

```
> data(dyouth)
> dy.m1 <- glm( Obs ~ Age*Depression*Gender + Age*Group,
               data=dyouth, family=poisson)
> anova(dy.m1, test="Chisq")
                     Df Deviance Resid. Df Resid. Dev  Pr(>Chi)
NULL                                    23     368.05
Age                   2   11.963        21     356.09  0.002525 **
Depression            1  168.375        20     187.71 < 2.2e-16 ***
Gender                1   58.369        19     129.34 2.172e-14 ***
Group                 1   69.104        18      60.24 < 2.2e-16 ***
Age:Depression        2    3.616        16      56.62  0.163964
Age:Gender            2    3.631        14      52.99  0.162718
Depression:Gender     1    7.229        13      45.76  0.007175 **
Age:Group             2   27.090        11      18.67 1.311e-06 ***
Age:Depression:Gender 2    8.325         9      10.35  0.015571 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The three-way interaction shows that the relationship between age and depression is different for males and females:

```
> Males   <- subset(dyouth, Gender=="M")
> Females <- subset(dyouth, Gender=="F")
> table.M <- prop.table( xtabs(Obs~Age+Depression, data=Males),   1)
> table.F <- prop.table( xtabs(Obs~Age+Depression, data=Females), 1)
> round(table.F * 100)   # FEMALES
       Depression
Age      H  L
  12-14 36 64
  15-16 31 69
  17-18 10 90
> round(table.M * 100)   # MALES
       Depression
Age      H  L
  12-14 20 80
  15-16 12 88
  17-18 20 80
```

Given the fitted model, collapsing the table into a simpler table would be misleading. The proportion tables show that the rate of high depression decreases with age for girls, especially for 17 years and older, whereas for males the rate of high depression decreases at age 15–16 then increases again for 17–18. This difference in pattern explains the three-way interaction detected by the analysis of deviance table.

The model also finds a significant interaction between `Age` and `Group`, meaning simply that the SED and LD groups contain different proportions of the age groups. This is not particularly of interest, but it is important to keep the `Age:Group` term in the model, so that the tests for interactions involving `Depression` should adjust for these demographic proportions.

Overall, the model shows an association between depression and age and gender, but no difference in depression rates between the two groups once the demographic variables have been taken into account. □

10.4.8 Structural Zeros in Contingency Tables

Contingency tables may contain cells with zero counts. Depending on the reason for a zero count, different approaches must be taken when modelling.

Sampling zeros or *random zeros* appear by chance, simply because no observations occurred in that category. Larger samples may produce non-zero counts in those cells. Computing fitted values for these cells is sensible; they are legitimate counts to be modelled like the other counts in the data. However, the presence of the zeros means the saddlepoint approximation is likely to be very poor. As a result, levels of one or more factors may be combined to increase the minimum count. For example, 'Strongly agree' and 'Agree' may be combined sensibly into a single 'Agreement' category.

Structural zeros appear because the outcome is impossible. For example, in a cross-tabulation of gender and surgical procedures, the cell corresponding to male hysterectomies *must* contain a zero count. Producing fitted values for these cells makes no sense. Structural zeros are not common in practice.

Structural zeros require special attention since computing expected counts for impossible events is nonsense. As a result, cells containing structural zeros are removed from the data before analysis.

Example 10.7. The types of cancer diagnosed in Western Australia in 1996 were recorded for males and females (Table 10.8; data set: `wacancer`) to ascertain whether the number of cancers differs between genders [20].

Three cells have zeros recorded. Two of these three cells are *structural zeros* since they are impossible—females cannot have prostate cancer, and males cannot have cervical cancer. Breast cancer is a possible, but very rare, disease among men (about 100 times as many cases in females compared to males, in the USA [34, Table 1]). The zero for male breast cancer is technically

Table 10.8 The number of cancers diagnosed by gender in Western Australia during 1996 (Example 10.7)

	Cancer type						
Gender	Prostate	Breast	Colorectal	Lung	Melanoma	Cervix	Other
Males	923	0	511	472	362	0	1406
Females	0	875	355	211	282	77	1082

a *sampling* zero. Since breast cancer is already *known* to be a rare disease for males, the analysis should focus on gender differences for other types of cancers, such as colorectal, lung, melanoma and other cancers.

To begin, we fit a model ignoring these complications:

```
> data(wacancer)
> wc.poor <- glm( Counts ~ Cancer*Gender, data=wacancer, family=poisson )
> anova( wc.poor, test="Chisq")
             Df Deviance Resid. Df Resid. Dev  Pr(>Chi)
NULL                             13     6063.7
Cancer        6   3281.5         7     2782.2 < 2.2e-16 ***
Gender        1     95.9         6     2686.2 < 2.2e-16 ***
Cancer:Gender 6   2686.2         0        0.0 < 2.2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

To compare, we now remove breast cancer, male cervical cancer and female prostate cancer from the analysis, and refit:

```
>    # Omit necessary cells of table:
> wc <- subset(wacancer, (Cancer!="Breast"))
> wc <- subset(wc, !(Cancer=="Cervix" & Gender=="M"))
> wc <- subset(wc, !(Cancer=="Prostate" & Gender=="F"))
> xtabs(Counts~Gender+Cancer, data=wc)  # Table *looks* similar
      Cancer
Gender Breast Cervix Colorectal Lung Melanoma Other Prostate
     F      0     77        355  211      282  1082        0
     M      0      0        511  472      362  1406      923
>    # Now fit the model
> wc.m1 <- glm( Counts ~ Cancer*Gender, data=wc, family=poisson )
> anova( wc.m1, test="Chisq")
             Df Deviance Resid. Df Resid. Dev  Pr(>Chi)
NULL                             9    2774.32
Cancer        5  2591.47         4     182.85 < 2.2e-16 ***
Gender        1   144.74         3      38.11 < 2.2e-16 ***
Cancer:Gender 3    38.11         0       0.00  2.68e-08 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

An alternative to explicitly removing these observations from the table is to set the corresponding prior weights `weights` to zero for these observations, and to one for other observations. Even though the prior weighs are defined to

be positive, R interprets a prior weight of zero to mean that the corresponding observation should be ignored in the analysis.

For both models, the interaction term is very significant, so the number of people diagnosed with the different types of cancers differs according to gender, even after eliminating prostate, breast and cervical cancer, which are obviously gender-linked. However, note that the degrees of freedom are different for the two models. □

10.5 Overdispersion

10.5.1 Overdispersion for Poisson GLMs

For a Poisson distribution, $\text{var}[y] = \mu$. However, in practice the apparent variance of the data often exceeds μ. This is called *overdispersion*, as has already been discussed for binomial GLMs (Sect. 9.8). Underdispersion also occurs, but is less common.

Overdispersion arises either because the mean μ retains some innate variability, even when all the explanatory variables are fixed, or because the events that are being counted are positively correlated. Overdispersion typically arises because the events being counted arise in clusters or are mutually supporting in some way. This causes the underlying events to be positively correlated, and overdispersion of the counts is the result.

The presence of overdispersion might or might not affect the parameter estimates $\hat{\beta}_j$, depending on the nature of the overdispersion, but the standard errors $\text{se}(\hat{\beta}_j)$ are necessarily underestimated. Consequently, tests on the explanatory variables will generally appear to be more significant that warranted by the data, and confidence intervals for the parameters will be narrower than warranted by the data.

Overdispersion is detected by conducting a goodness-of-fit test (as described in Sect. 7.4). If the residual deviance and Pearson goodness-of-fit statistics are much larger than the residual degrees of freedom, then either the fitted model is inadequate or the data are overdispersed. If lack of fit remains even after fitting the maximal possible explanatory model, and after eliminating any outliers, then overdispersion is the alternative explanation.

When the counts are very small, so asymptotic approximations to the residual deviance and Pearson statistics are suspect (Sect. 7.5, p. 276), then overdispersion may be difficult to judge. However the goodness-of-fit statistics are more likely to be underestimated than overestimated in small count situations, so large goodness-of-fit statistics should generally be taken to indicate lack of fit.

Table 10.9 The number of membrane pock marks at various dilutions of the viral medium (Example 10.9)

Dilution	Pock counts									
1	116	151	171	194	196	198	208	259		
2	71	74	79	93	94	115	121	123	135	142
4	27	33	34	44	49	51	52	59	67	92
8	8	10	15	22	26	27	30	41	44	48
16	5	6	7	7	8	9	9	9	11	20

Example 10.8. For the final model fitted to the kidney stone data (see Table 10.6), the residual deviance was 3.5 and the residual df was 2. A goodness-of-fit test does not reject the hypothesis that the model is adequate:

```
> pchisq(deviance(ks.noMO), df.residual(ks.noMO), lower.tail=FALSE)
[1] 0.1781455
```

□

Example 10.9. In an experiment [35] to assess viral activity, pock marks were counted at various dilutions of the viral medium (Table 10.9; data set: `pock`). We use the logarithm to base 2 of `Dilution` as a covariate, since the dilution levels are in increasing powers of 2 suggesting this was factored into the design. A plot of the data shows a definite relationship between the variables (Fig. 10.3, left panel), and that the variance increases with increasing mean (Fig. 10.3, right panel):

```
> data(pock)
> plot( Count ~ jitter(log2(Dilution)), data=pock, las=1,
    xlab="Log (base 2) of dilution", ylab="Pock mark count")
> mn <- with(pock, tapply(Count, log2(Dilution), mean) )  # Group means
> vr <- with(pock, tapply(Count, log2(Dilution), var) )   # Group variances
> plot( log(vr) ~ log(mn), las=1,
    xlab="Group mean", ylab="Group variance")
```

Intuitively, pock marks are more likely to appear in clusters rather than independently, so overdispersion would not be at all surprising. Indeed, the sample variance is much larger than the mean for each group, clear evidence of overdispersion:

```
> data.frame(mn, vr, ratio=vr/mn)
       mn         vr    ratio
0 186.625 1781.12500 9.543871
1 104.700  667.34444 6.373872
2  50.800  360.40000 7.094488
3  27.100  194.98889 7.195162
4   9.100   17.65556 1.940171
```

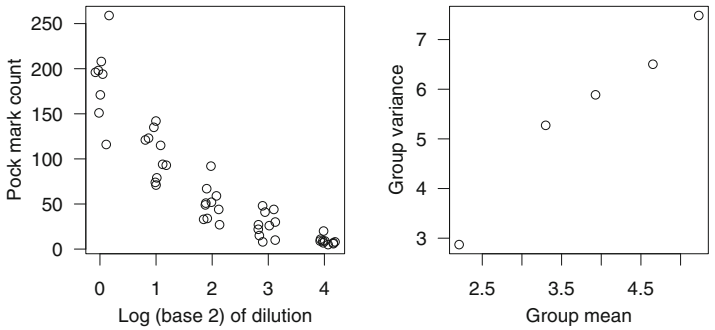

Fig. 10.3 The pock data. Left panel, the counts against the logarithm of dilution; right panel: the logarithm of the group variances against the logarithm of the group means (Example 10.9)

Not only are the variances greater than the means, but their ratio increases with the mean as well. The slope of the trend in the right panel of Fig. 10.3 is about 1.5:

```
> coef(lm(log(vr)~log(mn)))
(Intercept)      log(mn)
 0.02861162  1.44318666
```

This suggests a variance function approximately of the form $V(\mu) = \mu^{1.5}$. The mean–variance relationship here is in some sense intermediate between that for the Poisson ($V(\mu) = \mu$) and gamma ($V(\mu) = \mu^2$) distributions.

Fitting a Poisson GLM shows substantial lack of fit, as expected:

```
> m1 <- glm( Count ~ log2(Dilution), data=pock, family=poisson )
> X2 <- sum(residuals(m1, type="pearson")^2)
> c(Df=df.residual(m1), Resid.Dev=deviance(m1), Pearson.X2=X2)
        Df  Resid.Dev Pearson.X2
   46.0000   290.4387   291.5915
```

The saddlepoint approximation is satisfactory here as $\min\{y_i\} = 5$ is greater than 3. Indeed, the deviance and Pearson goodness-of-fit statistics are nearly identical. Two ways to model the overdispersion are discussed in Sects. 10.5.2 and 10.5.3. □

10.5.2 Negative Binomial GLMs

One way to model overdispersion is through a hierarchical model. Instead of assuming $y_i \sim \text{Pois}(\mu_i)$, we can add a second layer of variability by allowing μ_i itself to be a random variable. Suppose instead that

$$y_i|\lambda_i \sim \text{Pois}(\lambda_i) \quad \text{and} \quad \lambda_i \sim \text{G}(\mu_i, \psi)$$

where $\mathrm{G}(\mu_i, \psi)$ denotes a distribution with mean μ_i and coefficient of variation ψ. For example, we could imagine that the number of pock marks recorded in the pock data (Example 10.9) might follow a Poisson distribution for any given viral concentration, but that the viral concentration varies somewhat between replicates for any given dilution with a coefficient of variation ψ. It is straightforward to show, under the hierarchical model, that

$$\mathrm{E}[y_i] = \mu_i \quad \text{and} \quad \mathrm{var}[y_i] = \mu_i + \psi\mu_i^2,$$

so the variance contains an overdisperion term $\psi\mu_i^2$. The larger ψ, the greater the overdispersion.

A popular choice is to assume that the mixing distribution G is a gamma distribution. The coefficient of variation of a gamma distribution is its dispersion parameter, so the second layer of the hierachical model becomes $\lambda_i \sim \mathrm{Gam}(\mu_i, \psi)$. With this assumption, is it possible to show that y_i follows a *negative binomial distribution* with probability function

$$\mathcal{P}(y_i; \mu_i, k) = \frac{\Gamma(y_i + k)}{\Gamma(y_i + 1)\Gamma(k)} \left(\frac{\mu_i}{\mu_i + k}\right)^{y_i} \left(1 - \frac{\mu_i}{\mu_i + k}\right)^{k}, \qquad (10.11)$$

where $k = 1/\psi$ and $\Gamma()$ is the gamma function, so that $\mathrm{var}[y_i] = \mu_i + \mu_i^2/k$.

For any fixed value of k, it can be shown (Problem 10.1) that the negative binomial distribution is an EDM with unit deviance

$$d(y, \mu) = 2\left\{y \log \frac{y}{\mu} - (y + k) \log \frac{y + k}{\mu + k}\right\},$$

where the limit form (5.14) is used if $y = 0$. Hence the negative binomial distribution can be used to define a GLM for any given k. Note that negative binomial EDMs have dispersion $\phi = 1$, as do all EDMs for count data, because $\mathrm{var}[y_i]$ is determined by μ_i and k. In practice, k is rarely known and so negative binomial GLMs are usually used with an estimated value for k. In R, the function `glm.nb()` from package **MASS** can be used in place of `glm()` to fit the model. The function `glm.nb()` undertakes maximum likelihood estimation for both k and the GLM coefficients β_j simultaneously (see `?glm.nb`).

The estimation of k introduces an extra layer of uncertainty into a negative binomial GLM. However the maximum likelihood estimator $\hat{k}$ of k is uncorrelated with the $\hat{\beta}_j$, according to the usual asymptotical approximations. Hence the GLM fit tends to be relatively stable with respect to estimation of k.

Negative binomial GLMs give larger standard errors than the corresponding Poisson GLMs, depending on the size of $k = 1/\psi$. On the other hand, the coefficient estimates $\hat{\beta}_j$ from a negative binomial GLM may be similar to those produced from the corresponding Poisson GLM. The negative binomial GLM gives less weight to observations with large μ_i than does the Poisson GLM, and relatively more weight to observations with small μ_i, so the coefficients

will vary somewhat. Unlike `glm()`, where the default link function for every `family` is the canonical link, the default link function for `glm.nb()` is the logarithmic link function. Indeed the log-link is almost always used with negative binomial GLMs to ensure $\mu > 0$ for any value of the linear predictor. The function `glm.nb()` also allows the `"sqrt"` and `"identity"` link functions.

For negative binomial GLMs, the use of quantile residuals [12] is strongly recommended (Sect. 8.3.4.2).

Example 10.10. The pock data shows overdispersion (Example 10.9; data set: `pock`). We fit a negative binomial GLM, estimating k using the function `glm.nb()` in package **MASS** (note that `glm.nb()` uses `theta` to denote k):

```
> library(MASS)       # Provides the function  glm.nb()
> m.nb <- glm.nb( Count ~ log2(Dilution), data=pock )
> m.nb$theta          # This is the value of k (called theta in MASS)
[1] 9.892894
```

The output object `m.nb` includes information about the estimation of k. The output from `glm.nb()` model is converted to the style of output from `glm()` using `glm.convert()`:

```
> m.nb <- glm.convert(m.nb)
> printCoefmat(coef(summary(m.nb, dispersion=1)))
               Estimate Std. Error z value  Pr(>|z|)
(Intercept)     5.33284    0.08786  60.697 < 2.2e-16 ***
log2(Dilution) -0.72460    0.03886 -18.646 < 2.2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Note that we have to specify explicitly that the dispersion parameter is $\phi = 1$, because after using `glm.convert()`, R does not know automatically that the resulting GLM family should have dispersion equal to one.

Since $k \approx 10$, the negative binomial model is using the variance function $V(\mu) \approx \mu + \mu^2/10$. The coefficient of variation of the mixing distribution ($\psi = 1/k$) is estimated to be about 10%, a reasonable level for replicate to replicate variation. Comparing the Poisson and negative binomial models shows that the parameter estimates are reasonably close, but the standard errors are quite different:

```
> printCoefmat( coef( summary(m1)) )        # Poisson glm information
               Estimate Std. Error z value Pr(>|z|)
(Intercept)      5.2679     0.0226   233.6   <2e-16 ***
log2(Dilution)  -0.6809     0.0154   -44.1   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The diagnostic plots (Fig. 10.4, top panels) suggest the negative binomial model is adequate. No observations are particularly influential. □

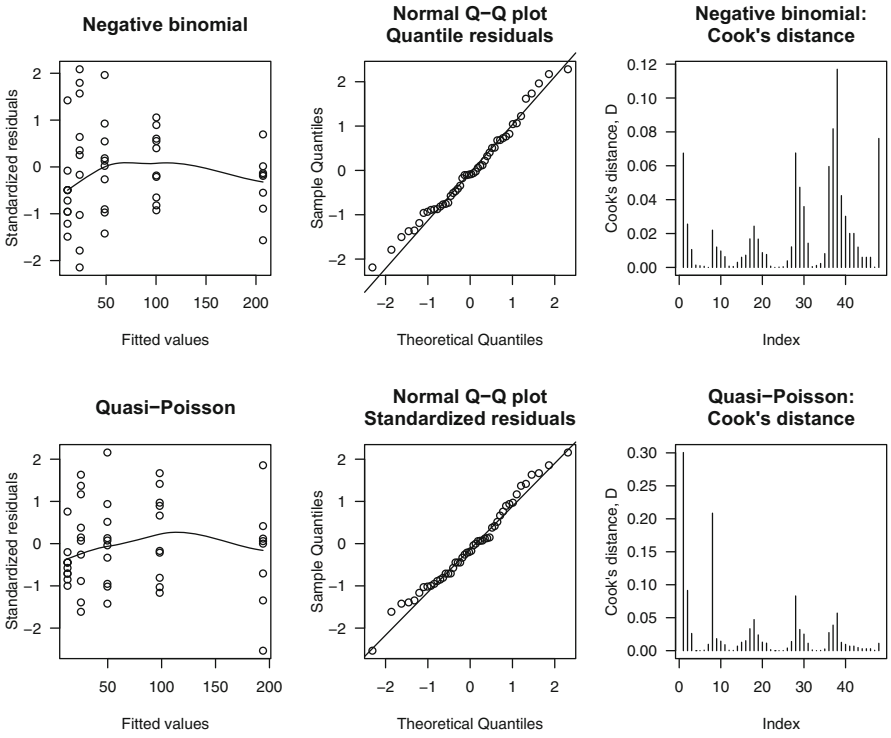

Fig. 10.4 Diagnostic plots from fitting the negative binomial model (top panels) and the quasi-Poisson models (bottom panels) to the pock data (Example 10.9)

10.5.3 Quasi-Poisson Models

The simplest to use, and therefore most commonly used, approach to overdispersed counts are quasi-Poisson models. Quasi-Poisson models keep the Poisson variance function $V(\mu) = \mu$ but simply allow a general positive dispersion parameter ϕ, so that $\text{var}[y_i] = \phi\mu_i$. Here $\phi > 1$ corresponds to overdispersion. This approach can be motivated in the same way as were quasi-binomial models (Sect. 9.8). Suppose that the counts y_i are counts of cases arising from a large population of size N, and the suppose that the individuals in the population are positively correlated. Then $\text{E}[y_i] = \mu_i = N\pi_i$, where π_i is the probability that a random individual is a case, and $\text{var}[y_i] = \phi N\pi_i(1 - \pi_i)$ where $\phi = 1 + (N - 1)\rho$ and ρ is the correlation between individuals. If N is large and the π_i are small, then $\text{var}[y_i] \approx \phi N\pi_i = \phi\mu_i$.

When $\phi \neq 1$, there is no EDM with this variance function that gives positive probability to integer values of y_i. Nevertheless, the quasi-likelihood methods of Sect. 8.10 still apply, so quasi-Poisson GLMs yield consistent estimators and

consistent standard errors for the β_j, provided only that $\mathrm{E}[y_i]$ and $\mathrm{var}[y_i]$ are correctly specified. Note that quasi-Poisson GLMs reduce to Poisson GLMs when $\phi = 1$.

The coefficient estimates from a quasi-Poisson GLM are identical to those from the corresponding Poisson GLM (since the estimates $\hat{\beta}_j$ do not depend on ϕ), but the standard errors are inflated by a factor of $\sqrt{\phi}$. Confidence intervals and statistics for testing hypotheses tests will change for the same reason.

Note that quasi-Poisson and the negative binomial model both produce overdispersion relative to the Poisson distribution but they assume different mean–variance relationships. Quasi-Poisson models assume a linear variance function ($V(\mu) = \phi\mu$) whereas negative binomial models uses a *quadratic* variance function ($V(\mu) = \mu + \mu^2/k$).

Quasi-Poisson models are fitted in R using `glm()` and specifying `family=quasipoisson()`. As for `family=poisson()`, the default link function is the `"log"` link, while `"identity"` and `"sqrt"` are also permitted. Since the quasi-Poisson model is not based on a probability model, the AIC is undefined. For the same reason, quantile residuals [12] cannot be computed for the quasi-Poisson GLM since no probability model is defined.

Example 10.11. The model fitted to the `pock` data shows overdispersion (Example 10.9), so an alternative solution is to fit a quasi-Poisson model:

```
> m.qp <- glm( Count ~ log2(Dilution), data=pock, family="quasipoisson")
```

The diagnostic plots (Fig. 10.4, bottom panels) suggest the quasi-Poisson model is broadly adequate, and no observations are particularly influential. It is discernible from the left panels of Fig. 10.4, however, that the negative binomial model tends to under-estimate slightly the variances of the low counts while the quasi-Poisson model does the same for large counts.

F-tests are used for model comparisons, since ϕ is estimated. Comparing the standard errors from the quasi-Poisson model to the standard errors produced from the Poisson GLM, the standard errors in the quasi-Poisson model are scaled by $\sqrt{\hat{\phi}}$:

```
> se.m1 <- coef(summary(m1))[, "Std. Error"]
> se.qp <- coef(summary(m.qp))[, "Std. Error"]
> data.frame(SE.Pois=se.m1, SE.Quasi=se.qp, ratio=se.qp/se.m1)
                  SE.Pois   SE.Quasi    ratio
(Intercept)    0.02255150 0.05677867 2.517733
log2(Dilution) 0.01544348 0.03888257 2.517733
> sqrt(summary(m.qp)$dispersion)
[1] 2.517733
```

Note that quantile residuals can be produced for the negative binomial GLM since a full probability function is defined, but quantile residuals cannot be computed for the quasi-Poisson GLM since no probability model is defined. For this reason, the residual plots for the quasi-Poisson model use standardized deviance residuals. The fitted systematic components are compared in

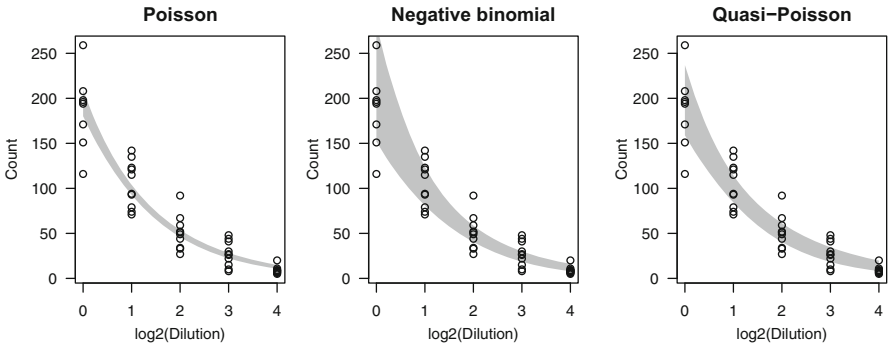

Fig. 10.5 Models fitted to the pock data, including the 99.9% confidence intervals for $\hat{\mu}$ (Example 10.11)

Fig. 10.5. Recall the Poisson and quasi-Poisson models produce identical parameter estimates, and hence fitted values.

```
> coef.mat <- rbind( coef(m1), coef(m.qp), coef(m.nb) )
> rownames(coef.mat) <- c("Poisson glm", "Quasi-Poisson", "Neg bin glm")
> coef.mat
              (Intercept) log2(Dilution)
Poisson glm      5.267932     -0.6809442
Quasi-Poisson    5.267932     -0.6809442
Neg bin glm      5.332844     -0.7245983
```

The plots in Fig. 10.5 show that the different approaches model the randomness differently.

We can now interpret the fitted model. The fitted models say that the expected number of pock marks decreased by a factor of about $\exp(-0.7) \approx 0.5$ for every 2-fold dilution. In other words, the expected number of pock marks is directly proportional to the concentration of the viral medium. □

10.6 Case Study

In a study of nesting female horseshoe crabs [1, 5], each with an attached male, the number of other nearby male crabs (called satellites) were counted (Table 10.10; data set: `hcrabs`). The colour of the female, the condition of her spine, her carapace width, and her weight were also recorded. The purpose of the study is to understand the factors that attract satellite crabs. Are they more attracted to larger females? Does the condition or colour of the female play a role?

Table 10.10 The horseshoe crab data (Example 10.6)

Colour	Spine condition	Carapace width (in cm)	Number of satellites	Weight (in g)
Medium	None OK	28.3	8	3050
Dark medium	None OK	22.5	0	1550
Light medium	Both OK	26.0	9	2300
Dark medium	None OK	24.8	0	2100
Dark medium	None OK	26.0	4	2600
Medium	None OK	23.8	0	2100
⋮	⋮	⋮	⋮	⋮

Colour is on a continuum from light to dark, and spine condition counts the number of intact sides, so we define both as ordered factors:

```
> data(hcrabs); str(hcrabs)
'data.frame':        173 obs. of  5 variables:
 $ Col   : Factor w/ 4 levels "D","DM","LM",..: 4 2 3 2 2 4 3 2 4 2 ...
 $ Spine: Factor w/ 3 levels "BothOK","NoneOK",..: 2 2 1 2 2 2 1 3 1 2 ...
 $ Width: num  28.3 22.5 26 24.8 26 23.8 26.5 24.7 23.7 25.6 ...
 $ Sat   : int  8 0 9 0 4 0 0 0 0 0 ...
 $ Wt    : int  3050 1550 2300 2100 2600 2100 2350 1900 1950 2150 ...
> hcrabs$Col <- ordered(hcrabs$Col, levels=c("LM", "M", "DM", "D"))
> hcrabs$Spine <- ordered(hcrabs$Spine,
                  levels=c("NoneOK", "OneOK", "BothOK"))
```

Plotting `Sat` against the other variables shows trends for more satellite crabs to congregate around females that are larger (in weight and width), are lighter in colour, and have no spinal damage (Fig. 10.6).

```
> with(hcrabs,{
   logSat <- log(Sat+1)
   plot( jitter(Sat) ~ Wt, ylab="Sat", las=1)
   plot( jitter(logSat) ~ log(Wt), ylab="log(Sat+1)", las=1)
   plot( logSat ~ Col, ylab="log(Sat+1)", las=1)
   plot( jitter(Sat) ~ Width, ylab="Sat", las=1)
   plot( jitter(logSat) ~ log(Width), ylab="log(Sat+1)", las=1)
   plot( logSat ~ Spine, ylab="log(Sat+1)", las=1)
 })
```

`jitter()` is used to avoid overplotting. Plots on the log-scale are preferable because the values of `Wt` and `Width` are distributed more symmetrically on the log-scale, and because the relationships between them and `Sat` are more likely to be relative rather than additive. `log(Sat+1)` is used to avoid taking logarithm of zero.

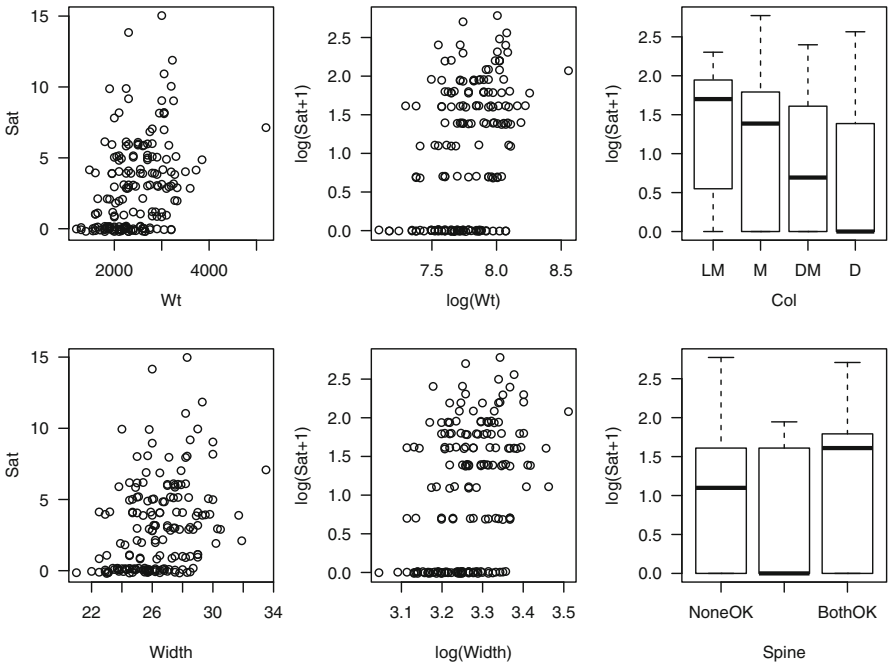

Fig. 10.6 The number of satellites on each female horseshoe crab plotted against the weight, colour, width and spine condition (Sect. 10.6)

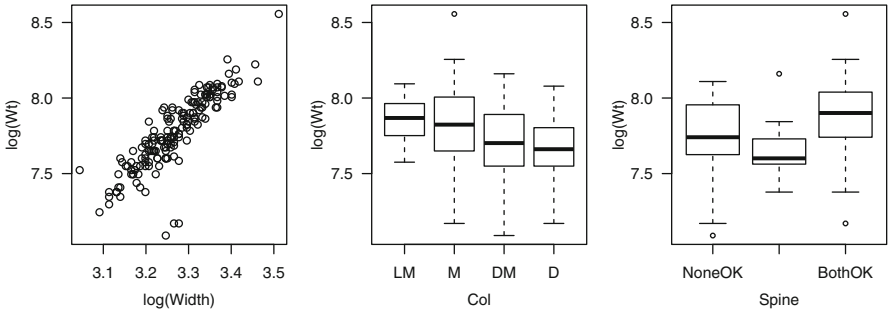

Fig. 10.7 Weight of each female horseshoe crab plotted against width, colour and spine condition (Sect. 10.6)

The explanatory variables are inter-related however; `Wt` is the most obvious overall summary of the size of each female. It turns out that lighter-coloured females are also typically heavier, as are females with no spine damage, so the relationships observed between `Sat` and `Col` and `Spine` might be explained by this (Fig. 10.7).

```
> with(hcrabs,{
   plot( log(Wt) ~ log(Width), las=1 )
   plot( log(Wt) ~ Col, las=1 )
   plot( log(Wt) ~ Spine, las=1 )
 })
> coef(lm( log(Wt) ~ log(Width), data=hcrabs ))
(Intercept)  log(Width)
      -0.60        2.56
```

`Wt` should be proportional to the volume of each female, hence should be approximately proportional to `Width^3`, if the females are all the same shape. Indeed, `log(Wt)` is nearly linearly related to `log(Width)` with a slope nearly equal to 3.

Crabs tend to congregate and interact with one another, rather than behaving independently, hence we should expect overdispersion *a priori* relative to Poisson for the counts of satellite crabs. We fit a quasi-Poisson GLM with log-link:

```
> cr.m1 <- glm(Sat ~ log(Wt) + log(Width) + Spine + Col,
              family=quasipoisson, data=hcrabs)
> anova(cr.m1, test="F")
           Df Deviance Resid. Df Resid. Dev     F  Pr(>F)
NULL                         172        633
log(Wt)     1     83.1       171        550 25.96 9.4e-07 ***
log(Width)  1      0.0       170        550  0.00    0.96
Spine       2      1.1       168        549  0.18    0.84
Col         3      7.6       165        541  0.79    0.50
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
> deviance(cr.m1)
[1] 541
> df.residual(cr.m1)
[1] 165
```

The residual deviance and Pearson X^2 are both more than three times the residual degrees of freedom, so our expectation of overdispersion seems confirmed. Using F-tests, `log(Wt)` is a highly significant predictor whereas none of the other variables are at all significant, after adjusting for `log(Wt)`. We adopt a model with just `Wt` as an explanatory variable:

```
> cr.m2 <- glm(Sat ~ log(Wt), family=quasipoisson, data=hcrabs)
> printCoefmat(coef(summary(cr.m2)), digits=3)
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  -12.568      2.664   -4.72  4.9e-06 ***
log(Wt)        1.744      0.339    5.15  7.0e-07 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

It is tempting to speculate on the biological implications. It might well be possible for a male crab to sense the overall weight of the female crab by smell or other chemical senses, because the amount of chemical emitted by

a female should be proportional to her size, whereas width, colour or spine damage would need vision. The results perhaps suggest that the crabs do not use vision as their primary sense.

We may worry that nearly half of the values of the response `Sat` are 0 or 1, which may suggest a problem for the distribution of the residual deviance and the evaluation of overdispersion. However a quick simulation shows that the chi-square approximation for the residual deviance is excellent:

```
> x <- log(hcrabs$Wt); dev <- rep(NA, 100)
> n <- length(hcrabs$Sat); mu <- fitted(cr.m2)
> for (i in 1:100) {
    y <- rpois(n, lambda=mu)  # Generate random Poisson values
    dev[i] <- glm(y~x, family=quasipoisson)$deviance
 }
> c(Mean.Dev=mean(dev), Std.Dev=sd(dev))
 Mean.Dev   Std.Dev
185.53962  19.61709
```

The mean and standard deviance of the residual deviance are close to their theoretical values of $\mathrm{df} = 171$ and $\sqrt{2 \times \mathrm{df}} = 18.5$ respectively, under the null hypothesis of Poisson variation. (Note: A χ^2 distribution with k degrees of freedom has mean k and standard deviation $\sqrt{2k}$.)

The diagnostics for this model suggest a reasonable model:

```
> plot( resid(cr.m2) ~ sqrt(fitted(cr.m2)), las=1,
    main="Deviance residuals", ylab="Deviance residuals",
    xlab="Square root of fitted values" )
> plot( cooks.distance(cr.m2), type="h", las=1,
    ylab="Cook's distance, D", main="Cook's distance")
> qqnorm( resid(cr.m2), las=1,
    main="Normal Q-Q plot\ndeviance residuals")
> qqline( resid(cr.m2))
```

Notice that quantile residuals cannot be used for the quasi-Poisson model; the trend in the bottom left of the Q–Q plot may be due to the use of deviance residuals (Fig. 10.8). No observation is identified as influential using Cook's distance or DFBETAS, but other criteria indicate influential observations:

```
> colSums( influence.measures(cr.m2)$is.inf )
  dfb.1_ dfb.l(W)    dffit    cov.r   cook.d      hat
       0        0        1        8        0        3
```

The quasi-Poisson model indicates that heavier crabs have more satellites on average. The fitted systematic component is

$$\log \mu = -12.57 + 1.744 \log W \quad \text{or equivalently} \quad \mu = 0.000003483 \times W^{1.744},$$

where W is the weight of the crabs in grams. If the regression coefficient for $\log W$ was 1, then the expected number of satellite crabs would be directly proportional to the weight of the female. The number of satellites seems to increase just a little faster than this.

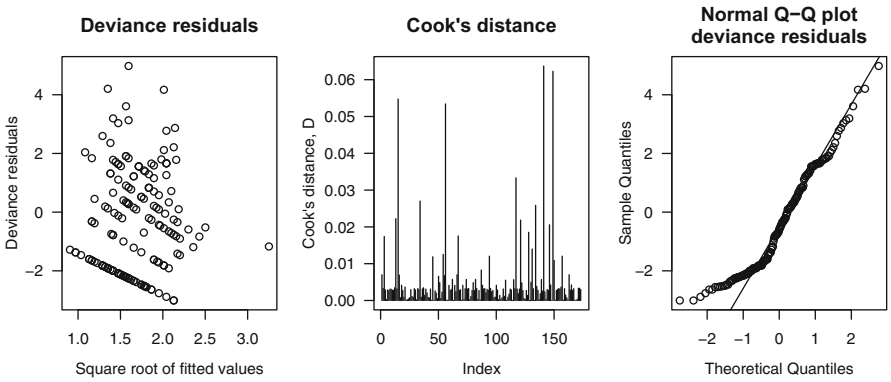

Fig. 10.8 Diagnostic plots for the quasi-Poisson model `cr.m2`. The deviance residuals against fitted values (left panel); Cook's distance (centre panel); a Q–Q plot of the quantile residuals (right panel) (Sect. 10.6)

An alternative model is to fit a negative binomial model:

```
> library(MASS)
> cr.nb <- glm.nb(Sat ~ log(Wt), data=hcrabs)
> cr.nb <- glm.convert(cr.nb)
> anova(cr.nb, dispersion=1, test="Chisq")


        Df Deviance Resid. Df Resid. Dev  Pr(>Chi)
NULL                     172     219.81
log(Wt)  1   23.339       171     196.47 1.358e-06 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1


> printCoefmat(coef(summary(cr.nb, dispersion=1)))
             Estimate Std. Error z value  Pr(>|z|)
(Intercept) -14.55581    3.10909 -4.6817 2.845e-06 ***
log(Wt)       1.99862    0.39839  5.0168 5.254e-07 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
> cr.nb$theta
[1] 0.9580286
```

The fitted negative binomial distribution uses $\hat{k} = 0.9580$. The diagnostic plots (not shown) indicate that the negative binomial model is also suitable. No observation is identified as influential using Cook's distance:

```
> colSums( influence.measures(cr.nb)$is.inf )
  dfb.1_ dfb.l(W)    dffit    cov.r   cook.d      hat
       0        0        0        6        0        3
```

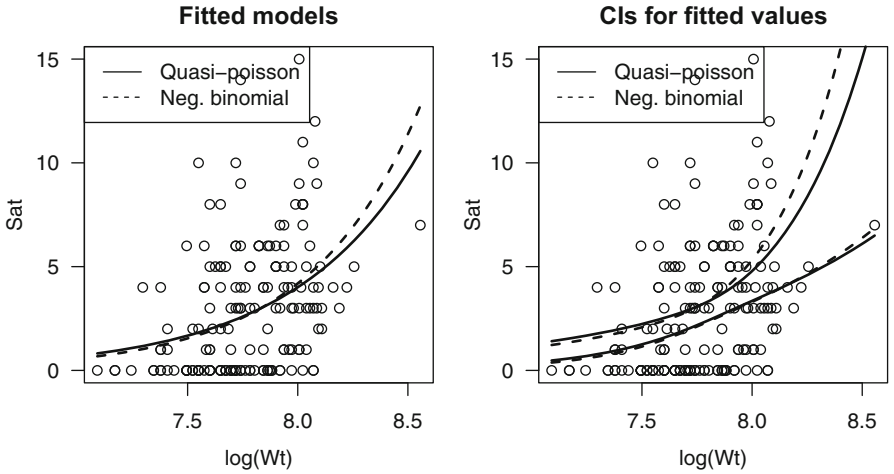

Fig. 10.9 Comparing the systematic components of the quasi-Poisson model and the negative binomial GLM (left panel) and the corresponding 95% confidence intervals (right panel) fitted to the horseshoe crab data. Solid lines represent the quasi-Poisson model, while dashed lines represent the negative binomial model

The differences between the two models becomes apparent for heavier crabs, for both the systematic components (Fig. 10.9, left panel) and the random components (Fig. 10.9, right panel). First, create predictions for a range of weights:

```
> newW <- seq( min(hcrabs$Wt), max(hcrabs$Wt), length=100)
> newS.qp <- predict(cr.m2, newdata=data.frame(Wt=newW), se.fit=TRUE)
> newS.nb <- predict(cr.nb, newdata=data.frame(Wt=newW), se.fit=TRUE,
                     dispersion=1)
> tstar <- qt(0.975, df=df.residual(cr.m2) ) # For a 95% CI
> ME.qp <- tstar * newS.qp$se.fit; ME.nb <- tstar * newS.nb$se.fit
> mu.qp <- newS.qp$fit; mu.nb <- newS.nb$fit
```

Then plot:

```
> par( mfrow=c(1, 2))
> plot( Sat~log(Wt), data=hcrabs, las=1, main="Fitted models")
> lines( exp(mu.qp) ~ log(newW), lwd=2 )
> lines( exp(mu.nb) ~ log(newW), lwd=2, lty=2 );
> legend("topleft", lty=1:2, legend=c("Quasi-poisson", "Neg. binomial") )
> #
> plot( Sat~log(Wt), data=hcrabs, las=1, main="CIs for fitted values")
> ci.lo <- exp(mu.qp - ME.qp); ci.hi <- exp(mu.qp + ME.qp)
> lines( ci.lo ~ log(newW), lwd=2); lines( ci.hi ~ log(newW), lwd=2)
> ci.lo <- exp(mu.nb - ME.nb); ci.hi <- exp(mu.nb + ME.nb)
> lines( ci.lo ~ log(newW), lwd=2, lty=2)
> lines( ci.hi ~ log(newW), lwd=2, lty=2)
> legend("topleft", lty=1:2, legend=c("Quasi-poisson", "Neg. binomial") )
```

10.7 Using R to Fit GLMs to Count Data

A Poisson GLM is specified in R using `glm(formula, family=poisson())` (note the lower case `p`). The link functions `"log"`, `"identity"`, and `"sqrt"` are permitted with Poisson distributions. Quasi-Poisson models are specified using `glm(formula, family=quasipoisson())`.

To fit negative binomial models, use `glm.nb()` from package **MASS** [37] when k is unknown (the usual situation). The output from `glm.nb()` is converted to the style of output from `glm()` using `glm.convert()`. Then, the usual `anova()` and `summary()` commands may be used, remembering to set `dispersion=1` when using `summary()`. See `?negative.binomial`, `?glm.nb`, and Sect. 10.5.2 for more information.

The function `gl()` is useful for generating factors occurring in a regular pattern, as is common in tabulated data. `gl(3, 2, 12)` produces a factor of length 12 with three levels (labelled `1`, `2` and `3` by default), appearing two at a time:

```
> gl(3, 2, 18, labels=c("A", "B", "C") )
 [1] A A B B C C A A B B C C A A B B C C
Levels: A B C
```

The functions `margin.table()` and `prop.table()` are useful for producing marginal tables and tables of proportions from raw data in tables (Sect. 10.4.5).

10.8 Summary

Chapter 10 considers fitting GLMs to count data. Counts are commonly modelled using the Poisson distribution (Sect. 10.2), where $\mu > 0$ is the expected count and $y = 0, 1, 2, \ldots$. Note that $\phi = 1$ and $V(\mu) = \mu$. The residual deviance $D(y, \hat{\mu})$ is suitably described by a $\chi^2_{n-p'}$ distribution if $\min\{y_i\} \geq 3$ (Sect. 10.2). The logarithmic link function is often used for Poisson GLMs (Sect. 10.2).

When any of the explanatory variables are quantitative, the fitted Poisson GLM is also called a Poisson regression model. When all the explanatory variables are qualitative, the fitted Poisson GLM is also called a log-linear model (Sect. 10.2).

Poisson GLMs can be used to model rates (such as counts of cancer cases per unit of population) by using a suitable offset in the linear predictor (Sect. 10.3).

Count data often appear cross-classified in tables, commonly called contingency tables (Sect. 10.4). Contingency tables may arise under various sampling schemes, each implying a different random component (Sect. 10.4). How-

ever, in all cases a Poisson GLM can be fitted provided the coefficients in the linear predictor corresponding to fixed margins are included in the model.

Three-dimensional tables may be interpreted, and possibly simplified, according to which interactions are present in the model (Sect. 10.4.4). If tables are collapsed incorrectly, the resulting tables may be misleading. Simpson's paradox is an extreme example (Sect. 10.4.5). Poisson GLMs fitted to higher-order tables may be difficult to interpret (Sect. 10.4.7).

Contingency tables may contain cells with zero counts (Sect. 10.4.8). Sampling zeros occur by chance, and larger samples may produce counts in these cells. Structural zeros appear for impossible events, so cells containing structural zeros must be removed from the analysis.

Overdispersion occurs when the variation in the responses is greater than expected under the Poisson model (Sect. 10.5). Possible causes are that the model is misspecified (in which case the model should be amended), the means are not constant, or the responses are not independent.

In cases of overdispersion relative to the Poisson GLM, a negative binomial distribution may be used, which is an EDM if k is known (Sect. 10.5.2). For the negative binomial distribution, $V(\mu) = \mu + \mu^2/k$ for $k > 0$. The value of k usually needs to be estimated (by $\hat{k}$) for a negative binomial GLM (Sect. 10.5.2). If overdispersion is observed, a quasi-Poisson model may be fitted also, which assumes $V(\mu) = \phi\mu$ (Sect. 10.5.3).

Problems

Selected solutions begin on p. 541.

10.1. Consider the negative binomial distribution, whose probability function is given in (10.11).

1. Show that the negative binomial distribution with known k is an EDM, by identifying θ, $\kappa(\theta)$ and ϕ. (See Sect. 5.3.6, p. 217.)
2. Show that the negative binomial distribution with known k has $\text{var}[y] = \mu + \mu^2/k$.
3. Deduce the canonical link function for the negative binomial distribution.
4. Show that, for the negative binomial distribution,

$$d(y, \mu) = 2\left\{ y \log \frac{y}{\mu} - (y + k) \log \frac{y + k}{\mu + k} \right\}$$

for $y > 0$. Also, deduce the unit deviance when $y = 0$.

10.2. If the fitted Poisson GLM includes a constant term, and the logarithmic link function is used, the sum over the observations of the second term in the expression for the residual deviance is zero. In other words, $\sum_{i=1}^{n}(y_i - \hat{\mu}_i) = 0$. Prove this result by writing the log-likelihood for a model with linear

predictor containing a constant term, say β_0, differentiating the log-likelihood with respect to β_0, setting to zero, and solving.

10.3. Sometimes, count data explicitly omit zero counts. Examples include the numbers of days patients spend in hospital (only patients who actually stay overnight in hospital are considered, and so the smallest possible count is one); the number of people per car using a rural road (the driver at least must be in the car); and a survey of the number of people living in each household (to respond, the households must have at least one person). Using a Poisson distribution is inadequate, as the zero counts will be modelled as true zero counts.

In these situations, the zero-truncated Poisson distribution may be suitable, with probability function

$$\mathcal{P}(y;\lambda) = \frac{\exp(-\lambda)\lambda^y}{\{1-\exp(-\lambda)\}y!},$$

where $y = 1, 2, \ldots$ and $\lambda > 0$.

1. Show that the truncated Poisson distribution is an EDM by identifying θ and $\kappa(\theta)$.
2. Show that $\mu = \mathrm{E}[y] = \lambda/\{1-\exp(-\lambda)\}$, and that $\mu > 1$.
3. Find the variance function for the truncated Poisson distribution.
4. Plot the truncated Poisson distribution and the Poisson distribution for $\lambda = 2$, and compare.

10.4. A study [25] used a Poisson GLM to model the number of politicians switching political parties in the USA. The response variable was the number of members of the House of Representatives who switched parties every year from 1802–1876.

1. Explain why the authors used a Poisson GLM to model the data.
2. The authors use eleven possible explanatory variables in the linear predictor. One of the explanatory variables is whether or not the year is an election year (election years are coded as 0, non-election years as 1). The coefficient for this explanatory variable is 1.051. Interpret the meaning of this coefficient.
3. The estimated standard error for the election year parameter is 0.320. Determine if the parameter is statistically significant.
4. Compute and interpret a 90% confidence interval for the election year parameter.

10.5. A study in the USA [22] examined the number of pregnancies in a stratified random sample of 1154 sexually-active teenage girls (7th to 12th grade). Details of the fitted Poisson GLM are shown in Table 10.11.

1. Explain why the years of sexual activity is used as an offset.

Table 10.11 The fitted Poisson GLMs for the teenage pregnancy data. The response variable is the number of pregnancies. All variables are binary (0: no; 1: yes) apart from age, which is measured in completed years. Years of sexual activity is used as an offset (Problem 10.5)

	df	$\hat{\beta}_j$	se($\hat{\beta}_j$)	Wald 95% confidence limits		Deviance
Intercept	1	−2.0420	0.9607	−3.9248	−0.1591	4.52
Current age (in years)	1	0.1220	0.0543	0.0156	0.2283	5.05
Race ('White' is the reference)						
African-American	1	0.6604	0.1287	0.4082	0.9126	26.33
Hispanic	1	0.2070	0.2186	−0.2215	0.6354	0.90
Asian	1	0.4896	0.3294	−0.1561	1.1352	2.21
Single	1	−0.9294	0.2080	−1.3371	−0.5218	19.97
College plans	1	−0.0871	0.0515	−0.1881	0.0139	2.86
Contraceptive self-efficacy	1	−0.2241	0.0845	−0.3897	−0.0585	7.04
Consistent use of contraceptives	1	−0.2729	0.0825	−0.4346	−0.1113	10.95
Residual df:	1144					
Residual deviance:	3359.9					

2. Use likelihood ratio tests to identify statistically significant explanatory variables.
3. Use the Wald statistics to identify statistically significant explanatory variables. Compare to the results of using the likelihood ratio test.
4. Interpret the coefficients in the model.
5. Show that overdispersion may be present.
6. Because of the possible overdispersion, estimate ϕ for the quasi-Poisson model. Hence compute $\hat{\beta}_j$ and se($\hat{\beta}_j$) for the quasi-Poisson GLM.
7. Form a 95% confidence interval for age using the quasi-Poisson GLM.

10.6. The brood sizes of blue tits were experimentally changed (increased or decreased) through three brooding seasons to study the survival of offspring [32, Table 2]. The hypothesis was that blue tits should produce the clutch size maximizing the survival of their offspring (so that manipulated broods should show less surviving offspring than unmanipulated broods). In other words, the number of eggs laid is optimum given the ability of the parents to rear the offspring (based on their body condition, food resources, age, etc.). A log-linear model for modelling the number of offspring surviving y produced the results in Table 10.12, where M is the amount of manipulation (ranging from taking ten eggs ($M = -10$) to adding four eggs ($M = 4$) to the clutch), and C is the original clutch size (ranging from two to 17 eggs).

1. Write down the fitted model from Table 10.12 (where $\hat{\beta}_0 = -2.928$).
2. Using likelihood ratio tests, determine which explanatory variables are significant.
3. Use Wald statistics to determine the significance of each parameter. Compare to the results from the likelihood ratio tests, and comment.

Table 10.12 The analysis of deviance table for a Poisson GLM fitted to the blue tits data. The response variable is the number of offspring surviving (Problem 10.6)

Model	Residual deviance	df	$\hat{\beta}_j$	se($\hat{\beta}_j$)
Null model	732.74	617		
$+ C$	662.25	616	0.238	0.028
$+ M$	649.01	615	0.017	0.035
$+ M^2$	637.22	614	−0.028	0.009

Table 10.13 Information about the fitted Poisson GLM for the *spina bifida* study. The response variable is the number of babies born with *spina bifida* (Problem 10.7)

Model	Residual deviance	df	$\hat{\beta}_j$	se($\hat{\beta}_j$)	
Null	554.11	200			
$+ \log B$	349.28	199	1.06	0.07	
$+ S$	305.32	197	−8.61	0.68	(routine screening)
			−8.18	0.67	(no routine screening)
			−8.43	0.68	(policy uncertain)
$+ C$	285.06	196	−0.11	0.03	
$+ U$	266.88	195	0.046	0.009	
$+ A$	256.03	194	0.039	0.011	

4. Compute and interpret the 95% confidence interval for the effect of the original clutch size C.
5. Comment on under- or overdispersion for this model.
6. Using the fitted model, determine the value of M maximizing expected offspring survival μ.
7. Determine if any manipulation of the clutch size decreases the survival chances of the young.

10.7. A study of *spina bifida* in England and Wales [27] examined the relationship between the number of babies born with *spina bifida* between 1983 and 1985 inclusive in various Regional Health Authorities (RHA), and explanatory variables such as the total number of live and still births between 1983–1985, B; the screening policy of the health authority in 1982, S (routine; non-routine; uncertain); the percentage of female residents born in the Caribbean, C; the percentage economically-active residents unemployed, U; the percentage of residents lacking a car, L; and the percentage of economically-active residents employed in agriculture, A. A Poisson GLM with a log-link was fitted (Table 10.13) to model the number of babies born with *spina bifida*.

1. Write down the fitted model. (Note that a different constant term is fitted for each screening policy.)
2. Using the standard errors, check which parameters are significantly different from zero.
3. Use likelihood ratio tests to determine which explanatory variables are significant in the model.

4. Interpret the effect of the unemployment rate U.
5. Compute and interpret the 95% confidence interval for the effect of the unemployment rate U.
6. Explain why using $\log B$ as an offset seems reasonable from the description of the data. Also explain why Table 10.13 supports this approach.
7. Is overdispersion likely to be a problem?

10.8. For the depressed youth data used in Sect. 10.4.7 (p. 393), fit the model used in that section as follows (data set: `dyouth`).

1. Show that the four-factor interaction is not significant.
2. Show that only one three-factor interaction is significant in the model.
3. Then show that four two-factor interactions are needed in the model (some because they are significant, some because of the marginality principle).
4. Show that the model is adequate by examining the model diagnostics.

10.9. Consider the Danish lung cancer data of Example 10.1 (data set: `danishlc`). In that example, a Poisson GLM was fitted to model the *number* of lung cancers per unit of population.

1. Fit a model for the *proportion* of lung cancers, based on the proportion `Cases/Pop`, and compare to the equivalent Poisson GLM fitted in Sect. 10.3.
2. Show that the conditions for the equivalence of the binomial and Poisson GLMs, as given in Sect. 10.4.6, are approximately satisfied.

10.10. In Sect. 8.12 (p. 322), a Poisson GLM was fitted to the noisy miner data [30] (data set: `nminer`) that was first introduced in Example 1.5 (p. 14). In Example 1.5, the only explanatory variable considered was the number of eucalypts `Eucs`, but the data frame actually contains a number of other explanatory variables: the number of buloke trees (`Bulokes`); the area in hectares of remnant patch vegetation at each site (`Area`); whether the area was grazed (`Grazed`: `1` means yes); and whether shrubs were present in the transect (`Shrubs`: `1` means yes).

1. Find a suitable Poisson regression model for modelling the number of noisy miners `Minerab`, including a diagnostic analysis.
2. Is the saddlepoint approximation likely to be accurate? Explain.

10.11. The number of deaths for 1969–1973 (1969–1972 for Belgium) due to cervical cancer is tabulated (Table 10.14; data set: `cervical`) by age group for four different countries [19, 38].

1. Plot the data, and discuss any prominent features.
2. Explain why an offset is useful when fitting a GLM to the data.
3. Fit a Poisson GLM with `Age` and `Country` as explanatory variables. Produce the plot of residuals against fitted values, and evaluated the model.

Table 10.14 The number of deaths y due to cervical cancer and woman-years at-risk T in various age groups, for four countries (Problem 10.11)

	25–34		35–44		45–54		55–64	
Country	y	T	y	T	y	T	y	T
England and Wales	192	15,399	860	14,268	2762	15,450	3035	15,142
Belgium	8	2328	81	2557	242	2268	268	2253
France	96	15,324	477	16,186	998	14,432	1117	13,201
Italy	45	19,115	255	18,811	621	16,234	839	15,246

Table 10.15 The number of women developing depression in a 1-year period in Camberwell, South London [15]. SLE refers to a 'Severe Life Event' (Example 6.2)

	Three children under 14		Other women	
	SLE	No SLE	SLE	No SLE
Depression	9	0	24	4
No depression	12	20	119	231

4. Fit the corresponding quasi-Poisson model. Produce the plot of residuals against fitted values, and evaluated the model.
5. Fit the corresponding negative binomial GLM. Produce the plot of residuals against fitted values, and evaluated the model.
6. Which model seems appropriate, if any?

10.12. In a study of depressed women [15], women were classified into groups (Table 10.15; data set: `dwomen`) based on their depression level (`Depression`), whether a severe life event had occurred in the last year (`SLE`), and if they had three children under 14 at home (`Children`). Model these counts using a Poisson GLM, and summarize the data if possible.

10.13. The number of severe and non-severe cyclones in the Australian region between 1970 and 2005 were recorded (Table 10.16; data set: `cyclones`), together with a climatic index called the *Ocean Niño Index*, or ONI. The ONI is a 3-month running mean of sea surface temperature anomalies; Table 10.16 shows the ONI at four times during each year.

1. Plot the number of severe cyclones against the ONI, and then plot the number of non-severe cyclones against the ONI. Comment.
2. Fit a Possion GLM to model the number of severe cyclones, and another Poisson GLM for the number of non-severe cyclones.
3. Interpret your final models.

10.14. A study [13, 18] of the species richness (the number of species) of ants at 22 sites in New England, USA, examined relationships with habitat (forest

Table 10.16 The number of severe and non-severe cyclones in the Australian region, with four values of the Ocean Niño Index (ONI) for each year (Problem 10.13)

	Number of cyclones		ONI			
Year	Severe	Non-severe	JFM	AMJ	JAS	OND
1969	3	7	1.0	0.6	0.4	0.8
1970	3	14	0.3	0.0	−0.8	−0.9
1971	9	7	−1.3	−0.8	−0.8	−1.0
1972	6	6	−0.4	0.5	1.3	2.0
1973	4	15	1.2	−0.6	−1.3	−2.0
1974	3	13	−1.7	−0.9	−0.5	−0.9
⋮	⋮	⋮	⋮	⋮	⋮	⋮

Table 10.17 Species richness of ants in New England, USA. Elevation is in metres (Problem 10.14)

		Species richness in:				Species richness in:	
Elevation	Latitude	Forest	Bog	Elevation	Latitude	Forest	Bog
41.97	389	6	5	42.57	335	10	4
42.00	8	16	6	42.58	543	4	2
42.03	152	18	14	42.69	323	5	7
42.05	1	17	7	43.33	158	7	2
42.05	210	9	4	44.06	313	7	3
42.17	78	15	8	44.29	468	4	3
42.19	47	7	2	44.33	362	6	2
42.23	491	12	3	44.50	236	6	3
42.27	121	14	4	44.55	30	8	2
42.31	95	9	8	44.76	353	6	5
42.56	274	10	8	44.95	133	6	5

or bog), elevation (in m) and latitude (Table 10.17; data set: `ants`). Find a suitable model for the data. Interpret your final model.

10.15. A study [14, 17, 33] compared the number polyps in patients with familial adenomatous polyposis (Table 10.18; data set: `polyps`), after treatment with a new drug (sulindac) or a placebo.

1. Plot the data and comment.
2. Find a suitable Poisson GLM for modelling the data, and show that overdispersion exists.
3. Fit a quasi-Poisson model to the data.
4. Fit a negative binomial GLM to the data.
5. Decide on a final model.

10.16. An experiment [21] compared the density of understorey birds at a series of sites in two areas either side of a stockproof fence (Table 10.19;

Table 10.18 The number of polyps in the treatment and placebo group for patients with famial adenomatous polyposis (Problem 10.15)

Treatment group				Placebo group			
Number	Age	Number	Age	Number	Age	Number	Age
1	22	17	22	7	34	44	19
1	23	25	17	10	30	46	22
2	16	33	23	15	50	50	34
3	23			28	18	61	13
3	23			28	22	63	20
4	42			40	27		

Table 10.19 The number of understorey-foraging birds observed in three 20-min surveys of 2 ha quadrats either side of a stockproof fence, before and after grazing (Problem 10.16)

Ungrazed				Grazed					
Before	After	Before	After	Before	After	Before	After	Before	After
0	1	37	5	2	6	0	0	30	4
3	10	7	5	0	2	1	3	13	14
1	10	10	4	0	0	0	7	0	6
19	29	11	4	1	11	4	17	2	8
8	21	1	6	4	7	0	7	0	18
		30	15	2	4	0	0	1	4
				3	3	2	7		

data set: `grazing`). One side had limited grazing (mainly from native herbivores), and the other was heavily grazed by feral herbivores, mostly horses. Bird counts were recorded at the sites either side of the fence (the 'before' measurements). Then the herbivores were removed, and bird counts recorded again (the 'after' measurements). The measurements are the total number of understorey-foraging birds observed in three 20-min surveys of 2 ha quadrats.

1. Plot the data, and explain the important features.
2. Fit a Poisson GLM with systematic component `Birds ~ When * Grazed`, ensuring a diagnostic analysis.
3. Show that overdispersion exists. Demonstrate by computing the mean and variance of each combination of the explanatory variables.
4. Fit a quasi-Poisson model.
5. Fit a negative binomial GLM.
6. Compare all three fitted models to determine a suitable model.
7. Interpret the final model.

10.17. An experiment [23, 36] recorded the time to failure of a piece of electronic equipment while operating in two different modes. In any session, the machine is run in both modes for varying amounts of time (Table 10.20; data

Table 10.20 Observations on electronic equipment failures. The time spent in each mode is measured in weeks (Problem 10.17)

Time spent in Mode 1	Time spent in Mode 2	Number of failures	Time spent in Mode 1	Time spent in Mode 2	Number of failures
33.3	25.3	15	116.3	53.6	27
52.2	14.4	9	131.7	56.6	23
64.7	32.5	14	85.0	87.3	18
137.0	20.5	24	91.9	47.8	22
125.9	97.6	27			

Table 10.21 The estimated number of deaths for the five leading cancer sites in Canada in 2000, by geographic region and gender (Problem 10.18)

	Ontario		Newfoundland		Quebec	
Cancer	Male	Female	Male	Female	Male	Female
Lung	3500	2400	240	95	3500	2000
Colorectal	1250	1050	60	50	1100	1000
Breast	0	2100	0	95	0	1450
Prostate	1600	0	80	0	900	0
Pancreas	540	590	20	25	390	410
Estimated population:	11,874,400		533,800		7,410,500	

set: `failures`). For each operating period, Mode 1 is the time spent operating in one mode and Mode 2 is the time spent operating in the other mode. The number of failures in each period is recorded, where each operating period is measured in weeks. The interest is in finding a model for the number of failures given the amount of time the equipment spends in the two modes.

1. Plot the number of failures against the time spent in Mode 1, and then against the time spent in Mode 2.
2. Show that an identity link function may be appropriate.
3. Fit the Poisson model, to model the number of failures as a function of the time spent in the two modes. Which mode appears to be the major source of failures?
4. Is there evidence of under- or overdispersion?
5. Interpret the final model.

10.18. A report on Canadian cancer statistics estimated the number of deaths from various types of cancer in Canada in 2000 [7]. The five leading cancer sites are studied here (Table 10.21; data set: `ccancer`).

1. Plot the cancer rates per thousand of population against each geographical location, and then against gender. Comment on the relationships.
2. Identify the zeros as systematic or sampling.
3. Find an appropriate model for the data using an appropriate offset. Do the cancer rates appear to differ across the geographic regions?
4. Interpret the fitted model.

Table 10.22 Health concerns of teenagers (Problem 10.20)

		Health concern			
Sex	Age group	Sex; relationships	Menstrual	How healthy	Nothing at all
Males	12–15	4	0	42	57
	16–17	2	0	7	20
Females	12–15	9	4	19	71
	16–17	7	8	10	31
Total		22	12	78	179

Table 10.23 Smoking and survival data for Whickham women (Problem 10.21)

	Smokers		Non-smokers	
Age (at first survey)	Alive	Dead	Alive	Dead
18–24	53	2	61	1
25–34	121	3	152	5
35–44	95	14	114	7
45–54	103	27	66	12
55–64	64	51	81	40
65–74	7	29	28	101
75+	0	13	0	64

10.19. In Problem 2.18 (p. 88), data were presented about children building towers out of building blocks (data set: `blocks`). One variable measured was the number of blocks needed to build a tower as high as possible. Find a model for the number of blocks, including a diagnostic analysis.

10.20. A study [6, 9, 16] asked teenagers about their health concerns, including sexual health. The data in Table 10.22 (data set: `teenconcerns`) are the number of teenagers who reported wishing to talk to their doctor about the indicated topic.

1. How would you classify the zeros? Explain.
2. Fit an appropriate log-linear model to the data.

10.21. A survey originally conducted in 1972–1974 [3, 10] asked women in Whickham in the north of England about their smoking habits and age, and recorded their survival (Table 10.23; data set: `wwomen`). A subsequent survey 20 years later followed up the women to determine how many women from the original survey had died.

1. Classify the zeros as sampling or structural zeros.
2. Plot the proportion of women alive at each age (treat age as continuous, using the lower boundary of each class), distinguishing between smokers and non-smokers. Comment.

3. Compute the *overall* percentage of smokers and non-smokers alive, and comment.
4. Compute the percentage of smokers and non-smokers *in each age group* who died. Compare to the previous answers. Comment and explain.
5. Fit a suitable log-linear model for the *number* of women alive. What evidence is there that the data should not be collapsed over age?

References

[1] Agresti, A.: An Introduction to Categorical Data Analysis, second edn. Wiley-Interscience, New York (2007)
[2] Andersen, E.B.: Multiplicative Poisson models with unequal cell rates. Scandinavian Journal of Statistics **4**, 153–158 (1977)
[3] Appleton, D.R., French, J.M., Vanderpump, M.P.J.: Ignoring a covariate: An example of Simpson's paradox. The American Statistician **50**, 340–341 (1996)
[4] Berkeley, E.C.: Right answers—a short guide for obtaining them. Computers and Automation **18**(10) (1969)
[5] Brockmann, H.J.: Satellite male groups in horseshoe crabs, *limulus polyphemus*. Ethology **102**, 1–21 (1996)
[6] Brunswick, A.F.: Adolescent health, sex, and fertility. American Journal of Public Health **61**(4), 711–729 (1971)
[7] Canadian Cancer Society: Canadian cancer statistics 2000. Published on the internet: www.cancer.ca/stats2000/tables/tab5e.htm (2000). Accessed 19 September 2001
[8] Charig, C.R., Webb, D.R., Payne, S.R., Wickham, J.E.A.: Comparison of treatment of renal calculi by open surgery, percutaneous nephrolithotomy, and extracorporeal shockwave lithotripsy. British Medical Journal **292**, 879–882 (1986)
[9] Christensen, R.: Log-Linear Models. Springer Texts in Statistics. Springer, New York (2013)
[10] Davison, A.C.: Statistical Models. Cambridge University Press, UK (2003)
[11] Dunn, P.K.: Contingency tables and log-linear models. In: K. Kempf-Leonard (ed.) Encyclopedia of Social Measurement, pp. 499–506. Elsevier (2005)
[12] Dunn, P.K., Smyth, G.K.: Randomized quantile residuals. Journal of Computational and Graphical Statistics **5**(3), 236–244 (1996)
[13] Ellison, A.M.: Bayesian inference in ecology. Ecology Letters **7**, 509–520 (2004)
[14] Everitt, B.S., Hothorn, T.: A Handbook of Statistical Analyses using, second edn. Chapman & Hall/CRC, Boca Raton, FL (2010)

[15] Everitt, B.S., Smith, A.M.R.: Interactions in contingency tables: A brief discussion of alternative definitions. Psychological Medicine **9**, 581–583 (1979)
[16] Fienberg, S.: The Analysis of Cross-Classified Categorical Data. Springer, New York (2007)
[17] Giardiello, F.M., Hamilton, S.R., Krush, A.J., Piantadosi, S., Hylind, L.M., Celano, P., Booker, S.V., Robinson, C.R., Johan, G., Offerhaus, A.: Treatment of colonic and rectal adenomas with sulindac in famial adenomatous polyposis. New England Journal of Medicine **328**(18), 1313–1316 (1993)
[18] Gotelli, N.J., Ellison, A.M.: Biogeography at a regional scale: Determinants of ant species density in bogs and forests of New England. Ecology **83**(6), 1604–1609 (2002)
[19] Hand, D.J., Daly, F., Lunn, A.D., McConway, K.Y., Ostrowski, E.: A Handbook of Small Data Sets. Chapman and Hall, London (1996)
[20] Health Department of Western Australia: Annual report 1997/1998—health of Western Australians—mortality and survival. Published on the internet: www.health.wa.gov.au/Publications/annualreport_9798/. Accessed 19 September 2001
[21] Howes, A.L., Maron, M., McAlpine, C.A.: Bayesian networks and adaptive management of wildlife habitat. Conservation Biology **24**(4), 974–983 (2010)
[22] Hutchinson, M.K., Holtman, M.C.: Analysis of count data using Poisson regression. Research in Nursing and Health **28**, 408–418 (2005)
[23] Jorgensen, D.W.: Multiple regression analysis of a Poisson process. Journal of the American Statistical Association **56**(294), 235–245 (1961)
[24] Julious, S.A., Mullee, M.A.: Confounding and Simpson's paradox. British Medical Journal **309**(1480), 1480–1481 (1994)
[25] King, G.: Statistical models for political science event counts: Bias in conventional procedures and evidence for the exponential Poisson regression model. American Journal of Political Science **32**(3), 838–863 (1988)
[26] Lindsey, J.K.: Modelling Frequency and Count Data. No. 15 in Oxford Statistical Science Series. Clarendon Press, Oxford (1995)
[27] Lovett, A.A., Gatrell, A.C.: The geography of *spina bifida* in England and Wales. Transactions of the Institute of British Geographers (New Series) **13**(3), 288–302 (1988)
[28] Luo, D., Wood, G.R., Jones, G.: Visualising contingency table data. The Australian Mathematical Society Gazette **31**(4), 258–262 (2004)
[29] Maag, J.W., Behrens, J.T.: Epidemiologic data on seriously emotionally disturbed and learning disabled adolescents: Reporting extreme depressive symptomatology. Behavioral Disorders **15**(1) (1989)
[30] Maron, M.: Threshold effect of eucalypt density on an aggressive avian competitor. Biological Conservation **136**, 100–107 (2007)

[31] Norton, J., Lawrence, G., Wood, G.: The Australian public's perception of genetically-engineered foods. Australasian Biotechnology pp. 172–181 (1998)

[32] Pettifor, R.A.: Brood-manipulation experiments. I. The number of offspring surviving per nest in blue tits (*Parus caeruleus*). Journal of Animal Ecology **62**, 131–144 (1993)

[33] Piantadosi, S.: Clinical Trials: A Methodologic Perspective, second edn. John Wiley and Sons, New York (2005)

[34] Siegel, R.L., Miller, K.D., Jemal, A.: Cancer statistics, 2015. CA: A Cancer Journal for Clinicians **65**(1), 5–29 (2015)

[35] Smith, P.T., Heitjan, D.F.: Testing and adjusting for departures from nominal dispersion in generalized linear models. Journal of the Royal Statistical Society, Series C **42**(1), 31–41 (1993)

[36] Smyth, G.K.: Australasian data and story library (OzDASL) (2011). URL http://www.statsci.org/data

[37] Venables, W.N., Ripley, B.D.: Modern Applied Statistics with S, fourth edn. Springer-Verlag, New York (2002). URL http://www.stats.ox.ac.uk/pub/MASS4

[38] Whittemore, A.S., Gong, G.: Poisson regression with misclassified counts: Applications to cervical cancer mortality rates. Journal of the Royal Statistical Society, Series C **40**(1), 81–93 (1991)

Chapter 11
Positive Continuous Data: Gamma and Inverse Gaussian GLMs

It has been said that data collection is like garbage collection: before *you collect it you should have in mind what you are going to do with it.*
Fox, Garbuny and Hooke [6, p. 51]

11.1 Introduction and Overview

This chapter considers models for positive continuous data. Variables that take positive and continuous values often measure the amount of some physical quantity that is always present. The two most common GLMs for this type of data are based on the gamma and inverse Gaussian distributions. Judicious choice of link function and transformations of the covariates ensure that a variety of relationships between the response and explanatory variables can be modelled. Modelling positive continuous data is introduced in Sect. 11.2, then the two most common EDMs for modelling positive continuous data are discussed: gamma distributions (Sect. 11.3) and inverse Gaussian distributions (Sect. 11.4). The use of link functions is then addressed (Sect. 11.5). Finally, estimation of ϕ is considered in Sect. 11.6.

11.2 Modelling Positive Continuous Data

Many applications have response variables which are continuous and positive. Such variables usually have distributions that are right skew, because the boundary at zero limits the left tail of the distribution. If the values of such a variable vary by orders of magnitude, then such skewness is inevitable. Another consequence of the boundary at zero is that the variance of the response must generally approach zero as the expected value approaches zero, provided the structure of the distribution remains otherwise the same (Sect. 4.2). Positive continuous data therefore usually shows an increasing mean–variance relationship.

P. K. Dunn, G. K. Smyth, *Generalized Linear Models with Examples in R*,
Springer Texts in Statistics, https://doi.org/10.1007/978-1-4419-0118-7_11

Table 11.1 Measurements from small-leaved lime trees in Russia, grouped by the origin of the tree. Foliage refers to the foliage biomass, and DBH refers to the 'diameter at breast height' (Example 11.1)

Natural			Coppice			Planted		
Foliage (in kg)	DBH (in cm)	Age (in years)	Foliage (in kg)	DBH (in cm)	Age (in years)	Foliage (in kg)	DBH (in cm)	Age (in years)
0.10	4.00	38	0.27	7.20	24	0.92	16.40	38
0.20	6.00	38	0.03	3.10	11	3.69	18.40	38
0.40	8.00	46	0.04	3.30	12	0.82	12.80	37
0.60	9.60	44	0.03	3.10	11	1.09	14.10	42
0.60	11.30	60	0.01	3.30	12	0.08	6.40	35
0.80	13.70	56	0.07	3.30	12	0.59	12.00	32
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮

Apart from $V(\mu) = \mu$, which we have already seen corresponds to count data, the simplest increasing variance function functions are $V(\mu) = \mu^2$ and $V(\mu) = \mu^3$, which correspond to the gamma and inverse Gaussian distributions respectively. For these reasons, GLMs based on the gamma and inverse Gaussian distributions are useful for modelling positive continuous data. The gamma distribution corresponds to ratio data with constant coefficient of variation. A gamma GLM is specified in R using `family=Gamma()`, and an inverse Gaussian GLM using `family=inverse.gaussian()`.

Example 11.1. A series of studies [22] sampled the forest biomass in Eurasia [21]. Part of that data, for small-leaved lime trees (*Tilia cordata*), is shown in Table 11.1 (data set: `lime`).

A model for the foliage biomass y is sought. The foliage mostly grows on the outer canopy, which could be crudely approximated as a spherical shape, so one possible model is that the mean foliage biomass μ may be related to the surface area of the approximately-spherical canopy. In turn, the canopy diameter may be proportional to the diameter of the tree trunk (or DBH), d. This suggests a model where μ is proportional to the surface area $4\pi(d/2)^2 = \pi d^2$; taking logs, $\log y \propto \log \pi + 2 \log d$. In addition, the tree diameter may be related to the age of the tree. However, since diameter measures some physical quantity and is easier to measure precisely, expect the relationship between foliage biomass and DBH to be stronger than the relationship between foliage biomass and age.

```
> library(GLMsData); data(lime); str(lime)
'data.frame':        385 obs. of  4 variables:
 $ Foliage: num  0.1 0.2 0.4 0.6 0.6 0.8 1 1.4 1.7 3.5 ...
 $ DBH    : num  4 6 8 9.6 11.3 13.7 15.4 17.8 18 22 ...
 $ Age    : int  38 38 46 44 60 56 72 74 68 79 ...
 $ Origin : Factor w/ 3 levels "Coppice","Natural",..: 2 2 2 2 2 2 2
   2 2 2 ...
```

```
>    #
>    # Plot Foliage against DBH
> plot(Foliage ~ DBH, type="n", las=1,
    xlab="DBH (in cm)", ylab="Foliage biomass (in kg)",
    ylim = c(0, 15), xlim=c(0, 40), data=lime)
> points(Foliage ~ DBH, data=subset(lime, Origin=="Coppice"),
    pch=1)
> points(Foliage ~ DBH, data=subset(lime, Origin=="Natural"),
    pch=2)
> points(Foliage ~ DBH, data=subset(lime, Origin=="Planted"),
    pch=3)
> legend("topleft", pch=c(1, 2, 3),
        legend=c("Coppice", "Natural","Planted"))
>    #
>    #  Plot Foliage against DBH, on log scale
> plot( log(Foliage) ~ log(DBH), type="n", las=1,
    xlab="log of DBH (in cm)", ylab="log of Foliage biomass (in kg)",
    ylim = c(-5, 3), xlim=c(0, 4), data=lime)
> points( log(Foliage) ~ log(DBH), data=subset(lime, Origin=="Coppice"),
    pch=1)
> points( log(Foliage) ~ log(DBH), data=subset(lime, Origin=="Natural"),
    pch=2)
> points( log(Foliage) ~ log(DBH), data=subset(lime, Origin=="Planted"),
    pch=3)
>    #
>    # Plot Foliage against Age
> plot(Foliage ~ Age, type="n", las=1,
    xlab="Age (in years)", ylab="Foliage biomass (in kg)",
    ylim = c(0, 15), xlim=c(0, 150), data=lime)
> points(Foliage ~ Age, data=subset(lime, Origin=="Coppice"), pch=1)
> points(Foliage ~ Age, data=subset(lime, Origin=="Natural"), pch=2)
> points(Foliage ~ Age, data=subset(lime, Origin=="Planted"), pch=3)
>    #
>    # Plot Foliage against Origin
> plot( Foliage ~ Origin, data=lime, ylim=c(0, 15),
      las=1, ylab="Foliage biomass (in kg)")
```

Clearly, the response is always positive. From Fig. 11.1, the variance in foliage biomass increases as the mean increases, and a relationship exists between foliage biomass and DBH, and between foliage biomass and age. The effect of origin is harder to see. □

11.3 The Gamma Distribution

The probability function for a gamma distribution is commonly written as

$$\mathcal{P}(y;\alpha,\beta) = \frac{y^{\alpha-1}\exp(-y/\beta)}{\Gamma(\alpha)\beta^{\alpha}},$$

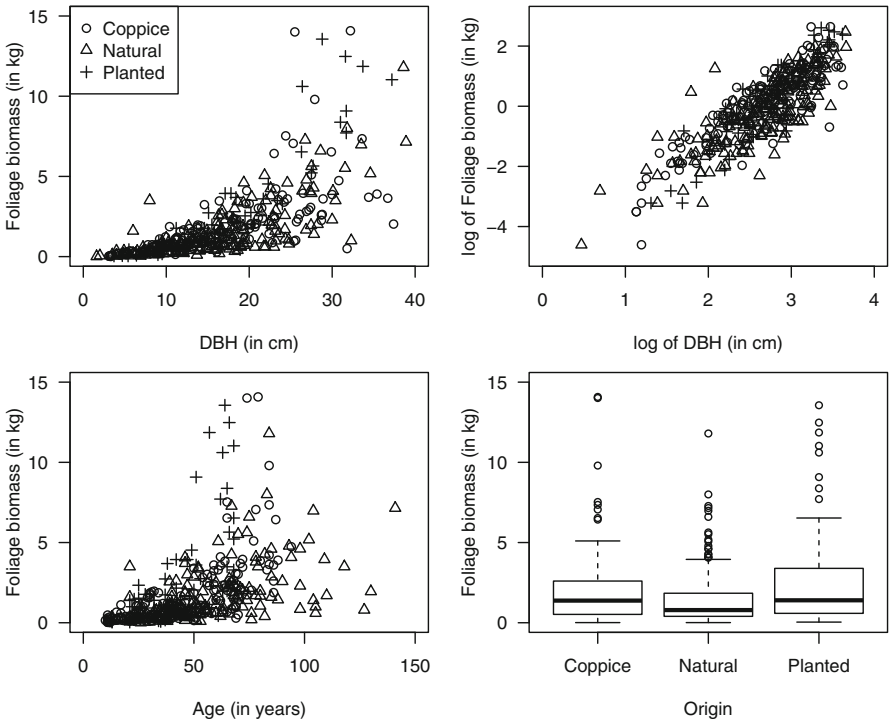

Fig. 11.1 The small-leaved lime data. Foliage biomass against DBH (diameter at breast height; top left panel); log of foliage biomass against the log of DBH (diameter at breast height; top right panel); foliage biomass against age (bottom left panel) foliage biomass against origin (bottom right panel) (Example 11.1)

for $y > 0$, $\alpha > 0$ (the shape parameter) and $\beta > 0$ (the scale parameter), where $\mathrm{E}[y] = \alpha\beta$ and $\mathrm{var}[y] = \alpha\beta^2$. Note that $\Gamma()$ is the gamma function (where, for example, if n is a non-negative integer then $\Gamma(n) = (n-1)!$). Writing in terms of μ and ϕ, the probability function becomes

$$\mathcal{P}(y;\mu,\phi) = \left(\frac{y}{\phi\mu}\right)^{1/\phi} \frac{1}{y} \exp\left(-\frac{y}{\phi\mu}\right) \frac{1}{\Gamma(1/\phi)}$$

for $y > 0$, and $\mu > 0$ and $\phi > 0$, where $\alpha = 1/\phi$ and $\beta = \mu\phi$. Plots of some example gamma probability functions are shown in Fig. 11.2. The variance function for the gamma distribution is $V(\mu) = \mu^2$. The *coefficient of variation* is defined as the ratio of the variance to the mean squared, and is a measure of the relative variation in the data. Therefore, the gamma distribution has a constant coefficient of variation, and consequently gamma GLMs are useful in situations where the coefficient of variation is (approximately) constant. Useful information about the gamma distribution appears in Table 5.1 (p. 221).

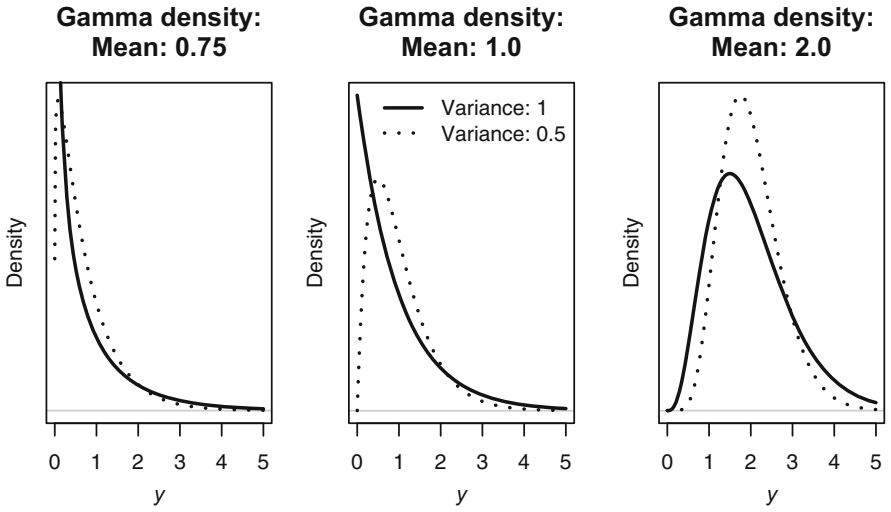

Fig. 11.2 Some example gamma probability functions (Sect. 11.3)

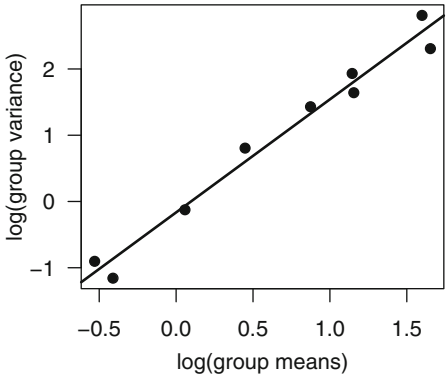

Fig. 11.3 The small-leaved limed data: the logarithm of group variances plotted against the logarithm of the group means (Example 11.2)

Example 11.2. For the small-leaved lime data (Example 11.1; data set: `lime`), the data can be split into smaller groups, and the mean and variance of each group calculated. Then, Fig. 11.3 shows that the variance increases as the mean increases:

```
>    # Define age *groups*
> lime$AgeGrp <- cut(lime$Age, breaks=4 )
>    # Now compute means and variances of each origin/age group:
> vr <- with( lime, tapply(Foliage, list(AgeGrp, Origin), "var" ) )
> mn <- with( lime, tapply(Foliage, list(AgeGrp, Origin), "mean" ) )
>    # Plot
> plot( log(vr) ~ log(mn), las=1, pch=19,
    xlab="log(group means)", ylab="log(group variance)")
```

```
> mf.lm <- lm( c(log(vr)) ~ c(log(mn)) )
> coef( mf.lm )
(Intercept)  c(log(mn))
  -0.165002    1.706453
> abline( coef( mf.lm ), lwd=2)
```

The slope of the line is a little less than 2, so approximately

$$\log(\text{group variance}) \propto 2 \times \log(\text{group mean}).$$

Re-arranging shows the group variance is approximately proportional to square of the group mean. In other words, $V(\mu) \approx \mu^2$ which corresponds to a gamma distribution (Sect. 5.3.6). □

For the gamma distribution, ϕ is almost always unknown and therefore must be estimated (Sect. 11.6.1), so likelihood ratio tests are based on F-tests (Sect. 7.6.4). Two common situations exist where ϕ is known. In situations where y follows a normal distribution, the sample variances can be modelled using a chi-square distribution, which is a gamma distribution with $\phi = 2$. Secondly, the exponential distribution (4.37), which has a history of its own apart from its connection with the gamma distribution, is a gamma distribution with $\phi = 1$ (see Problem 11.17).

The unit deviance for the gamma distribution is

$$d(y, \mu) = 2\left\{-\log\frac{y}{\mu} + \frac{y-\mu}{\mu}\right\}. \tag{11.1}$$

The residual deviance $D(y, \hat{\mu}) = \sum_{i=1}^{n} w_i d(y_i, \hat{\mu}_i) \sim \chi^2_{n-p'}$ approximately, by the saddlepoint approximation, for a model with p' parameters in the linear predictor. The saddlepoint approximation is adequate if $\phi \leq 1/3$ (Sect. 7.5, p. 276).

The canonical link function for the gamma distribution is the inverse (or reciprocal) link function $\eta = 1/\mu$. In practice, the logarithmic link function is often used because it avoids the need for constraints on the linear predictor in view of $\mu > 0$. The log-link often also leads to a useful interpretation where the impact of the explanatory variables is multiplicative (as discussed in the context of Poisson GLMs; see Sect. 10.2). Other link functions are used sometimes to produce desirable features (Sect. 11.5).

The gamma distribution can be used to describe the time between occurrences that follow a Poisson distribution. More formally, suppose an event occurs over a time interval of length T at the Poisson rate of λ events per unit time. Assuming the probability of more than one event in a very small time interval is small, then the number of events in the interval from time 0 to time T can be modelled using a Poisson distribution. Then the length of time y required for r events to occur follows a gamma distribution, with mean r/λ and variance r/λ^2. In this interpretation, r is an integer, which

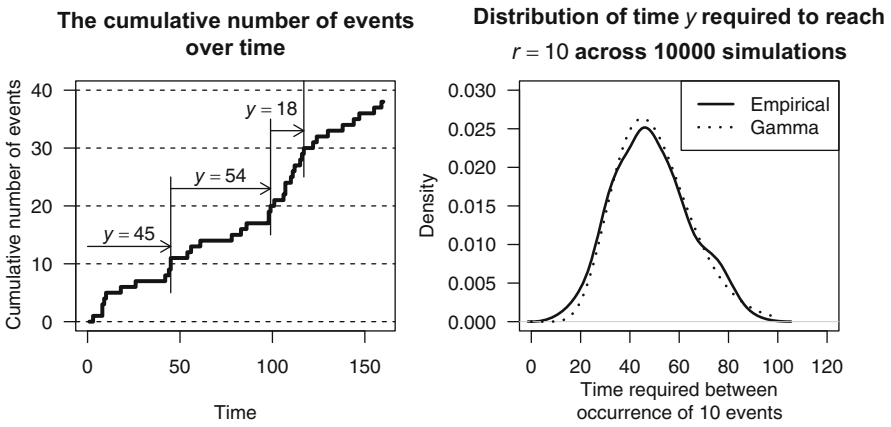

Fig. 11.4 The gamma distribution describes the time between Poisson events. Left panel: the occurrence of the Poisson events showing the time y between the occurrence of $r = 10$ Poisson events for the first three occurrences only. Right panel: the distribution of the time y between events has a gamma distribution (Example 11.3)

is not true in general for the gamma distribution. When r is an integer, the gamma distribution is also called the *Erlang distribution*.

Example 11.3. Suppose events occur over a time interval of $T = 1$ at the rate of $\lambda = 0.2$ per unit time. The length of time y for $r = 10$ events to occurs is shown in Fig. 11.4 (left panel) for the first three sets of $r = 10$ events. The distribution of these times has an approximate gamma distribution with mean $r/\lambda = 10/0.2 = 50$ and variance $r/\lambda^2 = 10/0.2^2 = 250$ (Fig. 11.4, right panel).

□

11.4 The Inverse Gaussian Distribution

The inverse Gaussian distribution may sometimes be suitable for modelling positive continuous data. The inverse Gaussian has the probability function

$$\mathcal{P}(y; \mu, \phi) = (2\pi y^3 \phi)^{-1/2} \exp\left\{-\frac{1}{2\phi}\frac{(y-\mu)^2}{y\mu^2}\right\} \tag{11.2}$$

where $y > 0$, for $\mu > 0$ and the dispersion parameter $\phi > 0$. The variance function is $V(\mu) = \mu^3$. The inverse Gaussian distribution is used when the responses are even more skewed than suggested by the gamma distribution. Plots of some example inverse Gaussian densities are shown in Fig. 11.5.

The canonical link function for the inverse Gaussian distribution is $\eta = \mu^{-2}$, though other link functions are almost always used in practice

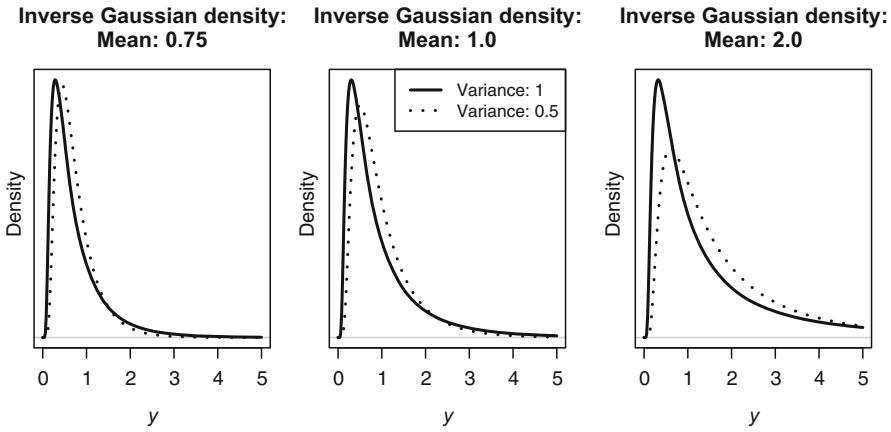

Fig. 11.5 Some example inverse Gaussian probability functions (Sect. 11.4)

(Sect. 11.5), often to ensure $\mu > 0$ and for interpretation purposes. The unit deviance for the inverse Gaussian distribution is

$$d(y, \mu) = \frac{(y-\mu)^2}{y\mu^2},$$

when the residual deviance is $D(y, \hat{\mu}) = \sum_{i=1}^{n} w_i d(y_i, \hat{\mu}_i)$, where the w_i are the prior weights. The unit deviance for the inverse Gaussian distribution is distributed exactly as χ^2_1 (Sect. 5.4.3), since the saddlepoint approximation is exact for the inverse Gaussian distribution (Problem 11.4). This means $D(y, \hat{\mu}) \sim \chi^2_{n-p'}$ exactly (apart from sampling error in estimating μ_i and ϕ) for a model with p' parameters in the linear predictor. Useful information about the inverse Gaussian distribution appears in Table 5.1 (p. 221). For the inverse Gaussian distribution, ϕ is almost always unknown and estimated (Sect. 11.6.2), so likelihood ratio tests are based on F-tests (Sect. 7.6.4).

The inverse Gaussian distribution has an interesting interpretation, connected to *Brownian motion*. Brownian motion is the name given to the random movement of particles over time. For a particle moving with Brownian motion with positive drift (the tendency to move from the current location), the inverse Gaussian distribution describes the distribution of the time taken for the particle to reach some point that is a fixed positive distance δ away. The normal distribution, also known as the Gaussian distribution, describes the distribution of *distance* from the origin at fixed time. The inverse Gaussian distribution gets its name from this relationship to the normal distribution.

To demonstrate these connections between the normal and inverse Gaussian distribution in R, consider a particle moving with Brownian motion with drift 0.5. We can measure both the time taken to exceed a fixed value $\delta = 5$ from the origin, and the distance of the particle from the origin after $T = 20$.

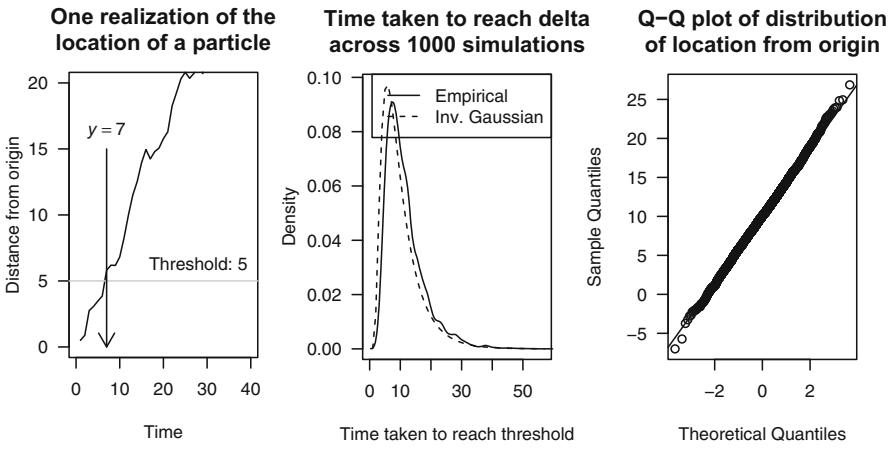

Fig. 11.6 The connection between Brownian motion, the inverse Gaussian distribution and the normal distribution. Left panel: the location of the particle x_t at time t; centre panel: the distribution of the time taken for the particle to exceed $\delta = 5$ follows an inverse Gaussian distribution; right panel: the distance of the particle from the origin after $T = 20$ follows a normal distribution (Sect. 11.4)

The distribution of the time taken closely resembles the expected inverse Gaussian distribution (Fig. 11.6, centre panel), and the distance of the particle from the origin closely follows a normal distribution (Fig. 11.6, right panel).

11.5 Link Functions

The logarithmic link function is the link function most commonly used for gamma and inverse Gaussian GLMs, to ensure $\mu > 0$ and for interpretation purposes (Sect. 10.2). For the gamma and inverse Gaussian distributions, R permits the link functions `"log"`, `"identity"` and `"inverse"` (the default for the gamma distribution). The link function `link="1/mu^2"` is also permitted for the inverse Gaussian distribution, and is the default (canonical) link function.

Example 11.4. For the small-leaved lime data in Example 11.1 (data set: `lime`), no turning points or asymptotes are evident. Consider using a gamma distribution with a variety of link functions, starting with the commonly-used logarithmic link function, and using the ideas developed in Example 11.1 for the model:

```
> lime.log <- glm( Foliage ~ Origin * log(DBH), family=Gamma(link="log"),
    data=lime)
```

We next try the inverse link function:

```
> lime.inv <- update(lime.log, family=Gamma(link="inverse") )

Error: no valid set of coefficients has been found: please supply starting
values
In addition: Warning message:
In log(ifelse(y == 0, 1, y/mu)) : NaNs produced
```

Using the inverse link function produces error messages: R cannot find suitable starting points (which may indicate a poor model). This is because the inverse link function does not restrict μ to be positive. To help R find a starting point for fitting the model, starting points may be supplied to `glm()` on the scale of the data (using the input `mustart`) or on the scale of the linear predictor (using the input `etastart`). For example, we can provide the fitted values from `lime.log` as a starting point:

```
> lime.inv <- update(lime.log, family=Gamma(link="inverse"),
             mustart=fitted(lime.log) )

Error: no valid set of coefficients has been found: please supply starting
values
In addition: Warning message:
In log(ifelse(y == 0, 1, y/mu)) : NaNs produced
```

The model still can not be fitted, so we do not consider this model further.

Finally, we try the identity link function:

```
> lime.id  <- update(lime.log, family=Gamma(link="identity"),
                     mustart = fitted(lime.log) )

Error: no valid set of coefficients has been found: please supply starting
values
In addition: Warning message:
In log(ifelse(y == 0, 1, y/mu)) : NaNs produced
```

Warning messages are displayed when fitting the model with the identity link function: the algorithm did not converge. Again, we could supply starting values to the algorithm to see if this helps:

```
> lime.id  <- update(lime.log, family=Gamma(link="identity"),
            mustart=fitted(lime.log) )
Error: no valid set of coefficients has been found: please supply starting
values
In addition: Warning message:
In log(ifelse(y == 0, 1, y/mu)) : NaNs produced
```

The GLM with the identity link function still does not converge, so we do not consider this model any further. The inverse-link and identity-link models are not very sensible in any case, given Fig. 11.1.

For the log-link model, standard residual plots (using quantile residuals [4]) show that the model seems appropriate (Fig. 11.7):

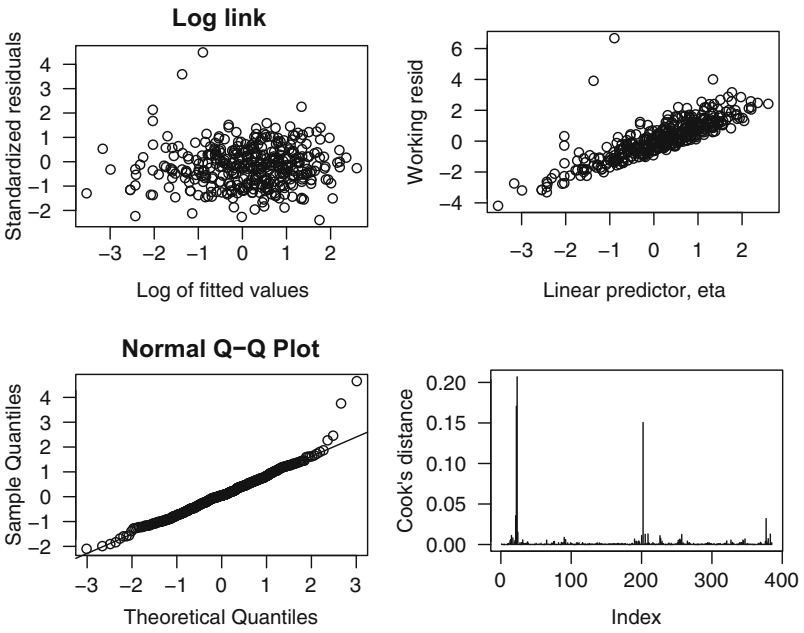

Fig. 11.7 Plots of the standardized residuals against the fitted values for two gamma GLMs fitted to the small-leaved lime data. Left panels: using a logarithmic link function; right panels: using an inverse link function; top panels: standardized residuals plotted against $\log \hat{\mu}$; centre panels: the working residuals e plotted against $\hat{\eta}$; bottom panels: Q–Q plots of the quantile residuals (Example 11.4)

```
>   ## STDIZD RESIDUALS vs FITTED VALUES on constant-info scale
> plot(rstandard(lime.log) ~ log(fitted(lime.log)), main="Log link", las=1,
    xlab="Log of fitted values", ylab="Standardized residuals")
>   ## CHECK LINEAR PREDICTOR
> eta.log <- lime.log$linear.predictor
> plot(resid(lime.log, type="working") + eta.log ~ eta.log, las=1,
      ylab="Working resid", xlab="Linear predictor, eta")
>   ## QQ PLOT OF RESIDUALS
> qqnorm( qr1 <- qresid(lime.log), las=1 ); qqline( qr1 )
>   ## COOK'S DISTANCE
> plot( cooks.distance(lime.log), ylab="Cook's distance", las=1, type="h")
```

Some observations produce large residuals, and some observations appear to give a value of Cook's distance larger than the others though none are deemed influential:

```
> colSums(influence.measures(lime.log)$is.inf)
  dfb.1_ dfb.OrgN dfb.OrgP dfb.1(DB dfb.ON:( dfb.OP:(    dffit    cov.r
       0        0        0        0        0        0        7       29
  cook.d      hat
       0       18
```

□

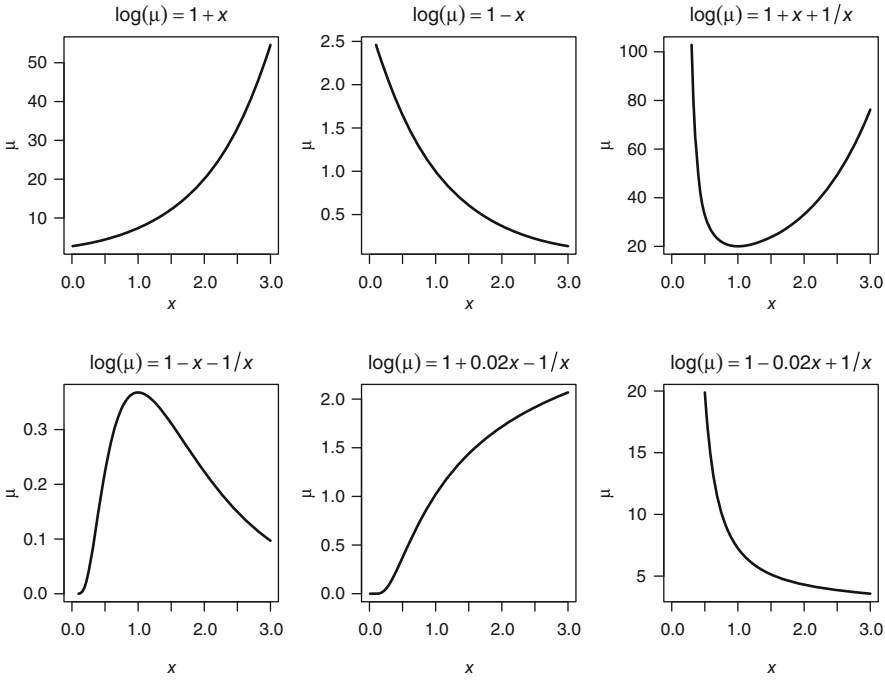

Fig. 11.8 Various logarithmic link function relationships, based on [15, Figure 8.4] (Sect. 11.5)

While the logarithmic link function is commonly used, judicious use of the logarithmic and inverse link functions with transformations of covariates accommodates a wide variety of relationships between the variables, including data displaying asymptotes (Figs. 11.8 and 11.9). Polynomial relationships cannot bound the value of μ, so non-polynomial linear predictors make more physical sense in applications where asymptotes are present. Yield–density experiments (Sect. 11.7.2) are one example where these relationships are used.

11.6 Estimating the Dispersion Parameter

11.6.1 Estimating ϕ for the Gamma Distribution

For the gamma distribution, the maximum likelihood estimate (MLE) of the dispersion parameter ϕ cannot be found in closed form. Defining the *digamma function* as $\psi(x) = \Gamma(x)'/\Gamma(x)$, the MLE of ϕ is the solution to

$$D(y, \hat{\mu}) = -2 \sum_{i=1}^{n} w_i \log \phi - w_i \log w_i + w_i \psi(w_i/\phi) \tag{11.3}$$

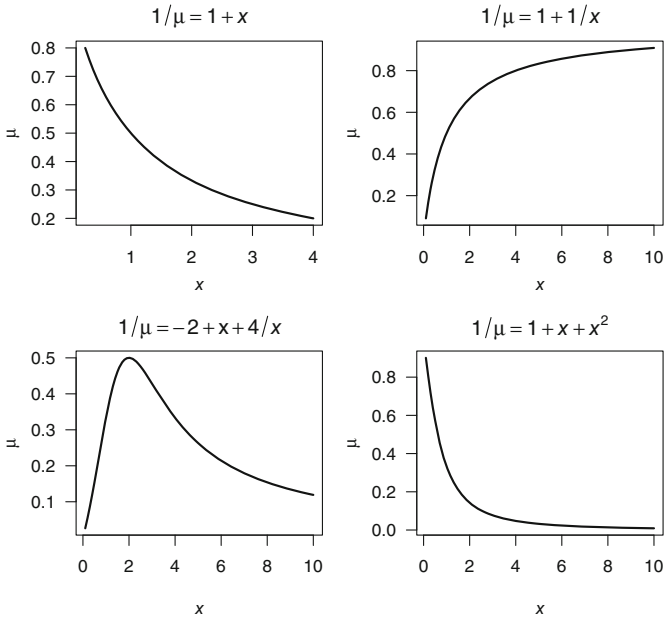

Fig. 11.9 Various inverse link function relationships, based on [15, Figure 8.4]) (Sect. 11.5)

where $D(y, \hat{\mu})$ is the residual deviance and n the sample size (Problem 11.1). Solving (11.3) for ϕ requires iterative numerical methods. This is one reason why the Pearson and deviance estimates are generally used.

Because the deviance is sensitive to very small values of y_i for gamma EDMs (Sect. 6.8.6), the Pearson estimator

$$\bar{\phi} = \frac{1}{n - p'} \sum_{i=1}^{n} \frac{w_i (y_i - \hat{\mu}_i)^2}{\hat{\mu}_i^2}$$

is recommended over the mean deviance estimator

$$\tilde{\phi} = \frac{D(y, \hat{\mu})}{n - p'}$$

for the gamma distribution when the accuracy of small values is in doubt, for example when observations have been rounded to a limited number of digits [15].

Example 11.5. Consider the gamma GLM `lime.log` fitted in Example 11.4 to the small-leaved lime data (data set: `lime`). Two estimates of ϕ are:

```
> phi.md <- deviance(lime.log)/df.residual(lime.log) # Mn dev estimate
> phi.pearson <- summary( lime.log )$dispersion    # Pearson estimate
> c( "Mean Deviance"=phi.md, "Pearson"=phi.pearson)
```

```
Mean Deviance       Pearson
    0.4028747     0.5443774
```

Using numerical methods (Problem 11.1), the MLE is 0.3736. □

Example 11.6. Using the model `lime.log` for the small-leaved lime data in Example 11.1 (data set: `lime`), the analysis of deviance table is:

```
> round(anova(lime.log, test="F"), 3)
                Df Deviance Resid. Df Resid. Dev       F Pr(>F)
NULL                              384     508.48
Origin           2    19.89       382     488.59  18.272 <2e-16 ***
log(DBH)         1   328.01       381     160.58 602.535 <2e-16 ***
Origin:log(DBH)  2     7.89       379     152.69   7.247  0.001 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

By default, R uses the Pearson estimate of ϕ to produce this output. An F-test is requested since ϕ is estimated. Other estimates of ϕ can be used also:

```
> round(anova(lime.log,test="F", dispersion=phi.md), 3)
                Df Deviance Resid. Df Resid. Dev       F    Pr(>F)
NULL                              384     508.48
Origin           2    19.89       382     488.59  24.690 < 2.2e-16 ***
log(DBH)         1   328.01       381     160.58 814.165 < 2.2e-16 ***
Origin:log(DBH)  2     7.89       379     152.69   9.793 < 2.2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The conclusions are very similar for either estimate of ϕ in this example. Retaining all model terms, the parameter estimates are:

```
> printCoefmat(coef(summary(lime.log)), 3)
                     Estimate Std. Error t value Pr(>|t|)
(Intercept)            -4.629      0.276  -16.79   <2e-16 ***
OriginNatural           0.325      0.388    0.84   0.4037
OriginPlanted          -1.528      0.573   -2.67   0.0079 **
log(DBH)                1.843      0.102   18.15   <2e-16 ***
OriginNatural:log(DBH) -0.204      0.143   -1.42   0.1554
OriginPlanted:log(DBH)  0.577      0.209    2.76   0.0061 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Notice that the reference level for `Origin` is `Coppice`, and that there is little evidence of a difference between the natural and coppice trees. From the model proposed in Example 11.1, the coefficient for DBH was expected to be approximately 2; the estimate above is close to this value, and a formal hypothesis tests could be conducted. □

11.6.2 Estimating ϕ for the Inverse Gaussian Distribution

For the inverse Gaussian distribution, the MLE of the dispersion parameter is exactly (Problem 11.3)

$$\hat{\phi} = \frac{D(y, \hat{\mu})}{n}.$$

As usual, the MLE of ϕ is biased. However the mean deviance estimator

$$\tilde{\phi} = \frac{D(y, \hat{\mu})}{n - p'}$$

is essentially the same as the modified profile likelihood estimator, and is very nearly unbiased. The mean deviance estimator has theoretically good properties, and it recommended when good quality data is available. The Pearson estimator is

$$\bar{\phi} = \frac{1}{n - p'} \sum_{i=1}^{n} \frac{w_i (y_i - \hat{\mu}_i)^2}{\hat{\mu}_i^3}.$$

As with the gamma distribution, the deviance is sensitive to rounding errors in very small values of y_i (Sect. 6.8.6), so the Pearson estimator may be better than mean deviance estimator when small values of y are recorded to less than two significant figures. As always, the Pearson estimator is used in R by default.

Example 11.7. For the small-leaved lime data (Example 11.1; data set: `lime`), an inverse Gaussian GLM could also be considered.

```
> lime.iG <- glm( Foliage ~ Origin * log(DBH),
    family=inverse.gaussian(link="log"), data=lime)
```

The estimates of ϕ are:

```
> phi.iG.mle <- deviance(lime.iG)/length(lime$Foliage)   # ML estimate
> phi.iG.md  <- deviance(lime.iG)/df.residual(lime.iG)   # Mean dev
> phi.iG.pearson <- summary( lime.iG )$dispersion        # Pearson
> c( "MLE"=phi.iG.mle, "Mean dev."=phi.iG.md, "Pearson"=phi.iG.pearson)
      MLE Mean dev.   Pearson
 1.056659  1.073387  1.255992
```

The AIC suggests the gamma GLM is preferred over the inverse Gaussian GLM:

```
> c( "Gamma:"=AIC(lime.log), "inv. Gauss.:"=AIC(lime.iG) )
      Gamma: inv. Gauss.:
    750.3267    1089.5297
```

□

11.7 Case Studies

11.7.1 Case Study 1

In a study of sheets of building materials [8, 12], the permeability of three sheets was measured on three different machines over nine days, for a total of 81 sheets, all of equal thickness. Each measurement is an average permeability of eight random pieces cut from each of the 81 sheets (Table 11.2; data set: `perm`). The inverse Gaussian model may be appropriate: particles move at random according to Brownian motion through the building material assuming uniform material, drifting across the sheet (Sect. 11.4). Plots of the data (Fig. 11.10) show that the variance increases with the mean, and shows one large observation that is a potential outlier:

```
> data(perm); perm$Day <- factor(perm$Day)
> boxplot( Perm ~ Day, data=perm, las=1, ylim=c(0, 200),
          xlab="Day", ylab="Permeability (in s)")
> boxplot( Perm ~ Mach, data=perm, las=1, ylim=c(0, 200),
          xlab="Machine", ylab="Permeability (in s)")
```

Because the inverse Gaussian distribution has a sensible interpretation for these data, we adopt the inverse Gaussian model. We also select the logarithmic link function, when the parameters are interpreted as having a multiplicative effect on the response:

```
> perm.log <- glm( Perm ~ Mach * Day, data=perm,
                   family=inverse.gaussian(link="log") )
> round( anova( perm.log, test="F"), 3)
         Df Deviance Resid. Df Resid. Dev      F Pr(>F)
NULL                        80      0.617
Mach      2    0.140        78      0.477 14.133 <2e-16 ***
Day       8    0.069        70      0.408  1.747  0.108
```

Table 11.2 The average permeability (in seconds) of eight sheets of building materials (Sect. 11.7.1)

	Machine				Machine				Machine		
Day	A	B	C	Day	A	B	C	Day	A	B	C
1	25.35	20.23	85.51	4	77.09	47.10	52.60	7	82.79	16.94	21.28
	22.18	42.26	47.21		30.55	23.55	33.73		85.31	32.21	63.39
	41.50	25.70	25.06		24.66	13.00	23.50		134.59	27.29	24.27
2	27.99	17.42	26.67	5	59.16	16.87	20.89	8	69.98	38.28	48.87
	37.07	15.31	58.61		53.46	24.95	30.83		61.66	42.36	177.01
	66.07	32.81	72.28		35.08	33.96	21.68		110.15	19.14	62.37
3	82.04	32.06	24.10	6	46.24	25.35	42.95	9	34.67	43.25	50.47
	29.99	37.58	48.98		34.59	28.31	40.93		26.79	11.67	23.44
	78.34	44.57	22.96		47.86	42.36	22.86		50.58	24.21	69.02

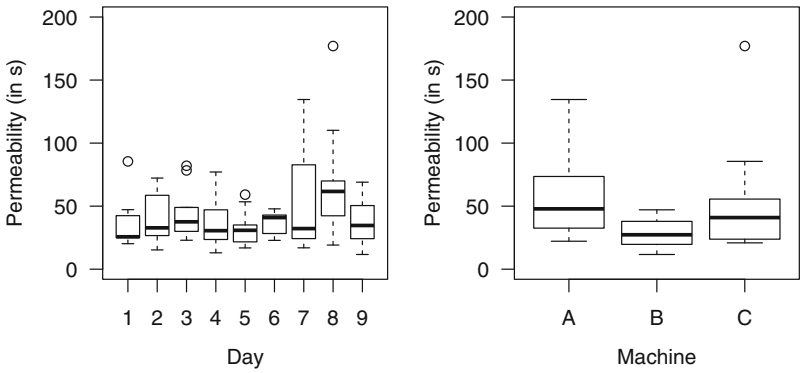

Fig. 11.10 The permeability data. Permeability plotted against the day (left panel), and permeability plotted against the machine (right panel) (Sect. 11.7.1)

```
Mach:Day 16    0.110        54      0.298  1.382  0.186
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Recall the deviance has an exact distribution for the inverse Gaussian distribution, so these results do not rely on small-dispersion or large-sample asymptotics. The interaction term is not necessary in the model. The effect of `Day` is marginal, and so we omit `Day` from the model also.

```
> perm.log <- update( perm.log, Perm ~ Mach)
```

In this case, the model is simply modelling the means of these three machines:

```
> tapply( perm$Perm, perm$Mach, "mean")          # Means from the data
       A        B        C
54.65704 28.84963 45.98037
> tapply( fitted(perm.log), perm$Mach, "mean")   # Fitted means
       A        B        C
54.65704 28.84963 45.98037
```

The final model is:

```
> printCoefmat(coef(summary(perm.log)))
            Estimate Std. Error t value  Pr(>|t|)
(Intercept)  4.00108    0.11694 34.2137 < 2.2e-16 ***
MachB       -0.63898    0.14455 -4.4205 3.144e-05 ***
MachC       -0.17286    0.15868 -1.0894    0.2794
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The model suggests the permeability measurements on Machine B are, on average, $\exp(-0.6390) = 0.5278$ times those for Machine A (the reference

level). Likewise, the permeability measurements on Machine C are, on average, $\exp(-0.1729) = 0.8413$ times those for Machine A. The output suggests Machine C is very similar to Machine A, but Machine B is different.

We can now examine the fitted model to determine if the large observation identified in Fig. 11.10 is an outlier, and if it is influential:

```
> range( rstudent(m1) )
[1] -2.065777  1.316577
> colSums(influence.measures(m1)$is.inf)
dfb.1_  dfb.x  dffit  cov.r cook.d    hat
     0      0      0      2      0      0
```

No residuals appear too large. No observations are influential according to Cook's distance or DFFITS.

11.7.2 Case Study 2

Consider results from an experiment [16] to test the yields of three new onion hybrids (Table 11.3; Fig. 11.11, left panel; data set: `yieldden`). This is an example of a *yield–density* experiment [2, §17.3], [15, §8.3.3].

Yield *per plant*, say z, and planting density, say x, usually exhibit an inverse functional relationship such that

$$\mathrm{E}[z] = \frac{1}{\beta_2 + \beta_0 x + \beta_1 x^2}. \tag{11.4}$$

Yield *per unit area*, $y = xz$, is usually of interest but is harder to measure directly than yield per plant z. However,

$$\mu = \mathrm{E}[y] = x\mathrm{E}[z] = \frac{x}{\beta_2 + \beta_0 x + \beta_1 x^2}. \tag{11.5}$$

Table 11.3 Plant yield density for an experiment with onion hybrids. The yields are the mean yields per plant (in g); the density is in plants per square foot. The yields are means over three plants, averaged on the log-scale (Example 11.7.2)

Variety 1		Variety 2		Variety 3	
Yield	Density	Yield	Density	Yield	Density
105.6	3.07	131.6	2.14	116.8	2.48
89.4	3.31	109.1	2.65	91.6	3.53
71.0	5.97	93.7	3.80	72.7	4.45
60.3	6.99	72.2	5.24	52.8	6.23
47.6	8.67	53.1	7.83	48.8	8.23
37.7	13.39	49.7	8.72	39.1	9.59
30.3	17.86	37.8	10.11	30.3	16.87
24.2	21.57	33.3	16.08	24.2	18.69
20.8	28.77	24.5	21.22	20.0	25.74
18.5	31.08	18.3	25.71	16.3	30.33

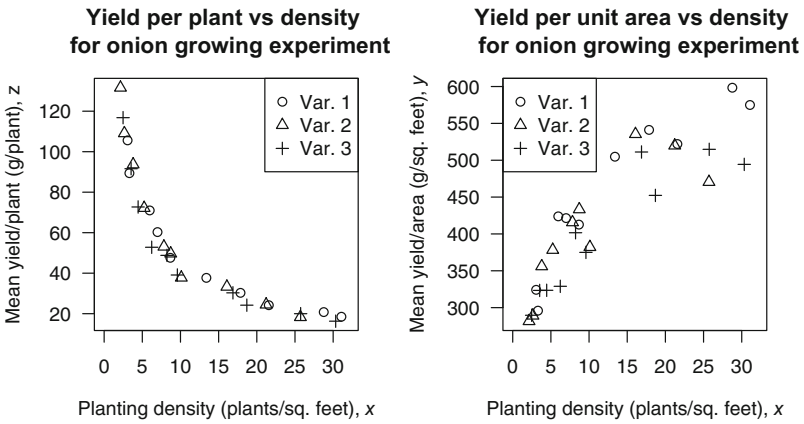

Fig. 11.11 The yield–density onion data. Yield *per plant* z against planting density x (left panel); yield *per unit area* y against planting density x (right panel) (Sect. 11.7.2)

Then inverting,

$$\frac{1}{\mu} = \beta_0 + \beta_1 x + \beta_2 \left(\frac{1}{x}\right) = \eta. \tag{11.6}$$

The bottom left panel of Fig. 11.9 (p. 437) also shows this relationship between the two variables is appropriate: $\mathrm{E}[z] \to 0$ as $x \to \infty$ (that is, as the planting density becomes very large the mean yield per *plant* diminishes) and $\mu \to 0$ as $x \to 0$ (that is, as the planting density becomes almost zero the mean yield per *unit area* diminishes). The plot of the mean yield per unit area (Fig. 11.11, right panel) shows that as density increases, the yield per unit area is more variable also. For this reason, we try using a gamma GLM. Hence, we model yield per unit area y using an inverse link function, with a gamma EDM:

```
> data(yieldden); yieldden$Var <- factor(yieldden$Var)
> yieldden$YD <- with(yieldden, Yield * Dens )
```

We adopt the theory-based model (11.6), adding interactions between the terms involving `Dens` and `Var` to the model (note the use of the `I()` function).

```
> yd.glm.int <- glm( YD ~ (Dens + I(1/Dens)) * Var,
    family=Gamma(link=inverse), data=yieldden )
> round( anova( yd.glm.int, test="F"), 2)
             Df Deviance Resid. Df Resid. Dev      F Pr(>F)
NULL                           29       1.45
Dens          1     1.00        28       0.45 191.67 <2e-16 ***
I(1/Dens)     1     0.27        27       0.18  51.28 <2e-16 ***
Var           2     0.06        25       0.12   5.48   0.01 **
Dens:Var      2     0.01        23       0.12   0.57   0.57
I(1/Dens):Var 2     0.01        21       0.11   0.53   0.59
```

```
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

None of the interaction terms are significant. Refit the model with no interactions:

```
> yd.glm <- update( yd.glm.int, . ~ Dens + I(1/Dens) + Var )
> round( anova(yd.glm, test="F"), 2)
          Df Deviance Resid. Df Resid. Dev      F Pr(>F)
NULL                        29       1.45
Dens       1     1.00        28       0.45 209.56 <2e-16 ***
I(1/Dens)  1     0.27        27       0.18  56.07 <2e-16 ***
Var        2     0.06        25       0.12   5.99   0.01 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The fitted model is:

```
> printCoefmat( coef(summary(yd.glm)), 5)
              Estimate  Std. Error t value  Pr(>|t|)
(Intercept)  1.9687e-03  1.3934e-04 14.1292 2.009e-13 ***
Dens        -1.2609e-05  5.1637e-06 -2.4419  0.022026 *
I(1/Dens)    3.5744e-03  4.9364e-04  7.2409 1.376e-07 ***
Var2         1.0015e-04  7.1727e-05  1.3963  0.174914
Var3         2.4503e-04  7.1187e-05  3.4420  0.002041 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

While an optimal planting density (in terms of yield per unit area) can be determined in principle (see Problem 11.6), Fig. 11.11 shows that the optimal planting density is far beyond the range of the available data in this problem so will probably be unreliable.

The diagnostics show that the model is adequate (Fig. 11.12):

```
> library(statmod)   # For quantile residuals
> scatter.smooth( rstandard(yd.glm) ~ log(fitted(yd.glm)), las=1,
    xlab="Log of fitted values", ylab="Standardized residuals" )
> plot( cooks.distance(yd.glm), type="h", las=1,
    ylab="Cook's distance, D" )
> qqnorm( qr <- qresid(yd.glm), las=1 ); qqline(qr)
> plot( rstandard(yd.glm) ~ yieldden$Var, las=1,
    xlab="Variety", ylab="Standardized residuals" )
```

The yield is modelled by a gamma distribution with the same dispersion parameter for all values of the planting density and all varieties:

```
> summary(yd.glm)$dispersion
[1] 0.004789151
```

Since the estimate of ϕ is small, the saddlepoint approximation will be very accurate (Sect. 7.5), and the distributional assumptions used in inferences are accurate also (Sect. 5.4.4).

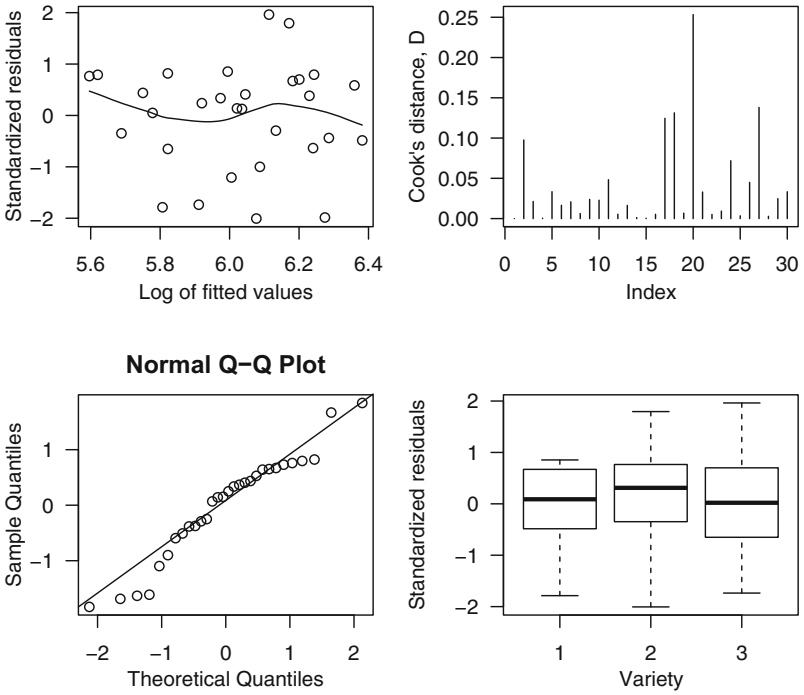

Fig. 11.12 The diagnostic plots from fitting model `yd.glm` to the yield–density onion data (Sect. 11.7.2)

11.8 Using R to Fit Gamma and Inverse Gaussian GLMs

Gamma GLMs are specified in R using `glm(formula, family=Gamma)` in the `glm()` call. (Note the capital `G`, since `gamma()` refers to the gamma function $\Gamma()$.) Inverse Gaussian GLMs are specified in R using `glm(family=inverse.gaussian)` (note all lower case) in the `glm()` call. The link functions `"inverse"`, `"identity"` and `"log"` are permitted for both gamma and inverse Gaussian distributions. The inverse Gaussian distribution also permits the link function `"1/mu^2"` (the canonical link for the inverse Gaussian distribution).

11.9 Summary

Chapter 11 considers fitting GLMs to positive continuous data. Positive continuous data often have the variance increasing with increasing mean (Sect. 11.2), so positive continuous data can be modelled using the gamma

distribution (Sect. 11.3) or, for data more skewed than that suggested by the gamma distribution, using the inverse Gaussian distribution (Sect. 11.4).

For the gamma distribution (Sect. 11.3), $V(\mu) = \mu^2$. The residual deviance $D(y, \hat{\mu})$ is suitably described by a $\chi^2_{n-p'}$ distribution if $\phi \leq 1/3$. For the inverse Gaussian distribution (Sect. 11.4), $V(\mu) = \mu^3$. The residual deviance $D(y, \hat{\mu})$ is described by a $\chi^2_{n-p'}$ distribution.

The gamma distribution models the waiting time between events that occur randomly according to a Poisson distribution (Sect. 11.3). The inverse Gaussian distribution is related to the first-passage time in Brownian motion (Sect. 11.4).

Commonly-used link functions are the logarithmic, inverse and identity link functions (Sect. 11.5). Careful choice of the link function and transformations of the covariates can be used to describe asymptotic relationships between y and x.

The Pearson estimate of ϕ is recommended for both the gamma and inverse Gaussian distributions, though the MLE of ϕ is exact for the inverse Gaussian distribution (Sect. 11.6).

Problems

Selected solutions begin on p. 544.

11.1. Consider estimating ϕ for a gamma GLM.

1. Prove the result (11.3) (p. 436).
2. When $w_i = 1$ for all observations i, show that the MLE of ϕ is the solution to $D(y, \hat{\mu}) = -2n\{\log \phi + \psi(1/\phi)\}$, where $\psi(x) = \Gamma(x)'/\Gamma(x)$ is the digamma function.
3. Write an R function for computing the MLE of ϕ for a gamma GLM with $w_i = 1$ for all i. (HINT: The digamma function $\psi(z)$ and the trigamma function $\psi_1(z) = \mathrm{d}\psi(z)/\mathrm{d}z$ are available in R as `digamma()` and `trigamma()` respectively.)
4. Using this R function, find the MLE of ϕ as given in Example 11.5 (p. 437).

11.2. If a fitted gamma GLM includes a constant term and the logarithmic link function is used, the sum over the observations of the second term in the expression (11.1) for the residual deviance is zero. In other words, $\sum_{i=1}^{n}(y_i - \hat{\mu}_i)/\hat{\mu}_i = 0$. Prove this result by writing the log-likelihood for a model with linear predictor containing the constant term β_0, differentiating the log-likelihood with respect to β_0, setting to zero and solving.

11.3. Show that the MLE of the dispersion parameter ϕ for an inverse Gaussian distribution is $\hat{\phi} = D(y, \hat{\mu})/n$.

11.4. In this problem we explore the distribution of the unit deviance for the inverse Gaussian and gamma distributions.

1. Use R to generate 2000 random numbers `y1` from an inverse Gaussian distribution (using `rinvgauss()` from the **statmod** package [7, 26]) with `dispersion=0.1` (that is, $\phi = 0.1$). Fit an inverse Gaussian GLM with systematic component `y1~1` and then compute the fitted unit deviances $d(y, \hat{\mu})$. By using `qqplot()`, show that these fitted unit deviances follow a χ^2_1 distribution.
2. Use R to generate 2000 random numbers `y2` from a gamma distribution (using `rgamma()`) with `shape=2` and `scale=1`. (This is equivalent to $\mu = 2$ and $\phi = 1/2$.) Fit a gamma GLM with systematic component `y2~1` and then compute the fitted unit deviances $d(y, \hat{\mu})$. By using `qqplot()`, show that these fitted unit deviances do not follow a χ^2_1 distribution.

11.5. Consider the inverse Gaussian distribution (Table 5.1, p. 221).

1. Show that the inverse Gaussian distribution with mean $\mu \to \infty$ (called the Lévy distribution) has the probability function

$$\mathcal{P}(y; \phi) = \frac{1}{\sqrt{2\pi\phi y^3}} \exp\{-1/(2y\phi)\} \quad \text{for } y > 0.$$

2. Show that the variance of the Lévy distribution is infinite.
3. Plot the Lévy probability function for $\phi = 0.5$ and $\phi = 2$.

11.6. Show that the maximum value for μ for a gamma GLM with a systematic component of the form $1/\mu = \beta_0 + \beta_1 x + \beta_2/x$ occurs at $x = \sqrt{\beta_2/\beta_1}$. Then, show that this maximum value is $\mu = 1/\left(\beta_0 + 2\sqrt{\beta_1\beta_2}\right)$.

11.7. A study of insurance claims [19] modelled the amount of insurance claims (for a total of 1975 claims) using a GLM(gamma; log) model, with five potential qualitative explanatory variables: policy-holder age P (five age groups); vehicle type T (five types); vehicle age V (four age groups); district D (five districts); and no-claims discount C (four levels). All main effects are significant, and the interactions are tested using the deviance (Table 11.4).

1. Determine the changes in degrees of freedom after fitting each interaction term.
2. Find an estimate of the dispersion parameter ϕ for the model with all two-factor interactions.
3. Determine which interaction terms are significant using likelihood ratio tests.
4. Interpret the meaning of the interaction term $T.P$.

11.8. The UK700 randomized trial [1] compared the 2-year costs (in dollars) of treating mentally-ill patients in the community using two different management approaches: intensive (caseload of 10–15 patients) and standard (caseload of 30–35 patients). Data for 667 patients are available. Numerous models were fitted, including those summarized in Table 11.5. For all these models, $g(\mu) = \beta_0 + \beta_1 x_1 + \beta_2 x_2$, where $x_1 = 1$ for the intensive group and is zero otherwise, and x_2 is the patient age in completed years.

Table 11.4 The analysis of deviance table from fitting a gamma GLM to claim severity data; read down the columns (Problem 11.7)

Terms	Residual deviance	Terms	Residual deviance
Main effects model	5050.9		
$+T.P$	4695.2	$+P.D$	4497.1
$+T.V$	4675.9	$+P.C$	4462.0
$+T.D$	4640.1	$+V.D$	4443.4
$+T.C$	4598.8	$+V.C$	4420.8
$+P.V$	4567.3	$+D.C$	4390.9

Table 11.5 Summaries of the GLMs fitted to the mental care cost data [1, Table 3], using identity and logarithmic link functions (Problem 11.8)

EDM	$g(\mu)$	$\hat{\beta}_1$	95% CI	$\hat{\beta}_2$	95% CI	AIC
Normal	Identity	2032	$(-1371, 5435)$	-3324	$(-4812, -1836)$	15,259
Gamma	Identity	1533	$(-1746, 4813)$	-2622	$(-3975, -1270)$	14,765
Inverse Gaussian	Identity	1361	$(-1877, 4601)$	-2416	$(-3740, -1091)$	15,924
Normal	Log	1.10	$(0.95, 1.27)$	0.84	$(0.79, 0.90)$	15,256
Gamma	Log	1.07	$(0.93, 1.24)$	0.88	$(0.82, 0.93)$	14,763
Inverse Gaussian	Log	1.07	$(0.93, 1.23)$	0.89	$(0.84, 0.95)$	15,924

1. Based on the AIC, which EDM seems most appropriate?
2. The constants in the models β_0 are not revealed. Nonetheless, write down the two models based on this EDM as comprehensively as possible.
3. Interpret the regression parameters for x_1 in both models.
4. Interpret the regression parameters for x_2 in both models.
5. Is the type of treatment significant for modelling cost? Explain.
6. Is the patient age significant for modelling cost? Explain.
7. Which interpretation (i.e. the use of which link function) seems most appropriate? Why?

11.9. For the small-leaved lime data in data set `lime`, the gamma GLM `lime.log` was fitted in Example 11.6 (p. 438). Consider fitting a similar gamma GLM with a log link, but using `DBH` as the explanatory variable in place of `log(DBH)`.

1. Produce the diagnostic plots for this model.
2. Interpret the fitted model.
3. Do the diagnostic plots suggest which model (using `DBH` or `log(DBH)`) is preferred?

11.10. For the small-leaved lime data in data set `lime`, the model in Example 11.1 proposed a relationship between `Foliage` and `log(DBH)`. Determine if a model that also includes `Age` improves the model.

Table 11.6 The average daily fat yields (in kg/day) each week for 35 weeks for a dairy cow (Problem 11.12)

Week	Yield	Week	Yield	Week	Yield	Week	Yield	Week	Yield
1	0.31	8	0.66	15	0.57	22	0.30	29	0.15
2	0.39	9	0.67	16	0.48	23	0.26	30	0.18
3	0.50	10	0.70	17	0.46	24	0.34	31	0.11
4	0.58	11	0.72	18	0.45	25	0.29	32	0.07
5	0.59	12	0.68	19	0.31	26	0.31	33	0.06
6	0.64	13	0.65	20	0.33	27	0.29	34	0.01
7	0.68	14	0.64	21	0.36	28	0.20	35	0.01

11.11. For the small-leaved lime data in data set `lime`, the model in Example 11.1 proposed that the coefficient for `log(DBH)` was expected to be approximately 2. For this problem, consider fitting a gamma GLM with only `log(DBH)` as an explanatory variable (that is, without `Origin`) to test this idea.

1. Test this hypothesis using a Wald test, and comment.
2. Test this hypothesis using a likelihood ratio test, and comment.

11.12. In the dairy science literature, Wood's lactation curve is the equation, justified biometrically, relating the production of milk fat y in week t:

$$y = at^b \exp(ct),$$

where the parameters a, b and c are estimated from the data. Lactation data [10] from one dairy cow are shown in Table 11.6 (data set: `lactation`).

1. Plot the data, and propose possible models based on the graphs shown in Sect. 11.5.
2. Fit models suggested above, plus the model suggested by Wood's lactation curve.
3. Plot the curves on the data, and comment.

11.13. A study of computer tomography (CT) interventions [23, 32] in the abdomen measured the total procedure time (in s) and the total radiation dose received (in rads) (Table 3.21; data set: `fluoro`). During these procedures, "one might postulate that the radiation dose received is related to... the total procedure time" [32, p. 61].

1. Find a suitable GLM for the data, ensuring a diagnostic analysis, and test the hypothesis implied by the above quotation.
2. Plot the fitted model, including the 95% confidence interval about the fitted line.

11.14. Nambe Mills, Santa Fe, New Mexico [3, 25], is a tableware manufacturer. After casting, items produced by Nambe Mills are shaped, ground,

buffed, and polished. In 1989, as an aid to rationalizing production of its 100 products, the company recorded the total grinding and polishing times and the diameter of each item (Table 5.3; data set: `nambeware`). In Chaps. 5–8 (Problems 5.26, 6.11, 7.5 and 8.12), only the item diameter was considered as an explanatory variable. Now, consider modelling price y as a function of all explanatory variables.

1. Plot the `Price` against `Type`, against `Diam` and against `Time`. What do the plots suggest about the relationship between the mean and the variance for the data?
2. What possible distribution could be used to fit a GLM? Justify your answer.
3. Determine a good model for `Price`, considering interactions. Perform a comprehensive diagnostic test of your model and comment on the structure of the fitted model.
4. Write down your final model(s).
5. Interpret your final model(s).

11.15. The lung capacity data [13] in Example 1.1 have been used in Chaps. 2 and 3 (data set: `lungcap`).

1. Plot the data, and identify possible relationships between `FEV` and the other variables.
2. Find a suitable GLM for the data, ensuring a diagnostic analysis.
3. Is there evidence that smoking affects lung capacity?
4. Interpret your model.

11.16. In a study of foetal size [20], the mandible length (in mm) and gestational age (in weeks) for 167 foetuses were measured from the 12th week of gestation onwards (Table 11.7; data set: `mandible`). According to the source [20, p. 437], the data for foetuses aged over 28 weeks should be discarded, because "the technique was difficult to perform and excessive measurement error was suspected".

1. Using the `subset()` command in R, create a data frame of the measurements for the 158 foetuses less than or equal to 28 weeks.

Table 11.7 The mandible length and foetal age (Problem 11.16)

Age (in weeks)	Length (in mm)
12.3	8
12.4	9
12.7	11
12.7	11
12.9	10
⋮	⋮

2. Plot this data subset, and identify the important features of the data.
3. Fit a suitable model for the data subset. Consider exploring different link functions, and including polynomial terms in age.
4. Plot the full data set (including foetuses older than 28 weeks of age), and then draw the systematic component on the same plot. Does the model fit well to these extra observations?
5. Find and interpret the 90% Wald confidence interval for the age parameter.

11.17. The times to death (in weeks) of two groups of leukaemia patients whose white blood cell counts were measured (Table 4.3; data set: `leukwbc`) were grouped according to a morphological variable called the AG factor [5].

1. Plot the survival time against white blood cell count (WBC), distinguishing between AG-positive and AG-negative patients. Comment on the relationship between WBC and survival time, and the AG factor.
2. Plot the survival time against $\log_{10}$ WBC, and argue that using $\log_{10}$ WBC is likely to be a better choice as an explanatory variable.
3. Fit a GLM(gamma; log) model to the data, including the interaction term between the AG factor and $\log_{10}$ WBC, and show that the interaction term is not necessary.
4. Refit the GLM without the interaction term, and evaluate the model using diagnostic tools.
5. Plot the fitted lines for each AG-factor on a plot of the observations.
6. The original source [5] uses an exponential distribution (4.37), which is a gamma distribution with $\phi = 1$. Does this seem reasonable?

11.18. The data in Table 11.8 come from a study [14] of the nitrogen content of soil, with three replicates at each dose (data set: `nitrogen`).

1. Plot the data, identifying the organic nitrogen source.

Table 11.8 The soil nitrogen (in kilograms of nitrogen per hectare) after applying different doses of fertilizer (in kilograms of nitrogen per hectare). The fertilizers are inorganic apart from the dose of 248 kg N ha^{-1}, whose source is organic (farmyard manure) (Problem 11.18)

Fertilizer dose	Soil N content		
Control	4.53	5.46	4.77
48	6.17	9.30	8.29
96	11.30	16.58	16.24
144	24.61	18.20	30.03
192	21.94	29.24	27.43
240	46.74	38.87	44.07
288	57.74	45.59	39.77
248	25.28	21.79	19.75

2. Find the mean and variance of each fertilizer dose. Then, plot the logarithm of the variance against the logarithm of the means, and show that a gamma distribution appears sensible.
3. Fit a suitable gamma GLM to the data, including a diagnostic analysis.

11.19. In Problem 2.18 (p. 88), data are given from an experiment where children were asked to build towers out of cubical and cylindrical blocks as high as they could [11, 24]. The number of blocks used and the time taken were recorded (Table 2.12; data set: `blocks`). In this problem, we examine the time taken to stack blocks.

1. Find a suitable gamma GLM for modelling the time taken to build the towers.
2. Find a suitable inverse Gaussian GLM for modelling the time taken to build the towers.
3. Using a diagnostic analysis, determine which of the two models is more appropriate.
4. Test the hypothesis that the time taken to stack the blocks differs between cubical and cylindrical shaped blocks.
5. Test the hypothesis that older children take less time to stack the blocks, for both cubes and cylinders.

11.20. Hardness of timber is difficult to measure directly, but is related to the density of the timber (which is easier to measure). To study this relationship [29], density and Janka hardness measurements for 36 Australian eucalyptus hardwoods were obtained (Table 11.9; data set: `hardness`). Venables [27] suggests that a GLM using a square-root link function with a gamma distribution fits the data well. Fit the suggested model, and use a diagnostic analysis to show that this model seems reasonable.

Table 11.9 The Janka hardness and density of Australian hardwoods, units unknown (Problem 11.20)

Density	Hardness	Density	Hardness	Density	Hardness
24.7	484	39.4	1210	53.4	1880
24.8	427	39.9	989	56.0	1980
27.3	413	40.3	1160	56.5	1820
28.4	517	40.6	1010	57.3	2020
28.4	549	40.7	1100	57.6	1980
29.0	648	40.7	1130	59.2	2310
30.3	587	42.9	1270	59.8	1940
32.7	704	45.8	1180	66.0	3260
35.6	979	46.9	1400	67.4	2700
38.5	914	48.2	1760	68.8	2890
38.8	1070	51.5	1710	69.1	2740
39.3	1020	51.5	2010	69.1	3140

11.21. In Problem 3.19, a study of urethral length L and mass M of various mammals [30] was discussed. For these data (data set: `urinationL`), one postulated relationship is $L = kM^{1/3}$ for some proportionality constant k. In that Problem, a weighted regression model was fitted to the data using a transformation of the relationship to linearity: $\log L = \log k + (\log M)/3$. Fit an approximately-equivalent GLM for modelling these data. Using this model, test the hypothesis again using both a Wald and likelihood-ratio test.

11.22. In Problem 3.11 (p. 150), data are given from a study of the food consumption of fish [17] (data set: `fishfood`). In Problem 3.11, the linear regression model fitted in the source is shown. Fit the equivalent gamma GLM for modelling the daily food consumption, and compare to the linear regression model in Problem 3.11.

11.23. In Problem 3.17, the daily energy requirements [9, 28, 31] and weight of 64 wethers (Table 2.11; data set: `sheep`) were analysed using a linear regression model, using the logarithm of the daily energy requirements as the response.

1. Fit the equivalent GLM.
2. Perform a diagnostic analysis of the GLM and compare to the regression model using the logarithm of the daily energy requirements as the response. Comment.
3. Plot the data and the fitted GLM, and add the 95% confidence intervals for the fitted values.
4. Interpret the GLM.

11.24. An experiment to investigate the initial rate of benzene oxidation [18] over a vanadium oxide catalyst used three different reaction temperatures and varied oxygen and benzene concentrations. A subset of the data is presented in Table 11.10 (data set: `rrates`) for a benzene concentration near 2×10^{-3} gmoles/L.

1. Plot the reaction rate against oxygen concentration, distinguishing different temperatures. What important features of the data are obvious?
2. Compare the previous plot to Fig. 11.8 (p. 436) and Fig. 11.9 (p. 437). Suggest two functional relationships between oxygen concentration and reaction rate that could be compared.
3. Fit the models identified above, and separately plot the fitted systematic components on the data. Select a model, explaining your choice.
4. For your chosen model, perform a diagnostic analysis, identifying potential problems with the model.
5. By looking at the data for each temperature separately, is it reasonable to assume the dispersion parameter ϕ is approximately constant? Explain.

Table 11.10 The initial reaction rate of benzene oxidation. Oxygen concentration [O] is $\times 10^4$ gmole/L; the temperature is in Kelvin; and the reaction rate is $\times 10^{19}$ gmole/g of catalyst/s (Problem 11.24)

Temp: 623 K		Temp: 648 K		Temp: 673 K	
[O]	Rate	[O]	Rate	[O]	Rate
134.5	218	23.3	229	16.0	429
108.0	189	40.8	296	23.5	475
68.6	192	140.3	547	132.8	1129
49.5	174	140.8	582	107.7	957
41.7	152	141.2	480	68.5	745
29.4	139	140.0	493	47.2	649
22.5	118	121.2	513	42.5	742
17.2	120	104.7	411	30.1	662
17.0	122	40.8	349	11.2	373
22.8	132	22.5	226	17.1	440
41.3	167	55.2	338	65.8	662
59.6	208	55.4	351	108.2	724
119.7	216	29.5	295	123.5	915
158.2	294	30.0	294	160.0	944
		16.3	233	66.4	713
		16.5	222	66.5	736
		20.8	239		
		20.6	217		

References

[1] Barber, J., Thompson, S.: Multiple regression of cost data: Use of generalized linear models. Journal of Health Services Research and Policy **9**(4), 197–204 (2004)

[2] Crawley, M.J.: GLIM for Ecologists. Blackwell Scientific Publications, London (1993)

[3] Data Desk: Data and story library (DASL) (2017). URL http://dasl.datadesk.com

[4] Dunn, P.K., Smyth, G.K.: Randomized quantile residuals. Journal of Computational and Graphical Statistics **5**(3), 236–244 (1996)

[5] Feigl, P., Zelen, M.: Estimation of exponential survival probabilities with concomitant information. Biometrics **21**, 826–838 (1965)

[6] Fox, R., Garbuny, M., Hooke, R.: The Science of Science. Walker and Company, New York (1963)

[7] Giner, G., Smyth, G.K.: statmod: probability calculations for the inverse Gaussian distribution. The R Journal **8**(1), 339–351 (2016)

[8] Hald, A.: Statistical Theory with Engineering Applications. John Wiley and Sons, New York (1952)

[9] Hand, D.J., Daly, F., Lunn, A.D., McConway, K.Y., Ostrowski, E.: A Handbook of Small Data Sets. Chapman and Hall, London (1996)

[10] Henderson, H.V., McCulloch, C.E.: Transform or link? Tech. Rep. BU-049-MA, Cornell University (1990)
[11] Johnson, B., Courtney, D.M.: Tower building. Child Development **2**(2), 161–162 (1931)
[12] Jørgensen, B.: Exponential dispersion models and extensions: A review. International Statistical Review **60**(1), 5–20 (1992)
[13] Kahn, M.: An exhalent problem for teaching statistics. Journal of Statistical Education **13**(2) (2005)
[14] Lane, P.W.: Generalized linear models in soil science. European Journal of Soil Science **53**, 241–251 (2002)
[15] McCullagh, P., Nelder, J.A.: Generalized Linear Models, second edn. Monographs on Statistics and Applied Probability. Chapman and Hall, London (1989)
[16] Mead, R.: Plant density and crop yield. Applied Statistics **19**(1), 64–81 (1970)
[17] Palomares, M.L., Pauly, D.: A multiple regression model for predicting the food consumption of marine fish populations. Australian Journal of Marine and Freshwater Research **40**(3), 259–284 (1989)
[18] Pritchard, D.J., Downie, J., Bacon, D.W.: Further consideration of heteroscedasticity in fitting kinetic models. Technometrics **19**(3), 227–236 (1977)
[19] Renshaw, A.E.: Modelling the claims process in the presence of covariates. ASTIN Bulletin **24**(2), 265–285 (1994)
[20] Royston, P., Altman, D.G.: Regression using fractional polynomials of continuous covariates: Parsimonious parametric modelling. Journal of the Royal Statistical Society, Series C **43**(3), 429–467 (1994)
[21] Schepaschenko, D., Shvidenko, A., Usoltsev, V.A., Lakyda, P., Luo, Y., Vasylyshyn, R., Lakyda, I., Myklush, Y., See, L., McCallum, I., Fritz, S., Kraxner, F., Obersteiner, M.: Biomass plot data base. PANGAEA (2017). DOI 10.1594/PANGAEA.871465. In supplement to: Schepaschenko, D et al. (2017): A dataset of forest biomass structure for Eurasia. Scientific Data, 4, 170070, doi:10.1038/sdata.2017.70
[22] Schepaschenko, D., Shvidenko, A., Usoltsev, V.A., Lakyda, P., Luo, Y., Vasylyshyn, R., Lakyda, I., Myklush, Y., See, L., McCallum, I., Fritz, S., Kraxner, F., Obersteiner, M.: A dataset of forest biomass structure for Eurasia. Scientific Data **4**, 1–11 (2017)
[23] Silverman, S.G., Tuncali, K., Adams, D.F., Nawfel, R.D., Zou, K.H., Judy, P.F.: CT fluoroscopy-guided abdominal interventions: Techniques, results, and radiation exposure. Radiology **212**, 673–681 (1999)
[24] Singer, J.D., Willett, J.B.: Improving the teaching of applied statistics: Putting the data back into data analysis. The American Statistician **44**(3), 223–230 (1990)
[25] Smyth, G.K.: Australasian data and story library (OzDASL) (2011). URL http://www.statsci.org/data

[26] Smyth, G.K.: statmod: Statistical Modeling (2017). URL https://CRAN.R-project.org/package=statmod. R package version 1.4.30. With contributions from Yifang Hu, Peter Dunn, Belinda Phipson and Yunshun Chen.

[27] Venables, W.N.: Exegeses on linear models. In: S-Plus User's Conference. Washington DC (1998). URL https://www.stats.ox.ac.uk/pub/MASS3/Exegeses.pdf

[28] Wallach, D., Goffinet, B.: Mean square error of prediction in models for studying ecological systems and agronomic systems. Biometrics **43**(3), 561–573 (1987)

[29] Williams, E.J.: Regression Analysis. Wiley, New York (1959)

[30] Yang, P.J., Pham, J., Choo, J., Hu, D.L.: Duration of urination does not change with body size. Proceedings of the National Academy of Sciences **111**(33), 11 932–11 937 (2014)

[31] Young, B.A., Corbett, J.L.: Maintenance energy requirement of grazing sheep in relation to herbage availability. Australian Journal of Agricultural Research **23**(1), 57–76 (1972)

[32] Zou, K.H., Tuncali, K., Silverman, S.G.: Correlation and simple linear regression. Radiology **227**, 617–628 (2003)

Chapter 12
Tweedie GLMs

. . . we cannot know if any statistical technique that we develop is useful unless we use it.
Box [5, p. 792]

12.1 Introduction and Overview

This chapter introduces GLMs based on Tweedie EDMs. Tweedie EDMs are distributions that generalize many of the EDMs already seen (the normal, Poisson, gamma and inverse Gaussian distributions are special cases) and include other distributions also. First, Tweedie EDMs are discussed in general (Sect. 12.2), and then two subsets of the Tweedie GLMs which are important are studied: Tweedie EDMs for modelling positive continuous data for which gamma and inverse Gaussian GLMs are special cases (Sect. 12.2.3), then Tweedie EDMs for modelling continuous data with exact zeros (Sect. 12.2.4). We then follow with a description of how to use these Tweedie EDMs to fit Tweedie GLMs (Sect. 12.3).

12.2 The Tweedie EDMs

12.2.1 Introducing Tweedie Distributions

Apart from the binomial and negative binomial distributions, the EDMs seen so far in this book have variance functions with similar forms:

- the normal distribution, where $V(\mu) = \mu^0 = 1$ (Chaps. 2 and 3);
- the Poisson distribution, where $V(\mu) = \mu^1$ (Chap. 10);
- the gamma distribution, where $V(\mu) = \mu^2$ (Chap. 11);
- the inverse Gaussian distribution, where $V(\mu) = \mu^3$ (Chap. 11).

These EDMs have power variance functions of the form $V(\mu) = \mu^\xi$, with $\xi = 0, 1, 2, 3$. More generally, any EDM with a variance function $V(\mu) = \mu^\xi$ is called a *Tweedie distribution*, or a *Tweedie* EDM, where ξ can take any real

P. K. Dunn, G. K. Smyth, *Generalized Linear Models with Examples in R*,
Springer Texts in Statistics, https://doi.org/10.1007/978-1-4419-0118-7_12

Table 12.1 Features of the Tweedie distributions for various values of the index parameter ξ, showing the support S (the permissible values of y) and the domain Ω for μ. The Poisson distribution ($\xi = 1$ and $\phi = 1$) is a special case of the discrete distributions, and the inverse Gaussian distribution ($\xi = 3$) is a special case of positive stable distributions. $\mathbb{R}$ refers to the real line; superscript + means positive real values only; subscript 0 means zero is included in the space (Sect. 12.2.1)

Tweedie EDM	ξ	S	Ω	Reference
Extreme stable	$\xi < 0$	$\mathbb{R}$	$\mathbb{R}^+$	Not covered
Normal	$\xi = 0$	$\mathbb{R}$	$\mathbb{R}$	Chaps. 2 and 3
No EDMs exist	$0 < \xi < 1$			
Discrete	$\xi = 1$	$y = 0, \phi, 2\phi, \ldots$	$\mathbb{R}^+$	Chap. 10 for $\phi = 1$
Poisson-gamma	$1 < \xi < 2$	$\mathbb{R}_0^+$	$\mathbb{R}^+$	Sect. 12.2.3
Gamma	$\xi = 2$	$\mathbb{R}^+$	$\mathbb{R}^+$	Chap. 11
Positive stable	$\xi > 2$	$\mathbb{R}^+$	$\mathbb{R}^+$	Sect. 12.2.4

value except $0 < \xi < 1$ [25]. ξ is called the *Tweedie index parameter* and is sometimes denoted by p. This power-variance relationship has been observed in natural populations for many years [36, 37]. Useful information about the Tweedie distribution appears in Table 5.1 (p. 221).

The four specific cases of Tweedie distributions listed above show that the Tweedie distributions are useful for a variety of data types (Table 12.1). More generally:

- For $\xi \leq 0$, the Tweedie distributions are suitable for modelling continuous data where $-\infty < y < \infty$. The normal distribution ($\xi = 0$) is a special case. When $\xi < 0$, the Tweedie distributions have the unusual feature that data y are defined on the entire real line, but $\mu > 0$. These Tweedie distributions with $\xi < 0$ have no known realistic applications, and so are not considered further.
- For $\xi = 1$ the Tweedie distributions are suitable for modelling discrete data where $y = 0, \phi, 2\phi, 3\phi, \ldots$. When $\phi = 2$, for example, a positive probability exists for $y = 0, 2, 4, \ldots$. The Poisson distribution is a special case when $\phi = 1$.
- For $1 < \xi < 2$, the Tweedie distributions are suitable for modelling positive continuous data with exact zeros. An example is rainfall modelling [12, 31]: when no rain falls, an exact zero is recorded, but when rain *does* fall, the amount is a continuous measurement. Plots of example probability functions are shown in Fig. 12.1. As $\xi \to 1$, the densities show local maxima corresponding to the discrete masses for the corresponding Poisson distribution.
- For $\xi \geq 2$, the Tweedie distributions are suitable for modelling positive continuous data. The gamma ($\xi = 2$) and inverse Gaussian ($\xi = 3$) distributions are special cases (Chap. 11). The distributions become more right skewed as ξ increases (Fig. 12.2).

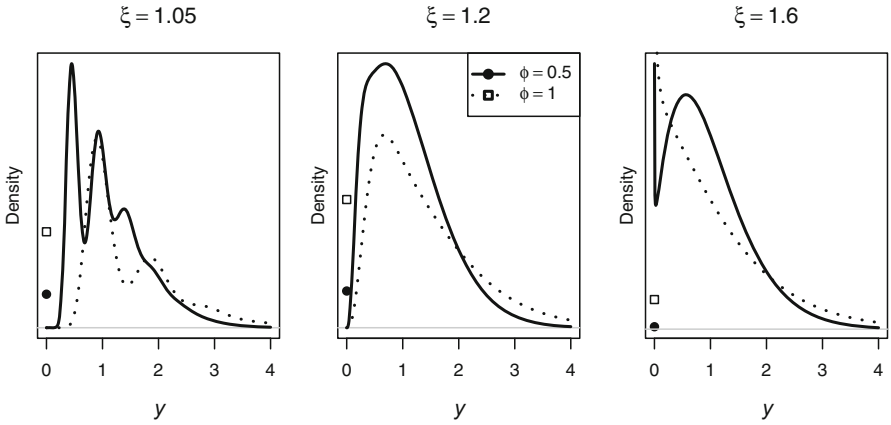

Fig. 12.1 Examples of Tweedie probability functions with $1 < \xi < 2$ and $\mu = 1$. The solid lines correspond to $\phi = 0.5$ and the dotted lines to $\phi = 1$. The filled dots show the probability of exactly zero when $\phi = 0.5$ and the empty squares show the probability of exactly zero when $\phi = 1$ (Sect. 12.2.1)

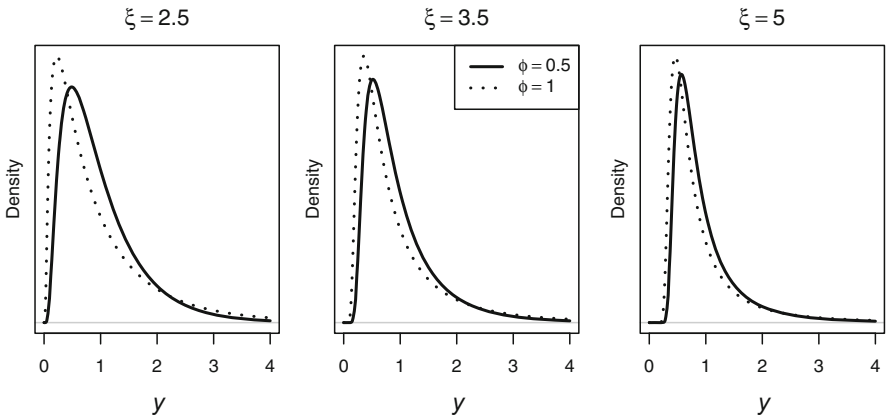

Fig. 12.2 Examples of Tweedie probability functions with $\xi > 2$ and $\mu = 1$. As ξ gets larger, the distributions become more skewed to the right. The solid lines correspond to $\phi = 0.5$; the dotted lines to $\phi = 1$ (Sect. 12.2.1)

ξ is called the *Tweedie index parameter* for the Tweedie distributions, and specifies the particular distribution in the Tweedie family of distributions. The two cases $1 < \xi < 2$ and $\xi \geq 2$ are considered in this chapter in further detail. (The special cases $\xi = 0, 1, 2, 3$ were considered earlier.)

12.2.2 The Structure of Tweedie EDMs

Tweedie distributions are defined as EDMs with variance function $V(\mu) = \mu^\xi$ for some given ξ. Using this relationship, θ and $\kappa(\theta)$ can be determined (following the ideas in Sect. 5.3.6). Setting the arbitrary constants of integration to zero, obtain (Problem 12.1)

$$\theta = \begin{cases} \dfrac{\mu^{1-\xi}}{1-\xi} & \text{for } \xi \neq 1 \\ \log \mu & \text{for } \xi = 1 \end{cases} \quad \text{and} \quad \kappa(\theta) = \begin{cases} \dfrac{\mu^{2-\xi}}{2-\xi} & \text{for } \xi \neq 2 \\ \log \mu & \text{for } \xi = 2 \end{cases}. \tag{12.1}$$

Other parameterizations are obtained by setting the constants of integration to other values. One useful parameterization ensures θ and $\kappa(\theta)$ are continuous functions of ξ [16] (Problem 12.2). The expressions for θ and $\kappa(\theta)$ contain ξ, so the Tweedie distributions are only EDMs if ξ is known. In practice, the value of ξ is usually estimated (Sect. 12.3.2). If y follows a Tweedie distribution with index parameter ξ, mean μ and dispersion parameter ϕ, write $y \sim \text{Tw}_\xi(\mu, \phi)$.

Based on these expressions for θ and $\kappa(\theta)$, the Tweedie probability function may be written in canonical form (5.1). Apart from the special cases identified earlier (the normal, Poisson, gamma and inverse Gaussian distributions), the normalizing constant $a(y, \phi)$ cannot be written in closed form. Consequently, accurate evaluation of the probability function for Tweedie EDMs in general requires numerical methods [15, 16].

The unit deviance is (Problem 12.3)

$$d(y,\mu) = \begin{cases} 2\left\{ \dfrac{\max(y,0)^{2-\xi}}{(1-\xi)(2-\xi)} - \dfrac{y\mu^{1-\xi}}{1-\xi} + \dfrac{\mu^{2-\xi}}{2-\xi} \right\} & \text{for } \xi \neq 1, 2; \\ 2\left\{ y \log \dfrac{y}{\mu} - (y-\mu) \right\} & \text{for } \xi = 1; \\ 2\left(-\log \dfrac{y}{\mu} + \dfrac{y-\mu}{\mu} \right) & \text{for } \xi = 2. \end{cases} \tag{12.2}$$

When $y = 0$, the unit deviance is finite for $\xi \leq 0$ and $1 < \xi < 2$. (Recall $y = 0$ is only admitted for $\xi \leq 0$ and $1 < \xi < 2$; see Table 12.1.)

The Tweedie probability function can be written in the form of a dispersion model (5.13) also, using the unit deviance (12.2). In this form, the normalizing constant $b(y, \phi)$ cannot be written in closed form, apart from the four special cases. By the saddlepoint approximation, $D(y, \hat{\mu}) \sim \chi^2_{n-p'}$ approximately for a model with p' parameters in the linear predictor. The saddlepoint approximation is adequate if $\phi \leq \min\{y\}^{2-\xi}/3$ for the cases $\xi \geq 1$ considered in this chapter (Prob. 12.4). One consequence of this is that the approximation

is likely to be poor if any $y = 0$ (when $1 < \xi < 2$). Also, recall that $\xi = 3$ corresponds to the inverse Gaussian distribution, for which the saddlepoint approximation is exact.

Of interest is the Tweedie rescaling identity [16]. Writing $\mathcal{P}_\xi(y; \mu, \phi)$ for the probability function of a Tweedie EDM with index parameter ξ, then

$$\mathcal{P}_\xi(y; \mu, \phi) = c\mathcal{P}_\xi(cy; c\mu, c^{2-\xi}\phi) \tag{12.3}$$

for all ξ, where $y > 0$ and $c > 0$.

12.2.3 Tweedie EDMs for Positive Continuous Data

In most situations, positive continuous responses are adequately modelled using a gamma or inverse Gaussian distribution (Chap. 11). In some circumstances, neither is adequate, especially for severely skewed data. However, all EDMs with variance functions of the form μ^ξ for $\xi \geq 2$ are suitable for positive continuous data. The gamma ($\xi = 2$) and inverse Gaussian ($\xi = 3$) distributions are just two special cases, and are the only examples of Tweedie EDMs with $\xi \geq 2$ with probability functions that can be written in closed form. One important example corresponds to $V(\mu) = \mu^4$, which is approximately equivalent to using the transformation $1/y$ as the response variable in a linear regression model.

Example 12.1. The survival times (in 10 h units) of animals subjected to three types of poison were measured [6] for four different treatments (Table 12.2; data set: `poison`). Four animals were used for each poison–treatment combination (Fig. 12.3, top panels):

```
> data(poison); summary(poison)
  Psn     Trmt        Time
 I  :16   A:12   Min.   :0.1800
 II :16   B:12   1st Qu.:0.3000
 III:16   C:12   Median :0.4000
          D:12   Mean   :0.4794
                 3rd Qu.:0.6225
                 Max.   :1.2400
```

Table 12.2 Survival times (in 10 h units) for animals under four treatments A, B, C and D, and three poison types I, II and III (Example 12.1)

Poison I				Poison II				Poison III			
A	B	C	D	A	B	C	D	A	B	C	D
0.31	0.82	0.43	0.45	0.36	0.92	0.44	0.56	0.22	0.30	0.23	0.30
0.45	1.10	0.45	0.71	0.29	0.61	0.35	1.02	0.21	0.37	0.25	0.36
0.46	0.88	0.63	0.66	0.40	0.49	0.31	0.71	0.18	0.38	0.24	0.31
0.43	0.72	0.76	0.62	0.23	1.24	0.40	0.38	0.23	0.29	0.22	0.33

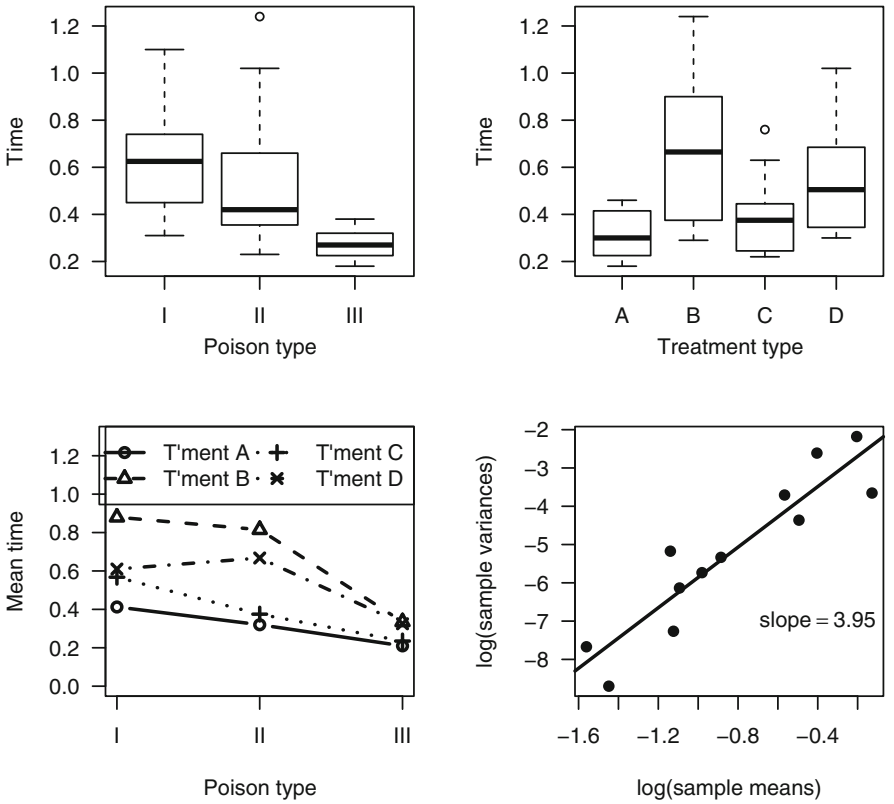

Fig. 12.3 The poison data. The time to death plotted against poison type (top left panel); the time to death plotted against treatment type (top right panel); the mean of the time to death by poison type and treatment type (bottom left panel); the logarithm of each treatment–poison group variance plotted against the logarithm of the group means (bottom right panel) (Example 12.1)

```
> plot( Time ~ Psn,  xlab="Poison type",    las=1, data=poison )
> plot( Time ~ Trmt, xlab="Treatment type", las=1, data=poison )
> GroupMeans <- tapply(poison$Time, list(poison$Psn, poison$Trmt), "mean")
> matplot( GroupMeans, type="b", xlab="Poison type", ylab="Mean time",
          pch=1:4, col="black", lty=1:4, lwd=2, ylim=c(0, 1.3), axes=FALSE)
> axis(side=1, at=1:3, labels=levels(poison$Psn))
> axis(side=2, las=1); box()
> legend("topright", lwd=2, lty=1:4, ncol=2, pch=1:4,
        legend=c("T'ment A", "T'ment B", "T'ment C", "T'ment D"))
```

Finding the variance and the mean of the four observations in each poison–treatment combination and plotting (Fig. 12.3, bottom right panel) shows that the variance is a function of the mean:

```
>    # Find mean and var of each poison/treatment combination
> mns <- tapply(poison$Time, list(poison$Psn, poison$Trmt), mean)
```

```
> vrs <- tapply(poison$Time, list(poison$Psn, poison$Trmt), var)
>    # Plot
> plot( log(c(vrs)) ~ log(c(mns)), las=1, pch=19,
    xlab="log(sample means)", ylab="log(sample variances)")
> mvline <- lm( log( c(vrs) ) ~ log( c(mns) ) )
> slope <- round( coef( mvline )[2], 2); abline( mvline, lwd=2)
> slope
log(c(mns))
       3.95
```

The slope of this line is 3.95, suggesting a Tweedie EDM with $\xi \approx 4$ may be appropriate. □

12.2.4 Tweedie EDMs for Positive Continuous Data with Exact Zeros

Tweedie EDMs with $1 < \xi < 2$ are useful for modelling continuous data with exact zeros. An example of this type of data is insurance claims data [26, 34]. Assume N claims are made in a particular company in a certain time frame, where $N \sim \text{Pois}(\lambda^*)$ where λ^* is the Poisson mean number of claims in the time frame. Observe that N could be zero if no claims are made. When $N > 0$, assume the amount of each claim $i = 1, \ldots, N$ is z_i, where z_i must be positive. Assume z_i follows a gamma distribution with mean μ^* and dispersion parameter ϕ^*, so that $z_i \sim \text{Gam}(\mu^*, \phi^*)$. The *total* insurance payout y is the sum of the N individual claims, such that

$$y = \sum_{i=1}^{N} z_i,$$

where $y = 0$ when $N = 0$. The total claim amount y has a Tweedie distribution with $1 < \xi < 2$. In this interpretation, y is a Poisson sum of gamma distributions, and hence these Tweedie distributions with $1 < \xi < 2$ are sometimes called Poisson–gamma distributions [31], though this term sometimes has another, but related, meaning [17].

Example 12.2. The Quilpie rainfall data were considered in Example 4.6 (data set: `quilpie`), where the probability of observing at least 10 mm of total July rainfall was the quantity of interest. In this example, we examine the total July rainfall in Quilpie. Observe that the total monthly July rainfall is continuous, with exact zeros:

```
> library(GLMsData); data(quilpie)
> head(quilpie)
  Year Rain   SOI Phase Exceed y
1 1921 38.4   2.7     2    Yes 1
2 1922  0.0   2.0     5     No 0
```

```
3 1923  0.0 -10.7     3     No 0
4 1924 24.4   6.9     2    Yes 1
5 1925  0.0 -12.5     3     No 0
6 1926  9.1  -1.0     4     No 0
> sum( quilpie$Rain==0 ) # How many months with exactly zero rainfall?
[1] 20
```

For these data, a Tweedie distribution with $1 < \xi < 2$ may be appropriate. The monthly rainfall could be considered as a Poisson sum of rainfall events each July, with each event producing rainfall amounts that follow a gamma distribution. □

The parameters of the fitted Tweedie EDM defined in Sect. 12.2.2, namely μ, ϕ and ξ, are related to the parameters of the underlying Poisson and gamma distributions by

$$\begin{aligned} \lambda^* &= \frac{\mu^{2-\xi}}{\phi(2-\xi)}; \\ \mu^* &= (2-\xi)\phi\mu^{\xi-1}; \\ \phi^* &= (2-\xi)(\xi-1)\phi^2\mu^{2(\xi-1)}. \end{aligned} \tag{12.4}$$

Tweedie EDMs with $1 < \xi < 2$ are continuous for $y > 0$, but have a positive probability π_0 at $y = 0$, where [15]

$$\pi_0 = \Pr(y = 0) = \exp(-\lambda^*) = \exp\left\{-\frac{\mu^{2-\xi}}{\phi(2-\xi)}\right\}. \tag{12.5}$$

To compute the MLE of π_0, the MLEs of μ, ξ and ϕ must be used in (12.5) (see the first property of MLEs in Sect. 4.9). The MLEs of μ, ξ and ϕ can be computed in R as shown in Sect. 12.3.2.

After computing the MLEs of μ, ϕ and ξ, the MLEs of λ^*, μ^* and ϕ^* can be computed using (12.4). These estimates give an approximate interpretation of the model based on the underlying Poisson and gamma models [7, 12, 15], and may sometimes be useful (see Sect. 12.7).

12.3 Tweedie GLMs

12.3.1 Introduction

GLMs based on the Tweedie distributions are Tweedie GLMs, specified as GLM(Tweedie, ξ; Link function). For both cases considered in this chapter (that is, $\xi > 2$ and $1 < \xi < 2$), we have $\mu > 0$ (Table 12.1). As a result, the usual link function used for Tweedie GLMs is the logarithmic link function. The dispersion parameter ϕ is usually estimated using the Pearson estimate

(though the MLE of ϕ is necessary for computing the MLE of the probability of exact zeros when $1 < \xi < 2$, as explained in Sect. 12.2.4).

To fit Tweedie GLMs, the particular distribution in the Tweedie family must be specified by defining the value of ξ, but usually the value of ξ is unknown and must be estimated before the Tweedie GLM is fitted (Sect. 12.3.2). The correlation between $\hat{\xi}$ and $\hat{\beta}$ is small, so using the estimate $\hat{\xi}$ has only a small effect on inference concerning β compared to knowing the true value of ξ.

Linear regression models using a Box–Cox transformation of the responses can be viewed as an approximation to the Tweedie GLM with the same underlying mean–variance relationship (Problem 12.7); see Sect. 5.8 (p. 232) and Table 5.2. In terms of inference, the normal approximation to the Box–Cox transformed responses can be quite poor when the responses cover a wide range, especially when the responses include exact zeros or near zeros. As a result, the Tweedie GLM approach can often give superior results.

12.3.2 Estimation of the Index Parameter ξ

As noted, fitting a Tweedie GLM requires that the value of the index parameter ξ be known, which identifies the specific Tweedie EDM to use. Since Tweedie distributions are defined as EDMs with $\text{var}[y] = \phi V(\mu) = \phi\mu^{\xi}$, then $\log(\text{var}[y]) = \log\phi + \xi\log\mu$. This shows that a simplistic method for estimating ξ is to divide the data into a small number of groups, and plot the logarithm of the group variances against the logarithm of the group means, as used in Example 12.1 and Example 5.9 (the noisy miner data). However, the estimate of ξ may depend upon how the data are divided.

Note that if exact zeros are present in the data, then $1 < \xi < 2$. However, if the data contains no exact zeros, then $\xi \geq 2$ is common but $1 < \xi < 2$ is still possible. In this situation, one interpretation is that exact zeros are feasible but simply not observed in the given data (Example 12.7).

Example 12.3. For the Quilpie rainfall data (data set: `quilpie`), the mean and variance of the monthly July rainfall amounts can be computed within each SOI phase, and the slope computed. An alternative approach is to compute the mean and variance of the rainfall amounts within each decade:

```
>    # Group by SOI Phase
> mn <- with( quilpie, tapply( Rain, Phase, "mean"))
> vr <- with( quilpie, tapply( Rain, Phase, "var"))
> coef( lm( log(vr) ~ log(mn) ) )
(Intercept)     log(mn)
   1.399527    1.553380
>    # Group by Decade
> Decade <- cut( quilpie$Year, breaks=seq(1920, 1990, by=10) )
> mn <- tapply( quilpie$Rain, Decade, "mean")
> vr <- tapply( quilpie$Rain, Decade, "var")
> coef( lm( log(vr) ~ log(mn) ) )
```

```
(Intercept)      log(mn)
  0.2821267    1.9459524
```

The two methods produce different estimates of ξ, but both satisfy $1 \leq \xi \leq 2$. □

A more rigorous method for estimating ξ, that uses the information in the explanatory variables and is not dependent on the arbitrary dividing of the data, is to compute the maximum likelihood estimator of ξ. A convenient way to organize the calculations is via the *profile likelihood* for ξ. Various values of ξ are chosen, then the Tweedie GLM is fitted for each value of ξ assuming that ξ is fixed, and the log-likelihood computed at each value of ξ. This gives the profile log-likelihood. The value of ξ giving the largest profile log-likelihood is the profile likelihood estimate. A plot of the profile log-likelihood against various values of ξ is often useful.

One difficulty with this method is that the likelihood function for the Tweedie EDMs must be computed, but the probability function for Tweedie EDMs does not have a closed form (Sect. 12.2.2) except in the well-known special cases. However, numerical methods exist for accurately evaluating the Tweedie densities [15, 16], and are used in the R function `tweedie.profile()` (in package **tweedie** [13]) for computing the profile likelihood estimate of ξ. The use of `tweedie.profile()` is demonstrated in Example 12.4, and briefly in Example 12.5. Sometimes, estimating ξ using `tweedie.profile()` may be slow, but once the estimate of ξ has been determined fitting the Tweedie GLM using `glm()` is fast (as computing the value of the likelihood is not needed for estimation).

Example 12.4. The total monthly July rainfall at Quilpie, considered in Example 12.2 (data set: `quilpie`), is continuous but has exact zeros. Following the conclusion in Sect. 4.12 (p. 202), we consider modelling the total July rainfall as a function of the SOI phase [35]. The SOI phase is clearly of some importance (Fig. 12.4, left panel):

```
> quilpie$Phase <- factor(quilpie$Phase)   # Declare  Phase  as a factor
> plot( Rain ~ Phase, data=quilpie, ylab="Total July rainfall",
       ylim=c(0, 100), las=1)
```

Also observe that the variation is greater for larger average rainfall amounts. A suitable estimate of ξ can be found using `tweedie.profile()`:

```
> library(tweedie)
> out <- tweedie.profile( Rain ~ Phase, do.plot=TRUE, data=quilpie)
```

The profile likelihood plot (Fig. 12.4, right panel) shows the likelihood is computed at a small number of ξ values as filled circles, then a smooth curve is drawn through these points. The horizontal dashed line is the value of the log-likelihood at which the approximate 95% confidence interval for ξ is located, using that, approximately,

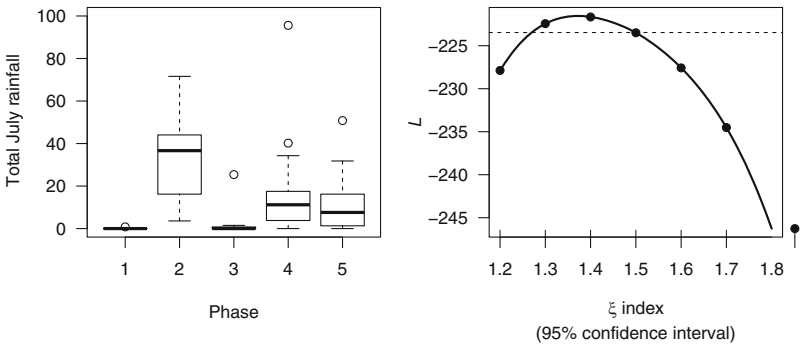

Fig. 12.4 The total July rainfall at Quilpie plotted against SOI phase (left panel), and the profile likelihood plot for estimating ξ (right panel) (Example 12.4)

$$2\left\{\ell(\hat{\xi}; y; \hat{\phi}, \hat{\mu}) - \ell(\xi; y; \hat{\phi}_\xi, \hat{\mu}_\xi)\right\} \sim \chi^2_1,$$

where $\ell(\xi; y; \hat{\phi}_\xi, \hat{\mu}_\xi)$ is the profile log-likelihood at ξ and $\ell(\hat{\xi}; y; \hat{\phi}, \hat{\mu})$ is the overall maximum.

The output object, named `out` in the above, contains a lot of information (see `names(out)`), including the estimate of ξ (as `xi.max`), the nominal 95% confidence interval for ξ (as `ci`), and the MLE of ϕ (as `phi.max`):

```
>    # The index parameter, xi
> xi.est <- out$xi.max
> c( "MLE of xi" = xi.est, "CI for xi" = out$ci )
 MLE of xi CI for xi1 CI for xi2
  1.371429   1.270144   1.499132
>    # Phi
> c("MLE of phi"=out$phi.max)
MLE of phi
  5.558709
```

□

A technical difficulty sometimes arises in estimating ξ, which has been observed by many authors [20, 23, 26]. Recall (Sect. 12.2) that the Tweedie distribution with $\xi = 1$ is suitable for modelling discrete data where $y = 0, \phi, 2\phi, 3\phi, \ldots$. If the responses y are rounded to, say, one decimal place, then the log-likelihood may be maximized by setting $\phi = 0.1$ and $\xi = 1$. Likewise, if the data are rounded to zero decimal places, then the log-likelihood may be maximized setting $\phi = 1$ and $\xi = 1$ (Example 12.5). Dunn and Smyth [15] discuss this problem in greater detail. In practice, the profile likelihood plot produced by `tweedie.profile()` should be examined, and values of ξ near 1 should be avoided as necessary.

Example 12.5. Consider 100 observations randomly generated from a Tweedie distribution with $\xi = 1.5$, $\mu = 2$ and $\phi = 0.5$.

```
> mu <- 2; phi <- 0.5; xi <- 1.5; n <- 100
> library(tweedie)
> rndm <- rtweedie(n, xi=xi, mu=mu, phi=phi)
```

We then estimate the value of ξ from the original data, and then after rounding to one and to zero decimal places (Fig. 12.5):

```
> xi.vec  <- seq(1.01, 1.75, by=0.05)
> out.est <- tweedie.profile( rndm ~ 1,           xi.vec=xi.vec)
> out.1   <- tweedie.profile( round(rndm, 1) ~ 1, xi.vec=xi.vec)
> out.0   <- tweedie.profile( round(rndm, 0) ~ 1, xi.vec=xi.vec)
```

Now compare the estimates of ξ and ϕ for the three cases:

```
> xi.max <- out.est$xi.max
> xi.1 <- out.1$xi.max
> xi.0 <- out.0$xi.max
> compare <- array( dim=c(2, 4))
> colnames(compare) <- c("True", "Estimate", "One d.p.", "Zero d.p.")
> rownames(compare) <- c("xi", "phi")
> compare[1,] <- c(xi, xi.max, xi.1, xi.0)
> compare[2,] <- c(phi, out.est$phi.max, out.1$phi.max, out.0$phi.max)
> round(compare, 3)
    True Estimate One d.p. Zero d.p.
xi   1.5    1.696    1.710     1.010
phi  0.5    0.411    0.407     1.003
```

For these data, rounding to one decimal place only makes a small difference to the log-likelihood, and to the estimate of ξ. However, rounding to zero decimal places produces an artificial maximum in the log-likelihood, where $\xi \to 1$ and $\phi \to 1$. □

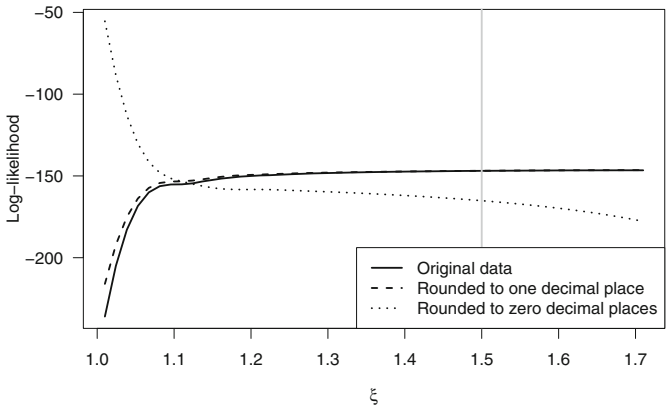

Fig. 12.5 Estimating ξ for some randomly generated data from a Tweedie distribution with $\xi = 1.5$. The gray vertical line is the true value of ξ (Example 12.5)

12.3.3 Fitting Tweedie GLMs

Once an estimate of ξ has been obtained, the Tweedie GLM can be fitted in R using the usual `glm()` function. The Tweedie distributions are denoted in R using `family=tweedie()` in the `glm()` call, after loading the **statmod** package. The call to `family=tweedie()` must specify which Tweedie EDM is to be used (that is, the value of ξ), using the input `var.power`; for example, `family=tweedie(var.power=3)` indicates the Tweedie EDM with $V(\mu) = \mu^3$ should be used. The link function is specified using the input `link.power`, where $\eta = \mu^{\texttt{link.power}}$. Usually, `link.power=0` which corresponds to the logarithmic link function. The logarithm link function is the most commonly-used link function with Tweedie GLMs. As usual, the default link function is the canonical link function.

Once the model has been fitted, quantile residuals [14] are recommended for diagnostic analysis, especially when $1 < \xi < 2$ when exact zeros may be present. Using more than one set of quantile residuals is recommended, due to the randomization used at $y = 0$ (Sect. 8.3.4.2).

Example 12.6. For the Quilpie rainfall data (data set: `quilpie`), the estimate of ξ found in Example 12.4 is $\xi \approx 1.37$. To fit this model in R:

```
> xi.est <- round(xi.est, 2); xi.est
[1] 1.37
> m.quilpie <- glm( Rain ~ Phase, data=quilpie,
                    family=tweedie(var.power=xi.est, link.power=0) )
> printCoefmat(coef(summary(m.quilpie)))
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  -2.1691     1.9560 -1.1089 0.271682
Phase2        5.6923     1.9678  2.8927 0.005239 **
Phase3        3.5153     2.0600  1.7064 0.092854 .
Phase4        5.0269     1.9729  2.5480 0.013287 *
Phase5        4.6468     1.9734  2.3547 0.021665 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

We can compare the Pearson, deviance and quantile residuals (Fig. 12.6):

```
> dres  <-  resid(m.quilpie)    # The default residual
> pres  <-  resid(m.quilpie, type="pearson")
> qres1 <- qresid(m.quilpie)   # Quantile resids, replication 1
> qres2 <- qresid(m.quilpie)   # Quantile resids, replication 2
> qqnorm(dres,  main="Deviance residuals", las=1); qqline(dres)
> qqnorm(pres,  main="Pearson residuals",  las=1); qqline(pres)
> qqnorm(qres1, main="Quantile residuals (set 1)", las=1); qqline(qres1)
> qqnorm(qres2, main="Quantile residuals (set 2)", las=1); qqline(qres2)
```

Compare the Q–Q plot of the deviance, Pearson and quantile residuals (Fig. 12.6): the exact zeros appear as bands in the bottom left corner when using the deviance residuals. When the data contain a large number of exact zeros, this feature makes the plots of the deviance residuals hard to read.

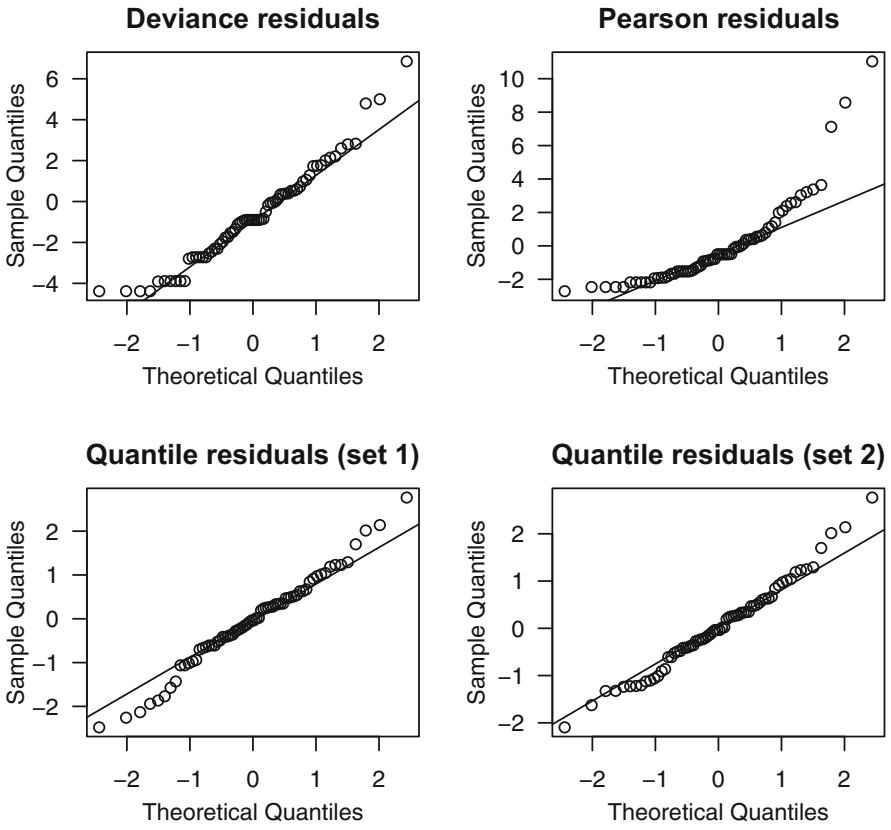

Fig. 12.6 Q–Q plots for the Pearson, deviance and quantile residuals for the Tweedie GLM fitted to the Quilpie rainfall data. Two realization of the quantile residuals are shown (Example 12.6)

The quantile residuals use a small amount of randomization (Sect. 8.3.4.2) to remove these bands. The Q–Q plot of the quantile residuals for these data suggest the model is adequate. Q–Q plots of the other residuals make it difficult to draw definitive conclusions. For this reason, the use of quantile residuals is strongly recommended for use with Tweedie GLMs with $1 < \xi < 2$.

Other model diagnostics (Fig. 12.7) also suggest the model is reasonable:

```
> plot( qres1 ~ fitted(m.quilpie), las=1,
       xlab="Fitted values", ylab="Quantile residuals" )
> plot( cooks.distance(m.quilpie), type="h", las=1,
       ylab="Cook's distance, D")
> plot( qresid(m.quilpie) ~ factor(quilpie$Phase), las=1,
       xlab="Phase", ylab="Quantile residuals" )
```

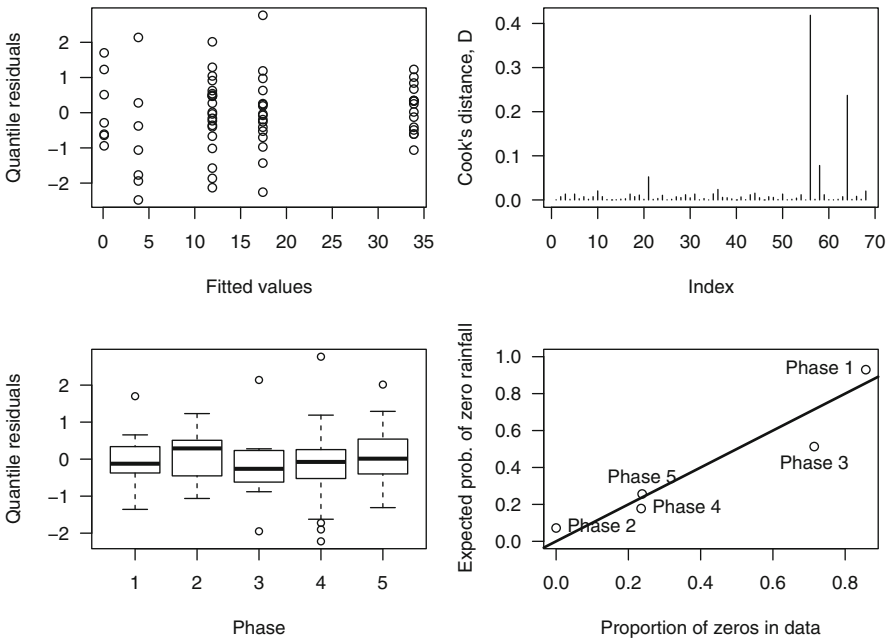

Fig. 12.7 The diagnostics for the Tweedie GLM fitted to the Quilpie rainfall data (Examples 12.6 and 12.7)

No observations are identified as influential using Cook's distance, though DFFITS identifies one observation as influential and CV identifies eight:

```
> q.inf <- influence.measures(m.quilpie)
> colSums(q.inf$is.inf)
  dfb.1_ dfb.Phs2 dfb.Phs3 dfb.Phs4 dfb.Phs5    dffit    cov.r   cook.d
       0        0        0        0        0        1        8        0
     hat
       0
```

□

As shown in Sect. 12.2.4, Tweedie GLMs with $1 < \xi < 2$ can be developed as a Poisson sum of gamma distributions. A fitted GLM can be interpreted on this basis too.

Example 12.7. For the Quilpie rainfall data (data set: `quilpie`), the predicted number of zero-rainfall months $\hat{\pi}_0$ for each SOI phase can be compared to the actual proportion of months in the data with zero rainfall for each SOI phase.

To find the MLE of π_0 using (12.5), the MLE of ϕ must be used, which was conveniently returned by `tweedie.profile()` as `phi.max` (Example 12.4). The plot of the expected probability of a zero against the proportion of zeros in the data for each SOI phase is shown in Fig. 12.7 (bottom right panel):

```
>    # Modelled probability of P(Y=0)
> new.phase <- factor( c(1, 2, 3, 4, 5) )
> mu.phase <- predict(m.quilpie, newdata=data.frame(Phase=new.phase),
                      type="response")
> names(mu.phase) <- paste("Phase", 1:5)
> mu.phase
   Phase 1    Phase 2    Phase 3    Phase 4    Phase 5
 0.1142857 33.8937500  3.8428573 17.4235294 11.9142857
> phi.mle <- out$phi.max
> pi0 <- exp( -mu.phase^(2 - xi.est) / (phi.mle * (2 - xi.est) ) )
>    #
>    # Observed probability of P(Y=0)
> prop0 <- tapply(quilpie$Rain, quilpie$Phase,
                  function(x){sum(x==0)/length(x)})
> #
> plot( pi0 ~ prop0, xlab="Proportion of zeros in data", ylim=c(0, 1),
       ylab="Expected prob. of zero rainfall", las=1 )
> abline(0, 1, lwd=2)  # The line of equality
> text(prop0, pi0,     # Adds labels to the points
    labels=paste("Phase", levels(quilpie$Phase)),
    pos=c(2, 4, 1, 4, 3))  # These position the labels; see  ?text
```

The proportion of months with zero rainfall are predicted with reasonable accuracy. The Tweedie GLM seems a useful model for the total July rainfall in Quilpie.

As suggested in Sect. 12.2.4 (p. 463), the estimated parameters of the GLM can be used to interpret the underlying Poisson and gamma distributions. To do so, use the `tweedie.convert()` function in package **tweedie**:

```
> out <- tweedie.convert(xi=xi.est, mu=mu.phase, phi=phi.mle)
> downscale <- rbind("Poisson mean"     = out$poisson.lambda,
                     "Gamma mean"       = out$gamma.mean,
                     "Gamma dispersion" = out$gamma.phi)
> colnames(downscale) <- paste("Phase", 1:5)
> downscale
                    Phase 1   Phase 2    Phase 3   Phase 4    Phase 5
Poisson mean     0.07281493  2.628215  0.6668339  1.728229  1.3602174
Gamma mean       0.16582834  1.362530  0.6088689  1.065178  0.9254371
Gamma dispersion 1.44678583 97.673944 19.5044793 59.694036 45.0588947
```

In the context of rainfall modelling, this interpretation in terms of λ^*, μ^* and ϕ^* is a form of *statistical downscaling* [11]. The estimates of the Poisson mean λ^* show the mean number of rainfall events in July when the SOI is in each phase, and the estimates of the gamma mean μ^* give the mean amount of rainfall in each rainfall event for each SOI phase. For Phase 2 the model predicts a mean of 2.628 rainfall events occur in July, with a mean of 1.363 mm in each. The mean monthly July rainfall predicted by the model agrees with the observed mean rainfall in the data:

```
> tapply( quilpie$Rain, quilpie$Phase, "mean")  # Mean rainfall from data
        1         2         3         4         5
 0.1142857 33.8937500  3.8428571 17.4235294 11.9142857
```

```
> mu.phase                                        # Mean rainfall from model
   Phase 1    Phase 2    Phase 3    Phase 4    Phase 5
 0.1142857 33.8937500  3.8428573 17.4235294 11.9142857
```

(Note that the boxplots in Fig. 12.4 show the *median* rainfall, not the *mean*.) The estimates of μ^* and ϕ^* are the mean and dispersion parameters for the gamma distribution fitted to the total July rainfall amount for each SOI phase.

Notice that $1 < \xi < 2$ since exact zeros are present in the data. However, exact zeros are not present in every SOI Phase:

```
> tapply(quilpie$Rain, quilpie$Phase, "min")
  1   2   3   4   5
0.0 3.6 0.0 0.0 0.0
```

In other words, even though no months with exactly zero rainfall were observed during Phase 2, the Tweedie GLM assigns a (small) probability that such an event could occur:

```
> round(out$p0, 2)
[1] 0.93 0.07 0.51 0.18 0.26
```

□

12.4 Case Studies

12.4.1 Case Study 1

A study of performance degradation of electrical insulation from accelerated tests [28, 29, 32] measured the dialetric breakdown strength (in kilovolts) for eight time periods (in weeks) and four temperatures (in degrees Celsius). Four measurements are given for each time–temperature combination (data set: `breakdown`), and the study can be considered as a 8×4 factorial experiment.

```
> data(breakdown)
> breakdown$Time <- factor(breakdown$Time)
> breakdown$Temperature <- factor(breakdown$Temperature)
> summary(breakdown)
    Strength          Time    Temperature
 Min.   : 1.00   1      :16   180:32
 1st Qu.:10.00   2      :16   225:32
 Median :12.00   4      :16   250:32
 Mean   :11.24   8      :16   275:32
 3rd Qu.:13.53   16     :16
 Max.   :18.50   32     :16
                 (Other):32
```

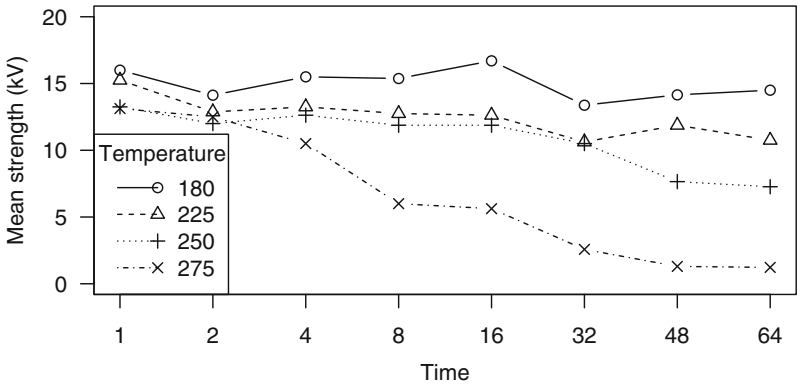

Fig. 12.8 A plot of the dialetric breakdown data (Sect. 12.4.1)

A plot of the data (Fig. 12.8) may suggest that a temperature of 275°C is different than the rest:

```
> bd.means <- with(breakdown,
                   tapply(Strength, list(Time, Temperature), "mean"))
> matplot( bd.means, type="b", col="black",
           pch=1:4, lty=1:4, las=1, ylim=c(0, 20),
           xlab="Time", ylab="Mean strength (kV)", axes=FALSE)
> axis(side=1, at=1:8, labels=levels(breakdown$Time))
> axis(side=2, las=2); box()
> legend("bottomleft", pch=1:4, lty=1:4, merge=FALSE,
    legend=levels(breakdown$Temperature), title="Temperature" )
```

The plot also seems to show that the variance increases as `Time` increases. To consider fitting a Tweedie GLM to the data, we use `tweedie.profile()` to find an estimate of ξ:

```
> bd.xi <- tweedie.profile(Strength~Time*Temperature, data=breakdown,
           do.plot=TRUE, xi.vec=seq(1.2, 2, length=11))
> bd.m <- glm( Strength~factor(Time) * factor(Temperature), data=breakdown,
           family=tweedie(link.power=0, var.power=bd.xi$xi.max))
> anova(bd.m, test="F")
```

Notice that $1 < \xi < 2$ even though all breakdown strengths are positive:

```
> bd.xi$xi.max
[1] 1.591837
```

The Q–Q plot (Fig. 12.9, right panel) suggests no major problems with the model:

```
> qqnorm( resid(bd.m), las=1 ); qqline( resid(bd.m) )
```

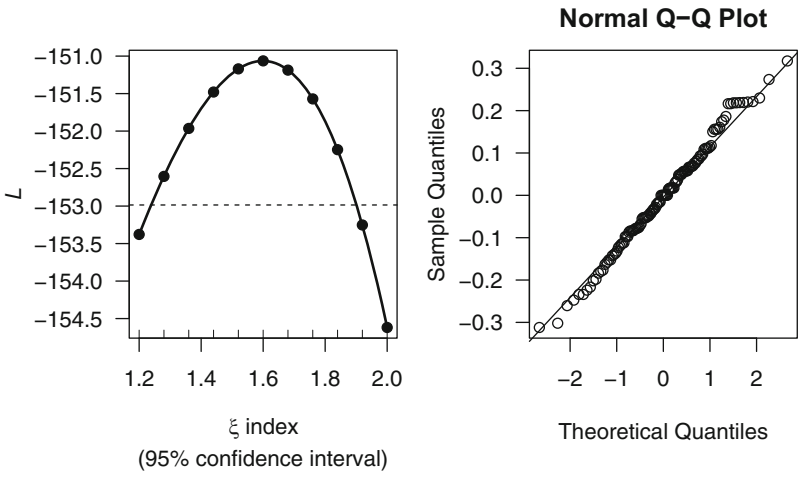

Fig. 12.9 The profile-likelihood plot (left panel) and Q–Q plot of quantile residuals (right panel) for the dialetric breakdown data (Sect. 12.4.1)

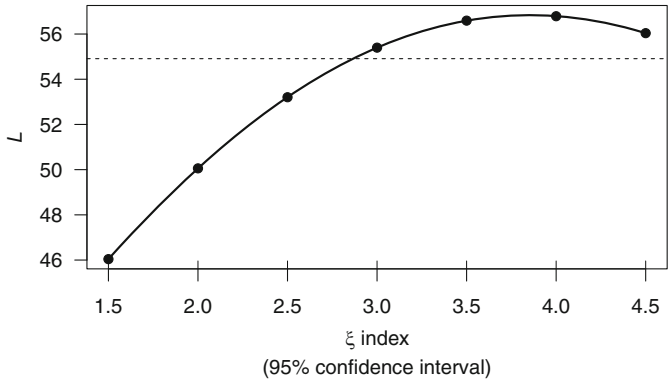

Fig. 12.10 The profile likelihood plot for estimating the value of the Tweedie index parameter ξ for the poison data (Sect. 12.4.2)

12.4.2 Case Study 2

Consider the survival times data first introduced in Example 12.1, where a Tweedie EDM with $\xi \approx 4$ was suggested for modelling the data (data set: `poison`). To find the appropriate Tweedie EDM for modelling the data more formally, initially determine an estimate of ξ using the profile likelihood (Fig. 12.10), using the R function `tweedie.profile()` from the package **tweedie**:

```
> data(poison)
> library(tweedie)   # To provide  tweedie.profile()
```

```
> pn.profile <- tweedie.profile( Time ~ Trmt * Psn, data=poison,
    do.plot=TRUE)
.......Done.
> c("xi: MLE"=pn.profile$xi.max, "xi: CI"=pn.profile$ci)
 xi: MLE  xi: CI1  xi: CI2
3.826531 2.866799       NA
```

These results suggest that fitting a Tweedie GLM using $\hat{\xi} = 4$ is not unreasonable:

```
> library(statmod)  # To provide the  tweedie()  family
> poison.m1 <- glm( Time ~ Trmt * Psn, data=poison,
    family=tweedie(link.power=0, var.power=4))
> anova( poison.m1, test="F")
         Df Deviance Resid. Df Resid. Dev       F    Pr(>F)
NULL                        47     62.239
Trmt      3   19.620        44     42.619 32.7270 2.189e-10 ***
Psn       2   32.221        42     10.398 80.6195 5.053e-14 ***
Trmt:Psn  6    2.198        36      8.199  1.8334      0.12
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The interaction is not significant. The fitted model without the interaction term is:

```
> poison.m2 <- update( poison.m1, . ~ Trmt + Psn )
> summary(poison.m2)
Call:
glm(formula = Time ~ Trmt + Psn, family = tweedie(link.power = 0,
    var.power = 4), data = poison)

Deviance Residuals:
     Min        1Q    Median        3Q       Max
-1.29925  -0.32135  -0.03321   0.20951   0.94121

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -0.82828    0.07938 -10.435 3.10e-13 ***
TrmtB        0.61792    0.08812   7.012 1.40e-08 ***
TrmtC        0.15104    0.06414   2.355   0.0233 *
TrmtD        0.49832    0.08053   6.188 2.13e-07 ***
PsnII       -0.22622    0.09295  -2.434   0.0193 *
PsnIII      -0.77091    0.08007  -9.628 3.43e-12 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for Tweedie family taken to be 0.2656028)

    Null deviance: 62.239  on 47  degrees of freedom
Residual deviance: 10.398  on 42  degrees of freedom
AIC: NA

Number of Fisher Scoring iterations: 8
```

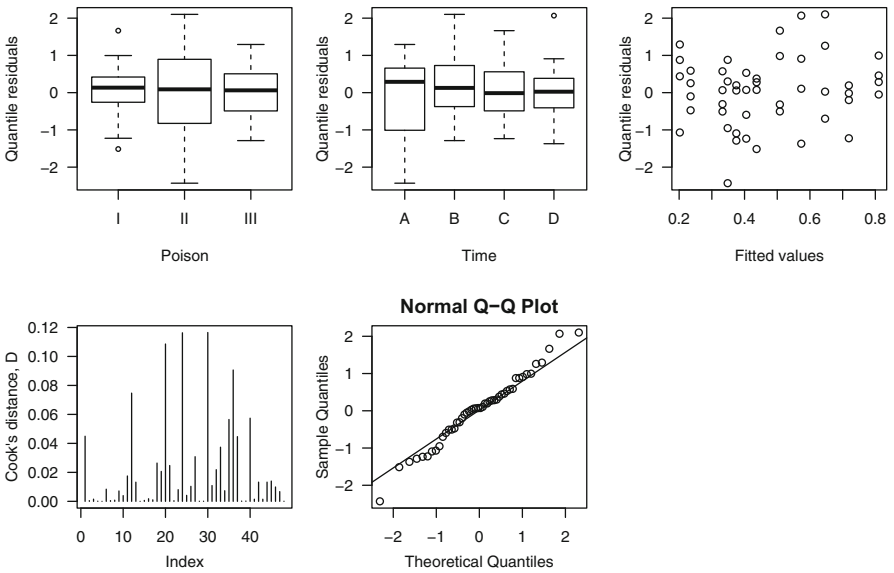

Fig. 12.11 The diagnostics for the final model `poison.m2` fitted to the poison data (Sect. 12.4.2)

Notice the AIC is not computed by default, because the necessary numerical computations may be time consuming. However, the AIC can be computed explicitly using the function `AICtweedie()` in package **tweedie**, suggesting the non-interaction model is preferred:

```
> c("With int"     = AICtweedie(poison.m1),
    "Without int." = AICtweedie(poison.m2))
    With int Without int.
   -87.57423    -88.32050
```

The diagnostic plots suggest model `poison.m2` is adequate (Fig. 12.11), though the residuals for Poison 2 are more variable than for other poisons:

```
> plot( qresid(poison.m2) ~ poison$Psn, las=1,
       xlab="Poison", ylab="Quantile residuals" )
> plot( qresid(poison.m2) ~ poison$Trmt, las=1,
    xlab="Time", ylab="Quantile residuals" )
> plot( qresid(poison.m2) ~ fitted(poison.m2), las=1,
       xlab="Fitted values", ylab="Quantile residuals" )
> plot( cooks.distance(poison.m2), type="h", las=1,
       ylab="Cook's distance, D")
> qqnorm( qr<-qresid(poison.m2), las=1 ); qqline(qr)
```

The final model is GLM(Tweedie, $\xi = 4$; log):

$$\begin{cases} y \sim \mathrm{Tw}_{\xi=4}(\hat{\mu}, \bar{\phi} = 0.2656) & \text{(random)} \\ \log \mathrm{E}[y] = \log \hat{\mu} = \hat{\beta}_0 + \hat{\beta}_1 x_1 + \hat{\beta}_2 x_2 + \hat{\beta}_3 x_3 + \hat{\beta}_4 x_4 + \hat{\beta}_5 x_5 & \text{(systematic)} \end{cases}$$

where the x_j represent dummy variables for the treatment type ($j = 1, 2, 3$) and poison type ($j = 4, 5$). Observe the Pearson estimate of ϕ is given in the output of `summary(poisson.m2)` as $\bar{\phi} = 0.2656$.

These data have also been analysed [6] using the Box–Cox transformation $\lambda = -1$, corresponding to $y^* = 1/y$. This transformation is the variance-stabilizing transformation approximating the Tweedie GLM with $\xi = 4$ (Table 5.2).

12.5 Using R to Fit Tweedie GLMs

Fitting Tweedie GLMs require extra R libraries to be installed (Sect. A.2.5):

- The **tweedie** package [13] is useful for estimating the appropriate value of ξ for a given data set using the function `tweedie.profile()`.
- The **statmod** package [33] is essential for fitting Tweedie GLMs, providing the `tweedie()` GLM family function. It also provides the function `qresid()` for computing quantile residuals, whose use is strongly recommended with Tweedie GLMs.

The `tweedie.profile()` function fixes the value of ξ and fits the Tweedie GLM, then computes the log-likelihood. After doing so for various values of ξ, the profile likelihood estimate of ξ is the value producing the largest value of the log-likelihood. The function may be slow for very large data sets.

The use of `tweedie.profile()` requires a `formula` for specifying the systematic component in the same form as used for `glm()`. Other important inputs are:

- `xi.vec`: The vector of ξ-values to consider. By default, if the response contains zeros then `xi.vec = seq(1.2, 1.8, by=0.1)`, and if the response does not contain zeros then `xi.vec = seq(1.5, 5, by=0.5)`. The likelihood function is smoothed by default (unless `do.smooth=FALSE`) through the likelihood values computed at these values of ξ given in `xi.vec`.
- `do.plot`: Indicates whether to produce a plot of the log-likelihood against ξ, called a *profile likelihood plot*. Producing the plot is recommended to ensure the function has worked correctly and to ensure the problem identified in Sect. 12.3.2 has not occurred. If the plot is not smooth, the `method` may need to be changed. The log-likelihood is evaluated numerically at the values of ξ in `xi.vec`, and these evaluations shown with a filled circle in the profile likelihood plot if `do.plot=TRUE` (by default, `do.plot=FALSE`). An interpolation spline is drawn if `do.smooth=TRUE` (the default).
- `method`: The method used for numerically computing the log-likelihood. Occasionally the method needs to be changed explicitly to avoid difficulties (errors messages may appear; the log-likelihood may be computed as $\pm\infty$ (shown as `Inf` or `-Inf` in R); or the plot of the log-likelihood against

ξ is not smooth). The options include `method = "series"`, `method = "inversion"`, or `method = "interpolation"`. The series method [15] often works well when the inversion method fails [16]. The interpolation method uses either the series or an interpolation of the inversion method results, so is often faster but may produce discontinuities in the profile likelihood plot when the computations change regimes.

- `do.ci`: Produces a nominal 95% confidence interval for the MLE of ξ when `do.ci=TRUE` (which is the default).

The function `tweedie.profile()` returns numerous quantities, the most useful of which are:

- `xi.max`: The profile likelihood estimate of ξ.
- `phi.max`: The MLE of ϕ.
- `ci`: The limits of the approximate 95% confidence interval for ξ (returned if `do.ci=TRUE`, which is the default).

See `?tweedie.profile` for further information.

After installing the **statmod** package, specify a Tweedie GLM in R using `glm(formula, family=tweedie(var.power, link.power))`, where the value of ξ is `var.power`, and `link.power` specifies the link function in the form $\mu^{\texttt{link.power}} = \eta$. Most commonly, `link.power` is zero, specifying the logarithmic link function. (The default link function is the canonical link function; Problem 12.5.) The AIC is not computed and shown in the model `summary()`, because the computations may be slow. If necessary, the AIC can be computed directly using `AICtweedie()` in package **tweedie**.

12.6 Summary

Chapter 12 focuses on fitting Tweedie GLMs to two types of data: Tweedie GLMs for positive continuous data, and Tweedie GLMs for positive continuous data with exact zeros.

The Tweedie distributions are EDMs with the variance function $V(\mu) = \mu^{\xi}$, for $\xi \notin (0, 1)$ (Sect. 12.2). Special cases of Tweedie distributions previously studied are the normal ($\xi = 0$), Poisson ($\xi = 1$ and $\phi = 1$), gamma ($\xi = 2$) and inverse Gaussian ($\xi = 3$) distributions (Sect. 12.2).

The unit deviance is given in (12.2). The residual deviance $D(y, \hat{\mu})$ is suitably described by a $\chi^2_{n-p'}$ distribution if $\phi \leq y^{2-\xi}/3$, but is exact when $\xi = 3$ (the inverse Gaussian distribution) (Sect. 12.2.2).

For $\xi \geq 2$, the Tweedie distributions, and hence Tweedie GLMs, are appropriate for positive continuous data. For $1 < \xi < 2$, the Tweedie distributions, and hence Tweedie GLMs, are appropriate for positive continuous data with exact zeros (Sect. 12.2).

The value of ξ is estimated using the `tweedie.profile()` function from the R package **tweedie** (Sect. 12.3).

Problems

Selected solutions begin on p. 547.

12.1. Deduce the expressions for θ and $\kappa(\theta)$ for the Tweedie EDMs, as given in (12.1) (p. 460), using that $V(\mu) = \mu^{\xi}$. Set the arbitrary constants of integration to zero. (HINT: Follow the approach in Sect. 5.3.6, p. 217.)

12.2. In Problem 12.1, expressions for θ and $\kappa(\theta)$ were found by setting the arbitrary constants of integration to zero. In this problem we consider an alternative parameterization [15].

1. By appropriately choosing the constants of integration, show that alternative expressions for θ and $\kappa(\theta)$ can be written as

$$\theta = \begin{cases} \dfrac{\mu^{1-\xi} - 1}{1-\xi} & \text{for } \xi \neq 1 \\ \log \mu & \text{for } \xi = 1 \end{cases} \quad \text{and} \quad \kappa(\theta) = \begin{cases} \dfrac{\mu^{2-\xi} - 1}{2-\xi} & \text{for } \xi \neq 2 \\ \log \mu & \text{for } \xi = 2 \end{cases} \tag{12.6}$$

2. Show that θ is continuous in ξ. (HINT: Use that $\lim_{\alpha \to 0}(x^{\alpha} - 1)/\alpha \to \log x$.)
3. Likewise, show that $\kappa(\theta)$ is continuous in ξ.

12.3. Deduce the unit deviance for the Tweedie EDMs given in (12.2) (p. 460).

12.4. Using the guideline presented in Sect. 5.4.5 (p. 226), show that the residual deviance $D(y, \hat{\mu})$ is likely to follow a $\chi^2_{n-p'}$ distribution when $\phi \leq y^{2-\xi}/3$ when $\xi \geq 1$. Hence show that the saddlepoint approximation is likely to be poor for continuous data with exact zeros.

12.5. Deduce the canonical link function for the Tweedie EDMs.

12.6. Consider the rescaling identity in (12.3).

1. Using this identity, deduce the Tweedie EDM for which the value of ϕ does not change when a change of measurement units (say, from grams to kilograms) is applied to the data y.
2. Using this identity, deduce the Tweedie EDM for which value of ϕ increases by the same factor as that used for a change of measurement units in the data y.
3. What does the identity reveal about the case of the inverse Gaussian distribution in the case of a change in measurement units in y?
4. Show that the probability function for any Tweedie EDM $\mathcal{P}_{\xi}(y; \mu, \phi)$ can be computed by an evaluation at $\mu = 1$ (that is, $\mathcal{P}_{\xi}(y^*; 1, \phi^*)$), by finding the appropriately-redefined values of y^* and ϕ^*.

12.7. Consider the Box–Cox transformation (Sect. 3.9, p. 116).

1. Show that the Box–Cox transformation for any λ approximates fitting a GLM based on a EDM with variance function $V(\mu) = \mu^{2(1-\lambda)}$ if $\mu > 0$. (Use a Taylor series of the transformation expanded about the mean μ, as in Sect. 5.8.)
2. No Tweedie EDMs exist when $0 < \xi < 1$. Use this result to show no equivalent power-variance GLM exists for the Box–Cox transformations corresponding to $0.5 < \lambda < 1$.

12.8. A study of monthly rainfall in Australia [22] fitted Tweedie GLMs to a number of different rainfall stations using $\hat{\xi} = 1.6$. For Bidyadanga monthly rainfall from 1912 to 2007, the fitted systematic component was

$$\log \hat{\mu}_m = 2.903 + 1.908 \sin(2\pi m/12) + 0.724 \cos(2\pi m/12),$$

where $m = 1, 2, \ldots 12$ corresponds to the month of the year (for example, February corresponds to $m = 2$). The standard errors for the parameter estimates are (respectively) 0.066, 0.090 and 0.085, and the MLE of ϕ is 8.33.

1. Compute the Wald statistic for testing if each regression parameter is zero.
2. Plot the value of $\hat{\mu}_m$ against m for $m = 1, \ldots, 12$ for Bidyadanga.
3. Plot the predicted value of π_0 against m for $m = 1, \ldots, 12$ for Bidyadanga.

12.9. A study [10] of the walking habits of adults living in south-east Queensland, Australia, compared different types of Statistical Areas classified by their *walk score* [9] as 'Highly walkable', 'Somewhat walkable', 'Car-dependent' or 'Very car-dependent' (Table 12.3). The Tweedie GLM was fitted using $\hat{\xi} = 1.5$.

1. Explain the differences between the predicted mean walking times in both sections of the table. Why are the predicted means all larger for the second model ('walking adults')?
2. A Tweedie GLM was fitted for 'All adults' and a gamma GLM for 'Walking adults'. Explain why these models may have been chosen.
3. The deviance from the fitted Tweedie GLM was 5976.08 on 1242 degrees of freedom. Use this information to find an estimate of ϕ.
4. Using the Tweedie GLM, find an estimate of the proportion of all adults who did no walking in each of the four types of walkability descriptions, and comment. Why are these values not the MLEs of the π_0?

12.10. A study of polythene use by cosmetic companies in the UK [19] hypothesized a relationship with company turnover (Table 12.4; data set: `polythene`). Consider two Tweedie GLMs models for the data, both using a logarithmic link function for the systematic component: the first using `Polythene~Turnover`, and the second using `Polythene~log(Turnover)`.

1. Find estimates of ξ for each model.

Table 12.3 Predicted mean number of minutes of walking per day in four types of regions, adjusted for work status, household car ownership and driver's license status (Problem 12.9)

	All adults		Walking adults	
	n	Predicted mean	n	Predicted mean
Highly walkable	214	7.5	155	25.5
Somewhat walkable	407	4.7	255	25.4
Car-dependent	441	2.9	254	21.2
Very car-dependent	187	2.5	90	18.3

Table 12.4 The company turnover and polythene use for 23 cosmetic companies in the UK (to preserve confidentiality, the data were scaled) (Problem 12.10)

Polythene use (in tonnes)	Turnover (in £00 000)	Polythene use (in tonnes)	Turnover (in £00 000)	Polythene use (in tonnes)	Turnover (in £00 000)
0.04	0.02	31.50	9.85	587.83	83.94
1.60	0.23	472.50	21.13	1068.92	106.13
0.00	3.17	0.00	24.40	676.20	156.01
0.00	3.46	94.50	30.18	1056.30	206.43
3.78	3.55	55.94	40.13	1503.60	240.51
29.40	4.62	266.53	68.40	1438.50	240.93
8.00	5.71	252.53	70.88	2547.30	371.68
95.13	7.77			4298.70	391.33

2. Fit the GLMs to the data, and interpret the models.
3. On two separate plots of polythene use against turnover, plot the systematic components of both models, including the 95% confidence interval for the fitted lines. Comment on the models.
4. Compute the AIC for both models, and comment.
5. Produce the appropriate diagnostic plots for both models.
6. Deduce a suitable model for the data.

12.11. Consider the permeability of building material data given in Table 11.2 (data set: `perm`). In Sect. 11.7 (p. 440), the positive continuous response was modelled using an inverse Gaussian GLM for interpretation reasons. Jørgensen [24] also considers a gamma ($\xi = 2$) GLM for the data.

1. Determine an estimate of ξ using `tweedie.profile()`. What EDM is suggested?
2. Fit a suitable Tweedie GLM ensuring an appropriate diagnostic analysis.

12.12. A study of human energy expenditure measured the energy expenditure y of 104 females over a 24-h period (Table 12.5; data set: `energy`), and also recorded their fat-tissue mass x_1 and non-fat tissue x_2 mass [18, 24]. A model for the energy expenditure is $\mathrm{E}[y] = \beta_1 x_1 + \beta_2 x_2$, assuming the

Table 12.5 The energy expenditure and mass of 104 females (units not given). Only the first six observations are shown (Problem 12.12)

Energy expenditure	Mass of fat tissue	Mass of non-fat tissue
60.08	17.31	43.22
60.08	34.09	43.74
63.69	33.03	48.72
64.36	9.14	50.96
65.37	30.73	48.67
66.05	20.74	65.31
⋮	⋮	⋮

energy expenditure for each tissue type is homogenous. Since the total mass is $M = x_1 + x_2$, divide by M and rewrite as $\mathrm{E}[\bar{y}] = \beta_2 + (\beta_1 - \beta_2)\bar{x}$, where $\bar{y} = y/M$ is the energy expenditure per unit mass, and $\bar{x} = x_1/M$ is the proportion of fat-tissue mass.

1. Plot $\bar{y}$ against $\bar{x}$ and confirm the approximate linear relationship between the variables.
2. Use `tweedie.profile()` to estimate ξ for the data. Which Tweedie EDMs is appropriate?
3. Find a suitable GLM for the data, ensuring a diagnostic analysis.

12.13. The data described in Table 12.6 (data set: `motorins1`) concern third party motor insurance claims in Sweden for the year 1977 [1, 21, 32]. The description of the data states that Swedish motor insurance companies "apply identical risk arguments to classify customers, and thus their portfolios and their claims statistics can be combined" [1, p. 413]. The data set contains 315 observations representing one of the zones in the country (covering Stockholm, Göteborg, and Malmö with surroundings).

For the remainder of the analysis, consider payments in millions of Kroner. Policies are categorized by kilometres of travel (five categories), the no-claim bonus (seven categories) and make of car (nine categories), for a total of 315 categories. Of these, 20 contain exactly zero claims, so the total payout in those categories is exactly zero; in other categories, the total payout can be consider continuous. Find an appropriate model for the data. (HINT: You will need to change the range of ξ values considered by `tweedie.profile()` using the `xi.vec` input.)

Using your fitted model, interpret the model using the parameters of the underlying Poisson and gamma distributions. (HINT: See (12.4), p. 464.)

12.14. The total monthly August rainfall for Emerald (located in Queensland, north eastern Australia) from 1889 to 2002 is shown in Table 12.7 (data set: `emeraldaug`) with the monthly average southern oscillation index (SOI). Negative values of the SOI often indicate El Niño episodes, which are often associated with reduced rainfall in eastern and northern Australia [27].

Table 12.6 A description of the variables used in the Swedish insurance claims data set (Problem 12.13)

Variable	Description
`Kilometres`:	Kilometres travelled per year: 1: Less than 1000 2: 1000–15,000 3: 15,000–20,000 4: 20,000–25,000 5: More than 25,000
`Bonus`:	No claims bonus; the number of years since last claim, plus one
`Make`:	1–8 represent eight different common car models. All other models are combined in class 9
`Insured`:	Number of insured in policy-years
`Claims`:	Number of claims
`Payment`:	Total value of payments in Skr (Swedish Kroner)

Table 12.7 The total monthly rainfall in August from 1889–2002 in Emerald, Australia, plus the monthly average SOI and corresponding SOI phases. The first five observations are shown (Problem 12.14)

Year	Rain (in mm)	SOI	SOI phase
1889	15.4	2.1	5
1890	47.5	−3.1	5
1891	45.7	−8.9	5
1892	0.0	5.9	2
1893	108.7	7.8	2
⋮	⋮	⋮	⋮

1. Argue that the Poisson–gamma models are appropriate for monthly rainfall data, along the lines of the argument in Sect. 12.2.4 (p. 463).
2. Perform a hypothesis test to address the relationship between rainfall and SOI given earlier in the question to see if it applies at Emerald: "Negative values of the SOI... are often associated with reduced rainfall in eastern and northern Australia."
3. Fit an appropriate EDM for modelling the total monthly August rainfall in Emerald from the SOI.
4. Compute the 95% confidence interval for the SOI parameter, and determine the practical importance of SOI for August rainfall in Emerald.
5. Fit an appropriate EDM for modelling the total monthly August rainfall in Emerald from the SOI phases.
6. Interpret the fitted model using SOI phases, using the parameters of the underlying Poisson and gamma distributions. (HINT: See (12.4), p. 464.)

Table 12.8 Data from 194 trawls in the South East Fisheries ecosystem regarding the catch of tiger flathead. Distance is measured north to south on the 100 m depth contour (Problem 12.15)

Longitude of trawl	Latitude of trawl	Depth (in m)	Distance (in m)	Swept area (in ha)	Number of tiger flathead	Biomass of tiger flathead (in kg)
149.06	−37.81	−33	91	4.72260	1	0.02
149.08	−37.83	−47	90	5.00040	0	0.00
149.11	−37.87	−74	89	6.11160	153	30.70
149.22	−38.02	−117	88	5.83380	15	7.77
149.27	−38.19	−212	88	3.04222	0	0.00
150.29	−37.41	−168	48	6.11160	25	6.90
150.19	−37.33	−113	48	5.83380	53	15.30
⋮	⋮	⋮	⋮	⋮	⋮	⋮

12.15. A study on the South East Fisheries ecosystem near Australia [4] collected data about the number of fish caught from fish trawl surveys. One analysis of these data [17] studied the number of tiger flathead (Table 12.8; data set: `flathead`).

1. The data record the number of flathead caught per trawl plus the total biomass of the flathead caught. Propose a mechanism for the total biomass that leads to the Tweedie GLM as a possible model (similar to that used in Sect. 12.2.4).
2. The paper that analysed the data [17] fits a Poisson GLM to model the number of tiger flathead caught. The paper states

   > ... the dependence on covariates, if any, is specified using orthogonal polynomials in the linear predictor. The dependency on depth used a second order polynomial and the dependency on along-coast used a third order polynomial... The log of the area swept variable was included as an offset (p. 542).

   Explain why area is used as an offset.
3. Based on the information above, fit an appropriate Poisson GLM for modelling the *number* of tiger flathead caught (using `Depth` and `Distance` as covariates, in the manner discussed in the quote above). Show that this model has large overdispersion, and hence fit a quasi-Poisson model. Propose a reason why overdispersion is observed.
4. Based on the above information, plot the logarithm of biomass against the depth and distance, and comment on the relationships.
5. The paper that analysed the biomass data [17] stated that

   > There is no reason to include an extra spatial dimension... as it would be highly confounded with depth (p. 541).

   Determine if any such correlation exists between depth, and the latitude and longitude.

Table 12.9 Feeding rates (in feeds per hour) of chestnut-crowed babblers (Problem 12.16)

Feeding rate	Observation time (h)	Sex	Chick age (days)	Non-breeding birds ages	Brood size
0.000	11.09	M	1	Adult	3
0.000	11.16	M	2	Adult	4
0.000	12.81	M	3	Adult	1
0.238	12.59	M	4	Adult	1
1.316	12.16	M	5	Adult	1
1.041	11.53	M	6	Adult	1
⋮	⋮	⋮	⋮	⋮	⋮
0.321	6.22	F	19	Adult	3
0.000	6.22	M	19	Yearling	3

6. The paper that analysed the biomass data [17] used a Tweedie GLM (using `Depth` and `Distance` as covariates, in the manner discussed in the quote above). Based on the above information, fit a suitable Tweedie GLM, and assess the model using diagnostics.
7. Compare the Q–Q plot of the deviance and quantile residuals from the Tweedie GLM, and comment.

12.16. Chestnut-crowned babblers are medium-sized Australian birds that live in social groups. A study of their feeding habits [8] recorded, among other things, the rates at which they fed, in feeds per hour (Table 12.9; data set: `babblers`). About 18% of the feeding rates are exact zeros. Fit a Tweedie GLM to the data to model the feeding rates.

12.17. A study comparing two different types of toothbrushes [2, 30] measured the plaque index for females and males before and after brushing (Table 12.10; data set: `toothbrush`). Smaller values mean cleaner teeth. The 26 subjects all used both toothbrushes. One subject received the same plaque index before and after brushing.

Assuming the plaque index cannot become worse after brushing, fit an appropriate GLM to the data for modelling the difference (Before − After), and deduce if the toothbrushes appear to differ in their teeth-cleaning ability, and if this seems related to the sex of the subject.

12.18. An experiment [3] to quantify the effect of ketamine (an anaesthetic) measured the amount of sleep (in min) for 30 guinea pigs, using five different doses (Table 12.11; data set: `gpsleep`).

1. Explain what the exact zeros mean.
2. Plot the data, and show that the variance increases with the mean.
3. Plot the logarithm of the group variances against the logarithm of the group means, where the groups are defined by the doses. Show this implies $\xi \approx 1$.

Table 12.10 The plaque index before and after brushing for two types of toothbrushes; smaller values indicate cleaner teeth (Problem 12.17)

Conventional brush				Hugger (new) brush			
Females		Males		Females		Males	
Before	After	Before	After	Before	After	Before	After
1.20	0.75	3.35	1.58	2.18	0.43	0.90	0.15
1.43	0.55	1.50	0.20	2.05	0.08	0.58	0.10
0.68	0.08	4.08	1.88	1.05	0.18	2.50	0.33
1.45	0.75	3.15	2.00	1.95	0.78	2.25	0.33
0.50	0.05	0.90	0.25	0.28	0.03	1.53	0.53
2.75	1.60	1.78	0.18	2.63	0.23	1.43	0.43
1.25	0.65	3.50	0.85	1.50	0.20	3.48	0.65
0.40	0.13	2.50	1.15	0.45	0.00	1.80	0.20
1.18	0.83	2.18	0.93	0.70	0.05	1.50	0.25
1.43	0.58	2.68	1.05	1.30	0.30	2.55	0.15
0.45	0.38	2.73	0.85	1.25	0.33	1.30	0.05
1.60	0.63	3.43	0.88	0.18	0.00	2.65	0.25
0.25	0.25			3.30	0.90		
2.98	1.03			1.40	0.24		

Table 12.11 Amount of sleep (in min) for 30 guinea pigs after receiving intravenous doses of ketamine (Problem 12.18)

0.60 mg/kg		1.04 mg/kg		1.44 mg/kg		2.00 mg/kg		2.75 mg/kg	
0.00	0.00	0.00	0.00	0.00	3.60	5.59	7.67	0.00	1.71
0.00	0.00	2.85	5.92	8.32	8.50	9.40	9.77	11.15	11.89
3.99	4.78	7.36	10.43	12.73	13.20	10.92	24.80	14.48	14.75

4. Using `tweedie.profile()`, show that $\hat{\xi} = 1.1$. (HINT: Try using `xi.vec = (1.02, 1.4, by=0.02)` to ensure you obtain a good estimate of ξ.)
5. Show that a quadratic Tweedie GLM in `Dose` is significantly better than the Tweedie GLM linear is `Dose`.
6. Also consider the linear and quadratic Tweedie GLM using `log(Dose)` in place of `Dose`.
7. Also consider a Tweedie GLM using a natural cubic spline, with `knots= quantile(Dose, c(0.33, 0.67))`.
8. Plot all five systematic component on a plot of the data, and comment.
9. Use the AIC to determine a model from the five considered, and show the quadratic model in `Dose` is the preferred model.

References

[1] Andrew, D.F., Herzberg, A.M.: Data: A Collection of Problems from Many Fields for the Student and Research Worker. Springer, New York (1985)

[2] Aoki, R., Achcar, J.A., Bolfarine, H., Singer, J.M.: Bayesian analysis of null-intercept errors-in-variables regression for pretest/post-test data. Journal of Applied Statistics **31**(1), 3–12 (2003)

[3] Bailey, R.C., Summe, J.P., Hommer, L.D., McCracken, L.E.: A model for the analysis of the anesthetic response. Biometrics **34**(2), 223–232 (1978)

[4] Bax, N.J., Williams, A.: Habitat and fisheries production in the South East fishery ecosystem. Final Report 1994/040, Fisheries Research and Development Corporation (2000)

[5] Box, G.E.P.: Science and statistics. Journal of the American Statistical Association **71**, 791–799 (1976)

[6] Box, G.E.P., Cox, D.R.: An analysis of transformations (with discussion). Journal of the Royal Statistical Society, Series B **26**, 211–252 (1964)

[7] Brown, J.E., Dunn, P.K.: Comparisons of Tobit, linear, and Poisson-gamma regression models: an application of time use data. Sociological Methods & Research **40**(3), 511–535 (2011)

[8] Browning, L.E., Patrick, S.C., Rollins, L.A., Griffith, S.C., Russell, A.F.: Kin selection, not group augmentation, predicts helping in an obligate cooperatively breeding bird. Proceedings of the Royal Society B **279**, 3861–3869 (2012)

[9] Carr, L.J., Dunsiger, S.I., Marcus, B.H.: Validation of Walk Score for estimating access to walkable amenities. British Journal of Sports Medicine **45**(14), 1144–1148 (2011)

[10] Cole, R., Dunn, P., Hunter, I., Owen, N., Sugiyama, T.: Walk score and Australian adults' home-based walking for transport. Health & Place **35**, 60–65 (2015)

[11] Connolly, R.D., Schirmer, J., Dunn, P.K.: A daily rainfall disaggregation model. Agricultural and Forest Meteorology **92**(2), 105–117 (1998)

[12] Dunn, P.K.: Precipitation occurrence and amount can be modelled simultaneously. International Journal of Climatology **24**, 1231–1239 (2004)

[13] Dunn, P.K.: tweedie: Tweedie exponential family models (2017). URL https://CRAN.R-project.org/package=tweedie. R package version 2.3.0

[14] Dunn, P.K., Smyth, G.K.: Randomized quantile residuals. Journal of Computational and Graphical Statistics **5**(3), 236–244 (1996)

[15] Dunn, P.K., Smyth, G.K.: Series evaluation of Tweedie exponential dispersion models. Statistics and Computing **15**(4), 267–280 (2005)

[16] Dunn, P.K., Smyth, G.K.: Evaluation of Tweedie exponential dispersion models using Fourier inversion. Statistics and Computing **18**(1), 73–86 (2008)
[17] Foster, S.D., Bravington, M.V.: A Poisson–gamma model for analysis of ecological data. Environmental and Ecological Statistics **20**(4), 533–552 (2013)
[18] Garby, L., Garrow, J.S., Jørgensen, B., Lammert, O., Madsen, K., Sørensen, P., Webster, J.: Relation between energy expenditure and body composition in man: Specific energy expenditure in *vivo* of fat and fat-free tissue. European Journal of Clinical Nutrition **42**(4), 301–305 (1988)
[19] Gilchrist, R.: Regression models for data with a non-zero probability of a zero response. Communications in Statistics—Theory and Methods **29**, 1987–2003 (2000)
[20] Gilchrist, R., Drinkwater, D.: Fitting Tweedie models to data with probability of zero responses. In: H. Friedl, A. Berghold, G. Kauermann (eds.) Statistical Modelling: Proceedings of the 14th International Workshop on Statistical Modelling, pp. 207–214. International Workshop on Statistical Modelling, Gräz (1999)
[21] Hallin, M., François Ingenbleek, J.: The Swedish automobile portfolio in 1997. Scandinavian Actuarial Journal pp. 49–64 (1983)
[22] Hasan, M.M., Dunn, P.K.: A simple Poisson–gamma model for modelling rainfall occurrence and amount simultaneously. Agricultural and Forest Meteorology **150**, 1319–1330 (2010)
[23] Jørgensen, B.: Exponential dispersion models (with discussion). Journal of the Royal Statistical Society, Series B **49**, 127–162 (1987)
[24] Jørgensen, B.: Exponential dispersion models and extensions: A review. International Statistical Review **60**(1), 5–20 (1992)
[25] Jørgensen, B.: The Theory of Dispersion Models. Monographs on Statistics and Applied Probability. Chapman and Hall, London (1997)
[26] Jørgensen, B., de Souza, M.C.P.: Fitting Tweedie's compound Poisson model to insurance claims data. Scandinavian Actuarial Journal **1**, 69–93 (1994)
[27] McBride, J.L., Nicholls, N.: Seasonal relationships between Australian rainfall and the southern oscillation. Monthly Weather Review **111**(10), 1998–2004 (1983)
[28] National Institute of Standards and Technology: Statistical reference datasets (2016). URL http://www.itl.nist.gov/div898/strd
[29] Nelson, W.: Analysis of performance-degradation data from accelerated tests. IEEE Transactions on Reliability **30**(2), 149–155 (1981)
[30] Singer, J.M., Andrade, D.F.: Regression models for the analysis of pretest/posttest data. Biometrics **53**, 729–725 (1997)

[31] Smyth, G.K.: Regression analysis of quantity data with exact zeros. In: Proceedings of the Second Australia–Japan Workshop on Stochastic Models in Engineering, Technology and Management, pp. 572–580. Technology Management Centre, University of Queensland, Brisbane (1996)

[32] Smyth, G.K.: Australasian data and story library (OzDASL) (2011). URL http://www.statsci.org/data

[33] Smyth, G.K.: statmod: Statistical Modeling (2017). URL https://CRAN.R-project.org/package=statmod. R package version 1.4.30. With contributions from Yifang Hu, Peter Dunn, Belinda Phipson and Yunshun Chen.

[34] Smyth, G.K., Jørgensen, B.: Fitting Tweedie's compound Poisson model to insurance claims data: Dispersion modelling. In: Proceedings of the 52nd Session of the International Statistical Institute. Helsinki, Finland (1999). Paper Meeting 68: Statistics and Insurance

[35] Stone, R.C., Auliciems, A.: SOI phase relationships with rainfall in eastern Australia. International Journal of Climatology **12**, 625–636 (1992)

[36] Taylor, L.R.: Aggregation, variance and the mean. Nature **189**, 732–735 (1961)

[37] Tweedie, M.C.K.: The regression of the sample variance on the sample mean. Journal of the London Mathematical Society **21**, 22–28 (1946)

Chapter 13
Extra Problems

Practice is the best of all instructors.
Publilius Syrus [19, Number 439]

13.1 Introduction and Overview

In previous chapters, problems were supplied relevant to the material in that chapter. In this final chapter, we present a series of problems without the chapter context, and often with less direction for modelling the data.

Problems

13.1. A study of pubertal timing of youths [5, Table III] tabulated the relationship between gender, when they matured, and the satisfaction with their current weight (Table 13.1; data set: `satiswt`).

1. Identify the zero as either structural or sampling.
2. Find a suitable model for the data, ensuring an appropriate diagnostic analysis.
3. Interpret the final model.

13.2. The data in Table 13.2 (data set: `toxo`) give the proportion of the population testing positive to toxoplasmosis y against the annual rainfall (in mm) x for 34 cities in El Salvador [7]. Plot the data, and describe the important features of the data. Then, find a suitable model for the data. (HINT: A complicated systematic component is necessary; see Problem 1.4.)

13.3. A study [15, 17] examined the effects of boric acid, a compound in household products and pesticides, on *in utero* embryo damage in mice (Table 13.3; data set: `boric`). Find a suitable model for modelling the effect of bromic acid on *in utero* damage in mice.

P. K. Dunn, G. K. Smyth, *Generalized Linear Models with Examples in R*,
Springer Texts in Statistics, https://doi.org/10.1007/978-1-4419-0118-7_13

Table 13.1 The number of youths classified by gender, when they matured, and their own opinions about their weight (Problem 13.1)

		Number who wish to be		
	Matured	Thinner	Same weight	Heavier
Girls	Late	91	171	74
	Mid	1170	861	177
	Early	84	36	0
Boys	Late	87	164	101
	Mid	418	1300	604
	Early	46	127	15

Table 13.2 The proportion of people testing positive to toxoplasmosis in 34 cities in El Salvador (Problem 13.2)

Rainfall (in mm)	Proportion	Sampled	Rainfall (in mm)	Proportion	Sampled
1735	0.50	4	1770	0.61	54
1936	0.30	10	2240	0.44	9
2000	0.20	5	1620	0.28	18
1973	0.30	10	1756	0.17	12
1750	1.00	2	1650	0.00	1
1800	0.60	5	2250	0.73	11
1750	0.25	8	1796	0.53	77
2077	0.37	19	1890	0.47	51
1920	0.50	6	1871	0.44	16
1800	0.80	10	2063	0.56	82
2050	0.29	24	2100	0.69	13
1830	0.00	1	1918	0.54	43
1650	0.50	30	1834	0.71	75
2200	0.18	22	1780	0.61	13
2000	0.00	1	1900	0.30	10
1770	0.54	11	1976	0.17	6
1920	0.00	1	2292	0.62	37

13.4. In the Birth to Ten study (BTT) from the greater Johannesburg–Soweto metropolitan area of South Africa during 1990, all mothers of singleton births (4019 births) who had a permanent address within a defined area were interviewed during a seven-week period between April and June 1990 [13]. (Singleton births are non-multiple births; that is, no twins, triplets, etc.) Five years later, 964 of these mothers were re-interviewed.

For further research to be useful, the mothers not followed-up five years later (Group 1) should have similar characteristics to those mothers who were followed-up five years later (Group 2). One of the factors for comparison was whether the mother had medical aid (similar to health insurance) at the time of the birth of the child. Table 13.4 (data set: `bttstudy`) supplies these data according to the mothers' race.

Table 13.3 The number of dead embryos D and total number of embryos T in mice at various doses of boric acid (as percentage of feed) (Problem 13.3)

Dose 0.0				Dose 0.1				Dose 0.2				Dose 0.4			
D	T	D	T	D	T	D	T	D	T	D	T	D	T	D	T
0	15	0	8	0	6	0	13	1	12	0	10	12	12	3	12
0	3	0	13	1	14	0	10	0	12	0	9	1	12	2	21
1	9	2	14	1	12	1	12	0	11	1	12	0	13	3	10
1	12	3	14	0	10	0	11	0	13	0	13	2	8	3	11
1	13	0	11	2	14	2	10	0	12	1	14	2	12	1	11
2	13	2	12	0	12	2	12	0	14	0	13	4	13	1	11
0	16	0	15	0	14	2	15	4	15	0	14	0	13	8	14
0	11	0	15	3	14	3	12	0	14	1	13	1	13	0	15
1	11	2	14	0	10	1	12	0	12	2	12	0	12	2	13
2	8	1	11	2	12	0	12	1	6	1	14	1	9	8	11
0	14	1	16	3	13	1	12	2	13	0	13	3	9	4	12
0	13	0	12	1	11	1	13	0	10	0	12	0	11	2	12
3	14	0	14	1	11	1	15	1	14	1	7	1	14		
1	13			0	11			1	12			0	10		

Table 13.4 Number of subjects whose mothers had medical aid by the race of the participants (Problem 13.4)

	White		Black	
	Group 1	Group 2	Group 1	Group 2
Had medical aid	104	10	91	36
Had no medical aid	22	2	957	368
Total	126	12	1048	404

1. Compute the percentage of mothers in each group with medical aid. Which group has a higher uptake of medical aid? (That is, produce a two-way table of Group against whether or not the mother had medical aid, combing both race categories.)
2. Compute the percentages of mothers in each group with and without medical aid according to race. Which group has a higher uptake of medical aid within each race? Contrast this with your answer above.
3. Explain the above paradox by fitting and interpreting the appropriate GLM for the data.

13.5. In Example 4.4, data were given regarding the time to service soft drink vending machine routes [12]. The main interest was in predicting the amount of time y required by the route driver to service the vending machines in an outlet. This service activity includes stocking the machine with beverage products and minor maintenance or housekeeping. In that example, the two most important variables were identified as the number of cases of product stocked x_1 and the distance walked by the route driver x_2 (Table 4.2; data set: `sdrink`).

Table 13.5 Canadian insurance data (Problem 13.6)

Merit	Class	Insured	Premium	Claims	Cost
3	1	2,757,520	159,108	217,151	63,191
3	2	130,535	7175	14,506	4598
3	3	247,424	15,663	31,964	9589
3	4	156,871	7694	22,884	7964
3	5	64,130	3241	6560	1752
2	1	130,706	7910	13,792	4055
2	2	7233	431	1001	380
2	3	15,868	1080	2695	701
2	4	17,707	888	3054	983
2	5	4039	209	487	114
1	1	163,544	9862	19,346	5552
1	2	9726	572	1430	439
1	3	20,369	1382	3546	1011
1	4	21,089	1052	3618	1281
1	5	4869	250	613	178
0	1	273,944	17,226	37,730	11,809
0	2	21,504	1207	3421	1088
0	3	37,666	2502	7565	2383
0	4	56,730	2756	11,345	3971
0	5	8601	461	1291	382

The dependence of time on the two covariates is likely to be directly linear, as seen in Fig. 4.1, because time should increase linearly with the number of cases or the distance walked. Fit a suitable GLM for modelling the delivery times.

13.6. A summary of the Canadian automobile insurance industry [1] for policy years 1956 and 1957 (as of June 30, 1959) are given in Table 13.5 (data set: `cins`). Virtually every insurance company operating in Canada is represented. The data are for private passenger automobile liability for non-farmers for all of Canada apart from Saskatchewan.

The factor `Merit` measures the number of years since the last policy claim (see `?cins` for the details). `Class` is a factor based on gender, age, use and marital status (see `?cins` for the details). `Insured` and `Premium` are two measures of the risk exposure of the insurance companies. `Insured` is measured in earned car-years; that is, a car insured for 6 months is 0.5 car-years. `Premium` is in thousands of dollars adjusted to the premium of cars written off at 2001 rates. The data also give the number of `Claims` and the total `Cost` of the claims in thousands of dollars.

1. Fit a GLM to model the number of claims.
2. Fit a GLM to model the cost per claim.
3. Fit a GLM to model the total cost.

In your models, you will need to consider using an offset.

Table 13.6 The number of revertant colonies for various doses of quinoline (in µg per plate) (Problem 13.7)

Dose	Colonies	Dose	Colonies	Dose	Colonies
0	15	33	16	333	33
0	21	33	26	333	38
0	29	33	33	333	41
10	16	100	27	1000	20
10	18	100	41	1000	27
10	21	100	60	1000	42

13.7. A study [2] used an Ames mutagenicity assay to count the number of revertant colonies (colonies that revert back to the original genotype) of TA98 *Salmonella* in rat livers (Table 13.6; data set: `mutagen`). Theory [2] suggests a good approximate model for the data is $\log(\mu) = \alpha + \beta \log(d + c) - d\gamma$ for dose d, where $\mu = \mathrm{E}[\mathrm{Counts}]$, $\gamma \geq 0$, and $c = 10$ in this case.

1. Plot the data, using logarithm of dose on the horizontal axis.
2. Fit the suggested model to the data, and summarize. Plot this model with the data.
3. Show that there is evidence of overdispersion.
4. Fit a negative binomial model (with the same systematic component) to the data, and summarize.
5. Compare the two models graphically, including confidence intervals for the fitted values.

13.8. To study the effect of trout eggs and the toxin potassium cyanate (KSCN) [9, 14], the toxin was applied at six different concentrations to vials of fish eggs. Each vial contained between 61 and 179 eggs. The eggs in half of the vials were allowed to water-harden for several hours after fertilization before the KSCN was applied. For the other vials, the toxin was applied immediately after fertilization. The number of eggs in the vial after 19 days was recorded (Table 13.7; data set: `trout`). Interest is in the effect of KSCN concentration on trout egg mortality.

Find an appropriate model for the proportion of eggs that do not survive, ensuring an appropriate diagnostic analysis. Interpret the model.

13.9. In 1990, the Water Board of New South Wales, Australia, gathered self-reported data from swimmers (Table 13.8; data set: `earinf`) about the number of self-diagnosed ear infections after swimming [9, 18] to determine if beach swimmers were more or less likely to report ear infections than non-beach swimmers. Swimmers reported their age group (`Age`, with levels `15-19`, `20-24` or `25-29`), sex (`Sex` with levels `Male` or `Female`), and the number of self-diagnosed ear infections (`NumInfec`), where they usually swam (`Loc`, with levels `Beach` or `NonBeach`), and whether they were a frequent ocean swimmer (`Swim`, with levels `Freq` (frequent) or `Occas` (occasional)).

Table 13.7 The effect on potassium cyanate concentration (in mg/L) on the mortality of trout eggs (Problem 13.8)

	Water hardening		No water hardening			Water hardening		No water hardening	
Conc.	Number	Dead	Number	Dead	Conc.	Number	Dead	Number	Dead
90	111	8	130	7	720	83	2	99	29
	97	10	179	25		87	3	109	53
	108	10	126	5		118	16	99	40
	122	9	129	3		100	9	70	0
180	68	4	114	12	1440	140	60	100	14
	109	6	149	4		114	47	127	10
	109	11	121	4		103	49	132	8
	118	6	105	0		110	20	113	3
360	98	6	102	4	2880	143	79	145	113
	110	5	145	21		131	85	103	84
	129	9	61	1		111	78	143	105
	103	17	118	3		111	74	102	78

Table 13.8 The number of self-reported ear infections from swimmers (Problem 13.9)

Males				Females			
Frequency of ocean swimming	Usual swimming location	Age group	Number of infections	Frequency of ocean swimming	Usual swimming location	Age group	Number of infections
Occasional	Non-beach	15–19	0	Occasional	Non-beach	15–19	0
Occasional	Non-beach	15–19	0	Occasional	Non-beach	15–19	0
Occasional	Non-beach	15–19	0	Occasional	Non-beach	15–19	4
Occasional	Non-beach	15–19	0	Occasional	Non-beach	15–19	10
Occasional	Non-beach	15–19	0	Occasional	Non-beach	20–24	0
Occasional	Non-beach	15–19	0	Occasional	Non-beach	20–24	0
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
Frequent	Beach	25–29	2	Frequent	Beach	25–29	2
Frequent	Beach	25–29	2	Frequent	Beach	25–29	2

The purpose of the study is to understand the factors that influence the number of ear infections. Find a suitable model for the data, and interpret this model.

13.10. A study of the root system of apple trees [6, 16] used three different root stocks (`Rstock` with levels `M26`, `Mark` and `MM106`) and two different spacing (`Spacing`, with levels `4x2` and `5x3`) for eight apple trees (`Plant`). Soil core samples were analysed, classified as coming from the inner or outer zone (`Zone`, with levels `Inner` and `Outer` respectively) relative to each plant (Table 13.9; data set: `fineroot`). The response variable is the density of fine roots (the root length density, `RLD`, in cm/cm^3); 38% of the `RLD` values are zero.

Table 13.9 The root length density (RLD) of apple trees, rounded to two decimals places (Problem 13.10)

M26				Mark				MM106			
Plant	Spacing	Zone	RLD	Plant	Spacing	Zone	RLD	Plant	Spacing	Zone	RLD
7	4×2	Outer	0	1	5×3	Inner	0	5	5×3	Outer	0
7	4×2	Inner	0	1	5×3	Outer	0	5	5×3	Outer	0
7	4×2	Outer	0	1	5×3	Inner	0	5	5×3	Outer	0
7	4×2	Inner	0	1	5×3	Outer	0	5	5×3	Outer	0
7	4×2	Outer	0	1	5×3	Inner	0	5	5×3	Inner	0
7	4×2	Inner	0	1	5×3	Outer	0	5	5×3	Outer	0
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
8	4×2	Outer	0.42	4	4×2	Inner	0.30	6	5×3	Outer	0.48
8	4×2	Inner	0.54	4	4×2	Inner	0.36	6	5×3	Outer	0.60

The design is not a full factorial design: not all plants are used with each root stock and spacing. The `Mark` rootstock is used with both plant spacings, but the other rootstocks are used at only one spacing each (`M26` at `4x2`, and `MM106` at `5x3`).

1. Plot the data and describe the potential relationships.
2. `Zone` is the only variable varying within `Plant`, so initially fit the model with `Plant` and `Zone`, and possibly the interaction. Find an estimate of ξ, then fit the corresponding Tweedie GLM.
3. Show that the model predicts the probability of zero RLD well, but slightly underestimates the probability for small values
4. Between plants, `Rstock` and `Spacing` vary. First, consider a Tweedie GLM with only `Rstock` and `Zone` together in the model (using the previously estimated value of ξ). Then add `Spacing`, `Plant` and their interaction, plus the `Plant:Zone` interaction to the model, and show only `Rstock` and `Zone` and the interaction are necessary in the model.
5. Deduce a possible model for the data, ensuring a diagnostic analysis.
6. For the final model, examine the mean RLD for each rootstock–zone combination, and interpret.

13.11. A study of the time it takes mammals of various masses to urinate [21] found that

> mammals above 3 kg in weight empty their bladders over nearly constant duration (p. 11,932).

In other words, the mass of the mammal is not related to urination time. The theory presented in the paper suggests that the authors were expecting a relationship between duration D of urination and the mass M of the form $D = kM^{1/3}$ for some proportionality constant k (data set: `urinationD`).

1. By using a transformation, fit an appropriate weighted linear regression model to all the data, and estimate the relationship between D and M.

Table 13.10 The number of live births and number of Downs Syndrome births for mothers in various age groups in British Columbia from 1961–1970 (Problem 13.12)

Mean age	Live births	Downs Synd. cases	Mean age	Live births	Downs Synd. cases	Mean age	Live births	Downs Synd. cases
17.0	13,555	16	27.5	19,202	27	37.5	5780	17
18.5	13,675	15	28.5	17,450	14	38.5	4834	15
19.0	18,752	16	29.5	15,685	9	39.5	3961	30
20.5	22,005	22	30.5	13,954	12	40.5	2952	31
21.5	23,896	16	31.5	11,987	12	41.5	2276	33
22.5	24,667	12	32.5	10,983	18	42.4	1589	20
23.5	24,807	17	33.5	9825	13	43.5	1018	16
24.5	23,986	22	34.5	8483	11	44.5	596	22
25.5	22,860	15	35.5	7448	23	45.5	327	11
26.5	21,450	14	36.5	6628	13	47.0	249	7

2. The paper suggests that no relationship exists between D and M for mammals heavier than 3 kg. Determine if those observation appear as influential in the fitted model above.
3. Fit the same model as above, but to mammals heavier than 3 kg only, as suggested by the quotation above. Are the paper's conclusions supported?

13.12. The number of Downs Syndrome births in British Columbia, Canada, from 1961–1970 is tabulated in Table 13.10 (data set: `downs`) [4, 8]. Fit an appropriate GLM to model the *number* of Downs Syndrome cases, and plot the systematic component on the plot of the data. Then, fit an appropriate GLM to model the *proportion* of Downs Syndrome cases as a function of age. Comment on the similarities and differences between the two models.

13.13. Blood haematology in athletes is of interest and importance at the elite level. To this end, the Australian Institute of Sport (AIS) gathered haematological information from 202 elite athletes across various sports [20] (data set: `AIS`). The aim of the study was stated as follows:

> The main aim of the statistical analysis was to determine whether there were any hematological differences, on average, between athletes from the various sports, between the sexes, and whether there was an effect of mass or height (p. 789).

Use the data to provide information for answering this question, focussing on haemoglobin concentration.

13.14. A study [11] exposed 96 rainbow trout to various concentrations of 3, 4-dichloroaniline (DCA). After 28 days, the weights of the trout were recorded (Table 13.11; data set: `rtrout`). The aim of the study was to "determine the concentration level which causes 25% inhibition [i.e. weight loss] from the control" [3, p. 161]. One analysis of the data [3] used a gamma GLM with a quadratic systematic component.

Table 13.11 The weight of rainbow trout (in grams) at various doses of DCA (in µg per litre) (Problem 13.14)

Dose of DCA (in µg per litre)					
Control	19	39	71	120	210
12.7	9.4	12.7	11.9	7.7	8.8
13.3	13.9	9.2	13.2	6.4	8.7
16.3	16.4	10.4	10.5	9.8	8.6
13.8	11.8	15.3	9.5	8.8	7.3
8.7	15.0	13.3	12.5	9.9	8.6
13.6	14.3	11.1	10.4	11.1	11.4
10.6	11.0	9.4	13.1	12.1	9.9
13.8	15.0	8.2	8.4	10.5	7.3
12.5	12.2	13.2	10.6	9.0	10.6
14.7	13.3	12.1	11.3	13.7	8.4
10.9	12.3	7.9	9.6	8.4	7.4
8.9	7.0	15.3	9.1	7.6	8.3
12.7	11.3	9.6	10.6	11.0	8.5
13.0	11.8		15.5	7.4	7.8
9.1	14.6	15.3	9.6	9.7	10.1
13.7	12.4	8.2	10.3	9.5	8.2

Fit and evaluate the fitted model, suggesting another model if appropriate. Then, using this model, estimate the dose as described in the aim.

13.15. Consider the Galápagos Islands species data (Table 13.12; data set: `galapagos`) [10]. Find factors that seem to influence (a) the number of endemic species, and (b) the proportion of the species on each island which are endemic. Summarize your results. Here are some hints:

- The number of species, and the proportion of endemics, are obviously non-normal variables. You will need to choose appropriate response distributions for them.
- All of the explanatory variables are highly skew, and no regression method could be expected to be successful without transforming them. Whenever an explanatory variable is strictly positive and varies by a factor of 10 or more, it is a good idea to pre-emptively apply a logarithmic transformation before undertaking any analysis. Even if the logarithmic transformation doesn't eventually turn out to the best transformation, it will be a big step in the right direction. For a variable like `StCruz` which contains an exact zero, you could use `log(StCruz+0.1)`, where 0.1 is the smallest unit in which the distances are recorded.

Table 13.12 The Galápagos Islands species data. See the help file (`?galapagos`) for information on the variables (Problem 13.15)

Island	Plants	PlantEnd	Finches	FinchEnd	FinchGenera	Area	Elevation	Nearest	StCruz	Adjacent
Baltra	58	23			4	25.09	100	0.6	0.6	1.84
Bartolome	31	21				1.24	109	0.6	26.3	572.33
Caldwell	3	3				0.21	114	2.8	58.7	0.78
Champion	25	9				0.10	46	1.9	47.4	0.18
Coamano	2	1				0.05	25	1.9	1.9	903.82
Daphne Major	18	11				0.34	50	8.0	8.0	1.84
Darwin	10	7	4	2	2	2.33	168	34.1	290.2	2.85
Eden	8	4				0.03	50	0.4	0.4	17.95
Enderby	2	2				0.18	112	2.6	50.2	0.10
Espanola	97	26	3	2	2	58.27	198	1.1	88.3	0.57
Fernandina	93	35	9	0	5	634.49	1494	4.3	95.3	4669.32
Gardner (near Española)	58	17				0.57	49	1.1	93.1	58.27
Gardner (near Santa Maria)	5	4				0.78	227	4.6	62.2	0.21
Genovesa	40	19	4	3	2	17.35	76	47.4	92.2	129.49
Isabela	347	89	10	1	5	4669.32	1707	0.7	28.1	634.49
Marchena	51	23	7	1	4	129.49	343	29.1	85.9	59.56
Onslow	2	2				0.01	25	3.3	45.9	0.10
Pinta	104	37	9	2	4	59.56	777	29.1	119.6	129.49
Pinzon	108	33	9	0	5	17.95	458	10.7	10.7	0.03
Las Plazas	12	9				0.23	50	0.5	0.6	25.09
Rabida	70	30	9	0	5	4.89	367	4.4	24.4	572.33
San Cristobal	280	65	7	3	5	551.62	716	45.2	66.5	0.57
San Salvador	237	81	10	0	5	572.33	906	0.2	19.8	4.89
Santa Cruz	444	95	10	0	5	903.82	864	0.6	0.0	0.52
Santa Fe	62	28	7	1	3	24.08	259	16.5	16.5	0.52
Santa Maria	285	73	9	2	4	170.92	640	2.6	49.2	0.10
Seymour	44	16				1.84	50	0.6	9.6	25.09
Tortuga	16	8				1.24	186	6.8	50.9	17.95
Wolf	21	12	5	1	2	2.85	253	34.1	254.7	2.33

References

[1] Bailey, R.A., Simon, L.J.: Two studies in automobile insurance ratemaking. ASTIN Bulletin **I**(IV), 192–217 (1960)

[2] Breslow, N.E.: Extra-Poisson variation in log-linear models. Applied Statistics **33**(1), 38–44 (1984)

[3] Crossland, N.O.: A method to evaluate effects of toxic chemicals on fish growth. Chemosphere **14**(11–12), 1855–1870 (1985)

[4] Davison, A.C., Hinkley, D.V.: Bootstrap Methods and their Application. Cambridge University Press (1997)

[5] Duncan, P.D., Ritter, P.L., Dornbusch, S.M., Gross, R.T., Carlsmith, J.M.: The effects of pubertal timing on body image, school behavior, and deviance. Journal of Youth and Adolescence **14**(3), 227–235 (1985)
[6] Dunn, P.K., Smyth, G.K.: Series evaluation of Tweedie exponential dispersion models. Statistics and Computing **15**(4), 267–280 (2005)
[7] Efron, B.: Double exponential families and their use in generalized linear regression. Journal of the American Statistical Association **81**(395), 709–721 (1986)
[8] Geyer, C.J.: Constrained maximum likelihood exemplified by isotonic convex logistic regression. Journal of the American Statistical Association **86**(415), 717–724 (1991)
[9] Hand, D.J., Daly, F., Lunn, A.D., McConway, K.Y., Ostrowski, E.: A Handbook of Small Data Sets. Chapman and Hall, London (1996)
[10] Johnson, M.P., Raven, P.H.: Species number and endemism: The Galápagos Archipelago revisited. Science **179**(4076), 893–895 (1973)
[11] Maul, A.: Application of generalized linear models to the analysis of toxicity test data. Environmental Monitoring and Assessment **23**(1), 153–163 (1992)
[12] Montgomery, D.C., Peck, E.A.: Introduction to Regression Analysis. Wiley, New York (1992)
[13] Morrell, C.H.: Simpson's paradox: An example from a longitudinal study in South Africa. Journal of Statistics Education **7**(3) (1999)
[14] O'Hara Hines, R.J., Carter, M.: Improved added variable and partial residual plots for the detection of influential observations in generalized linear models. Applied Statistics **42**(1), 3–20 (1993)
[15] Price, J.J., Field, C.J., Field, E.A., Marr, M.C., Myers, C.B., Morrisse, R.E., Schwetz, B.A.: Developmental toxicity of boric acid in mice and rats. Fundamental and Applied Toxicology **18**, 266–277 (1992)
[16] de Silva, H.N., Hall, A.J., Tustin, D.S., Gandar, P.W.: Analysis of distribution of root length density of apple trees on different dwarfing rootstocks. Annals of Botany **83**, 335–345 (1999)
[17] Slaton, T.L., Piergorsch, W.W., Durham, S.D.: Estimation and testing with overdispersed proportions using the beta-logistic regression model of Heckman and Willis. Biometrics **56**(1), 125–133 (2000)
[18] Smyth, G.K.: Australasian data and story library (OzDASL) (2011). URL http://www.statsci.org/data
[19] Publius Syrus: The Moral Sayings of Publius Syrus, a Roman Slave: from the Latin. L. E. Bernard & co (Translated by Darius Lyman) (1856)
[20] Telford, R.D., Cunningham, R.B.: Sex, sport, and body-size dependency of hematology in highly trained athletes. Medicine and Science in Sports and Exercise **23**(7), 788–794 (1991)
[21] Yang, P.J., Pham, J., Choo, J., Hu, D.L.: Duration of urination does not change with body size. Proceedings of the National Academy of Sciences **111**(33), 11 932–11 937 (2014)

Appendix A
Using R for Data Analysis

The data analyst knows more than the computer.
Henderson and Velleman [7, p. 391]

A.1 Introduction and Overview

This chapter introduces the R software package. We start by discussing how to obtain and install R and the R packages needed for this book (Sect. A.2). We then introduce the basic use of R, including working with vectors, loading data, and writing functions in R (Sect. A.3).

A.2 Preparing to Use R

A.2.1 Introduction to R

R is a powerful and convenient tool for fitting the models presented in this book. Rather than a menu-driven statistical package, R is a powerful *environment* for statistically and graphically analyzing data. R is free to install and use.

While R itself is not a menu-driven package, some graphical front-ends are available, such as R Commander [4, 5, 6] (http://www.rcommander.com/). RStudio (https://www.rstudio.com/products/RStudio/) provides an environment for working with R which includes an integrated console, coding, graphics and help windows. R Commander is free, and free versions of RStudio also exist.

The use of R is explained progressively throughout this book for use with linear regression models and GLMs. In this appendix, some basics of using R are described. A more comprehensive treatment of using R can be found in the following books, among others:

P. K. Dunn, G. K. Smyth, *Generalized Linear Models with Examples in R*,
Springer Texts in Statistics, https://doi.org/10.1007/978-1-4419-0118-7

- Dalgaard [1] is a gentle introduction to using R for basic statistics.
- Maindonald and Braun [8] introduces R and covers a variety of statistical techniques.
- Venables and Ripley [13] is an authoritative book discussing the implementation of a variety of statistical techniques in R and the closely-related commercial program S-PLUS.

A.2.2 Important R Websites

Two websites are particularly important for R users:

- The R Project for Statistical Computing (http://www.r-project.org/) is the R homepage. This web site contains documentation, general information, links to searchable R mailing list archives, and much more.
- The Comprehensive R Archive Network, known as CRAN, contains the files necessary for downloading R and add-on packages. A link to CRAN is given from the R homepage: go to the R homepage, and select CRAN from the menu. Clicking this link forces the user to select a mirror site. (Selecting a mirror site near to you may make for faster downloads.) Clicking on an appropriate mirror site then directs the browser to CRAN, where R can be downloaded.

Another useful webpage is rseek.org, which provides a search facility dedicated to R.

A.2.3 Obtaining and Installing R

R can be downloaded from CRAN (follow the instructions in Sect. A.2.2 to locate CRAN). The procedure for then installing R depends on your operating system (Windows; Mac OS X; linux; etc.). The easiest approach for most users is to go to CRAN, then click on 'Download and Install R', then download the pre-compiled binaries for your operating system. Then install these pre-compiled binaries in the usual way for your operating system.

CRAN maintains current documentation for installing R. Click on the 'Manuals' link on the left (on either the CRAN website or the R homepage), and read the manual *R Installation and Administration*. (Another manual, the document *An Introduction to R*, may also prove useful for learning to use R.)

A.2.4 Downloading and Installing R Packages

Packages are collections of R functions that add extra functionality to R. Some packages come with R, but other packages must be separately downloaded and installed before use. An important package used in this book is the **GLMsData** package [3], which contains the data sets used in this book. Using the R code in this book requires the **GLMsData** package to be

downloaded and installed, so we demonstrate the process of downloading and installation of R packages using the **GLMsData** packages. More information about the **GLMsData** package appears in Appendix B (p. 525).

For Windows and Mac OS X users, packages can be installed by starting R and using the menu system:

Windows: Click Packages, then Install package(s). Select a CRAN mirror, then select the package you wish to install, and then press OK.

Mac OS X: Click Packages & Data, and select CRAN (binaries) from the drop-down menu. Clicking Get List creates a list of the packages that can be installed from CRAN; make your selection, then press Update All.

Users of RStudio can install packages through the RStudio menus (under Tools).

Alternatively, packages can be downloaded directly from CRAN; Sect. A.2.2 contain instructions to locate your nearest CRAN mirror. From the CRAN homepage, select 'Packages', then locate and click on the name of the package you wish to install. Here, we use the package **GLMsData** to demonstrate, but the instructions are the same for downloading any R package. After clicking on the package name in the CRAN list, click on the file to download for your operating system (for example, Windows users click on the file next to 'Windows binary'). The file will be then downloaded. To then install:

- Windows: Choose Packages from the Menu, then Install package(s) from local zip files.... Locate the package to install.
- Mac OS X: Click Packages & Data, select Local Binary Package, then press Install.... Locate the package to install.
- Linux: Open a terminal and type `sudo R CMD INSTALL GLMsData`, for example, in the directory where the package was downloaded, assuming the appropriate permissions exist.

Packages can also be installed using `install.packages()` from the R command line; for example, `install.packages("GLMsData")`. Reading the document *R Installation and Administration*, available at http://cran.r-project.org/doc/manuals/R-admin.pdf, may prove useful.

A.2.5 Using R Packages

Any package, whether downloaded and installed or a package that comes with R, must be *loaded* before being used in any R session:

- **Loading:** To load an installed package and so make the extra functionality *available* to R, type (for example) `library(GLMsData)` (or `library("GLMsData")`) at the R prompt.
- **Using:** After loading the package, the functions in the package can be used like any other function or data set in R.
- **Obtaining help:** To obtain help about the **GLMsData** package, even if the package is not loaded (but is installed), type `library(help=`

`GLMsData)` (or `library(help="GLMsData")`) at the R prompt. To obtain help about particular function or data set in the package, type (for example) `?lungcap` at the R prompt after the package is loaded.

A.2.6 The R Packages Used in This Book

We have purposely kept the number of packages needed for this book to a minimum. These packages are used in this book:

GLMsData: The **GLMsData** package [3] is essential for running the R code in this book, as it provides most of the necessary data.

MASS: The **MASS** package [13] supplies the `boxcox()` function (Sect. 3.9), the `dose.p()` function and functions used for fitting negative binomial GLMs (Sect. 10.5.2). **MASS** comes with all R distributions, and does not need to be downloaded and installed as described above.

splines: The **splines** package [10] is used to fit regression splines (Sect. 3.12). **splines** comes with all R distributions, and does not need to be downloaded and installed as described above.

statmod: The **statmod** package [12] provides the `tweedie()` family function used to fit Tweedie GLMs (Chap. 12), for computing quantile residuals (Sect. 8.3.4), and for evaluating the probability function for the inverse Gaussian distribution. **statmod** does *not* come with R distributions, and must be downloaded and installed as described above.

tweedie: The **tweedie** package [2] provides functions for estimating the Tweedie index parameter ξ for fitting Tweedie GLMs, is used by `qresid()` to compute quantile residuals for Tweedie GLMs, and is used for other computations related to Tweedie GLMs (Chap. 12, p. 457). **tweedie** does *not* come with R distributions, and must be downloaded and installed as described above.

The packages are loaded for use (after being downloaded and installed if necessary) by typing `library(statmod)` (for example) at the R prompt.

A.3 Introduction to Using R

A.3.1 Basic Use of R as an Advanced Calculator

After starting R, a command line is presented indicating that R is waiting for the user to enter commands. This command line usually looks like this:

```
>
```

Instruct R to perform basic arithmetic by issuing commands at the command line, and pressing the ENTER or RETURN key. After starting R, enter this command, and then press ENTER (do not type the `>` as this is the R prompt):

```
> 2 - 9 * (1 - 3)
```

Note that * indicates multiplication. R responds with the answer:

```
[1] 20
>
```

After giving the answer, R then awaits your next instruction. Note that the answer here is preceded by [1], which indicates the first item of output, and is of little use here where the output consists of one number. Sometimes R produces many numbers as output, when the [1] proves useful, as seen later (Sect. A.3.5). Other examples:

```
> 2 * pi                  # pi is 3.1415...
[1] 6.283185
> -8 + ( 2^3 )            # 2^3 means 2 raised to the power 3
[1] 0
> 10/4000000              # 10 divided by a big number
[1] 2.5e-06
> 1 + 2 * 3               # Note the order of operations
[1] 7
```

Note the use of #: the # character is a comment character, so that # and all text after it is ignored by R. (You don't need to type the # or the text that follows.) The output from the final expression 2.5e-06 is R's way of displaying 2.5×10^{-6}. Very large or very small numbers can be entered using this notation also:

```
> 6.02e23          # Avogadro constant
[1] 6.02e+23
```

Standard mathematical functions are also defined in R:

```
> exp( 1 )     # exp(x) means e raised to the power x where e = 2.71828...
[1] 2.718282
> log( 10 )    # Notice that  log  is the natural log
[1] 2.302585
> log10( 10 )  # This is log to base 10
[1] 1
> log2(32)     # This is log to base 2
[1] 5
> sin( pi )    # The result is zero to computer precision
[1] 1.224647e-16
> sqrt( 45 )   # The square root
[1] 6.708204
```

Issuing incomplete R commands forces R to wait for the command to be completed. Suppose you wish to evaluate 2 * pi * 7.4, but enter this incomplete command:

```
> 2 * pi *
```

R will continue to wait for you to complete the command. The prompt changes from > to + to indicate R is waiting for further input. Continue by entering 7.4 and pressing RETURN. The complete interaction looks like this:

```
> 2 * pi *
+ 7.4          # DO NOT type the "+" sign: R is asking for more info
[1] 46.49557
```

Note that `2 * pi` is a complete command, so if `2 * pi` is issued at the R prompt, R provides the answer and does not expect any further input.

A.3.2 Quitting R

To finish using R, enter the command `q()` at the command prompt:

```
> q()          # This will close R
```

The empty parentheses are necessary. R asks if you wish to Save workspace image? If you respond with Yes, then R will save your work, so that next time R is started you can continue your previous R session. If you respond with No, R starts a fresh session the next time R is started.

A.3.3 Obtaining Help in R

The following commands can be used to obtain help in R:

- `help.search("glm")`: search the R help system for the text `glm`.
- `?glm`: obtain help for the function `glm()`; equivalent to `help("glm")`.
- `help.start()`: opens R's on-line documentation in a browser.
- `RSiteSearch("generalized linear model")`, if you are connected to the Internet: Search wider R resources, such as R-help mailing list archives, R manuals and R help pages, and displays the results in a browser window.
- `example("glm")`: show an example of using `glm()`.

A.3.4 Variable Names in R

Importantly, answers computed by R can be assigned to variables using the two-character combination <- as shown below:

```
> radius <- 0.605
> area <- pi * radius^2
> area
[1] 1.149901
```

Notice that when `<-` is used, the answer is not displayed. Typing the name of a variable shows its value. The equal sign `=` can be used in place of `<-` to make assignments, though `<-` is traditional:

```
> radius = 0.605
```

Spacing in the input is not important to R. All these commands mean the same to R, but the first is easiest to read and is recommended:

```
> area <- pi * radius^2
> area         <-        pi    *radius^      2
> area<-pi*radius^2
```

Variable names can consist of letters, digits, the underscore character, and the dot (period). Variable names cannot start with digits; names starting with dots should be avoided. Variable names are also case sensitive: `HT`, `Ht` and `ht` are different variables. Many possible variables names are already in use by R, such as `log` as used above. Problems may result if these are used as variable names. Common variables names to avoid include `t` (for transposing matrices), `c` (used for combining objects), `q` (for quitting R), `T` (is a logical true), `F` (is a logical false), and `data` (makes data sets available to R).

These are all valid variables names: `plant.height`, `dose2`, `Initial_Dose`, `PatientAge`, and `circuit.2.AM`. In contrast, these are *not* valid variables names: `Before-After` (the `-` is illegal), and `2ndTrial` (starts with a digit).

A.3.5 Working with Vectors in R

R works especially well with a group of numbers, called a *vector*. Vectors are created by grouping items together using the function `c()` (for 'combine' or 'concatenate'):

```
> x <- c(1, 2, 3, 4, 5, 6, 7, 8, 9, 10)
> log(x)
 [1] 0.0000000 0.6931472 1.0986123 1.3862944 1.6094379 1.7917595 1.9459101
 [8] 2.0794415 2.1972246 2.3025851
```

Notice that when the output is long, R identifies the element of each list in the left column, starting with `[1]`. Element 8 (which is `2.0794415`) starts the second line of output.

A long sequence of equally-spaced values is often useful, especially in plotting. Rather than the cumbersome approach adopted above, consider these simpler approaches:

```
> seq(0, 10, by=1)           # The values are separated by distance 1
 [1]  0  1  2  3  4  5  6  7  8  9 10
> 0:10                       # Same as above
 [1]  0  1  2  3  4  5  6  7  8  9 10
> seq(0, 10, length=9)       # The result has length 9
[1]  0.00  1.25  2.50  3.75  5.00  6.25  7.50  8.75 10.00
```

Variables do not have to be numerical to be grouped together; text and logical variables can be used also:

```
> day <- c("Sun", "Mon", "Tues", "Wed", "Thurs", "Fri", "Sat")
> hours.work  <- c(0,  8, 11.5, 9.5, 8, 8, 3)
> hours.sleep <- c(8,  8,    9, 8.5, 6, 7, 8)
> do.exercise <- c(TRUE, TRUE, TRUE, FALSE, TRUE, FALSE, TRUE)
> hours.play <- 24 - hours.work - hours.sleep
> hours.awake <- hours.work + hours.play
```

Single or double quotes are possible for defining text variables, though double quotes are preferred (which enables constructs like `"O'Neil"` and `"Don't know"`).

Specific elements of a vector are identified using square brackets:

```
> hours.play[3]; day[ 2 ]
[1] 3.5
[1] "Mon"
```

As shown, commands can be issued together on one line if separated by a `;` (a semi-colon). To find the value of `hours.work` on Fridays, consider the following:

```
> day == "Fri"                 # A logic statement
[1] FALSE FALSE FALSE FALSE FALSE  TRUE FALSE
> hours.work[ day == "Fri" ]
[1] 8
> hours.sleep[ day == "Fri" ]
[1] 7
> do.exercise[ day == "Thurs"]
[1] TRUE
```

Notice that `==` is used for logical comparisons. Other logical comparisons are also possible:

```
> day[ hours.work > 8 ]           #  >   means "greater than"
[1] "Tues" "Wed"
> day[ hours.sleep < 8 ]          #  <   means "less than"
[1] "Thurs" "Fri"
> day[ hours.work >= 8 ]          #  >=  means "greater than or equal to"
[1] "Mon"   "Tues"  "Wed"   "Thurs" "Fri"
> day[ hours.work <= 8 ]          #  <=  means "less than or equal to"
[1] "Sun"   "Mon"   "Thurs" "Fri"   "Sat"
> day[ hours.work != 8 ]          #  !=  means "not equal to"
[1] "Sun"  "Tues" "Wed"  "Sat"
> day[ do.exercise & hours.work>8 ]     #   &  means "and"
[1] "Tues"
> day[ hours.play>9 | hours.sleep>9 ]   #   |  means "or"
[1] "Sun"   "Thurs" "Sat"
```

Comparing real numbers using == should be avoided, because of the way computers store floating-point numbers. (This is true for all computer languages.) Instead, use `all.equal()`:

```
> expr1 <- 0.5 - 0.3   # These two expressions should be the same
> expr2 <- 0.3 - 0.1
> c(expr1, expr2)      # They *look* the same, but...
[1] 0.2 0.2
> expr1 == expr2       # ...Not exactly the same in computer arithmetic
[1] FALSE
> all.equal(expr1, expr2)  # ...so use  all.equal()
[1] TRUE
```

A.3.6 Loading Data into R

In statistics, data are usually stored in computer files, which must be loaded into R. R requires data files to be arranged with variables in columns, and cases in rows. Columns may have headers containing variable names; rows may have headers containing case labels.

In R, data are usually treated as a *data frame*, a set of variables (numeric, text, logical, or other types) grouped together. For the data entered in Sect. A.3.5, a single data frame named `my.week` could be constructed:

```
> my.week <- data.frame(day, hours.work, hours.sleep,
                        do.exercise, hours.play, hours.awake)
> my.week
    day hours.work hours.sleep do.exercise hours.play hours.awake
1   Sun        0.0         8.0        TRUE       16.0        16.0
2   Mon        8.0         8.0        TRUE        8.0        16.0
3  Tues       11.5         9.0        TRUE        3.5        15.0
4   Wed        9.5         8.5       FALSE        6.0        15.5
5 Thurs        8.0         6.0        TRUE       10.0        18.0
6   Fri        8.0         7.0       FALSE        9.0        17.0
7   Sat        3.0         8.0        TRUE       13.0        16.0
```

Entering data directly into R is only feasible for small amounts of data (and is demonstrated, for example, in Sect. 10.4.2). Usually, other methods are used for loading data into R:

1. If the data set comes with R, load the data using the command `data(trees)` (for example), as in Example 3.14 (p. 125). Type `data()` at the R prompt to see a list of all the data files that come with R.
2. If the data are in an installed R package (Sect. A.2.5), load the package, then use `data()` to load the data. For example (assuming the **GLMsData** is installed), load the package by typing `library(GLMsData)`, then load the data frame `lungcap` using `data(lungcap)` (Sect. 1.1).

3. If the data are stored as a text file (either on a storage device or on the Internet), R provides a set of functions for loading the data:

   `read.csv()`: Reads comma-separated text files. In files where the comma is a decimal point and fields are separated by a semicolon, use `read.csv2()`.
   `read.delim()`: Reads delimited text files, where fields are delimited by tabs by default. In files where the comma is a decimal point, use `read.delim2()`.
   `read.table()`: Reads files where the data in each line is separated by one or more spaces, tabs, newlines or carriage returns. `read.table()` has numerous options for reading delimited files.
   `read.fwf()`: Reads data from files where the data are in a fixed width format (that is, the data are in fields of known widths in each line of the data file).

   These functions are used by typing, for example:

   ```
   > mydata <- read.csv("filename.csv")
   ```

   Many other inputs are also available for these functions (see the relevant help files). All these functions load the data into R as a data frame. These functions can be used to load data directly from a web page (providing you are connected to the Internet) by providing the URL as the filename. For example, the data in Table 10.20 (p. 420) are also found in tab-delimited format at the OzDASL webpage [11], with variable names in the first row (called a `header`):

   ```
   > modes <- read.delim("http://www.statsci.org/data/general/twomodes.txt",
     header=TRUE)
   ```

4. For data stored in file formats from other software (such as SPSS, Stata, and so on), first load the package **foreign** [9], then see `library(help=foreign)`. Not all functions in the **foreign** package load the data as data frames by default (such as `read.spss()`).

Most data sets used in this book are available in the **GLMsData** package. Assuming the **GLMsData** package is installed, the `lungcap` data frame used in Example 1.1 (p. 1) is loaded and used as follows:

```
> library(GLMsData)  # Loads the  GLMsData  package
> data(lungcap)      # Makes the data set  lungcap  available for use
> names(lungcap)     # Shows the names of the variables in  lungcap
[1] "Age"    "FEV"    "Ht"     "Gender" "Smoke"
> head(lungcap)      # Shows the first six observations
  Age   FEV Ht Gender Smoke
1   3 1.072 46      F     0
2   4 0.839 48      F     0
3   4 1.102 48      F     0
4   4 1.389 48      F     0
5   4 1.577 49      F     0
6   4 1.418 49      F     0
```

```
> tail(lungcap)      # Shows the last six observations
    Age   FEV   Ht Gender Smoke
649  16 4.070 69.5      M     1
650  16 4.872 72.0      M     1
651  17 3.082 67.0      M     1
652  17 3.406 69.0      M     1
653  18 4.086 67.0      M     1
654  18 4.404 70.5      M     1
> str(lungcap)       # Shows the structure of the data frame
'data.frame':        654 obs. of  5 variables:
 $ Age   : int  3 4 4 4 4 4 5 5 5 ...
 $ FEV   : num  1.072 0.839 1.102 1.389 1.577 ...
 $ Ht    : num  46 48 48 48 49 49 50 46.5 49 49 ...
 $ Gender: Factor w/ 2 levels "F","M": 1 1 1 1 1 1 1 1 1 1 ...
 $ Smoke : int  0 0 0 0 0 0 0 0 0 0 ...
```

A summary of the variables in a data frame is produced using `summary()`:

```
> summary(lungcap)   # Summaries of each variable in  lungcap
      Age              FEV              Ht        Gender
 Min.   : 3.000   Min.   :0.791   Min.   :46.00   F:318
 1st Qu.: 8.000   1st Qu.:1.981   1st Qu.:57.00   M:336
 Median :10.000   Median :2.547   Median :61.50
 Mean   : 9.931   Mean   :2.637   Mean   :61.14
 3rd Qu.:12.000   3rd Qu.:3.119   3rd Qu.:65.50
 Max.   :19.000   Max.   :5.793   Max.   :74.00
     Smoke
 Min.   :0.00000
 1st Qu.:0.00000
 Median :0.00000
 Mean   :0.09939
 3rd Qu.:0.00000
 Max.   :1.00000
```

Notice that the `summary()` is different for numerical and non-numerical variables.

A.3.7 Working with Data Frames in R

Data loaded from files (using `read.csv()` and similar functions) or using the `data()` command are loaded as a *data frame*. A data frame is a set of variables (numeric, text, or other types) grouped together, as previously explained. For example, the data frame `lungcap` contains the data used in Example 1.1 (p. 1). The data frame contains the variables `FEV`, `Age`, `Height`, `Gender` and `Smoke`, as shown in Sect. A.3.6 in the output from the `names()` command.

The data frame `lungcap` is visible to R, but the individual variables within this data frame are not visible:

```
> library(GLMsData); data(lungcap)
> Age
Error: object "Age" not found
```

The objects visible to R are displayed using `objects()`:

```
> objects()
[1] "lungcap"
```

To refer to individual variables in the data frame `lungcap`, use `$` between the data frame name and the variable name, as follows:

```
> head(lungcap$Age)
[1] 3 4 4 4 4 4
```

This construct can become tedious to use all the time. An alternative is to use `with()`, by noting the data frame in which the command should executed:

```
> with( lungcap, head(Age) )
[1] 3 4 4 4 4 4
> with( lungcap, mean(Age) )
[1] 9.931193
> with( lungcap, {
    c( mean(Age), sd(Age) )
  })
[1] 9.931193 2.953935
> with( lungcap, {
    median(Age)
    IQR(Age)    # Only the last is displayed
 })
[1] 4
```

Another alternative is to *attach* the data frame so that the individual variables are visible to R (though this can have unintended side-effects and so the use of `attach()` is *not recommended*):

```
> attach(lungcap)
> head(Age)
[1] 3 4 4 4 4 4
```

When finished using the data frame, *detach* it:

```
> detach(lungcap)
```

A.3.8 Using Functions in R

Working with R requires using R functions. R contains a large number of functions, and the many additional packages add even more functions. Many R functions have been used already, such as `q()`, `read.table()`, `seq()` and

`log()`. Input arguments to R functions are enclosed in round brackets (parentheses), as previously seen. All R functions must be followed by parentheses, even if they are empty (recall the function `q()` for quitting R).

Many functions allow several input arguments. Inputs to R functions can be specified as *positional* or *named*, or even both in the one call. Positional specification means the function reads the inputs in the order in which function is defined to read them. For example, the R help for the function `log()` contains this information in the Usage section:

```
log(x, base = exp(1))
```

The help file indicates that the first argument is always the number for which the logarithm is needed, and the second (if provided) is the base for the logarithm.

Previously, `log()` was called with only one input, not two. If input arguments are not given, defaults are used when available. The above extract from the help file shows that the default `base` for the logarithm is $e \approx 2.71828\ldots$ (that is, `exp(1)`). In contrast, there is no default value for `x`. This means that if `log()` is called with only one input argument, the result is a natural logarithm (since `base=exp(1)` is used by default). To specify a logarithm to a different base, say base 2, a second input argument is needed:

```
> log(8, 2)    # Same as log2(8)
[1] 3
```

This is an example of specifying the inputs by *position*. Alternatively, all or some of the arguments can be *named*. For example, all these commands are identical, computing $\log_2 8$:

```
> log(x=8, base=2)   # All inputs are *named*
[1] 3
> log(8, 2)          # Inputs specified by position
[1] 3
> log(base=2, x=8)   # Inputs named can be given in any order
[1] 3
> log(8, base=2)     # Mixing positional and named inputs
[1] 3
```

A.3.9 Basic Statistical Functions in R

Basic statistical functions are part of R:

```
> library(GLMsData); data(lungcap)
> names(lungcap)                          # The variable names
[1] "Age"    "FEV"    "Ht"     "Gender" "Smoke"
> length( lungcap$Age )                   # The number of observations
```

```
[1] 654
> sum(lungcap$Age) / length(lungcap$Age)  # The mean, the long way
[1] 9.931193
> mean( lungcap$Age )                     # The mean, the short way
[1] 9.931193
> median( lungcap$Age )                   # The median
[1] 10
> sd( lungcap$Age )                       # The sample std deviation
[1] 2.953935
> var( lungcap$Age )                      # The sample variance
[1] 8.725733
```

A.3.10 Basic Plotting in R

R has very rich and powerful mechanisms for producing graphics. (In fact, there are different ways to produce graphics, including using the **ggplot2** package [14].) Simple plots are easily produced, but very fine control over many graphical parameters is possible. Consider a simple plot for the FEV data (Fig. A.1, left panel):

```
> data(lungcap)
> plot( lungcap$FEV ~ lungcap$Age )
```

The `~` command (`~` is called a 'tilde') can be read as 'is described by'. The variable on the left of the tilde appears on the vertical axis. Equivalent commands to the above `plot()` command (Fig. A.1, centre panel, p. 517) are:

```
> plot( FEV ~ Age, data=lungcap )
```

and

```
> with( lungcap, plot(FEV ~ Age) )
```

Notice the axes are labelled differently. As a general rule, R functions that use the formula interface (that is, constructs such as `FEV ~ Age`) allow an input called `data`, giving the data frame containing the variables.

The `plot()` command can also be used without using a formula interface:

```
> plot( lungcap$Age, lungcap$FEV )
```

This also produces Fig. A.1 (left panel). Using this approach, the variable appearing as the second input is plotted on the vertical axis.

Plots can be enhanced in many ways. Compare the result of the following code (the right panel of Fig. A.1) with the output of the previous code (the left and centre panels of Fig. A.1):

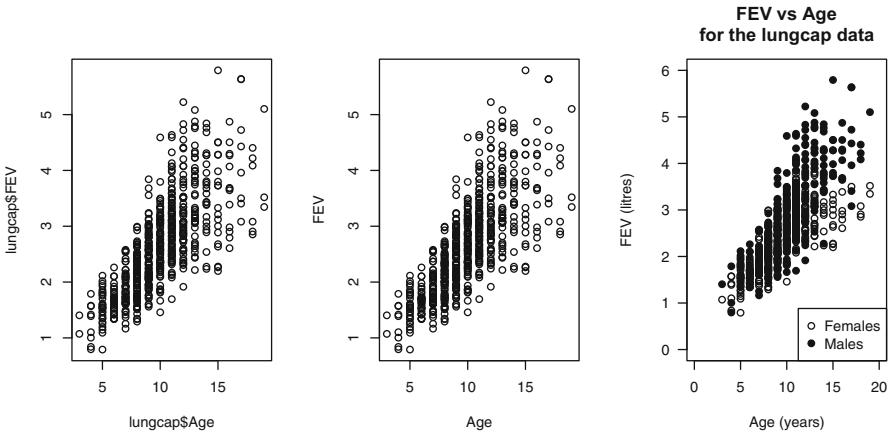

Fig. A.1 Plots of the FEV data. Left panel: a simple plot; centre panel: a simple plot produced using the `data` input; right panel: an enhanced plot using some of R's graphical parameters (Sect. A.3.10)

```
> plot( FEV ~ Age,              # Plot FEV against Age
    data=lungcap,               # The data frame to use
    las=1,                      # Ensure both axis labels are horizontal
    ylim=c(0, 6),               # Sets the limits of the vertical axis
    xlim=c(0, 20),              # Sets the limits of the horizontal axis
    xlab="Age (years)",         # The horizontal axis label
    ylab="FEV (litres)",        # The vertical axis label
    main="FEV vs Age\nfor the lungcap data",        # The main title
    pch=ifelse(Gender=="F", 1, 19) )                # (See below)
> legend("bottomright", pch=c(1, 19),               # Add legend
    legend=c("Females", "Males") )
```

Notice that the use of `\n` in the main title specifies a line break.

The construct `pch=ifelse(Gender=="F", 1, 19)` needs explanation. The input `pch` is used to select the plotting character. For example, `pch=1` plots the points with an open circle, and `pch=19` plots the points with a filled circle. The complete list of plotting characters is shown by typing `example(points)`. Further, `pch="F"` (for example) would use an `F` as the plotting character. The construct `pch=ifelse(Gender=="F", 1, 19)` is interpreted as follows:

- For each observation, determine if `Gender` has the value `"F"` (that is, if the gender is female). Note that the quotes are needed, otherwise R will look for a variable named `F`, which is the same as the logical `FALSE`. Also recall that `==` is used to make logical comparisons.
- If `Gender` does have the value `"F"`, then use `pch=1` (an open circle) to plot the observation.
- If `Gender` does not have the value `"F"` (that is, the gender is male), then use `pch=19` (a filled circle) to plot the observation.

An alternative to using `ifelse()`, which would be useful if three or more categories were to be plotted, is as follows. Begin by preparing the 'canvas' for plotting:

```
> plot( FEV ~ Age,
    type="n",       # Sets up the plot, but plots "n"othing
    data=lungcap, las=1, ylim=c(1.5, 5),
    xlab="Age (years)", ylab="FEV (litres)",
    main="FEV vs Age\nfor the lungcap data")
```

Using `type="n"` sets up the canvas for plotting, but plots nothing on the plot itself. Points are then added using `points()`:

```
> points( FEV~Age, pch=1,  subset=(Gender=="F"), data=lungcap )
> points( FEV~Age, pch=19, subset=(Gender=="M"), data=lungcap )
```

These two commands then add the points in two separate steps. The first call to `points()` plots the females only (by selecting the data subset `subset=(Gender=="F")`), using open circles (defined as `pch=1`). The second call to `points()` plots the males only (`subset=(Gender=="M")`), using filled circles (`pch=19`). Clearly, further points could be added for any number of groups using this approach. In a similar way, lines can be added to an existing plot using `lines()`.

A.3.11 Writing Functions in R

One advantage of R is that functionality is easily extended by writing new functions. Writing functions is only needed occasionally in this book.

As a simple and trivial example, consider writing a function to covert a decimal number into a percentage:

```
> as.percentage <- function(x){
   # Args:
   #  x:  The decimal value to be turned into a percentage
   # Returns:
   #   The value of x as a percentage

   x * 100
 }
```

(This R code can be typed directly into R.)

This function, called `as.percentage`, takes one input called `x`. The R instruction inside the brackets `{` and `}` shows what the function actually does. The lines beginning with the `#` are comments and can be omitted, but make the function easier to understand. This function simply multiplies the value of `x` by 100. The function `as.percentage` can be used like any other R function:

```
> item.cost <- c(110, 42, 25 )
> item.tax <-  c( 10, 4, 2.5)
> as.percentage( item.tax / item.cost )
[1]  9.090909  9.523810 10.000000
```

In R functions, the value of the last unassigned expression is the value returned by the function. Alternatively, the output can be assigned to a variable:

```
> out <- as.percentage( item.tax / item.cost ); out
[1]  9.090909  9.523810 10.000000
```

As a more advanced example, consider adapting the function `as.percentage` to return the percentage to a given number of significant figures. In a text editor (such as Notepad in Windows; TextEdit in Mac OS X; vi or Emacs in linux), enter:

```
as.percentage <- function(x, sig.figs=2){
  # Args:
  #  x:  The value to be turned into a decimal
  #  sig.figs: The number of significant figures
  # Returns:
  #   The value of x as a percentage, rounded to the requested number of
  #    significant figures and the value with a "%" sign added at the end
  percent <- signif( x * 100, sig.figs)
  percent.withsymbol <- paste( percent, "%", sep="")
  return( list(PC=percent, PC.symbol=percent.withsymbol ) )
}
```

The first line

```
  as.percentage <- function(x, sig.figs=2){
```

defines the name of the function as `as.percentage`, and declares that it needs two inputs: the first is called `x` (with no default value), and the second is called `sig.figs` (with a default value of `2`). The opening parenthesis `{` declares where the instructions begin to declare what the function does; obviously, the final closing parenthesis `}` shows where the function definition ends.

The lines that follow starting with `#` are again comments to aid readability. The next line computes the percentage rounded to the requested number of significant figures:

```
  percent <- signif( x * 100, sig.figs)
```

The next line adds the percentage symbol `%` after converting the number of a character:

```
  percent.withsymbol <- paste( percent, "%", sep="")
```

The final line is more cryptic:

```
  return( list(PC=percent, PC.symbol=percent.withsymbol ) )
```

This line determines what values the function will `return()` when finished. This `return()` command returned two values named `PC` and `PC.withsymbol`, combined together in a `list()`. When the function returns an answer, one output variable is called `PC`, which is assigned the value of `percent`, and the second output variable is called `PC.symbol`, which is assigned the value of `percent.withsymbol`. You can copy and paste the function into your R session, and use it as follows:

```
> out <- as.percentage( item.tax / item.cost )
> out
$PC
[1]  9.1  9.5 10.0

$PC.symbol
[1] "9.1%" "9.5%" "10%"
> out <- as.percentage( item.tax / item.cost, sig.figs=3 )
> out
$PC
[1]  9.09  9.52 10.00

$PC.symbol
[1] "9.09%" "9.52%" "10%"
```

Functions in R can be very long and complicated (for example, including code that detects for bad input such as trying to convert text into a percentage, or how to handle missing values). Writing functions are only required in a few cases in this book, and these functions are relatively simple. For more information on writing functions in R, see, for example, Venables and Ripley [13] or Maindonald and Braun [8].

* *A.3.12 Matrix Arithmetic in R*

R performs matrix arithmetic using some special functions. A matrix is defined using `matrix()`, where the matrix elements are given with the input `data`, the number of rows with `nrow` or columns with `ncol` (or both), and optionally whether to fill down columns (the default) or across rows (by setting `byrow=TRUE`):

```
> Amat <- matrix( c(1, 2, -3, -2), ncol=2) # Fills by columns (by default)
> Amat
     [,1] [,2]
[1,]    1   -3
[2,]    2   -2
> Bmat <- matrix( c(1, 5, -10, 15, -20, -25), nrow=2, byrow=TRUE) # By row
> Bmat
     [,1] [,2] [,3]
[1,]    1    5  -10
[2,]   15  -20  -25
```

Standard matrix operations can be performed:

```
> dim( Amat )    # The dimensions of matrix  Amat
[1] 2 2
> dim( Bmat )    # The dimensions of matrix  Bmat
[1] 2 3
> t(Bmat)        # The transpose of matrix  Bmat
     [,1] [,2]
[1,]    1   15
[2,]    5  -20
[3,]  -10  -25
> -2 * Bmat      # Multiply by scalar
     [,1] [,2] [,3]
[1,]   -2  -10   20
[2,]  -30   40   50
```

Matrix multiplication of conformable matrices requires the special function %*% to be used:

```
> Cmat <- Amat %*% Bmat; Cmat
     [,1] [,2] [,3]
[1,]  -44   65   65
[2,]  -28   50   30
```

Multiplying non-conformable matrices produces an error:

```
> Bmat %*% Amat
Error in Bmat %*% Amat : non-conformable arguments
```

Powers of matrices are produced by repeatedly using %*%:

```
> Amat^2           # Each *element* of Amat is squared
     [,1] [,2]
[1,]    1    9
[2,]    4    4
> Amat %*% Amat    # Correct way to compute  Amat  squared
     [,1] [,2]
[1,]   -5    3
[2,]   -2   -2
```

The usual multiplication operator * is for multiplication of scalars, not matrices:

```
> Amat * Bmat    # FAILS!!
Error in Amat * Bmat : non-conformable arrays
```

The * operator can also be used for multiplying the corresponding elements of matrices of the same size:

```
> Bmat * Cmat
     [,1]  [,2] [,3]
[1,]  -44   325 -650
[2,] -420 -1000 -750
```

The diagonal elements of matrices are extracted using `diag()`:

```
> diag(Cmat)
[1] -44  50
> diag(Bmat)     # diag()  even works for non-square matrices
[1]   1 -20
```

`diag()` can also be used to *create* diagonal matrices:

```
> diag( c(1, -1, 2) )
     [,1] [,2] [,3]
[1,]    1    0    0
[2,]    0   -1    0
[3,]    0    0    2
```

In addition, `diag()` can be used to create identity matrices easily:

```
> diag( 3 )       # Creates the 3x3 identity matrix
     [,1] [,2] [,3]
[1,]    1    0    0
[2,]    0    1    0
[3,]    0    0    1
```

To determine if a square matrix is singular or not, compute the determinant using `det()`:

```
> det(Amat)
[1] 4
> Dmat <- t(Bmat) %*% Bmat; Dmat
     [,1] [,2] [,3]
[1,]  226 -295 -385
[2,] -295  425  450
[3,] -385  450  725
> det(Dmat)        # Zero to computer precision
[1] -2.193801e-09
```

Zero determinants indicate singular matrices without inverses. (Near-zero determinants indicate near-singular matrices for which inverses may be difficult to compute.) The inverse of a non-singular matrix is found using `solve()`:

```
> Amat.inv <- solve(Amat); Amat.inv
     [,1] [,2]
[1,] -0.5 0.75
[2,] -0.5 0.25
> Amat.inv %*% Amat
     [,1] [,2]
[1,]    1    0
[2,]    0    1


> solve(Dmat)  # Not possible: Dmat is singular
Error in solve.default(Dmat) :
  system is computationally singular: reciprocal
  condition number = 5.0246e-18
```

The use of `solve()` to find the inverse is related to the use of `solve()` in solving matrix equations of the form $\mathbf{Ax} = \mathbf{b}$ where A is a square matrix, and $\mathbf{x}$ unknown. For example, consider the matrix equation

$$\begin{bmatrix} 1 & -3 \\ 2 & 2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 1 \\ -3 \end{bmatrix}.$$

In R:

```
> bvec <- matrix( c(1, -3), ncol=1); bvec
     [,1]
[1,]    1
[2,]   -3
> xvec <- solve(Amat, bvec); xvec     # Amat plays the role of matrix A
      [,1]
[1,] -2.75
[2,] -1.25
```

To check the solution:

```
> Amat %*% xvec
     [,1]
[1,]    1
[2,]   -3
```

This use of `solve()` also works if `bvec` is defined without using `matrix()`. However, the solution returned by `solve()` in that case is not a matrix either:

```
> bvec <- c(1, -3); x.vec <- solve(Amat, bvec); x.vec
[1] -2.75 -1.25
> is.matrix(x.vec)  # Determines if  x.vec  is an R matrix
[1] FALSE
> is.vector(x.vec)  # Determines if  x.vec  is an R vector
[1] TRUE
```

References

[1] Dalgaard, P.: Introductory Statistics with R, second edn. Springer Science and Business Media, New York (2008)

[2] Dunn, P.K.: tweedie: Tweedie exponential family models (2017). URL https://CRAN.R-project.org/package=tweedie. R package version 2.3.0

[3] Dunn, P.K., Smyth, G.K.: GLMsData: Generalized linear model data sets (2017). URL https://CRAN.R-project.org/package=GLMsData. R package version 1.0.0

[4] Fox, J.: The R Commander: A basic statistics graphical user interface to R. Journal of Statistical Software **14**(9), 1–42 (2005)

[5] Fox, J.: Using the R Commander: A Point-and-Click Interface for R. Chapman and Hall/CRC Press, Boca Raton FL (2017)
[6] Fox, J., Bouchet-Valat, M.: Rcmdr: R Commander (2016). URL http://socserv.socsci.mcmaster.ca/jfox/Misc/Rcmdr/. R package version 2.3.1
[7] Henderson, H.V., Velleman, P.F.: Building multiple regression models interactively. Biometrics **37**(2), 391–411 (1981)
[8] Maindonald, J.H., Braun, J.: Data Analysis and Graphics using R, third edn. Cambridge University Press, UK (2010)
[9] R Core Team: foreign: Read Data Stored by Minitab, S, SAS, SPSS, Stata, Systat, Weka, dBase, ... (2017). URL https://CRAN.R-project.org/package=foreign. R package version 0.8-69
[10] R Core Team: R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing, Vienna, Austria (2017). URL https://www.R-project.org/
[11] Smyth, G.K.: Australasian data and story library (OzDASL) (2011). URL http://www.statsci.org/data
[12] Smyth, G.K.: statmod: Statistical Modeling (2017). URL https://CRAN.R-project.org/package=statmod. R package version 1.4.30. With contributions from Yifang Hu, Peter Dunn, Belinda Phipson and Yunshun Chen.
[13] Venables, W.N., Ripley, B.D.: Modern Applied Statistics with S, fourth edn. Springer-Verlag, New York (2002)
[14] Wickham, H.: ggplot2: Elegant Graphics for Data Analysis. Springer-Verlag New York (2009)

Appendix B
The **GLMsData** package

If you have only pretend data, you can only pretend to analyze it.
Watkins, Scheaffer and Cobb [2, p. x]

Almost all of the data files used in this book are collated in the R package **GLMsData** [1]. This package is available from CRAN, and is downloaded and installed like any other R package (Sect. A.2.5). The version of **GLMsData** used to prepare this book is 1.0.0. Since the publication of this book, the contents of the **GLMsData** package may have been updated.

A list of the 97 data files in the **GLMsData** package appear below, with a brief description. For more details about the **GLMsData** package in general, enter `library(help = "GLMsData")` at the R prompt, assuming the **GLMsData** package is installed. For more information about any individual data set, say `lungcap`, enter `?lungcap` at the R prompt (assuming the **GLMsData** package is installed and loaded).

`AIS`	`Australian Institute of Sports (AIS) data`
`ants`	`Ants species richness`
`apprentice`	`Apprentice migration to Edinburgh`
`babblers`	`Feeding rates of babblers`
`belection`	`British election candidates`
`blocks`	`Blocks stacked by children`
`boric`	`Dead embryos after exposure to boric acid`
`breakdown`	`Dialetric breakdown data`
`bttstudy`	`The South African Birth to Ten (BTT) study`
`budworm`	`Insecticide doses and tobacco budworm`
`butterfat`	`Butterfat and dairy cattle`
`ccancer`	`Canadian cancers`
`ceo`	`CEO salaries`
`cervical`	`Deaths from cervical cancer`
`cheese`	`Tasting cheese`
`cins`	`Canadian car insurance data`
`crawl`	`The age at which babies start to crawl`
`cyclones`	`Cyclones near Australia`
`danishlc`	`Danish lung cancer`
`dental`	`Decayed, missing and filled teeth`
`deposit`	`Insecticides`
`downs`	`Downs Syndrome cases in British Columbia`

P. K. Dunn, G. K. Smyth, *Generalized Linear Models with Examples in R*, Springer Texts in Statistics, https://doi.org/10.1007/978-1-4419-0118-7

dwomen	Depression and children
dyouth	Depression in adolescents
earinf	Ear infections in swimmers
emeraldaug	August monthly rainfall in Emerald
energy	Energy expenditure
failures	Failures of electronic equipment
feedrates	Feeding rates of birds
fineroot	The root length density of apple trees
fishfood	Food consumption for fish
flathead	Tiger flathead from trawls
flowers	The average number of meadowfoam flowers
fluoro	The time of fluoroscopy and total radiation
galapagos	Gal\'apagos Island species data
germ	Germination of seeds
germBin	Germination of seeds
gestation	Gestation time
gforces	G-induced loss of consciousness
gopher	Clutch sizes of Gopher tortoises
gpsleep	Sleep times for guinea pigs
grazing	Bird abundance in grazing areas
hcrabs	Males attached to female horseshoe crabs
heatcap	Heat capacity of hydrobromic acid
humanfat	Human age and fatness
janka	Janka hardness
kstones	Treating kidney stones
lactation	Lactation of dairy cows
leafblotch	Percentage leaf area of leaf blotch
leukwbc	Leukaemia survival times
lime	Small-leaved lime trees
lungcap	Lung capacity and smoking in youth
mammary	Adult mammary stem cells
mandible	Mandible length and gestational age
manuka	Manuka honey and wound healing
motorins	Swedish third-party car insurance
mutagen	Mutagenicity assay
mutantfreq	Cell mutant frequencies in children
nambeware	Nambeware products
nhospital	Naval hospital maintenance
nitrogen	Soil nitrogen
nminer	Noisy miner abundance
paper	The tensile strength of paper
perm	Permeability of building materials
phosphorus	Soil phosphorus
pock	Pock counts
poison	Survival times of animals
polyps	The number of polyps and suldinac
polythene	Cosmetic company use of polythene
punting	Football punting
quilpie	Total July rainfall at Quilpie
ratliver	Drugs present in rat livers
rootstock	Rootstock data
rrates	Oxidation rate of benzene
rtrout	Weights of rainbow trout
ruminant	Energy in ruminant's diets

satiswt	Satisfaction with weight in youth
sdrink	Soft drink delivery times
seabirds	Counts of seabirds
serum	Mice surviving doses of antipneumococcus serum
setting	Heat evolved by setting cement
sharpener	Sharpener data
sheep	The daily energy requirements for wethers
shuttles	O-rings on the space shuttles
teenconcerns	Concerns of teenagers
toothbrush	Effectiveness of toothbrushes
toxo	Toxoplasmosis and rainfall
triangle	Artificial data from triangles
trout	The effect of potassium cyanate on trout eggs
turbines	Fissures in turbine wheels
urinationD	Urination time
urinationL	Urethral length
wacancer	Cancer in Western Australia
wheatrain	Annual rainfall in the NSW wheat belt
windmill	Power generation by windmills
wwomen	Smoking and survival
yieldden	Yield of onions at various densities

References

[1] Dunn, P.K., Smyth, G.K.: GLMsData: Generalized linear model data sets (2017). URL https://CRAN.R-project.org/package=GLMsData. R package version 1.0.0

[2] Watkins, A.E., Sheaffer, R.L., Cobb, G.W.: Statistics in Action, second edn. Key Curriculum Press (2008)

Selected Solutions

Research has shown that it is effective to combine example study and problem solving in the initial acquisition of cognitive skills.
Renkl [3, p. 293]

The data used generally come from the **GLMsData** [2] package. We do not explicitly load this package each time it is needed.

```
> library(GLMsData)
```

Solutions to Problems from Chap. 1

1.1 The more complex quartic model is similar to the cubic. The cubic is possibly superior to the quadratic, so we probably prefer the cubic.

1.4 The proportion testing positive is between zero and one. The cubic regression model is not good—it permits proportions outside the physical range; the cubic GLM is preferred.

1.5 ***1.*** Linear in the parameters; suitable for linear regression and GLMs. ***2.*** Not linear in parameters. ***3.*** Linear in the parameters; suitable for GLMs. ***4.*** Linear in the parameters; suitable for GLMs.

1.6

```
> data(turbines)
>    ### Part 1
> names(turbines)
>    ### Part 4
> summary(turbines)
>    ### Part 5
> plot(Fissures/Turbines ~ Hours, data=turbines, las=1)
```

2. All variables are quantitative. ***3.*** Clearly the number of hours run is important for knowing the proportion of fissures. The proportion must be between 0 and 1 obviously.

1.9

```
> data(blocks); blocks$Trial <- factor(blocks$Trial)
> blocks$cutAge <- cut(blocks$Age, breaks=c(0, median(blocks$Age), Inf))
>    ### Part 1
```


P. K. Dunn, G. K. Smyth, *Generalized Linear Models with Examples in R*,
Springer Texts in Statistics, https://doi.org/10.1007/978-1-4419-0118-7

```
> summary(blocks)
>    ### Part 2
> par( mfrow=c(2, 4))
> plot( Time~Shape, data=blocks, las=1)
> plot( Time~Trial, data=blocks, las=1)
> plot( Time~Age, data=blocks, las=1)
> with(blocks, interaction.plot(Shape, cutAge, Time))
>    ### Part 4
> plot( Number~Shape, data=blocks, las=1)
> plot( Number~Trial, data=blocks, las=1)
> plot( Number~Age, data=blocks, las=1)
> with(blocks, interaction.plot(Shape, cutAge, Number))
```

3. For both responses: shape seems important; trial number doesn't; age possibly. ***5.*** Perhaps interactions.

Solutions to Problems from Chap. 2

2.1 ***1.*** β_0 is the predicted value when $x = 0$. ***2.*** α_0 is the predicted value when x is equal to the mean of x (that is, $\overline{x}$). The second form may allow a better interpretation of the constant, since $x = 0$ may be far from the values of x used to fit the model.

2.2 Solve the equations. Note that $\sum w_i(x_i - \bar{x}_w)^2 = \sum w_i x_i^2 - \left(\sum w_i x_i\right)^2 / \sum w_i$. and $\sum w_i(x_i - \bar{x}_w)y_i = \sum w_i x_i y_i - \sum w_i x_i \bar{y}_w$, which makes the connection to the given formula a bit easier to see.

2.4 ***1.*** Expand $S = (\mathbf{y} - \mathrm{X}\boldsymbol{\beta})^T \mathrm{W}(\mathbf{y} - \mathrm{X}\boldsymbol{\beta})$ to get the result. ***2.*** Differentiating with respect to $\boldsymbol{\beta}$ gives $\partial S/\partial\boldsymbol{\beta} = -2\mathrm{X}^T\mathrm{W}\mathbf{y} + 2\mathrm{X}^T\mathrm{WX}\boldsymbol{\beta}$. ***3.*** Setting the differential to zero and solving gives $\mathrm{X}^T\mathrm{W}\mathbf{y}^T = \mathrm{X}^T\mathrm{WX}\boldsymbol{\beta}$. Pre-multiplying by $(\mathrm{X}^T\mathrm{WX})^{-1}$ gives the result.

2.6 $\mathrm{E}[\hat{\boldsymbol{\beta}}] = (\mathrm{X}^T\mathrm{WX})^{-1}\mathrm{X}^T\mathrm{WE}[\mathbf{y}] = (\mathrm{X}^T\mathrm{WX})^{-1}\mathrm{X}^T\mathrm{W}(\mathrm{X}\boldsymbol{\beta}) = \boldsymbol{\beta}$.

2.8 Substituting for R^2 on the right in terms of SS gives $\{\mathrm{SS}_{\mathrm{REG}}/(p'-1)\}/\{\mathrm{SS}_{\mathrm{T}}/(n-p')\}$, which is F.

2.9 ***1.*** A: $\begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 & 0 \end{bmatrix}^T$; B: $\begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & -1 & -1 \end{bmatrix}^T$; C: $\begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 0.5 & 0 & -0.5 & -1 \end{bmatrix}^T$.
2. Then, use that $\mathrm{var}[\hat{\mu}] = \mathbf{x}_g(\mathrm{X}^T\mathrm{X})^{-1}\mathbf{x}_g^T$ with $\mathbf{x}_g = [1 \quad x]$ to obtain $\mathrm{var}[\hat{\mu}_A] = (1/4) + x^2/5$; $\mathrm{var}[\hat{\mu}_B] = (5 - 6x + 5x^2)/16$; $\mathrm{var}[\hat{\mu}_B] = (1 - 2x^2)/5$.

```
> x  <- seq(-1, 1, length=100)
> xA <- c(1, 1, -1, -1, 0)
> xB <- c(1, 1, 1, 1, -1)
> xC <- c(1, 0.5, 0, -0.5, -1)
> varA <- function(x){0.25 + x^2/5}
> varB <- function(x){(5 - 6*x + 5*x^2)/16}
> varC <- function(x){(1+2*x^2)/5}
> vA <- varA(x); vB <- varB(x); vC <- varC(x)
> plot( range(c(vA, vB, vC)) ~ range(x), type="n", ylim=c(0, 1.2),
       ylab="Var. of predictions", xlab="x values", las=1)
> lines(varA(x) ~ x, lty=1, lwd=2)
> lines(varB(x) ~ x, lty=2, lwd=2)
```

```
> lines(varC(x) ~ x, lty=3, lwd=2)
> legend("top", lwd=2, lty=1:3, legend=c("Design A", "Design B",
    "Design C"))
```

As would be expected from the location of the x values: A produces the most uniform small prediction errors; B produces smaller prediction errors for larger x values; C produces smaller prediction errors in the middle of the range of x values.

2.10 ***1.*** The Taylor series expansion is $f(x) = f(\bar{x}) + \mathrm{d}f/\mathrm{d}x(x-\bar{x}) + \mathrm{d}^2 f/\mathrm{d}x^2(x-\bar{x})^2/2 + \cdots$. ***2.*** $f(x)$ is linear in x, if $x - \bar{x}$ is small. ***3.*** Any function can be considered *locally* approximately linear.

2.15 ***2.*** The relationship between the number of flowers per plant and light intensity has different intercepts for the different timings, but the same slope. ***3.*** The relationship between the number of flowers per plant and light intensity has different intercepts and different slopes for the different timings. ***4.*** Interaction term doesn't seem necessary. ***5.*** Makes no difference to the parameter estimates or standard errors. However, the estimate of σ is different. ***6.*** The interaction term does not seem needed.

```
> data(flowers)
> wts <- rep(10, length(flowers$Light) )
>    ### Part 1
> plot(Flowers~Light, data=flowers, pch=ifelse(Timing=="PFI", 1, 19))
> legend("topright", pch=c(1, 19), legend=c("PFI","Before PFI"))
>    ### Part 3
> m1    <- lm(Flowers~Light*Timing, data=flowers, weights=wts); anova(m1)
> m2    <- lm(Flowers~Light+Timing, data=flowers, weights=wts); anova(m2)
>    ### Part 5
> m1.nw <- lm(Flowers~Light*Timing, data=flowers);           anova(m1.nw)
> m2.nw <- lm(Flowers~Light+Timing, data=flowers);           anova(m2.nw)
> summary(m1); summary(m1.nw)
>    ### Part 6
> abline(coef(m2)[1], coef(m2)[2], lty=1)
> abline(sum(coef(m2)[c(1, 3)]), coef(m2)[2], lty=2)
```

2.18

```
> data(blocks)
>    ### Part 5
> m0 <- lm( Time ~ Shape, data=blocks);                anova( m0 )
> mA <- lm( Time ~ Trial + Age + Shape, data=blocks); anova( mA )
>    ### Part 6
> mB <- update(mA, . ~ Trial + Age*Shape);             anova( mB )
> t.test(Time~Shape, data=blocks)
> summary(m0)
>    ### Part 7
> m1 <- lm( Time~Shape, data=blocks);                  anova(m1)
```

1. Possible increasing variance. Perhaps non-linear? ***2.*** The relationship between age and time has different intercepts and slopes for the two shapes. ***3.*** Time depends on age and trial number, and the effect of age depends on the trial number. ***4.*** Time depends on age and shape, and both depend on the trial number. ***8.*** On average, the time taken to stack cylinders is 14.45 s less than for cubes.

Solutions to Problems from Chap. 3

3.2 Expand the expressions, simplify, and the results follow.

3.8

```
> data(lungcap)
>    ### Part 1
> m1 <- lm(FEV~factor(Smoke), data=lungcap)
>    ### Part 2
> m2 <- lm(FEV~factor(Smoke)+Age+Ht+factor(Gender), data=lungcap)
>    ### Part 3
> m3 <- lm(log(FEV)~factor(Smoke)+Age+Ht+factor(Gender), data=lungcap)
>    ### Part 4
> summary(m1); summary(m2); summary(m3); anova(m3) # Prefer m3
```

1. Smokers have a *larger* FEV by an average of 0.7107 L. ***2.*** Smokers have a *smaller* FEV by an average of −0.08725 L. ***3.*** Smokers have a *smaller* FEV by, on average, a factor of 0.9165.

3.10

```
> data(cheese)
> m4 <- lm( log(Taste) ~ log(H2S) + Lactic + Acetic, data=cheese )
> scatter.smooth( rstandard(m4) ~ fitted(m4) )
> qqnorm( rstandard(m4) ); qqline( rstandard(m4) )
> plot( cooks.distance(m4), type="h")
```

3.11

```
> data(fishfood); par(mfrow=c(2, 3))
>    ### Part 1
> m1 <- lm( log(FoodCon) ~ log(MaxWt) + log(Temp) + log(AR) + Food,
           data=fishfood); anova(m1)
>    ### Part 2
> plot(rstandard(m1)~fitted(m1)); qqnorm(rstandard(m1))
> plot( cooks.distance(m1), type="h") # Model looks OK
> m2 <- update(m1, . ~ log(MaxWt) * log(Temp) * Food * log(AR))
> m3 <- step(m2); anova(m1, m3) # Model m3 a bit better
> plot(rstandard(m3)~fitted(m3)); qqnorm(rstandard(m3))
> plot( cooks.distance(m3), type="h") # Model looks OK
```

3. Unravelling, the model has the form $\hat{\mu} = \exp(\beta_0) x_1^{\beta_1} x_2^{\beta_2} \cdots$. ***4.*** The interaction model is slightly better if the automated procedure can be trusted, by the ANOVA test (and AIC).

3.13

```
> data(flowers)
> m1 <- lm(Flowers~Light+Timing, data=flowers)
>    ### Part 1
> scatter.smooth( rstandard(m1) ~ fitted(m1) )
> qqnorm( rstandard(m1) ); qqline( rstandard(m1) )
> plot( cooks.distance(m1), type="h")
> plot( rstandard(m1) ~ flowers$Light)
>    ### Part 2
> rowSums(influence.measures(m1)$is.inf)
```

2. No observations reported as influential.

3.16

```
> data(blocks); par(mfrow=c(2, 4))
> m1 <- lm( Time~Shape, data=blocks); anova(m1)
>    ### Part 1
> plot( rstandard(m1) ~ fitted(m1) )
> qqnorm( rstandard(m1) ); qqline( rstandard(m1) )
> plot( cooks.distance(m1), type="h")
> plot( rstandard(m1) ~ blocks$Shape)
> rowSums(influence.measures(m1)$is.inf)
>    ### Part 2
> m2 <- lm( log(Time)~Shape*Age, data=blocks); anova(m2)
> m2 <- update(m2, .~Shape+Age);               anova(m2)
> m2 <- update(m2, .~Shape);                   anova(m2)
> plot( rstandard(m2) ~ fitted(m2) )
> qqnorm( rstandard(m2) ); qqline( rstandard(m2) )
> plot( cooks.distance(m2), type="h")
> plot( rstandard(m2) ~ blocks$Shape)
> rowSums(influence.measures(m2)$is.inf)
```

1. The model includes only `Shape`. The Q–Q plot shows non-normality; the variance is different between cubes and cylinders. Perhaps influential observations. *2.* The model diagnostics appear better, if not perfect, after applying a log-transform.

3.21

```
> data(paper)
>    ### Part 1
> plot( Strength~Hardwood, data=paper)
>    ### Part 2
> m1 <- lm(Strength ~ poly(Hardwood, 5),  data=paper); summary(m1)
>    ### Part 3
> m2 <- lm(Strength ~ ns(Hardwood, df=7), data=paper); summary(m2)
>    ### Part 4
> newH <- seq( min(paper$Hardwood), max(paper$Hardwood), length=100)
> newy1 <- predict( m1, newdata=data.frame(Hardwood=newH))
> newy2 <- predict( m2, newdata=data.frame(Hardwood=newH))
> lines(newy1~newH)
> lines(newy2~newH, lty=2)
```

3.23

```
> data(gopher)
>    ### Part 1
> par( mfrow=c(2, 2))
> plot( ClutchSize ~ Temp, data=gopher)
> plot( ClutchSize ~ Evap, data=gopher)
>    ### Part 3
> gt.lm <- lm( ClutchSize ~ Temp + Evap, weights=SampleSize, data=gopher)
> summary(gt.lm)
>    ### Part 4
> anova(gt.lm)
>    ### Part 5
> cor(cbind(gopher$ClutchSize, gopher$Temp, gopher$Evap, gopher$Latitude))
```

```
>    ### Part 6
> plot( Evap ~ Latitude,    data=gopher)
> plot( Temp ~ Latitude,    data=gopher)
> m1 <- lm(ClutchSize~Evap, data=gopher)
> par(mfrow=c(2, 2))
> plot( rstandard(m1) ~ gopher$Latitude)
> plot( rstandard(m1) ~ fitted(m1))
> plot(cooks.distance(m1), type="h")
> qqnorm( rstandard(m1)); qqline( rstandard(m1))
```

1. Some reasonable positive relationships. ***2.*** Each site has a different number of clutches. ***3.*** No significant explanatory variables. ***4.*** No significant explanatory variables. ***6.*** Evaporation and temperature look related to latitude.

3.25

```
> data(ratliver)
>    ### Part 1
> plot( DoseInLiver ~ BodyWt,  data=ratliver)
> plot( DoseInLiver ~ LiverWt, data=ratliver)
> plot( DoseInLiver ~ Dose,    data=ratliver)
>    ### Part 2
> m1  <- lm(DoseInLiver ~ BodyWt + LiverWt + Dose, data=ratliver)
>    ### Part 3
> summary(m1); anova(m1)
>    ### Part 4
> influence.measures(m1)
> infl <- which.max(cooks.distance(m1))
>    ### Plot 5
> plot(BodyWt ~ Dose, data=ratliver)
> points(BodyWt ~ Dose, subset=(infl), pch=19, data=ratliver)
>    ### Plot 6
> m2  <- update(m1, subset=(-infl) ); summary(m2); anova(m2)
```

1. Possible relationships.

Solutions to Problems from Chap. 4

4.2 Apply the derivatives and the results follow.

4.5

1. For one observation: $\ell = -\log\mu - y/\mu$.
2. $U(\mu) = -n/\mu + \sum y_i/\mu^2 = n(\hat{\mu} - \mu)/\mu^2$.
3. $\hat{\mu} = \sum y_i/n$.
4. $\mathcal{J}(\mu) = (-n\mu + 2\sum y_i)/\mu^3 = -n(\mu - 2\hat{\mu})/\mu^3$; $\mathcal{I}(\mu) = n/\mu^2$.
5. $\text{se}(\hat{\mu}) = \mathcal{I}(\hat{\mu})^{-1/2} = \mu/\sqrt{n}$.
6. $W = n(\hat{\mu} - 1)^2/\hat{\mu}^2$.
7. $S = n(\hat{\mu} - 1)^2$.
8. $L = 2n(\hat{\mu} - \log\hat{\mu} - 1)$.
10. W, S and L are similar near $\hat{\mu}$, but dissimilar far away from $\hat{\mu}$. For larger values of n, the curves are sharper at $\hat{\mu}$, so there is more information.

```
> par(mfrow=c(1, 2))
> muhat <- seq(0.5, 2, length=200)
>    ### Part 9
> n <- 10
> W <- (muhat-1)^2/(muhat^2/n)
> S <- n*(muhat-1)^2
> L <- 2*n*(muhat-log(muhat)-1)
> plot(range(W)~range(muhat), type="n", main="n = 10", xlab="x", ylab="")
> lines(W~muhat)
> lines(S~muhat, lty=2)
> lines(L~muhat, lty=3)
> legend("top", lty=1:3, legend=c("Wald","Score","LRT"))
> abline(v=1, lty=4)
>    ### Part 10
> n <- 100
> W <- (muhat-1)^2/(muhat^2/n)
> S <- n*(muhat-1)^2
> L <- 2*n*(muhat-log(muhat)-1)
> plot(range(W)~range(muhat), type="n", main="n = 100", xlab="x", ylab="")
> lines(W~muhat)
> lines(S~muhat, lty=2)
> lines(L~muhat, lty=3)
> legend("top", lty=1:3, legend=c("Wald","Score","LRT"))
> abline(v=1, lty=4)
```

4.6

```
> set.seed(252627)
> n <- 200; yy <- rexp(n, 1); len.mu <- 250
> #Part 1:
> muhat.vec <- seq(0.75, 1.25, length=len.mu)
> llh <- array(dim=len.mu)
> for (i in (1:length(muhat.vec))){
    llh[i] <- sum( log( dexp(yy, rate=1/muhat.vec[i]) ) )
 }
> plot(llh~muhat.vec, type="l", lwd=2, las=1, xlab="mu")
> muhat <- mean(yy); critical <- qchisq(1-0.05, df=1)
> abline(v=1); abline(v=muhat); abline(h=max(llh)- critical, lty=2)
> # Part 2:
> W <- (muhat-1)^2/(muhat^2/n); S <- n * (muhat-1)^2
> L <- 2*n*(muhat - log(muhat)-1)
> c(W, S, L); pexp( c(W, S, L), rate=1, lower.tail=FALSE)
> # Part 3:
> W <- (muhat.vec-1)^2/(muhat.vec^2/n); S <- n * (muhat.vec-1)^2
> L <- 2*n*(muhat.vec - log(muhat.vec)-1)
> plot(W~muhat.vec, type="l", lwd=2, ylab="Test statistic", xlab="mu hat")
> lines(S~muhat.vec, lty=2, lwd=2); lines(L~muhat.vec, lty=3, lwd=2)
> abline(v=1); abline(v=muhat); abline(h=critical)
> legend("top", lty=1:3, legend=c("Wald","Score","L. Ratio"),
    lwd=2, bg="white")
> # Parts 4 and 5
> se <- sqrt(muhat/n); se; c(muhat - se*1.960, muhat+se*1.960)
```

Solutions to Problems from Chap. 5

5.1 ***2.*** Geometric: $\theta = \log(1-p)$; $\kappa(\theta) = \log\{(1-p)/p\}$; $\phi = 1$. ***5.*** Strict arcsine: $\theta = \log p$; $\kappa(\theta) = \arcsin p$; $\phi = 1$.

5.4 Apply the formula.

5.7 $K''(t) = \phi\kappa''(\theta + t\phi)$; on setting $t = 0$ the results follow.

5.13 $\tau = 1 \times y/(y-0)^2 = (1/y) \leq (1/3)$.

5.16

1. Proceed:

$$M_Y(t) = \sum_{y=0}^{\infty} \exp(-\lambda)\lambda^y/y! \times e^{ty} = \exp(-\lambda)\sum_{y=0}^{\infty}\{\lambda \exp t\}^y \lambda^y/y! = \exp(-\lambda + \lambda e^t).$$

2. $K_Y(t) = \log M_Y(t) = -\lambda + \lambda e^t$.
3. Differentiating and setting $t = 0$ gives the required results.

5.17 ***1.*** $M_{\bar{y}}(t) = \mathrm{E}[\exp\{t(y_1 + \cdots y_n)/n\}] = \mathrm{E}[\exp(ty/n)]^n = M_y(t/n)^n$ since the y_i are iid. ***2.*** Then, $K_{\bar{y}}(t) = \log M_{\bar{y}}(t) = n \log M_y(t/n) = nK_y(t/n) = n\{\kappa(\theta + t\phi/n) - \kappa(\theta)\}/\phi$. ***3.*** This is the CGF of $\mathrm{EDM}(\mu, \phi/n)$.

5.18

1. Follow Sect. 5.3.6 (p. 217): $\theta = \arctan\mu$; $\kappa(\theta) = -\log(\cos\theta) = \{\log(1+\mu^2)\}/2$.
2. $d(y,\mu) = 2[y(\arctan y - \arctan\mu) - (1/2)\log\{(1+y^2)/(1+\mu^2)\}]$.
3. The saddlepoint approximation: $\tilde{\mathcal{P}}(y;\mu,\phi) = 1/\sqrt{2\pi\phi(1+y^2)}\exp\{-d(y,\mu)/(2\phi)\}$.
4. Saddlepoint approx. expected to be OK if $\phi(1+y^2)/y^2 \leq 1/3$; or $y^2 \geq -3/2$ when $\phi = 1$, or $y^2 \geq -3$ when $\phi = 0.5$. These expressions are true for all y.
5. The canonical link function has $\eta = \theta$, which is $\eta = \arctan\mu$.

```
> y <- seq(-4, 2, length=200); phi<-0.5; phi2 <- 1; mu <- -1
> b <- 1/sqrt(2*pi*phi*(1+y^2)); b2 <- 1/sqrt(2*pi*phi2*(1+y^2))
> dev <- 2*(y*(atan(y) - atan(mu))-(1/2)*log((1+y^2)/(1+mu^2)))
> plot(  b * exp(-dev/(2*phi ))~y, type="l")
> lines( b2* exp(-dev/(2*phi2))~y, lty=2)
> legend("topright", lty=1:2, legend=c("phi=0.5","phi=1"))
```

5.22 $M_y(t) = \int_0^\infty \exp(ty)\exp(-y)\,dy = 1/(1-t)$, provided $t < 1$ (otherwise the limit as $y \to \infty$ is not defined). Taking logs, $K_y(t) = \log M_y(t) = -\log(1-t)$, provided $t < 1$. Differentiating, $K'_y(t) = (1-t)^{-1}$, so $K'_y(0) = 1$. Likewise for the variance.

5.24 $g(\mu) = |\mu|$ is not valid (not differentiable when $-\infty < \mu < \infty$). $g(\mu) = \mu^2$ is not valid when $-\infty < \mu < \infty$ (not a monotonic function).

5.25

```
> data(blocks)
>    ### Part 1
> par(mfrow=c(1, 2))
> plot(jitter(Number)~Age, data=blocks)
> plot( Number~cut(Age, 3), data=blocks)
```

Responses are counts; variance increases with mean. Poisson GLM?

Solutions to Problems from Chap. 6

6.3 Consider $w_i(y_i - \mu_i)^2/V(\mu)^2$. Here, μ is constant, then taking expected values $w_i/V(\mu)^2 \mathrm{E}[(y_i - \mu_i^2)^2]$. By definition, the expected value of $(y_i - \mu_i)^2$ is $\mathrm{var}[y] = \phi V(\mu)/w_i$, so the expression simplifies to ϕ. Thus the expected value of the Pearson estimator is $1/(n - p') \times \sum_{i=1}^{n} \phi = \{n/(n - p')\}\phi$ with p' estimated regression parameters, approximately unbiased. With μ known and hence no unknown regression parameters, $p' = 0$ and then the expected value is ϕ, so the estimate is unbiased.

6.6

1. Using $\dfrac{\partial \ell^2}{\partial \beta_k \partial \beta_j} = \dfrac{\partial U(\beta_j)}{\partial \mu} \dfrac{\partial \mu}{\partial \beta_k}$. The first derivative comes from Problem 6.5. For the second, using that the canonical link function is $g(\mu) = \eta = \log\{\mu/(1 - \mu)\}$, we get that $\mathrm{d}\eta/\mathrm{d}\mu = 1/\{\mu(1 - \mu)\}$ and $\partial\mu/\partial\beta_k = \mu(1 - \mu)x_k$. Combining,

$$\mathcal{I}_{jk} = -\frac{\partial \ell^2}{\partial \beta_k \partial \beta_j} = \sum_{i=1}^{n} w_i \mu_i (1 - \mu_i) x_{ji} x_{ki}.$$

2. $z = \log\{\mu/(1 - \mu)\} + (y - \mu)/\{\mu(1 - \mu)\}$.

6.9

1. Using $\eta = \log \mu$, then $\mathrm{d}\eta/\mathrm{d}\mu = 1/\mu$. Hence $W_i = w_i/\mu_i$ and $U_j = \sum_{i=1}^{n} w_i(y_i - \mu_i)x_{ji}/(\phi\mu^2)$.
2. $z_i = \log \mu_i + (y_i - \mu_i)/\mu_i$.
3. Finding ℓ and differentiating with respect to ϕ leads to $\hat{\phi} = D(y, \mu)/n$.
4. $\tilde{\phi} = D(y, \mu)/(n - p')$.
5. $\bar{\phi} = X^2/(n - p')$ where $X^2 = \sum_{i=1}^{n} w_i(y_i - \hat{\mu}_i)^2/\hat{\mu}_i^3$.

6.10

```
> data(blocks)
> m1 <- glm(Number~Age, data=blocks, family=poisson)
> m1; deviance(m1); summary(m1)
```

Solutions to Problems from Chap. 7

7.1

```
>    ### Part 1
> L <- c(0.602, 14.83, 2.83)
> p.LRT <- pchisq(L, df=1,  lower.tail=FALSE)
>    ### Part 2
> beta <- c(0.143, 1.247, -0.706)
> se <- c(0.19, 0.45, 0.45)
> Wald <- beta/se
> p.Wald <- pnorm(abs(Wald), lower.tail=FALSE)*2
> cbind(p.LRT, p.Wald)
>    ### Part 4
> zstar <- qnorm(0.975)
> margin.err <- zstar*0.45
> c( 1.247 - margin.err, 1.247, 1.247 + margin.err)
```

7.3

```
>    ### Part 1
> ppois( 7, 1.8)  # Small probably of exceeding seven
> ppois( 7, 2.5)  # Small probably of exceeding seven
>    ### Part 2
> beta <- c(0.23, 0.04, 0.06, 0.01, 0.09, 0.05, 0.30)
> se <-   c(0.13, 0.04, 0.05, 0.03, 0.06, 0.02, 0.07)
> z <- beta/se; pvals <- (1-pnorm(abs(z)))*2
> round(pvals, 3)
```

1. The counts have an upper limit: weeks have a maximum of seven days. However, the means are relatively small, so a Poisson GLM may be OK.
2. Wald test: $z = 0.30/0.07 \approx 4.3$, which is highly significant. There is evidence of a difference.
3. Junior Irish legislators spend an average of 0.3 more days per week in their constituency.
4. $0.30 \pm 1.960 \times 0.07$.
5. 'Geographic proximity' and 'Nation' are statistically significant.
6. The systematic component:

$$\log \mu = 0.23 + 0.04x_1 + 0.06x_2 + 0.01x_3 + 0.09x_4 + 0.05x_5 + 0.30x_6;$$

the random component: $y_i \sim \text{Pois}(\mu_i)$.

7.4

```
> data(blocks); library(statmod)
> m1 <- glm(Number~Age, data=blocks, family=poisson)
> m0 <- update(m1, .~1)
>    ### Part 1
> z.Wald <- coef(summary(m1))[2, 3]
> P.Wald <- coef(summary(m1))[2, 4]
>    ### Part 2
> z.score <- glm.scoretest(m0, blocks$Age)
> P.score <- 2*(1-pt(abs(z.score), df=df.residual(m1)))
>    ### Part 3
> chisq.LRT <- anova(m1)[2, 2]
>  P.LRT <- anova(m1, test="Chisq")[2, 5]
>   # Part 4
> round(c(z.Wald, z.score, sqrt(chisq.LRT)), 4)
> round(c(P.Wald, P.score, P.LRT), 4); min(blocks$Number)
>    ### Part 8
> newA <- seq( min(blocks$Age), max(blocks$Age), length=100)
> newB <- predict( m1, newdata=data.frame(Age=newA), type="response",
                   se.fit=TRUE)
> plot( jitter(Number)~Age, data=blocks)
> lines(newB$fit ~ newA, lwd=2)
> t.star <- qt(p=0.975, df=df.residual(m1))
> ci.lo <- newB$fit - t.star * newB$se.fit
> ci.hi <- newB$fit + t.star * newB$se.fit
> lines(ci.lo~newA, lty=2)
> lines(ci.hi~newA, lty=2)
```

5. For a Poisson GLM, expect the saddlepoint approximation to be sufficient if the smallest $y \geq 3$; here the minimum is 3, so expect the saddlepoint approximation to be

OK. ***6.*** For a Poisson GLM, expect the CLT approximation to be sufficient if the smallest $y \geq 5$; here the minimum is 3 (and there are ten counts of 4), so the CLT approximation may be insufficiently accurate.

Solutions to Problems from Chap. 8

8.3 $r_D = \text{sign}(y_i - \mu_i)\sqrt{2\left[y\log(y/\mu) + (1-y)\log\{(1-y)(1-\mu)\}\right]}$. The result follows from substituting $y = 0$ and $y = 1$, and using that $\lim_{t\to 0} t\log t = 0$.

8.7 ***1.*** $r_P = (y-\mu)/\mu = (y/\mu) - 1$. $r_D = 2\sqrt{-\log(y/\mu) + (y-\mu)/\mu}$. Since $\mathcal{F}(y;\mu) = 1 - \exp(-y/\mu)$, $r_Q = \Phi^{-1}\left[1 - \exp(-y/\mu)\right]$. Hence $r_P = -0.571$; $r_D = -0.552$; $r_Q = \Phi^{-1}(0.34856) = -0.389$. ***2.*** Then $r_D = 0$; $r_D = 0$; $r_Q = \Phi^{-1}(0.632) = 0.337$. $r_Q \neq 0$ even though $y = \mu$. ***3.*** While quantile residual have a normal distribution, they do not necessarily report a zero residual when $y = \mu$. (They are best used for identifying patterns.)

8.11

```
> data(blocks); library(statmod)
> m1 <- glm(Number~Age, data=blocks, family=poisson)
> par(mfrow=c(2, 2))
> plot( rstandard(m1)~fitted(m1))
> plot(cooks.distance(m1), type="h")
> qqnorm(rstandard(m1)); qqnorm(qresid(m1))
> colSums(influence.measures(m1)$is.inf)
```

8.13

```
> data(triangle)
>    ### Part 2
> m1 <- glm( y~I(x1^2) + I(x2^2), data=triangle,
            family=quasi(link=power(lambda=2), variance="constant"))
> m2 <- glm( y~I(x1^2) + I(x2^2), data=triangle,
            family=quasi(link=power(lambda=2), variance="mu^2"))
> plot( rstandard(m1)~fitted(m1)); qqnorm(rstandard(m1))
> plot(cooks.distance(m1), type="h")
> plot( rstandard(m2)~fitted(m2)); qqnorm(rstandard(m2))
> plot(cooks.distance(m2), type="h")
> colSums(influence.measures(m1)$is.inf)
> colSums(influence.measures(m2)$is.inf)
```

1. $\mu^2 = x_1^2 + x_2^2$ so that the link function is $g(\mu) = \mu^2$.

Solutions to Problems from Chap. 9

9.1 The Taylor series expansion: $\sin^{-1}\sqrt{y} = \sin^{-1}\sqrt{\mu} + (y-\mu)/\left\{2\sqrt{(1-\mu)\mu}\right\} + \cdots$. On computing the variance, $\text{var}[\sin^{-1}\sqrt{y}] \approx \text{var}[y]/\{4(1-\mu)\mu\}$, which is equivalent to $\text{var}[y]$ being a constant times $(1-\mu)\mu$, the binomial variance function.

9.5

```
>    ### Part 2
> beta <- c(-6.949, 0.805, 0.161, 0.332, 0.116)
> se <- c(0.377, 0.0444, 0.113, 0.0393, 0.0204)
> z <- beta/se
>    ### Part 3
> ci <- cbind( beta-1.96*se, beta+1.96*se)
> pvals <- (1-pnorm(abs(z)))*2; OddsRatio <- exp(beta)
> round( cbind(beta, se, z, ci, pvals, OddsRatio), 3)
```

1. $\log\{\mu/(1-\mu)\} = -6.949 + 0.805x_1 + 0.161x_2 + 0.332x_3 + 0.116x_4$, with the x_j as defined in the problem. ***4.*** For example, the odds of having an apnoea-hyponoea index of 1 is 1.123 greater than the odds that the index is 0, after adjusting for the other variables.

9.7

```
> library(statmod)
> data(shuttles)
>    ### Part 1
> plot( Damaged/6 ~ Temp, data=shuttles)
>    ### Part 2
> shuttle.m <- glm(Damaged/6 ~ Temp, weights=rep(6, length(Temp)),
    family=binomial, data=shuttles)
>    ### Part 3
> qqnorm( qresid(shuttle.m))
> colSums(influence.measures(shuttle.m)$is.inf)
>    ### Part 4
> predict(shuttle.m, newdata=data.frame(Temp=31), type="response")
```

5. The temperature at which 50% of the O-rings fail. Since we do not want O-rings to fail, probably a higher threshold would be more useful.

9.9

```
> library(MASS); data(budworm)
>    ### Part 1
> budworm$Prop.Killed <- budworm$Killed/budworm$Number
> plot( Prop.Killed ~ log2(Dose),
       pch=ifelse(Gender=="F", 1, 19), data=budworm)
>    ### Part 2
> m1.logit    <- glm( Prop.Killed ~ Gender * log2(Dose)-1, weights=Number,
                    family=binomial(link=logit), data=budworm )
> anova(m1.logit, test="Chisq")
> m1.logit    <- glm( Prop.Killed ~ Gender + log2(Dose)-1, weights=Number,
                    family=binomial(link=logit), data=budworm )
>    ### Part 3
> newD <- seq( min(budworm$Dose), max(budworm$Dose), length=100)
> newP.F <- predict( m1.logit, newdata=data.frame(Dose=newD, Gender="F"),
    type="response" )
> newP.M <- predict( m1.logit, newdata=data.frame(Dose=newD, Gender="M"),
    type="response" )
> lines( newP.F ~ log2(newD), lty=1)
> lines( newP.M ~ log2(newD), lty=2)
> legend("topleft", lty=1:2, legend=c("Females", "Males"))
>    ### Part 4 and 5
```

```
> summary(m1.logit)
>    ### Part 6
> LD50.F <- dose.p(m1.logit, c(1, 3)); LD50.M <- dose.p(m1.logit, c(2, 3))
> exp(c(LD50.F, LD50.M))
>    ### Part 7
> confint( m1.logit, level=.90)
```

3. Model for males looks better than model for females.

9.11

```
> li <- factor( c(0, 0, 0, 0, 1, 1, 1, 1), labels=c("Absent", "Present") )
> m  <- c(3, 2, 4, 1, 5, 5, 9, 17); y <- c(3, 2, 4, 1, 5, 3, 5, 6)
> gender <- gl(2, 2, 8, labels=c("Female", "Male"))
> par( mfrow=c(1, 3))
>    ### Part 1
> plot(y/m~li); plot(y/m~gender)
> interaction.plot(li, gender, y/m)
>    ### Part 2
> m1 <- glm( y/m ~ gender,    weights=m, family=binomial)
> m2 <- glm( y/m ~ li+gender, weights=m, family=binomial)
> m3 <- glm( y/m ~ gender+li, weights=m, family=binomial)
> summary(m2)
>    ### Part 3
> anova(m2, test="Chisq"); anova(m3, test="Chisq")
>    ### Part 4
> z.score <- glm.scoretest(m1, as.numeric(li))
> p.score <- 2*(1-pnorm(abs(z.score)))
> c(z.score, p.score)
```

5. Wald test results show nothing greatly significant; the others do. The Hauck–Donner effect, since y/m is always 1 when `li` is `Absent`.

Solutions to Problems from Chap. 10

10.1

1. $\theta = \log\{\mu/(\mu+k)\}$; $\kappa(\theta) = k\log(\mu+k)$.
2. The mean is $\mathrm{d}\kappa/\mathrm{d}\theta = \mathrm{d}\kappa/\mathrm{d}\mu \times \mathrm{d}\mu/\mathrm{d}\theta$; hence $\mathrm{d}\theta/\mathrm{d}\mu = k/\{\mu(\mu+k)\}$. Expanding, the mean is μ (as expected). Variance:

$$\mathrm{d}^2\kappa/\mathrm{d}\theta^2 = \mathrm{d}/\mathrm{d}\theta(\mathrm{d}\kappa/\mathrm{d}\theta) = \mathrm{d}/\mathrm{d}\mu(\mathrm{d}\mu/\mathrm{d}\theta)\mathrm{d}\kappa/\mathrm{d}\theta = \mu(\mu+k)/k,$$

   as to be shown.
3. The canonical link is $\eta = \log\{\mu/(\mu+k)\}$.

10.3

1. $\theta = \log\lambda$ and $\kappa(\theta) = \lambda + \log\{1-\exp(-\lambda)\}$.
2. $\mathrm{d}\theta/\mathrm{d}\lambda = 1/\lambda$; $\mathrm{d}\kappa(\theta)/\mathrm{d}\lambda = 1/\{1-\exp(-\lambda)\}$, and the result follows.
3. $\mathrm{var}[y] = V(\mu) = \lambda\{1-\exp(-\lambda) - \lambda\exp(-\lambda)\}/\{1-\exp(-\lambda)\}^2$.

```
>    ### Part 4
> y <- 1:10; lambda <- 2
> p <- exp(-lambda) * lambda^y / ( (1-exp(-lambda)) * factorial(y) )
> plot(p~y, type="h", xlim=c(0, 10), xlab="Prob.", las=1, main="lambda=2")
> y1 <- 0:10; p1 <- dpois(y, lambda=lambda)
> points(p1~y, pch=19)
> legend("topright", pch=c(NA, 19), lty=c(1, NA),
    legend=c("Truncated", "Standard"))
```

10.9

```
> data(danishlc)
> danishlc$Rate <- danishlc$Cases / danishlc$Pop * 1000  # Rate per 1000
> danishlc$Age <- ordered(danishlc$Age, # Preserve age-order
   levels=c("40-54", "55-59", "60-64", "65-69", "70-74", ">74") )
> danishlc$City <- abbreviate(danishlc$City, 1)
>    ### Part 1
> dlc.bin <- glm( cbind(Cases, Pop-Cases) ~ Age,
 family=binomial, data=danishlc)
> dlc.psn <- glm( Cases ~ offset( log(Pop) ) + Age,
 family=poisson, data=danishlc)
```

The binomial and Poisson models give nearly identical results:

```
> data.frame( coef(dlc.bin), coef( dlc.psn))
> c( Df=df.residual(dlc.bin),
 Dev.Bin=deviance(dlc.bin),
 Dev.Poisson=deviance(dlc.psn) )
```

The conditions are satisfied, so the binomial and Poisson models are equivalent:

```
> max( fitted( dlc.bin) )    ### Small pi
> min( danishlc$Pop )        ### Large m
```

10.4 ***1.*** The number of politicians switching parties is a count. ***2.*** In non-election years, $\exp(1.051) = 2.86$ times more politicians switch on average. ***3.*** $z = 1.051/0.320 = 3.28$, and so $P = 0.00026$. ***4.*** Use $z = 1.645$ and then $1.051 \pm (1.645 \times 0.320)$, or 1.051 ± 0.5264.

10.6

```
>    ### Part 2
> ResDev <- c(732.74, 662.25, 649.01, 637.22)
> Dev <- abs(diff(ResDev))
> p.lrt <- round( pchisq(Dev, df=1, lower.tail=FALSE), 3)
>    ### Part 3
> beta <- c(0.238, 0.017,-0.028)
> se <- c(0.028, 0.035, 0.009)
> z <- beta/se
> p.wald <- round( 2*(1 - pnorm( abs(z) ) ), 3)
>    ### Part 5
> cbind(p.lrt, p.wald); pchisq(ResDev[4], df=614, lower.tail=FALSE)
```

1. $\log \hat{\mu} = -2.928 + 0.238C + 0.017M - 0.028M^2$. ***5.*** The residual deviance (637.22) is only slightly larger than the residual df (614). ***6. and 7.*** Write $\eta = \beta_0 + \beta_1 C + \beta_2 M + \beta_3 M^2$; solving shows the maximum occurs at $M = -\beta_1/(2\beta_2) = 0.15$. This is small (and far less than the minimum possible manipulation of one whole egg), suggesting that

manipulating the clutch-size in any way will reduce the number of offspring surviving, supporting the hypothesis.

10.11

```
> data(cervical)
> cervical$AgeNum <- rep( c(25, 35, 45, 55), 4)
> par( mfrow=c(2, 2))
>    ### Part 1
> with( cervical, {
  plot( Deaths/Wyears ~ AgeNum, type="n")
  lines(Deaths/Wyears ~ AgeNum, lty=1,
        subset=(Country==unique(Country)[1]) )
  lines(Deaths/Wyears ~ AgeNum, lty=2,
        subset=(Country==unique(Country)[2]) )
  lines(Deaths/Wyears ~ AgeNum, lty=3,
        subset=(Country==unique(Country)[3]) )
  lines(Deaths/Wyears ~ AgeNum, lty=4,
        subset=(Country==unique(Country)[4]) )
  legend("topleft", lty=1:4, legend=unique(cervical$Country) )
 })
>    ### Part 3
> cc.m0 <- glm( Deaths ~ offset(log(Wyears)) + Age + Country,
    data=cervical, family=poisson )
> plot( rstandard(cc.m0) ~ fitted(cc.m0), main="Poisson glm" )
>    ### Part 4
> cc.m0Q <- glm( Deaths ~ offset(log(Wyears)) + Age + Country,
    data=cervical, family=quasipoisson )
> plot( rstandard(cc.m0Q) ~ fitted(cc.m0Q), main="Quasi-Poisson model" )
>    ### Part 5
> cc.m0NB <- glm.nb( Deaths ~ offset(log(Wyears)) + Age + Country,
    data=cervical)
> cc.m0NB <- glm.convert(cc.m0NB)
> plot( rstandard(cc.m0NB) ~ fitted(cc.m0NB), main="Neg. bin. glm" )
```

2. To account for the exposure. ***5.*** All models seem to have a large negative outlier, but clearly the Poisson model does not accommodate the variation correctly.

10.13

```
> data(cyclones)
> par(mfrow=c(2, 2))
> scatter.smooth(cyclones$JFM, cyclones$Severe, ylim=c(0, 15))
> scatter.smooth(cyclones$AMJ, cyclones$Severe, ylim=c(0, 15))
> scatter.smooth(cyclones$JAS, cyclones$Severe, ylim=c(0, 15))
> scatter.smooth(cyclones$OND, cyclones$Severe, ylim=c(0, 15))
> par(mfrow=c(2, 2))
> scatter.smooth(cyclones$JFM, cyclones$NonSevere, ylim=c(0, 15))
> scatter.smooth(cyclones$AMJ, cyclones$NonSevere, ylim=c(0, 15))
> scatter.smooth(cyclones$JAS, cyclones$NonSevere, ylim=c(0, 15))
> scatter.smooth(cyclones$OND, cyclones$NonSevere, ylim=c(0, 15))
>    ### Best models...?
> mS <- glm(Severe~1, data=cyclones, family=poisson)
> mNS <- glm(NonSevere~1, data=cyclones, family=poisson)
```

10.15

```
> data(polyps); library(MASS); library(statmod)
>    ### Part 2
> par(mfrow=c(2, 2))
> plot( Number ~ Age, pch=ifelse(Treatment=="Drug", 1, 19), data=polyps)
>    ### Part 2
> m1 <- glm(Number ~ Age * Treatment, data=polyps, family=poisson)
> plot(qresid(m1) ~ fitted(m1)); plot(cooks.distance(m1), type="h")
> qqnorm( qresid(m1)); anova(m1, test="Chisq")
> c( deviance(m1), df.residual(m1) ) #  Massive overdispersion
>    ### Part 3
> m2 <- glm(Number ~ Age * Treatment, data=polyps, family=quasipoisson)
>    ### Part 4
> m3 <- glm.convert( glm.nb(Number ~ Age * Treatment, data=polyps) )
> anova(m2, test="F"); anova(m3, test="F")
> par(mfrow=c(1, 1))
```

10.19

```
> data(blocks)
> with(blocks,{
    m0 <- glm(Number~1,   family=poisson)
    m1 <- glm(Number~Age, family=poisson)
    coef(m1)
    anova(m1, test="Chisq")
    glm.scoretest(m0, blocks$Age)
 })
```

Solutions to Problems from Chap. 11

11.3 Differentiating the log-likelihood with respect to ϕ gives $\partial\ell/\partial\phi = -n/(2\phi) + 1/(2\phi)\sum_{i=1}^{n}(y-\hat{\mu})^2/(y\hat{\mu}^2)$; solving yields the required answer.

11.5 ***1.*** As $\mu \to \infty$, the expression in the exponent becomes $-1/(2\phi y)$, and the result follows. ***2.*** $\text{var}[y] = \phi\mu^3 \to \infty$ as $\mu \to \infty$.

```
>    ### Part 3
> y <- seq(0.00001, 8, length=500)
> dlevy <- function(y, phi){ exp(-1/(2*y*phi))/sqrt(2*pi*phi*y^3)}
> fy1 <- dlevy(y, phi=0.5)
> fy2 <- dlevy(y, phi=1)
> fy3 <- dlevy(y, phi=2)
> plot(fy3~y, type="l", xlab="y", ylab="Density")
> lines(fy2~y, lty=2)
> lines(fy1~y, lty=3)
> legend("topright", lty=1:3, legend=c("phi = 2","phi = 1","phi = 0.5"))
> abline(h=0, col="gray")
```

11.7 Note: The main-effects terms contribute 19 df also.

```
>    ### Part 1
> DiffDf <- c(16, 12, 16, 12, 12, 16, 12, 12, 9, 12)
>    ### Part 2
> phi <- 4390.9 / (1975-sum(DiffDf) - 19) # Mean deviance estimate
>    ### Part 3
> Dev <- c(5050.9, 4695.2, 4675.9, 4640.1, 4598.8, 4567.3,
          4497.1, 4462.0, 4443.4, 4420.8, 4390.9)
> DiffDev <- abs(diff(Dev))
> F <- (DiffDev/DiffDf)/phi
> ps <- pf(DiffDev, df1=DiffDf, df2=1975-sum(DiffDf) - 19,
     lower.tail=FALSE)
> ps
```

11.9

```
> data(lime)
>    ### Part 1
> lime.log <- glm( Foliage ~ Origin * log(DBH),
                family=Gamma(link="log"), data=lime)
> lime.m2  <- glm( Foliage ~ Origin * DBH,
                family=Gamma(link="log"), data=lime)
> par(mfrow=c(2, 3))
>    ### Part 2
> scatter.smooth( log(fitted(lime.log)), rstandard(lime.log),
       col="gray", lwd=2 )
> qqnorm( qresid(lime.log)); plot(cooks.distance(lime.log), type="h")
> scatter.smooth( log(fitted(lime.m2)), rstandard(lime.m2),
       col="gray", lwd=2 )
> qqnorm( qresid(lime.m2));
> plot(cooks.distance(lime.m2), type="h")
> colSums(influence.measures(lime.log)$is.inf)
> colSums(influence.measures(lime.m2)$is.inf)
```

Prefer gamma GLM with `log(DBH)`; see the plot of standardized residuals against fitted values (on constant-information scale).

11.13

```
> data(fluoro)
>    ### Part 1
> par(mfrow=c(2, 2))
> m1 <- glm(Dose~Time, family=Gamma(link="log"), data=fluoro)
> plot( rstandard(m1) ~ fitted(m1))
> qqnorm(rstandard(m1))
> plot( cooks.distance(m1), type="h")
>    ### Part 2
> plot(Dose~Time, data=fluoro)
> newT <- seq(min(fluoro$Time), max(fluoro$Time), length=100)
> new.df <- data.frame(Time=newT)
> newD <- predict(m1, newdata=new.df, se.fit=TRUE)
> tstar <- qt(0.975, df=df.residual(m1))
> m.err <- tstar*newD$se.fit
> ci.lo <- exp(newD$fit - m.err); ci.hi <- exp(newD$fit + m.err)
> lines(exp(newD$fit)~newT, lwd=2)
> lines(ci.lo~newT, lty=2)
> lines(ci.hi~newT, lty=2)
```

P-values are similar.

11.15

```
> data(lungcap)
> lungcap$Smoke <- factor(lungcap$Smoke, labels=c("NonSmoker", "Smoker"))
>    ### Part 1
> par(mfrow=c(3, 3))
> plot( FEV~Age, data=lungcap)
> plot(FEV~Smoke, data=lungcap)
> plot( FEV~Ht,  data=lungcap)
> plot(FEV~Gender, data=lungcap)
> interaction.plot(    lungcap$Smoke,   lungcap$Gender, lungcap$FEV)
> interaction.plot(cut(lungcap$Age, 3), lungcap$Gender, lungcap$FEV)
> interaction.plot(cut(lungcap$Ht,  3), lungcap$Gender, lungcap$FEV)
> interaction.plot(cut(lungcap$Age, 2), lungcap$Smoke,  lungcap$FEV)
> interaction.plot(cut(lungcap$Ht,  2), lungcap$Smoke,  lungcap$FEV)
>    ### Part 2
> m1 <- glm(FEV~Age*Ht*Gender*Smoke, family=Gamma(link="log"),
    data=lungcap)
> anova(m1, test="F")
> m2 <- glm(FEV~Age*Ht*Gender+Smoke, family=Gamma(link="log"),
    data=lungcap)
> anova(m2, test="F")
> par(mfrow=c(2, 4))
> plot(m1); plot(m2)
> colSums(influence.measures(m1)$is.inf)
> colSums(influence.measures(m2)$is.inf) # Prefer m2
```

11.17

```
> data(leukwbc); leukwbc$WBCx <- (leukwbc$WBC/1000)
> par( mfrow=c(1, 2))
>    ### Part 1
> plot( Time ~ WBCx, data = leukwbc, las=1,
       pch=ifelse(leukwbc$AG==1, 3, 1))
> legend("topright", c("AG positive","AG negative"), pch=c(3, 1) )
>    ### Part 2
> plot( Time ~ log(WBCx), data = leukwbc, las=1,
      pch=ifelse(leukwbc$AG==1, 3, 1))
> legend("topright", c("AG positive","AG negative"), pch=c(3, 1) )
>    ### Part 3
> m1 <-  glm( Time ~ AG * log10(WBCx), family=Gamma(link="log"),
            data=leukwbc)
> anova(m1, test="F")
>    ### Part 4
> m2 <-  update(m1, . ~ AG + log10(WBCx))
> anova(m2, test="F")
>    ### Part 5
> newW <- seq( min(leukwbc$WBCx), max(leukwbc$WBCx), length=100)
> newTP <- predict( m2, newdata=data.frame(WBCx=newW, AG=1),
                  type="response")
> newTN <- predict( m2, newdata=data.frame(WBCx=newW, AG=2),
                  type="response")
> par( mfrow=c(1, 2))
> plot( Time ~ WBCx, data = leukwbc, las=1,
    pch=ifelse(leukwbc$AG==1, 3, 1))
```

```
> lines( newTP ~ (newW), lty=1)
> lines( newTN ~ (newW), lty=2)
> legend("topright", c("AG +ive","AG -ive"), pch=c(3, 1), lty=c(1, 2))
> plot( Time ~ log10(WBCx), data = leukwbc, las=1,
    pch=ifelse(leukwbc$AG==1,3, 1))
> lines( newTP ~ log10(newW), lty=1)
> lines( newTN ~ log10(newW), lty=2)
> legend("topright", c("AG +ive","AG -ive"), pch=c(3,1), lty=c(1,2))
>    ### Part 6
> summary(m2)$dispersion # Exponential seems reasonable
```

11.19

```
> data(blocks)
>    ### Part 1
>    ### Trial and Age (or interactions) are not significant
> glm1 <- glm(Time~Shape, data=blocks, family=Gamma(link=log))
>    ### Part 2
> glm2 <- update(glm1, family=inverse.gaussian(link=log))
>    ### Part 3
> plot(glm1)
> plot(glm2)
> summary(glm2)
> c(extractAIC(glm1), extractAIC(glm2))
```

11.22

```
> data(fishfood)
> m1 <- lm(FoodCon ~ log(MaxWt) + log(Temp) + log(AR) + Food,
       data=fishfood)
> glm1 <- glm( FoodCon ~ log(MaxWt) + log(Temp) + log(AR) + Food,
      data=fishfood, family=Gamma(link="log"))
> anova(m1)
> anova(glm1, test="F")
> summary(glm1)
> par(mfrow=c(2, 4))
> plot(m1); plot(glm1)
> c(AIC(m1), AIC(glm1))
```

Solutions to Problems from Chap. 12

In this chapter, we do not explicitly load the `tweedie` package [1] each time it is needed.

```
> library(tweedie)
```

12.1 Perform the indicated integrations.

12.7 Proceed as in Sect. 5.8 (p. 232).

12.11

```
> data(perm); perm$Day <- factor(perm$Day)
>    ### Part 1
> out <- tweedie.profile( Perm ~ factor(Mach)+factor(Day),
          do.plot=TRUE, data=perm)
> out$p.max; out$ci # inverse Gaussian seems appropriate
```

12.13

```
> data(motorins1); motorins1$Km <- factor(motorins1$Kilometres)
> motorins1$Bns <- factor(motorins1$Bonus)
> motorins1$Make <- factor(motorins1$Make)
> out <- tweedie.profile(Payment ~ Km * Bns, data=motorins1, do.plot=TRUE,
    xi.vec=seq(1.6, 1.95, by=0.05)); xi <- out$xi.max; xi; out$ci
> ins.m1A <- glm(Payment ~ Km + Bns + Make + Km:Bns + Km:Make + Bns:Make,
          data = motorins1, family=tweedie(var.power=xi, link.power=0) )
> ins.m1B <- glm(Payment ~ Km + Bns + Make + Km:Bns + Bns:Make + Km:Make,
          data = motorins1, family=tweedie(var.power=xi, link.power=0) )
> ins.m1C <- glm(Payment ~ Km + Bns + Make + Km:Make + Bns:Make + Km:Bns,
          data = motorins1, family=tweedie(var.power=xi, link.power=0) )
> ins.m1D <- glm(Payment ~ Km + Bns + Make + Bns:Make + Km:Bns + Km:Make,
          data = motorins1, family=tweedie(var.power=xi, link.power=0) )
> anova( ins.m1A, test="F")
```

12.17

```
> data(toothbrush)
> toothbrush$Diff <- with(toothbrush, Before - After)
> with(toothbrush, interaction.plot(Sex, Toothbrush, Diff))
> out <- tweedie.profile(Diff~Sex*Toothbrush,
     xi.vec=seq(1.05, 1.6, length=15),
         data=toothbrush, do.plot=TRUE); xi <- round(out$xi.max, 2)
> m1 <- glm(Diff~Sex*Toothbrush, data=toothbrush,
                  family=tweedie(link.power=0, var.power=xi))
> anova(m1, test="F")
> summary(m1)
```

Solutions to Problems from Chap. 13

13.1

```
> data(satiswt)
>    ### Part 2
> m1 <- glm( Counts~Gender+WishWt+Matur, family=poisson, data=satiswt)
> drop1( glm( Counts~Gender*WishWt*Matur, family=poisson,
    data=satiswt), test="Chisq") # Need full model!
```

13.3

```
> data(boric)
> boric$Prob <- boric$Dead/boric$Implants
> plot( Prob~Dose, data=boric)
> m1 <- glm(Prob~Dose, weights=Implants, data=boric, family=binomial)
> m2 <- update(m1, .~log(Dose+1))
> newD <- seq(min(boric$Dose), max(boric$Dose), length=100)
> newP1 <- predict( m1, type="response", newdata=data.frame(Dose=newD))
> newP2 <- predict( m2, type="response", newdata=data.frame(Dose=newD))
> lines(newP1~newD, lwd=2, lty=1)
> lines(newP2~newD, lwd=2, lty=2)
> infl1 <- max( cooks.distance(m1))
> infl2 <- max( cooks.distance(m1))
> c(infl1, infl2)
```

13.5 The delivery times are strictly positive values, so a gamma or inverse Gaussian EDM may be appropriate for modelling the random component. Combining the systematic and random components, a possible model for the data is:

$$\begin{cases} y \sim \text{Gamma}(\mu; \phi) & \text{(random component)} \\ \mu = \beta_0 + \beta_1 x & \text{(systematic component).} \end{cases} \tag{B.1}$$

```
> data(sdrink)
> model.sdrink <- glm( Time ~ Cases + Distance, data=sdrink,
                       family=Gamma(link="identity") )
> model.sdrink.iG <- glm( Time ~ Cases + Distance, data=sdrink,
                       family=inverse.gaussian(link="identity") )
> printCoefmat(coef(summary(model.sdrink.iG)))
> plot( rstandard(model.sdrink) ~ log( fitted(model.sdrink) ),
    main="Gamma glm",
    ylab="Standardized residual", las=1, pch=19  )
> plot( cooks.distance(model.sdrink), type="h",
    ylab="Cook's distance", las=1)
> qqnorm( qresid(model.sdrink), las=1)
> qqline( qresid(model.sdrink))
> plot( rstandard(model.sdrink.iG) ~ log( fitted(model.sdrink.iG) ),
    main="Inverse Gaussian glm",
    ylab="Standardized residual", las=1, pch=19  )
> plot( cooks.distance(model.sdrink.iG), type="h",
    ylab="Cook's distance", las=1)
> qqnorm( qresid(model.sdrink.iG), las=1)
> qqline( qresid(model.sdrink.iG))
```

While neither model looks particularly poor, the gamma GLM is probably more suitable.

```
> c( Gamma=AIC( model.sdrink), iG=AIC(model.sdrink.iG))
> c( Gamma=BIC( model.sdrink), iG=BIC(model.sdrink.iG))
```

References

[1] Dunn, P.K.: tweedie: Tweedie exponential family models (2017). URL https://CRAN.R-project.org/package=tweedie. R package version 2.3.0
[2] Dunn, P.K., Smyth, G.K.: GLMsData: Generalized linear model data sets (2017). URL https://CRAN.R-project.org/package=GLMsData. R package version 1.0.0
[3] Renkl, A., Atkinson, R.K., Maier, U.H., Staley, R.: From example study to problem solving: Smooth transitions help learning. The Journal of Experimental Education **70**(4), 293–315 (2002)

Index: Data sets

Data is not information, Information is not knowledge, Knowledge is not understanding, Understanding is not wisdom.
(Attributed to Cliff Stoll and Gary Schubert in M. R. Keeler. Nothing to hide: Privacy in the 21st century. *iUniverse, 2006.)*

P. K. Dunn, G. K. Smyth, *Generalized Linear Models with Examples in R*, Springer Texts in Statistics, https://doi.org/10.1007/978-1-4419-0118-7

Index: R commands

Instruction ends in the schoolroom, but education ends only with life.
(Rev. F. W. Robertson. Sermons preached at Trinity Chapel, Brighton. *Bernhard Tauchnitz, 1866.)*

P. K. Dunn, G. K. Smyth, *Generalized Linear Models with Examples in R*, Springer Texts in Statistics, https://doi.org/10.1007/978-1-4419-0118-7

Index: General topics

Knowledge is of two kinds. We know a subject ourselves, or we know where we can find information upon it. (Attributed to Samuel Johnson in J. Boswell and R. W. Chapman. Life of Johnson. *Oxford World's Classics. Oxford University press, third edition, 1988.)*

P. K. Dunn, G. K. Smyth, *Generalized Linear Models with Examples in R*, Springer Texts in Statistics, https://doi.org/10.1007/978-1-4419-0118-7