Guangyuan Gao

Bayesian Claims Reserving Methods in Non-life Insurance with Stan

An Introduction

Springer

Bayesian Claims Reserving Methods in Non-life Insurance with Stan

Guangyuan Gao

Bayesian Claims Reserving Methods in Non-life Insurance with Stan

An Introduction

Springer

Guangyuan Gao
School of Statistics
Renmin University of China
Beijing, China

ISBN 978-981-13-3608-9   ISBN 978-981-13-3609-6 (eBook)
https://doi.org/10.1007/978-981-13-3609-6

Library of Congress Control Number: 2018963287

© Springer Nature Singapore Pte Ltd. 2018
This work is subject to copyright. All rights are reserved by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed.
The use of general descriptive names, registered names, trademarks, service marks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.
The publisher, the authors and the editors are safe to assume that the advice and information in this book are believed to be true and accurate at the date of publication. Neither the publisher nor the authors or the editors give a warranty, express or implied, with respect to the material contained herein or for any errors or omissions that may have been made. The publisher remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

This Springer imprint is published by the registered company Springer Nature Singapore Pte Ltd.
The registered company address is: 152 Beach Road, #21-01/04 Gateway East, Singapore 189721, Singapore

Preface

Bayesian models are very popular in non-life claims reserving. This monograph provides a review of Bayesian claims reserving models and their underlying Bayesian inference theory. It investigates three types of claims reserving models in Bayesian framework: chain ladder models, basis expansion models involving tail factor, and multivariate copula models. One of the core techniques in Bayesian modeling is inferential methods. This monograph largely relies on Stan, a specialized software environment which applies Hamiltonian Monte Carlo method and variational Bayes. This monograph has the following three distinguishing features:

- It has a thorough review of various aspects of Bayesian statistics and relates them to claims reserving problems.
- It addresses three important points in claims reserving: tail development, stochastic version of payments per claim incurred method, and aggregation of liabilities from correlated portfolios.
- It provides explicit Stan code for non-life insurance claims reserving.

Beijing, China
September 2018

Guangyuan Gao

Acknowledgements

I am very grateful to Borek Puza, Richard Cumpston, Hanlin Shang, Tim Higgins, Bronwen Whiting, Steven Roberts, and Xu Shi at the Australian National University. The discussion with them helps improve the quality of this monograph significantly. I am also very grateful to Chong It Tan and Yanlin Shi at the Macquarie University, and Shengwang Meng at the Renmin University of China. They provide constructive comments. I would like to thank the Research School of Finance, Actuarial Studies and Statistics at the Australian National University for the financial support. I also would like to thank the School of Statistics at the Renmin University of China for the research funds including National Social Science Fund of China (Grant No. 16ZDA052), MOE National Key Research Bases for Humanities and Social Sciences (Grant No. 16JJD910001), and the funds for building world-class universities (disciplines).

Contents

Chapter 1
Introduction

Abstract This chapter briefly reviews Bayesian statistics, Markov chain Monte Carlo methods, and non-life insurance claims reserving methods. Some of the most influential literature are listed in this chapter. Two Bayesian inferential engines, BUGS and Stan, are introduced. At the end the monograph structure is given and the general notation is introduced.

1.1 Bayesian Inference and MCMC

The foundation of Bayesian data analysis is Bayes' theorem, which derives from Bayes (1763). Although Bayes' theorem is very useful in principle, Bayesian statistics developed more slowly in the 18th and 19th centuries than in the 20th century. Statistical analysis based on Bayes' theorem was often daunting because of the extensive calculations, such as numerical integrations, required. Perhaps the most significant advances to Bayesian statistics in the period just after Bayes' death were made by Laplace (1785, 1810).

In the 20th century, the development of Bayesian statistics continued, characterised by Jeffreys (1961), Lindley (1965) and Box and Tiao (1973). At the time these monographs were written, computer simulation methods were much less convenient than they are now, so they restricted their attention to conjugate families and devoted much effort to deriving analytic forms of marginal posterior densities.

Thanks to advances in computing, millions of calculations can now be performed easily in a single second. This removes the prohibitive computational burden involved in much Bayesian data analysis. At the same time, computer-intensive sampling methods have revolutionized statistical computing and hence the application of Bayesian methods. They have profoundly impacted the practice of Bayesian statistics by allowing intricate models to be posited and used in disciplines as diverse as biostatistics and economics.

Bayesian inference

Compared with the frequentist approach, the Bayesian paradigm has the advantages of intuitive interpretation of confidence interval, fully defined predictive distributions and a formal mathematical way to incorporate the expert's prior knowledge of the

© Springer Nature Singapore Pte Ltd. 2018

G. Gao, *Bayesian Claims Reserving Methods in Non-life Insurance with Stan*,
https://doi.org/10.1007/978-981-13-3609-6_1

parameters. For example, a Bayesian interval for an unknown quantity of interest can be directly regarded as having a high probability of containing the unknown quantity. In contrast, a frequentist confidence interval may strictly be interpreted only in relation to a sequence of similar inferences that might be made in repeated practice.

The central feature of Bayesian inference, the direct quantification of uncertainty, means that there is no impediment in principle to fitting models with many parameters and complicated multi-layered probability specifications. The freedom to set up complex models arises in large part from the fact that the Bayesian paradigm provides a conceptually simple method for dealing with multiple parameters. In practice, the problems that do exist are ones of setting up and computing with such large models and we devote a large part of this monograph to recently developed, and still developing, techniques for handling these modelling and computational challenges.

Markov chain Monte Carlo methods

Among Bayesian computational tools, Markov chain Monte Carlo (MCMC) methods (Metropolis et al. 1953; Hastings 1970) are the most popular. The Metropolis algorithm (Metropolis et al. 1953) was first used to simulate a liquid in equilibrium with its gas phase. Hastings (1970) generalized the Metropolis algorithm, and simulations following his scheme are said to use the Metropolis-Hastings (M-H) algorithm. A special case of the Metropolis-Hastings algorithm was introduced by Geman and Geman (1984). Simulations following their scheme are said to use the Gibbs sampler. Gelfand and Smith (1990) made the wider Bayesian community aware of the Gibbs sampler, which up to that time had been known only in the spatial statistics community. It was rapidly realized that most Bayesian inference could be done by MCMC. Green (1995) generalized the M-H algorithm, as much as it can be generalized.

In the context of a Bayesian model, MCMC methods can be used to generate a Markov chain whose stationary distribution is the posterior distribution of the quantity of interest. Statisticians and computer scientists have developed software packages such as BUGS (Lunn et al. 2012) and Stan (Gelman et al. 2014) to implement MCMC methods for user-defined Bayesian models. Hence, practitioners from other areas without much knowledge of MCMC can create Bayesian models and perform Bayesian inference with relative ease.

The BUGS project started in 1989 at the MRC Biostatistics Unit in Cambridge, parallel to and independent of the classic MCMC work of Gelfand and Smith (1990). Nowadays there are two versions of BUGS: WinBUGS and OpenBUGS. WinBUGS is an older version and will not be further developed. OpenBUGS represents “the future of the BUGS project”.

Stan is a relatively new computing environment which applies Hamiltonian Monte Carlo (Duane et al. 1987; Neal 1994) and variational Bayes (Jordan et al. 1999). Stan was first introduced in Gelman et al. (2014). The BUGS examples (volume 1–3) are translated into Stan as shown in the Stan GitHub Wiki. In this monograph, we largely rely on Stan for doing Bayesian inference.

1.2 Bayesian Claims Reserving Methods

Recent attempts to apply enterprise risk management (ERM) principles to insurance have placed a high degree of importance on quantifying the uncertainty in the various necessary estimates, using stochastic models. For general insurers, the most important liability is the reserve for unpaid claims. Over the years a number of stochastic models have been developed to address this problem (Taylor 2000; Wüthrich and Merz 2008, 2015).

In many countries, loss reserves are the single largest liability on the insurance industry's balance sheet. The delayed and stochastic nature of the timing and amount of loss payments makes the insurance industry unique, and it effectively dominates or defines much of the financial management and risk and opportunity management of an insurance company. For example, insurers are typically hesitant to utilize a significant amount of debt in their capital structure, as their capital is already leveraged by reserves. Also, the characteristics of unpaid loss liabilities heavily influence insurer investment policy.

The claims reserving problem is not only about the expected value of claims liability, but also the distribution of claims liability (Taylor 2000; Wüthrich and Merz 2008). The predictive distribution of unpaid claims is vital for risk management, risk capital allocation and meeting the requirements of Solvency II (Christiansen and Niemeyer 2014) etc.

A feature of most loss reserve models is that they are complex, in the sense that they have a relatively large number of parameters. It takes a fair amount of effort to derive a formula for the predictive distribution of future claims from a complex model with many parameters (Mack 1993, 1999, 2008). Taking advantage of ever-increasing computer speeds, England and Verrall (2002) pass the work on to computers using a bootstrapping methodology with the over-dispersed Poisson model. With the relatively recent introduction of MCMC methods (Gelfand and Smith 1990), complex Bayesian stochastic loss reserve models are now practical in the current computing environment.

Bayesian inference can often be viewed in terms of credibility theory, where the posterior distribution is a weighted average of the prior and likelihood. The idea of credibility was widely used in actuarial science a long time ago (Whitney 1918; Longley-Cook 1962; Bühlmann 1967). Often reasonable judgements by experienced actuaries can override the signals in unstable data. Also, an insurance company may not have enough "direct" data available to do a "credible" analysis. Bayesian credibility theory provides a coherent framework for combining the "direct" data with either subjective judgements or collateral data so as to produce a useful "credibility estimate" (Mayerson 1964).

Setting a median reserve will lead to a half chance of insolvency, which definitely violates the policyholders' interest and will not meet the regulators' requirements. The insurers care more about the tail behaviour of future claims. Normally they hold the economic capital defined as a remote quantile of future claims distribution so as to ensure a low probability of insolvency.

Furthermore, the insurers may have several lines of business, such as automobile, commercial general liability, commercial property, homeowners etc. It is good for such multi-line insurers to know not only which lines have higher net profit but also which are riskier so they can compare the risk-adjusted return between lines. The risk cannot be characterised just by standard errors, since the claims amounts are always heavy-tailed. We are more interested in the tail-based risk measures such as value-at-risk (Brehm et al. 2007), which can be estimated from the predictive distribution of future claims.

Each line of insurance is typically modelled with its own parameters, but ultimately the distribution of the sum of the lines is needed. To get the distribution of the sum, the dependencies among the lines must be taken into account. For example, if there are catastrophic events, all of the property damage lines could be hit at the same time. Legislation changes could hit all of the liability lines. When there is the possibility of correlated large losses across lines, the distribution of the sum of the lines gets more probability in the right tail.

Unfortunately, even though the univariate distribution of the sum is the core requirement, with dependent losses the multivariate distribution of the individual lines is necessary to obtain the distribution of the sum. That quickly leads to the realm of copulas (Joe 2014), which provide a convenient way to combine individual distributions into a single multivariate distribution.

1.3 Monograph Structure

Two chapters of this monograph focus on Bayesian methodology and three chapters on the application of Bayesian methods to claims reserving in non-life insurance.

In Chap. 2, we provide a broad overview of Bayesian inference, making comparisons with the frequentist approach where necessary. Model assessment and selection in the Bayesian framework are reviewed. Some toy examples are used to illustrate the main concepts.

In Chap. 3, Bayesian computational methods are reviewed. These computational methods will be employed later in the monograph. As we mentioned before, the popularity of Bayesian modelling is largely due to the development of Bayesian computational methods and advances in computing. A knowledge of Bayesian computational methods lets us feel more confident with using a “black box” such as OpenBUGS or Stan. Moreover, with the computational methods at our disposal, we may develop our own algorithm for some special models which cannot be solved by any available package. To end this chapter, we do a full Bayesian analysis of a hierarchical model for biology data in Gelfand et al. (1990). This model has a connection with random effects models discussed in Chap. 4.

The next three chapters constitute an application of Bayesian methods to a data set from WorkSafe Victoria which provides the compulsory workers compensation insurance for all companies in Victoria except the self-insured ones. The data set includes claims histories of various benefit types from June 1987 to June 2012.

In Chap. 4, the parametric Bayesian models for the run-off triangle are investigated. We first review the time-honoured Mack's chain ladder models (Mack 1993, 1999) and Bornhuetter-Ferguson models (Bornhuetter and Ferguson 1972), which have been widely used in actuarial science for decades. Then the more recent Bayesian chain ladder models with an over-dispersed Poisson error structure (England et al. 2012) are studied. Reversible jump Markov chain Monte Carlo (RJMCMC) is discussed in this chapter for the purpose of dealing with the tail development component in the models. Finally, we apply the models discussed above to estimate the claims liabilities for the weekly benefit and the doctor benefit in WorkSafe Victoria. For the doctor benefit, we propose a compound model as a stochastic version of the payments per claim incurred (PPCI) method.

Chapter 5 investigates Bayesian basis expansion models with shrinkage priors and their applications to claims reserving. We first summarize some aspects of basis expansion models (Hastie et al. 2009). Among all the basis expansion models, the Bayesian natural cubic spline basis expansion model with shrinkage priors is our favourite. Two simulated examples are studied to illustrate two advantages of this model: the shorter computational time and the better tail extrapolation. The second simulated example is designed to mimic the mechanism of claims payments. Finally, we reanalyze the doctor benefit using the proposed Bayesian basis expansion model and compare the results with those in Chap. 4 and the PwC report (Simpson and McCourt 2012).

In Chap. 6, Bayesian copula models are used to aggregate the estimated claims liabilities from two correlated run-off triangles. In the first section, we review Sklar's theorem, several parametric copulas, and inferential methods. A simulated example is used to demonstrate the inference functions for margins (IFM) method (Joe and Xu 1996). In the second section, we discuss the usefulness of copulas in modelling risk dependence. Ignorance of risk dependence does not affect the aggregated mean too much, but it will affect the more interesting tail-based risk measures significantly. In the third section, we aggregate two correlated benefits in WorkSafe Victoria: the doctor benefit and the hospital benefit. The marginal regression for each benefit is the same as in Chap. 5.

Chapter 7 provides a summary of the monograph and discusses limitations and further research topics. It includes remarks about the three most useful stochastic claims reserving models in the monograph and suggests alternative Bayesian modelling procedures.

There are two appendices. Appendix A supplies the technical complements to support the examples in Chaps. 2 and 3. Appendix B lists some Bayesian computational methods not included in Chap. 3 and relevant proofs.

In each chapter, all figures and tables appear together at the end, in that order.

1.4 The General Notation Used

By default, vectors are column vectors. If we write $\theta = (\alpha, \beta)$, we mean θ is a column vector with two elements. A lower case letter is a column vector or a scalar. A matrix is denoted by a bold upper case letter.

Data

Bold and lower case Roman letters represent the observed data vector. For example, $\boldsymbol{y}$ might be an n-vector of observed response values. A bold and upper case Roman letter could represent a design matrix. For example, $\boldsymbol{X}$ might represent an $n \times p$ matrix of observed predictors.

Parameters

Non-bold and lower case Greek letters represent the parameters. For example, θ can be a vector containing p parameters. Bold and upper case Greek letters might represent a covariance matrix. $\boldsymbol{\Sigma}$ can be a $p \times p$ covariance matrix.

Functions

Unless stated otherwise, all the probability density (or mass) functions are represented by p and all the cumulative distribution functions are represented by F. Other generic functions are typically represented by f, g, h, π.

Conditional distributions

The distribution of data is conditional on the parameters and the prior of parameters is conditional on the hyperparameters. For example, a normal-normal-gamma model with unknown mean and variance is formally written as follows:

$$\begin{aligned} y|\mu, \sigma^2 &\sim \mathrm{N}(\mu, \sigma^2) \\ \mu|\sigma^2 &\sim \mathrm{N}(\mu_0, \sigma_0^2) \\ \sigma^2 &\sim \text{Inv-Gamma}(\alpha, \beta). \end{aligned}$$

For compactness, we will typically assume an implicit conditioning on the parameters going down the page. For example the normal-normal-gamma model above could also be written as follows:

$$\begin{aligned} y &\sim \mathrm{N}(\mu, \sigma^2) \\ \mu &\sim \mathrm{N}(\mu_0, \sigma_0^2) \\ \sigma^2 &\sim \text{Inv-Gamma}(\alpha, \beta). \end{aligned}$$

For the posterior distributions, we always include the conditioning parts to emphasize the meaning of "posterior". For example, the posterior distribution of μ is denoted by $p(\mu|\boldsymbol{y})$, the full conditional posterior distribution of μ is denoted by $p(\mu|\boldsymbol{y}, \sigma)$ or $p(\mu|\cdot)$, and the posterior predictive distribution is denoted by $p(y'|\boldsymbol{y})$.

References

Bayes, T. (1763). An essay towards solving a problem in the doctrine of chances. *Philosophical Transactions of the Royal Society,* 330–418.

Bornhuetter, R. L., & Ferguson, R. E. (1972). The actuary and IBNR. *Proceedings of the Casualty Actuarial Society*, *59*, 181–195.

Box, G. E., & Tiao, G. C. (1973). *Bayesian inference in statistical analysis*. New York: Wiley Classics.

Brehm, P. J., Gluck, S., Kreps, R., Major, J., Mango, D., & Shaw, R., et al. (2007). *Enterprise risk analysis for property and liability insurance companies: A practical guide to standard models and emerging solutions*. New York: Guy Carpenter & Company.

Bühlmann, H. (1967). Experience rating and credibility. *ASTIN Bulletin*, *4*, 199–207.

Christiansen, M. C., & Niemeyer, A. (2014). The fundamental definition of the solvency capital requirement in Solvency II. *ASTIN Bulletin*, *44*, 501–533.

Duane, S., Kennedy, A. D., Pendleton, B. J., & Roweth, D. (1987). Hybrid Monte Carlo. *Physics Letters B*, *195*, 216–222.

England, P. D., & Verrall, R. J. (2002). Stochastic claims reserving in general insurance. *British Actuarial Journal*, *8*, 443–518.

England, P. D., Verrall, R. J., & Wüthrich, M. V. (2012). Bayesian over-dispersed poisson model and the Bornhuetter-Ferguson claims reserving method. *Annals of Actuarial Science*, *6*, 258–283.

Gelfand, A. E., & Smith, A. F. M. (1990). Sampling-based approaches to calculating marginal densities. *Journal of the American Statistical Association*, *85*, 398–409.

Gelfand, A. E., Hills, S. E., Racinepoon, A., & Smith, A. F. M. (1990). Illustration of Bayesian-inference in normal data models using Gibbs sampling. *Journal of the American Statistical Association*, *85*, 972–985.

Gelman, A., Carlin, J. B., Stern, H. S., & Rubin, D. B. (2014). *Bayesian data analysis* (3rd ed.). Boca Raton: Chapman & Hall.

Geman, S., & Geman, D. (1984). Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *6*, 721–741.

Green, P. J. (1995). Reversible jump Markov chain Monte Carlo computation and Bayesian model determination. *Biometrika*, *82*, 711–732.

Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The elements of statistical learning: Data mining, inference, and prediction* (2nd ed.). Springer, New York.

Hastings, W. K. (1970). Monte Carlo sampling methods using Markov chains and their applications. *Biometrika*, *57*, 97–109.

Jeffreys, H. (1961). *Theory of probability* (3rd ed.). London: Oxford University Press.

Joe, H., & Xu, J. J. (1996). The estimation method of inference functions for margins for multivariate models. http://hdl.handle.net/2429/57078.

Joe, H. (2014). *Dependence modeling with copulas*. New York: Chapman & Hall.

Jordan, M. I., Ghahramani, Z., Jaakkola, T. S., & Saul, L. K. (1999). An introduction to variational methods for graphical models. *Machine Learning*, *37*, 183–233.

Laplace, P. S. (1785). Memoire sur les approximations des formules qui sont fonctions de tres grands nombres. In *Memoires de l'Academie Royale des Sciences*.

Laplace, P. S. (1810). Memoire sur les approximations des formules qui sont fonctions de tres grands nombres, et sur leur application aux probabilites. In *Memoires de l'Academie des Science de Paris*.

Lindley, D. V. (1965). *Introduction to probability and statistics from Bayesian viewpoint* (Vol. 2). Cambridge: Cambridge University Press.

Longley-Cook, L. H. (1962). An introduction to credibility theory. *Proceedings of the Casualty Actuarial Society*, *49*, 194–221.

Lunn, D., Jackson, C., Best, N., Thomas, A., & Spiegelhalter, D. (2012). *The BUGS book: A practical introduction to Bayesian analysis*. Boca Raton: Chapman & Hall.

Mack, T. (1993). Distribution-free calculation of the standard error of chain ladder reserve estimates. *ASTIN Bulletin*, *23*, 213–225.

Mack, T. (1999). The standard error of chain-ladder reserve estimates, recursive calculation and inclusion of a tail factor. *ASTIN Bulletin*, *29*, 361–366.

Mack, T. (2008). The prediction error of Bornhuetter-Ferguson. *ASTIN Bulletin*, *38*, 87.

Mayerson, A. L. (1964). A Bayesian view of credibility. *Proceedings of the Casualty Actuarial Society*, *51*, 7–23.

Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H., & Teller, E. (1953). Equation of state calculations by fast computing machines. *Journal of Chemical Physics*, *21*, 1087–1092.

Neal, R. M. (1994). An improved acceptance procedure for the hybrid Monte Carlo algorithm. *Journal of Computational Physics*, *111*, 194–203.

Simpson, L., & McCourt, P. (2012). *Worksafe Victoria actuarial valuation of outstanding claims liability for the scheme as at 30 June 2012*, Technical report. PricewaterhouseCoopers Actuarial Pty Ltd.

Taylor, G. (2000). *Loss reserving: An actuarial perspective*. Boston: Kluwer Academic Publishers.

Whitney, A. W. (1918). The theory of experience rating. *Proceedings of the Casualty Actuarial Society*, *4*, 274–292.

Wüthrich, M. V., & Merz, M. (2008). *Stochastic claims reserving methods in insurance*. Chichester: Wiley & Sons.

Wüthrich, M. V., & Merz, M. (2015). Stochastic claims reserving manual: Advances in dynamic modelling. *SSRN*, ID 2649057.

Chapter 2
Bayesian Fundamentals

Abstract Bayesian statistics is a field of study with a long history (Bayes 1763). It has the features of straightforward interpretation and simple underlying theory, at least in principle. Analogous to the maximum likelihood estimates and confidence intervals in the frequentist framework, we have point estimates and interval estimates based on posterior distributions in the Bayesian framework. We also have similar diagnostic tools for model assessment and selections such as residual plots and information criteria. In Sect. 2.1, we review Bayesian inference including the posterior distribution, the posterior predictive distribution and the associated point estimates and interval estimates. We also summarize the usefulness of different priors and state the asymptotic normality of the posterior distribution for large samples. In Sect. 2.2, Bayesian model assessment and selections are discussed. For the model assessment, the posterior predictive p-value is an alternative to the frequentist p-value. For model selection, we turn to the several information criteria including DIC, WAIC and LOO cross-validation.

2.1 Bayesian Inference

In contrast to frequentist statistics, where parameters are treated as unknown constants, Bayesian statistics treats parameters as random variables with specified prior distributions that reflect prior knowledge (information and subjective beliefs) about the parameters before the observation of data. Given the observed data, the prior distribution of the parameters is updated to the posterior distribution from which Bayesian inference is made. In the following, the model with a single parameter is considered first, and then extensions are made to the multi-parameter case.

2.1.1 The Single-Parameter Case

Denote an observed sample of size n as $\boldsymbol{y} = (y_1, y_2, \dots, y_n)$, the parameter as θ (assumed to be a scalar), the *prior* density function of θ as $p(\theta)$, the parameter space as Θ, the *likelihood* function (sometimes called *sampling distribution*) as $p(\boldsymbol{y}|\theta)$,

© Springer Nature Singapore Pte Ltd. 2018

G. Gao, *Bayesian Claims Reserving Methods in Non-life Insurance with Stan*,
https://doi.org/10.1007/978-981-13-3609-6_2

and the *posterior* density function of θ as $p(\theta|\boldsymbol{y})$. According to *Bayes' theorem*, the three functions $p(\theta|\boldsymbol{y})$, $p(\boldsymbol{y}|\theta)$ and $p(\theta)$ have the following relationship:

$$p(\theta|\boldsymbol{y}) = \frac{p(\theta, \boldsymbol{y})}{p(\boldsymbol{y})} = \frac{p(\boldsymbol{y}|\theta)p(\theta)}{\int_{\Theta} p(\boldsymbol{y}|\theta)p(\theta)d\theta} \propto p(\boldsymbol{y}|\theta)p(\theta), \tag{2.1}$$

where $p(\theta, \boldsymbol{y})$ is the *unconditional joint density function* of parameters and observations, and $p(\boldsymbol{y})$ is the unconditional density function (sometimes called *marginal distribution*) of $\boldsymbol{y}$ which averages the likelihood function over the prior.

An important concept associated with the posterior distribution is *conjugacy*. If the prior and posterior distributions are in the same family, we call them conjugate distributions and the prior is called a *conjugate prior* for the likelihood. We will see in Example 2.1 that the Beta distribution is the conjugate prior for the Bernoulli likelihood.

An aim of frequentist inference is to seek the "best" estimates of fixed unknown parameters; for Bayesian statistics, the counterpart aim is to seek the "exact" distribution for parameters and Eq. (2.1) has realized this aim.

2.1.1.1 Point Estimation

The fundamental assumption of Bayesian statistics is that parameters are random variables, but we are still eager to find a single value or an interval to summarize the posterior distribution in Eq. (2.1). Intuitively, we want to use the mean, median or mode of the posterior distribution to indicate an estimate of the parameter. We define the posterior mean of θ as

$$\hat{\theta} := \mathbb{E}(\theta|\boldsymbol{y}) = \int_{\Theta} \theta p(\theta|\boldsymbol{y})d\theta,$$

where Θ is the domain of θ determined by the prior $p(\theta)$. The posterior median of θ is defined as

$$\ddot{\theta} := \text{median}(\theta|\boldsymbol{y}) = \{t : \Pr(\theta \geq t|\boldsymbol{y}) \geq 0.5 \text{ and } \Pr(\theta \leq t|\boldsymbol{y}) \geq 0.5\}.$$

The posterior mode of θ is defined as

$$\tilde{\theta} := \text{mode}(\theta|\boldsymbol{y}) = \underset{\theta \in \Theta}{\text{argmax}}\, p(\theta|\boldsymbol{y}).$$

2.1.1.2 Interval Estimation

An interval covering the most likely values is called the *highest posterior density region* (HPDR). It is defined as

$$\text{HPDR}(\theta|\boldsymbol{y}) := \text{the shortest interval in } \mathscr{S},$$

where

$$\mathscr{S} = \{S : \Pr(\theta \in S|\boldsymbol{y}) \geq 1 - \alpha \text{ and } p(\theta = s|\boldsymbol{y}) \geq p(\theta = t|\boldsymbol{y}) \text{ for any } s \in S, t \in S^c\}.$$

Another interval, called the *central posterior density region* (CPDR), covers the central values of a distribution. It is defined as the following interval:

$$\text{CPDR}(\theta|\boldsymbol{y}) := (\sup\{z : \Pr(\theta < z|\boldsymbol{y}) \leq \alpha/2\}, \inf\{z : \Pr(\theta > z|\boldsymbol{y}) \leq \alpha/2\}),$$

where α is the significance level. Note that when θ is continuous, the above is simplified as $\text{CPDR}(\theta|\boldsymbol{y}) = \left(F_{\theta|\boldsymbol{y}}^{-1}(\alpha/2), F_{\theta|\boldsymbol{y}}^{-1}(1-\alpha/2)\right)$, where $F_{\theta|\boldsymbol{y}}^{-1}$ is the inverse of the cumulative posterior distribution function of θ.

2.1.1.3 Decision Analysis/Theory

When selecting a point estimate, it is of interest and value to quantify the consequences of that estimate being wrong to a certain degree. To this end, we may consider a specified *loss function* $L(\theta^*, \theta)$ as a measure of the information "cost" due to using an estimate θ^* of the "true" value θ. We want θ^* to minimize the "overall cost", $\mathbb{E}(L(\theta^*, \theta))$, namely the *Bayes risk*. According to the law of total expectation, we have the following relationship:

$$\mathbb{E}(L(\theta^*, \theta)) = \mathbb{E}_{\boldsymbol{y}}\{\mathbb{E}_{\theta|\boldsymbol{y}}\left(L\left(\theta^*, \theta\right)|\boldsymbol{y}\right)\} = \mathbb{E}_{\theta}\{\mathbb{E}_{\boldsymbol{y}|\theta}(L(\theta^*, \theta)|\theta)\}.$$

We define the *posterior expected loss* (PEL) and the *risk function* respectively as follows:

$$\begin{aligned} \text{PEL}(\theta^*) &:= \mathbb{E}_{\theta|\boldsymbol{y}}(L(\theta^*, \theta)|\boldsymbol{y}) = \int_{\Theta} L(\theta^*, \theta)p(\theta|\boldsymbol{y})d\theta \\ R(\theta^*, \theta) &:= \mathbb{E}_{\boldsymbol{y}|\theta}(L(\theta^*, \theta)|\theta) = \int L(\theta^*, \theta)p(\boldsymbol{y}|\theta)d\boldsymbol{y}. \end{aligned}$$

Hence $\mathbb{E}(L(\theta^*, \theta)) = \mathbb{E}_{\boldsymbol{y}}(\text{PEL}(\theta^*)) = \mathbb{E}_{\theta}(R(\theta^*, \theta))$. If θ^* minimizes $\text{PEL}(\theta^*)$ for all data $\boldsymbol{y}$, then it also minimizes the Bayesian risk. Such θ^* is called the *Bayesian estimate* with respect to the loss function $L(\theta^*, \theta)$. Consider the following three loss functions:

1. Quadratic error loss function: $L_q(\theta^*, \theta) = (\theta^* - \theta)^2$.
2. Absolute error loss function: $L_a(\theta^*, \theta) = |\theta^* - \theta|$.
3. Zero-one error loss function: $L_z = 1_{\{0\}^c}(\theta^* - \theta)$.

It can be proved that the posterior mean $\hat{\theta}$ minimizes the quadratic error loss function, the posterior median $\ddot{\theta}$ minimizes the absolute error loss function, and the posterior mode $\tilde{\theta}$ minimizes the zero-one error loss function. Hence, the point estimates discussed before are the Bayesian estimates with respect to these loss functions.

2.1.1.4 Prediction

Before the data $\boldsymbol{y}$ is observed, the distribution of the unknown but observable y is

$$p(y) = \int_{\Theta} p(y, \theta) d\theta = \int_{\Theta} p(y|\theta) p(\theta) d\theta.$$

This is called the *marginal distribution*, the *prior predictive distribution* or the *unconditional distribution* of y since it is not conditional on a previous observation.

After the data $\boldsymbol{y}$ has been observed, we can predict an unknown observable y'. The distribution of y' is called the *posterior predictive distribution*, since it is conditional on the data $\boldsymbol{y}$:

$$p(y'|\boldsymbol{y}) = \int_{\Theta} p(y', \theta|\boldsymbol{y}) d\theta = \int_{\Theta} p(y'|\theta) p(\theta|\boldsymbol{y}) d\theta.$$

Example 2.1 (A single-parameter Bernoulli-Beta model) Consider the following Bayesian Bernoulli-Beta model:

$$\begin{aligned} y_i &\sim \text{Bern}(\theta), i = 1, \ldots, n \\ \theta &\sim \text{Beta}(\alpha, \beta). \end{aligned}$$

According to Bayes' theorem, the posterior distribution of θ is

$$p(\theta|\boldsymbol{y}) \propto \theta^{\alpha - 1 + \sum\limits_{i=1}^{n} y_i} (1 - \theta)^{\beta - 1 + n - \sum\limits_{i=1}^{n} y_i}, \tag{2.2}$$

which implies the posterior distribution of θ is Beta$(\alpha + \sum_{i=1}^{n} y_i, \beta + n - \sum_{i=1}^{n} y_i)$. The posterior mean of θ is $\hat{\theta} = (\alpha + \sum_{i=1}^{n} y_i)/(\alpha + \beta + n)$, and it can be interpreted as an upgrade from the prior mean of $\alpha/\alpha + \beta$ due to observation $\boldsymbol{y}$. And we can continually upgrade $\hat{\theta}$ as more observations become available.

If we choose $\alpha = 1, \beta = 1$, i.e., the prior of θ is an uniform distribution on [0, 1] reflecting no favourite of a particular value of θ, then the posterior mean $\hat{\theta} = (1 + \sum_{i=1}^{n} y_i)/(2 + n)$. In the case when $\alpha = 0, \beta = 0$, the prior is *improper* (discussed later). However, the resulting posterior is still *proper* and $\hat{\theta} = n^{-1} \sum_{i=1}^{n} y_i$, which is equal to the MLE.

To illustrate the point estimates and interval estimates in the Bayesian framework, we assume the true underlying parameter as $\theta_{True} = 0.3$, then simulate a data set $\boldsymbol{y} = (0, 0, 0, 1, 0, 0, 1, 0, 1, 1, 0, 1, 0, 0, 1, 0, 0, 0, 1, 0)$. The prior of θ is assumed

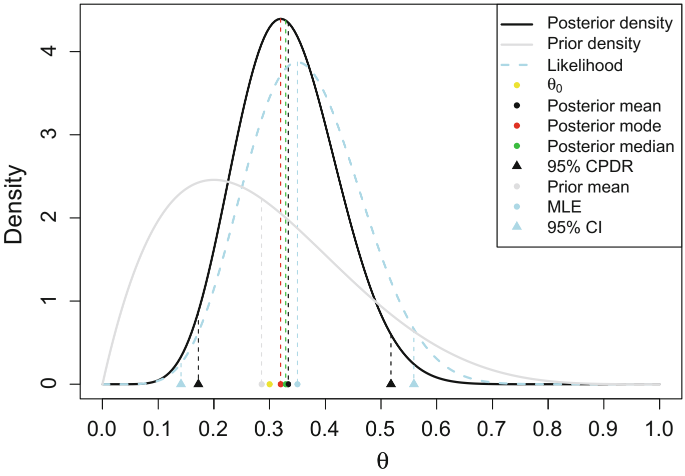


Fig. 2.1 The prior, posterior and likelihood of θ

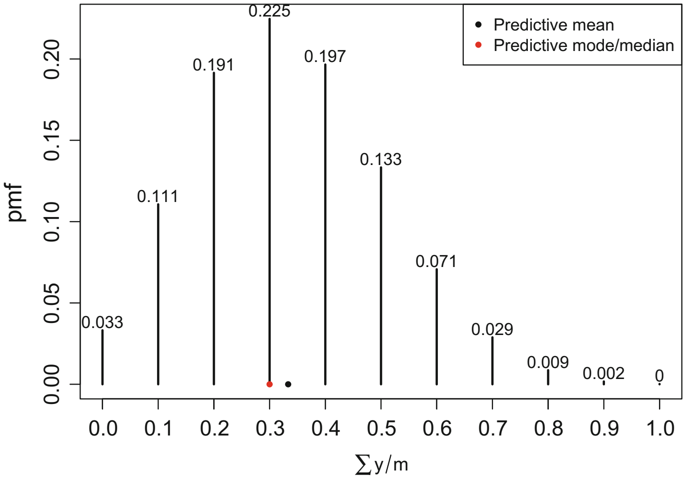


Fig. 2.2 The posterior predictive distribution of $\sum_{j=1}^{10} y_j'/10$

to be $Beta(2, 5)$, because suppose we had previously observed 2 successes in 7 trials before our $\boldsymbol{y}$ was observed. In Fig. 2.1, we show the prior distribution, the likelihood, the posterior distribution, three point estimates, the 95% CPDR, the MLE and the 95% confidence interval. The posterior distribution is a kind of weighting between the prior distribution and the likelihood. The predictive distribution of the proportion of successes in the next 10 trials, $\sum_{j=1}^{10} y_j'/10$, is given in Fig. 2.2, together with the predictive mean, mode and median.

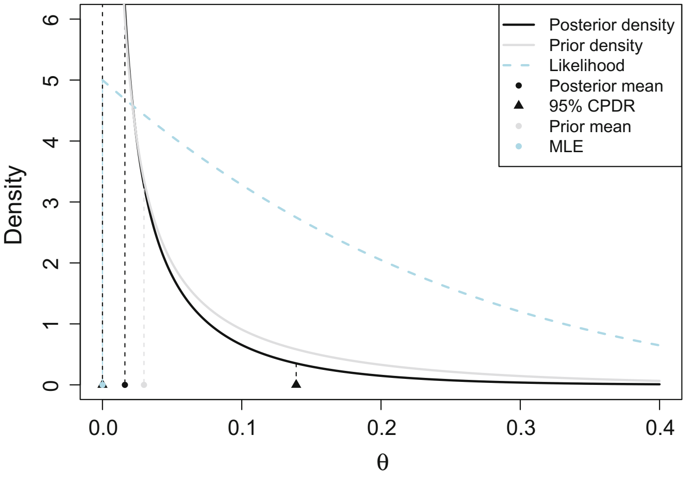


Fig. 2.3 The prior, posterior and likelihood of θ

Example 2.2 (Number of positive lymph nodes) This example is adjusted from Berry and Stangl (1996). About 75% of the lymph from the breasts drains into the axillary lymph nodes, making them important in the diagnosis of breast cancer. A doctor will usually refer a patient to a surgeon to have an axillary lymph node dissection to see if cancer cells have been trapped in the nodes. The presence of cancer cells in the nodes increases the risk of metastatic breast cancer.

Suppose a surgeon removes four axillary lymph nodes from a woman with breast cancer and none tests positive (i.e., no cancer cells). Suppose also that the probability of a node testing positive has a distribution of Beta(0.14, 4.56) Berry and Stangl (1996). The question is, what is the probability that the next four nodes are all negative?

Denote a random variable by y with the sample space of $\{0, 1\}$, where 0 represents negative and 1 represents positive for a tested node. We know $y \sim \text{Bern}(\theta)$. Now we have a data set $\mathbf{y} = (0, 0, 0, 0)$, so according to Eq. (2.2) our knowledge of θ is upgraded as the posterior distribution of $\text{Beta}(0.14 + \sum_{i=1}^{4} y_i, 4.56 + 4 - \sum_{i=1}^{4} y_i) = \text{Beta}(0.14, 8.56)$. Figure 2.3 shows how the observation shifts the prior to the posterior. In this example, the number of successes is zero, so the 95% CI is not well defined while the 95% CPDR still exists. The posterior mean is $\hat{\theta} = 0.01609$, the posterior median is $\ddot{\theta} = 0.0005460$, the posterior mode is $\tilde{\theta} = 0$ and the 95% CPDR of θ is (0, 0.14).

The posterior predictive distribution of y' is given by:

$$\Pr(y' = 1|\mathbf{y}) = \int_0^1 \theta p(\theta|\mathbf{y})d\theta = \hat{\theta} = 0.016$$

$$\Pr(y' = 0|\mathbf{y}) = \int_0^1 (1 - \theta)p(\theta|\mathbf{y})d\theta = 1 - \hat{\theta} = 0.984,$$

where $p(\theta|\boldsymbol{y})$ is the density function of Beta(0.14, 8.56). Hence $y'|\boldsymbol{y} \sim \text{Bern}(0.016)$. Now denote the status of next four nodes by y_5, y_6, y_7, y_8. The probability that the next four nodes are all negative is

$$
\begin{aligned}
&\Pr(y_5, y_6, y_7, y_8 = 0|\boldsymbol{y})\\
=&\Pr(y_8 = 0|y_5, y_6, y_7 = 0, \boldsymbol{y}) \Pr(y_7 = 0|y_5, y_6 = 0, \boldsymbol{y}) \Pr(y_6 = 0|y_5 = 0, \boldsymbol{y})\\
&\Pr(y_5 = 0|\boldsymbol{y})\\
=&0.946.
\end{aligned}
$$

Note that $\Pr(y_5 = 0|\boldsymbol{y}) = 0.984$ and the other terms are obtained from the updating procedure just described in two previous paragraphs.

2.1.2 The Multi-parameter Case

We extend a single parameter θ to multiple parameters θ and assume the parameter vector $\theta = (\theta_1, \ldots, \theta_m)$ distributed as a joint prior $p(\theta)$ with parameter space $\theta \subseteq \mathbb{R}^m$. The left hand side of Eq. (2.1) becomes a joint posterior distribution of $\theta = (\theta_1, \ldots, \theta_m)$.

Unlike the single parameter case, we cannot make inferences about a parameter directly from Eq. (2.1). We need to further find the *marginal posterior distribution* by integrating the joint posterior distribution $p(\theta|\boldsymbol{y})$ over all the parameters except the parameter of interest, θ_k, as follows:

$$
p(\theta_k|\boldsymbol{y}) = \int p(\theta|\boldsymbol{y}) d\theta_{-k}, \tag{2.3}
$$

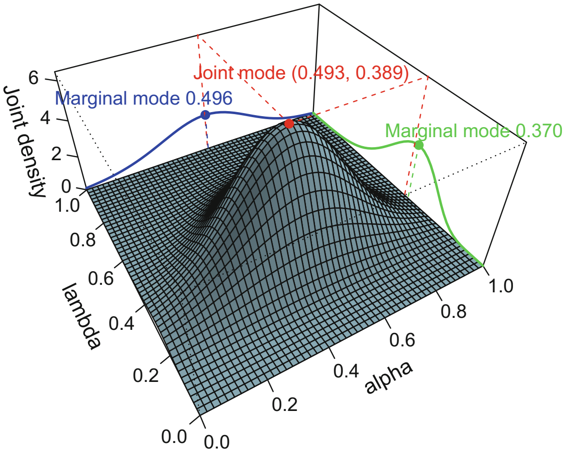
where $\theta_{-k} = (\theta_1, \ldots, \theta_{k-1}, \theta_{k+1}, \ldots, \theta_m)$. Now the definitions of posterior mean, median, mode, HPDR and CPDR from the previous section can be applied to $p(\theta_k|\boldsymbol{y})$. For the posterior predictive distribution, multiple integration is required since $p(\theta|\boldsymbol{y})$ is a joint distribution. We also define the *full conditional posterior distribution* of θ_k as $p(\theta_k|\boldsymbol{y}, \theta_{-k}) \propto p(\theta|\boldsymbol{y})$ for $1 \le k \le m$.

Example 2.3 (An autoregressive process of order one) Consider the following Bayesian model for an autoregressive process of order one:

$$
\begin{aligned}
x_t &= \alpha x_{t-1} + e_t, t = 1, \ldots, n\\
e_t &\sim \text{N}(0, \lambda^{-1})\\
\alpha &\sim \text{U}(-1, 1)\\
p(\lambda) &\propto 1/\lambda,
\end{aligned}
$$

where λ is the *precision parameter*. We simulate a sample of size n, assuming $\alpha_0 = 0.7, \lambda_0 = 0.25$ and $n = 20$. The joint posterior density of α and λ is

Fig. 2.4 The joint posterior distribution of α and λ



$$p(\alpha, \lambda) = h_0 \lambda^{n/2-1}(1-\alpha^2)^{1/2} \exp\left(-\frac{\lambda}{2} h(\boldsymbol{x}, \alpha)\right),$$

where h_0 is called the *normalizing constant* and $h(\boldsymbol{x}, \alpha) = (x_n - \alpha x_{n-1})^2 + \cdots + (x_2 - \alpha x_1)^2 + (1-\alpha^2)x_1^2$.

In Fig. 2.4 we show the joint posterior distribution, two marginal distributions, the joint mode and two marginal modes. There is a slight difference between joint modes and marginal modes. Similar to the single parameter case, in Fig. 2.5 we show the inferences made from two marginal posterior distributions. Under the non-informative priors, Bayesian inference is quite close to the frequentist inference. This is guaranteed by the asymptotic theory, which will be discussed in Sect. 2.1.4.

Finally for the prediction, $\hat{x}_{20+1} = \mathbb{E}(x_{20+1}|\boldsymbol{x}) = \mathbb{E}(\alpha x_{20}|\boldsymbol{x}) = x_{20}\mathbb{E}(\alpha|\boldsymbol{x}) = x_{20}\hat{\alpha} = 0.3517$. The analytic solution to the predictive distribution requires a double integral with respect to α and λ. We will estimate the posterior predictive distribution in Sect. 3.1.2 using the MCMC methods. See details in Appendix A on page 187.

2.1.3 Choice of Prior Distribution

Here we will discuss three types of priors: informative priors, non-informative priors and weakly informative priors Gelman et al. (2014).

2.1.3.1 Informative Priors

In Example 2.1, comparing $p(\theta)$ and $p(\theta|\boldsymbol{y})$ suggests that the prior is equivalent to $\alpha - 1$ prior successes and $\beta - 1$ prior failures. The parameters of the prior distribution are often referred to as *hyperparameters*. If we had past trials, we could

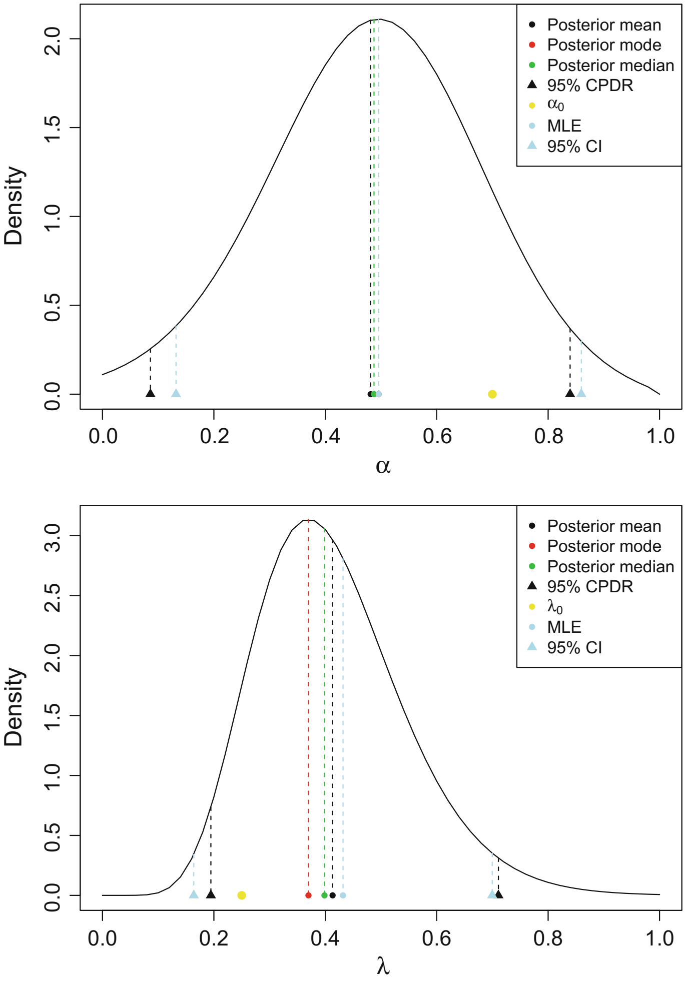


Fig. 2.5 The marginal posterior distributions of α and λ

summarize the past information about θ into an informative prior. Every time we use an informative prior we can treat the prior as the summary from past data. An informative prior is equivalent to adding some observations to a non-informative prior.

Sometimes informative priors are called strong priors, in the sense that they affect the posterior distribution more strongly, relative to the data, than other priors. The distinction between strong priors and weak priors is vague, and a strong prior may become a weak prior as more data comes in to counterbalance the strong prior. It is better to look at the prior together with the likelihood of data.

2.1.3.2 Non-informative Priors

There has been a desire for priors that can be guaranteed to play a minimal role, ideally no role at all, in the posterior distribution. Such priors are variously called *non-informative priors*, *reference priors* (Berger et al. 2009), *vague priors*, *flat priors*, or *diffuse priors*. The rationale for using a non-informative prior is often given as letting the data speak for themselves.

The Bernoulli-Beta model

In Example 2.1, Beta(1, 1) is a non-informative prior, since it assumes that θ is distributed uniformly on [0, 1]. The posterior distribution under this prior is the same as the likelihood. The posterior mode will be equal to the maximum likelihood estimate $\sum_{i=1}^{n} y_i/n$. Note that the posterior mean is not equal to the posterior mode.

If we want the posterior mean equal to the MLE, we need to specify $\alpha, \beta = 0$. This prior is called a *improper* non-informative prior since the integral of this prior's pdf is not 1. When we use an improper non-informative prior, we need to check whether the resulting posterior is proper. Fortunately, the posterior here is proper.

The normal-normal model with known variance

Another example is the normal model with unknown mean but known variance, shown as follows:

$$\begin{aligned} y_i &\sim \mathrm{N}(\mu, \sigma^2), i = 1, \ldots, n \\ \mu &\sim \mathrm{N}(\mu_0, \tau_0^2). \end{aligned}$$

If $\tau_0^2 \to \infty$, the prior is proportional to a constant, and is improper. But the posterior is still proper, $p(\mu|\mathbf{y}) \approx \mathrm{N}(\mu|\bar{\mathbf{y}}, \sigma^2/n)$. Here $\mathrm{N}(\mu|\bar{\mathbf{y}}, \sigma^2/n)$ is used to represent the probability density function for variable μ, a normal distribution with mean of $\bar{\mathbf{y}}$ and variance of σ^2/n.

The normal-normal model with known mean

Now assume the mean is known and variance is unknown. We know that the conjugate prior for variance is inverse-gamma distribution, i.e., σ^{-2} follows a gamma distribution, Gamma(α, β). The non-informative prior is obtained as $\alpha, \beta \to 0$.

Here we parameterize it as a scaled inverse$-\chi^2$ distribution with scale σ_0^2 and ν_0 degrees of freedom; i.e., the prior distribution of σ^2 is taken to be the distribution of $\sigma_0^2\nu_0/X$, where X is a $\chi^2_{\nu_0}$ random variable. The model can be written as follows:

$$\begin{aligned} y_i &\sim \mathrm{N}(\mu, \sigma^2), i = 1, \ldots, n \\ \sigma^2 &\sim \text{Inv-}\chi^2(\nu_0, \sigma_0^2). \end{aligned}$$

The resulting posterior distribution of σ^2 can be shown as

$$\sigma^2|\mathbf{y} \sim \text{Inv-}\chi^2\left(\nu_0 + n, \frac{\nu_0\sigma_0^2 + n\nu}{\nu_0 + n}\right),$$

where $\nu = 1/n \sum_{i=1}^{n}(y_i - \mu)^2$.

The non-informative prior is obtained as $\nu_0 \to 0$, which is improper and proportional to the inverse of the variance parameter. This non-informative prior is sometimes written as $p(\log \sigma^2) \propto 1$. The resulting posterior distribution is proper, with the density function of $p(\sigma^2|\mathbf{y}) \approx \text{Inv-}\chi^2(\sigma^2|n, \nu)$. The uniform prior distribution on σ^2, i.e., $p(\sigma^2) \propto 1$, will lead to an improper posterior.

Jeffreys' priors

Finally, there is a family of non-informative priors called *Jeffreys' priors*. The idea is that the non-informative priors should have the same influence as likelihood on the parameters. It can be shown that the Jeffreys' prior of θ is proportional to the squared root of *Fisher information*; i.e., $p(\theta) \propto J(\theta)^{1/2}$, where

$$J(\theta) = \mathbb{E}\left(\left(\frac{d \log p(\mathbf{y}|\theta)}{d\theta}\right)^2 \middle| \theta\right) = -\mathbb{E}\left(\frac{d^2 \log p(\mathbf{y}|\theta)}{d\theta^2} \middle| \theta\right). \tag{2.4}$$

As a simple justification, the Fisher information measures the curvature of the log-likelihood, and high curvature occurs wherever small changes in parameter values are associated with large changes in the likelihood. So the proportional relationship ensures that Jeffreys' prior gives more weight to these parameters. In Example 2.1, the Fisher information is $J(\theta) = n/(\theta(1-\theta))$. Hence, Jeffreys' prior is $p(\theta) \propto \theta^{-1/2}(1-\theta)^{-1/2}$, which is Beta(0.5, 0.5).

2.1.3.3 Weakly Informative Priors

A weakly informative prior lies between informative priors and non-informative priors. It is proper, but is set up so that the information it provides is intentionally weaker than whatever actual prior knowledge is available. We do not use weakly informative priors here. For more discussion, please refer to Gelman et al. (2014) on page 55.

Example 2.4 (A single-parameter Bernoulli-Beta model) We continue with Example 2.1 and consider the effects of two non-informative priors, Beta(1, 1) and Beta(0.5, 0.5), on the posterior distributions. Under the uniform distribution Beta(1, 1), the posterior distribution is equal to the scaled likelihood, so the posterior mode is equal to the MLE. Under the Jeffreys' prior Beta(0.5, 0.5), the posterior distribution is quite close to the scaled likelihood. In both cases, the effect of the priors on the posterior distribution is negligible. In Fig. 2.6, we plot the likelihood, the prior, and the posterior distribution. As we expect, under the two non-informative priors the scaled likelihood is quite close to the posterior distribution.

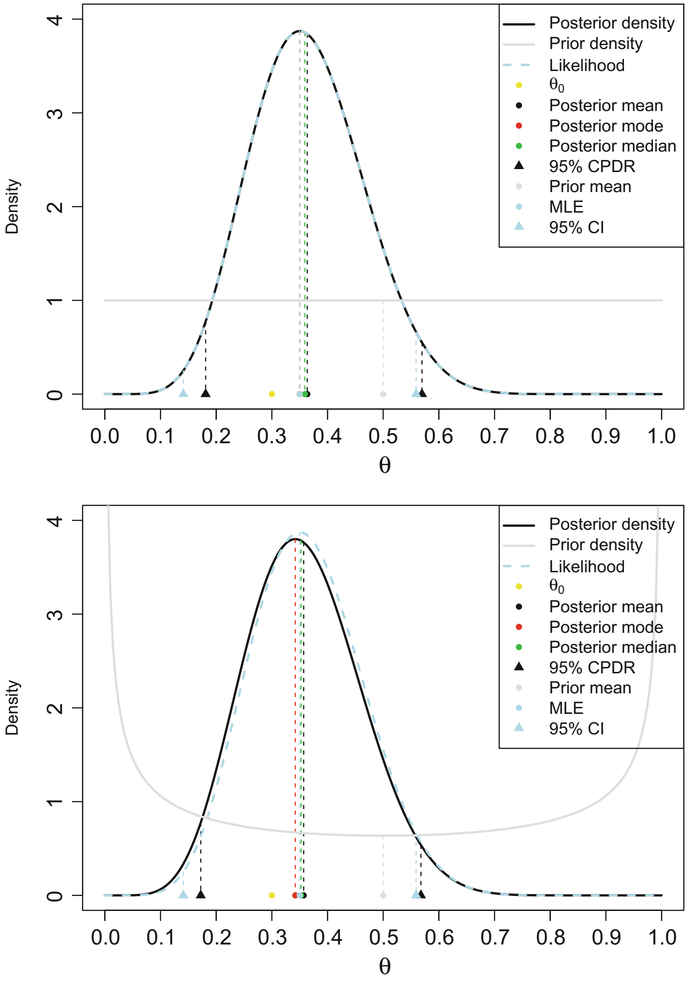


Fig. 2.6 The effect of two non-informative priors, Beta(1, 1) and Beta(0.5, 0.5), on the posterior distribution

2.1.4 Asymptotic Normality of the Posterior Distribution

Suppose $y_1, \ldots, y_n$ are outcomes sampled from a distribution $f(y)$. We model the data by a parametric family $p(y|\theta) : \theta \in \Theta$, where θ is distributed as $p(\theta)$. The result of large-sample Bayesian inference is that as more and more data arrive, i.e., $n \to \infty$, the posterior distribution of the parameter vector approaches multivariate normal distribution.

We label θ_0 as the value of θ that minimizes the Kullback-Leibler divergence $\mathrm{KL}(\theta)$ of the likelihood $p(\boldsymbol{y}|\theta)$ relative to the true distribution $f(\boldsymbol{y})$. The Kullback-Leibler divergence is defined as a function of θ as follows:

$$\mathrm{KL}(\theta) := \mathbb{E}_f\left(\log\left(\frac{f(\boldsymbol{y})}{p(\boldsymbol{y}|\theta)}\right)\right) = -\int \log\left(\frac{f(\boldsymbol{y})}{p(\boldsymbol{y}|\theta)}\right) f(\boldsymbol{y})d\boldsymbol{y}.$$

2.1.4.1 When the True Distribution is in the Parametric Family

If the true data distribution is included in the parametric family, i.e., $f(y) = p(y|\theta_{\mathrm{True}})$ for some θ_{True}, then θ_0 will approach θ_{True} as $n \to \infty$. The posterior distribution of θ approaches normality with mean θ_0 and variance $nJ(\theta_0)^{-1}$, where $J(\theta_0)$ is the Fisher information defined in Eq. (2.4).

The proof of asymptotic normality is based on the *Taylor series expansion* of log posterior distribution, $\log p(\theta|\boldsymbol{y})$, centred at the posterior mode up to the quadratic term. As $n \to \infty$, the likelihood dominates the prior, so we can just use the likelihood to obtain the mean and variance of the normal approximation.

2.1.4.2 When the True Distribution is Not in the Parametric Family

The above discussion is based on the assumption that the true distribution is included in the parametric family, i.e., $f(y) \in \{p(y|\theta) : \theta \in \Theta\}$. When this assumption fails, there is no true value $\theta_{\mathrm{True}} \in \Theta$, and its role in the theoretical result is replaced by the value θ_0 which minimizes the Kullback-Leibler divergence. Hence, we still have the similar asymptotic normality that the posterior distribution of θ approaches normality with mean θ_0 and variance $nJ(\theta_0)^{-1}$. But now $p(y|\theta_0)$ is no longer the true distribution $f(y)$.

2.2 Model Assessment and Selection

In this section, we review the model diagnostic tools including posterior predictive checking and residual plots. We also discuss the model selection criteria including several information criteria and cross-validation.

2.2.1 Posterior Predictive Checking

In the classical framework, the testing error is preferred since it is calculated on a testing data set which is not used to train the model. In the Bayesian framework, ideally we want to split the data into a training set and a testing set and do the posterior predictive checking on the testing data set. Alternatively, we can choose a test statistic whose predictive distribution does not depend on unknown parameters in the model but primarily on the assumption being checked. Then there is no need to have a separate testing data set. Nevertheless, when the same data are used for both fitting and checking the model, this needs to be carried out with caution, as the procedure can be conservative.

In frequentist statistics, p-value is typically defined as

$$p := \Pr(T(\boldsymbol{y}') \geq T(\boldsymbol{y})|H_0),$$

where T is the function of data that generates the test statistic, $\boldsymbol{y}'$ is the future observation (random variable), and $\boldsymbol{y}$ is the observed sample. Note that $T(\boldsymbol{y})$ is regarded as a constant. The probability is calculated over the sampling distribution of y under the null hypothesis. It is well known that p can be calculated exactly only in the sense that $T(\boldsymbol{y})$ is a *pivotal quantity*.

Meng (1994) explored the posterior predictive p-value (p_B), a Bayesian version of the classical p-value. p_B is defined as the probability

$$p_B := \Pr\{T(\boldsymbol{y}', \theta) \geq T(\boldsymbol{y}, \theta)|\boldsymbol{y}, H_0\},$$

where $\boldsymbol{y}'$ is the future data, and $T(\boldsymbol{y}, \theta)$ is a *discrepancy measure* that possibly depends on θ. This probability is calculated over the following distribution:

$$p(\boldsymbol{y}', \theta|\boldsymbol{y}, H_0) = p(\boldsymbol{y}'|\theta)p(\theta|\boldsymbol{y}, H_0),$$

where the form of $p(\theta|\boldsymbol{y}, H_0)$ depends on the nature of the null hypothesis. Following Meng (1994), we consider the two null hypotheses: a point hypothesis and a composite hypothesis. For the completion of discussion, please refer to Robins et al. (2000). They mentioned some problems associated with the posterior predictive p-value under a composite hypothesis.

2.2.1.1 When the Null Hypothesis is a Point Hypothesis

Suppose the null hypothesis is $\theta_k = a$ and the prior under the null hypothesis is $p(\theta_{-k}|\theta_k = a)$ with the parameter space $\Theta \subset \mathbb{R}^{m-1}$. Then the posterior density of θ under the null hypothesis is

$$p\left(\theta|\boldsymbol{y}, H_0\right) = \frac{p\left(\boldsymbol{y}|\theta_{-k}, \theta_k = a\right) p\left(\theta_{-k}|\theta_k = a\right)}{\int_\Theta p\left(\boldsymbol{y}|\theta_{-k}, \theta_k = a\right) p\left(\theta_{-k}|\theta_k = a\right) d\theta_{-k}}.$$

The posterior predictive p-value is calculated as

$$\begin{aligned} p_B &= \Pr\left\{T\left(\boldsymbol{y}', \theta\right) \geq T\left(\boldsymbol{y}, \theta\right)|y, H_0\right\} \\ &= \int_\Theta \Pr\left\{T\left(\boldsymbol{y}', \theta\right) \geq T\left(\boldsymbol{y}, \theta\right)|\theta\right\} p\left(\theta|\boldsymbol{y}, H_0\right) d\theta_{-k}. \end{aligned}$$

2.2.1.2 When the Null Hypothesis is a Composite Hypothesis

Suppose the null hypothesis is $\theta_k \in A$ and the prior under the null hypothesis is $p(\theta_{-k}|\theta_k)p(\theta_k)$. Then the posterior density of θ under the null hypothesis is

$$p\left(\theta|\boldsymbol{y}, H_0\right) = p\left(\theta_{-k}|\boldsymbol{y}, \theta_k\right) p\left(\theta_k\right) = \frac{p\left(\boldsymbol{y}|\theta\right) p\left(\theta_{-k}|\theta_k\right)}{\int_{\Theta} p\left(\boldsymbol{y}|\theta\right) p\left(\theta_{-k}|\theta_k\right) d\theta_{-k}} p\left(\theta_k\right).$$

The posterior predictive p-value is calculated as

$$\begin{aligned} p_B &= \Pr\left\{T\left(\boldsymbol{y}', \theta\right) \geq T\left(\boldsymbol{y}, \theta\right)|\boldsymbol{y}, H_0\right\} \\ &= \int_{\Theta}\int_{A} \Pr\left\{T\left(\boldsymbol{y}', \theta\right) \geq T\left(\boldsymbol{y}, \theta\right)|\theta\right\} p\left(\theta_{-k}|\boldsymbol{y}, \theta_k\right) p\left(\theta_k\right) d\theta_k d\theta_{-k}. \end{aligned}$$

2.2.1.3 Choice of $T(\boldsymbol{y}, \boldsymbol{\theta})$

Recall that in the frequentist theory, the most powerful test in a composite test, $H_0 : \theta_k \in A$ versus $H_1 : \theta_k \notin A$, is based on the *generalized likelihood ratio* defined as follows:

$$\Lambda_{\mathrm{g}}\left(\boldsymbol{y}\right) := \frac{\sup_{\theta_k \notin A} p(\boldsymbol{y}|\theta_k)}{\sup_{\theta_k \in A} p(\boldsymbol{y}|\theta_k)}.$$

Meng (1994) suggested using the *conditional likelihood ratio* and the *generalized likelihood ratio*, defined respectively as follows:

$$\begin{aligned} \mathrm{CLR}\left(\boldsymbol{y}, \theta\right) = T^{C}\left(\boldsymbol{y}, \theta\right) &:= \frac{\sup_{\theta_k \notin A} p\left(\boldsymbol{y}|\theta\right)}{\sup_{\theta_k \in A} p\left(\boldsymbol{y}|\theta\right)} \\ \mathrm{GLR}\left(\boldsymbol{y}\right) = T^{G}\left(\boldsymbol{y}\right) &:= \frac{\sup_{\theta_k \notin A} \sup_{\theta_{-k}} p\left(\boldsymbol{y}|\theta\right)}{\sup_{\theta_k \in A} \sup_{\theta_{-k}} p\left(\boldsymbol{y}|\theta\right)}. \end{aligned}$$

Because a probability model can fail to reflect the process that generated the data in any number of ways, p_B can be computed for a variety of discrepancy measures T in order to evaluate more than one possible model failure.

Example 2.5 (A one-sample normal mean test using p_B) This example is extracted from Meng (1994). Suppose we have a sample of size n from $\mathrm{N}(\mu, \sigma^2)$, and we test the null hypothesis that $\mu = \mu_0$ with σ^2 unknown. Recall that in classical testing, the pivotal test statistic is $\sqrt{n}(\bar{\boldsymbol{x}} - \mu_0)/s$, where $\bar{\boldsymbol{x}}$ is the sample mean and s^2 is the sample variance. We know this test statistic follows a t_{n-1} distribution. So $p = \Pr(t_{n-1} \geq \sqrt{n}(\bar{\boldsymbol{x}} - \mu_0)/s)$.

In the Bayesian framework, we assume μ and σ^2 are independent and σ^2 has a non-informative prior (i.e., $p(\sigma^2) \propto 1/\sigma^2$). We can find CLR and GLR as

$$\text{CLR}\left(\boldsymbol{x},\sigma^2\right)=T^C\left(\boldsymbol{x},\sigma^2\right)=\frac{n(\bar{\boldsymbol{x}}-\mu_0)^2}{\sigma^2}$$
$$\text{GLR}\left(\boldsymbol{x}\right)=T^G\left(\boldsymbol{x}\right)=\frac{n(\bar{\boldsymbol{x}}-\mu_0)^2}{s^2}.$$

Using the two discrepancy measures, we calculate p_B as

$$p_B^C=\Pr\{T^C\left(\boldsymbol{x}',\sigma^2\right)\geqslant T^C\left(\boldsymbol{x},\sigma^2\right)|\boldsymbol{x},\mu_0\}=\Pr\left(F_{1,\text{n}}\geqslant\frac{n(\bar{\boldsymbol{x}}-\mu_0)^2}{s_0^2}\right)$$
$$p_B^G=\Pr\{T^G\left(\boldsymbol{x}'\right)\geqslant T^G\left(\boldsymbol{x}\right)|\boldsymbol{x},\mu_0\}=\Pr\{F_{1,\text{n}-1}\geqslant T^G\left(\boldsymbol{x}\right)\},$$

where $s_0^2=\sum_{i=1}^n(x_i-\mu_0)^2$. Note that $p=p_B^G\neq p_B^C$; p_B is equal to the classical p-value when using GLR. See details in Appendix A on page 189.

Example 2.6 (Number of runs) Suppose we have a data set $\boldsymbol{x}=(x_1,x_2,\ldots,x_{10})=(1,1,1,0,0,\ 0,0,0,1,1)$, resulting from $n=10$ Bernoulli trials with success probability θ which has an non-informative improper prior, Beta (0, 0). Now we want to test the null hypothesis that the trials are independent of each other.

We use the number of runs in $\boldsymbol{x}$ as the test statistic, denoted by $r(\boldsymbol{x})$. Note that a run is defined as a subsequence of elements of one kind immediately preceded and succeeded by elements of the other kind. So in this example we have $r(\boldsymbol{x})=3$, and θ is treated as a *nuisance parameter*. It is easy to find that the posterior distribution of θ is Beta (6, 6) under H_0. To calculate $p_B=\Pr\{r(\boldsymbol{x}')\leq 3|H_0\}$, we apply the exact density of $r(\boldsymbol{x}')$.

According to Kendall and Stuart (1961), assuming n_1 1's and n_2 0's are randomly placed in a row, the number of runs, denoted by $R\ (n_1,n_2)$, has the following probability mass functions for $0\leq n_2\leq n_1$ and $2\leq R\leq n_1+n_2$:

$$\Pr\{R=2s\}=\frac{2\binom{n_1-1}{s}\binom{n_2-1}{s-1}}{\binom{n_2-1}{s-1}},\qquad \text{for } s=1,\ldots,n_2$$
$$\Pr\{R=2s-1\}=\frac{\binom{n_1-1}{s-2}\binom{n_2-1}{s-1}+\binom{n_1-1}{s-1}\binom{n_2-1}{s-2}}{\binom{n_2-1}{s-1}},\qquad \text{for } s=2,3,\ldots,n_2.$$

However, this probability mass function is not complete, missing the case when $R=2n_2+1$ (i.e., R is odd and $s=n_2+1$). For completeness, we add the two special cases and their associated probabilities as in Table 2.1. Applying the exact density of $R\ (n_1,n_2)$, p_B is calculated as

$$p_B=\int_0^1\left(\sum_{i=0}^{10}\sum_{j=1}^{3}\Pr\{R\,(i,10-i)=j\}\Pr\,(n_1=i|\theta)\right)p(\theta|\boldsymbol{x})d\theta=0.1630,\tag{2.5}$$

Table 2.1 Special cases for the probability of $R(n_1, n_2)$

n_1	n_2	s	$\Pr(R(n_1, n_2) = 2s - 1)$
≥ 1	0	1	1
$\geq n_2 + 1$	$n_2 \geq 1$	$n_2 + 1$	$\binom{n_1-1}{n_2}/\binom{n_1+n_2}{n_1}$

Table 2.2 p_B's for other observations

Case	Sample $\boldsymbol{x}$	n	$r(\boldsymbol{x})$	p_B
(i)	(1,1,1,1,1, 1,1,1,1,1)	10	1	0.5293
(ii)	(0,0,0,0,0, 1,1,1,1,1)	10	2	0.0462
(iii)	(0,1,1,0,1, 1,0,1,1,1)	10	6	0.6066
(iv)	(0,1,0,1,0, 1,0,1,0,1)	10	10	1
(v)	(1,0,1,1,1, 0,0,1,1,0, 1,0,1,1,0)	15	10	0.9354
(vi)	(1,1,1,1,1, 0,0,0,0,0, 1,1,1,1,1, 1,1,1,1,1, 1,1,1,0,0)	25	4	0.0248
(vii)	(1,0,1,0,1, 0,1,0,1,0, 1,0,1,0,1, 0,1,0,1,0, 1,0,1,0,1)	25	25	1

which implies that under H_0 the number of runs of a future observed sample would be smaller than 3 with probability of 0.163. See details in Appendix A on page 192. Furthermore, we list p_B's calculated for other observations in Table 2.2. Note that the sample test statistics in cases (iv) and (vii) reach the maximum number of runs, so p_B is definitely 1. However, we cannot conclude that x's are definitely independent of each other, as these observations indicate that 1's are most likely followed by 0's. We consider any p_B smaller than 0.1 or larger than 0.9 as indicating the violation of H_0.

2.2.2 *Residuals, Deviance and Deviance Residuals*

In the Bayesian framework, we can generate a set of residuals for one realization of posterior parameters. So there are four choices of residuals:

- Choose the posterior mean of parameters and find one set of residuals.
- Randomly choose a realization of parameters and find one set of the residuals.
- Get the posterior mean of residuals.
- Get the posterior distribution of residuals.

In the following, we will review Pearson residuals, deviance and deviance residuals. A *Pearson residual* is defined as

$$r_i(\theta) := \frac{y_i - \mathbb{E}(y_i|\theta)}{\sqrt{\operatorname{Var}(y_i|\theta)}}.$$

The *deviance* is defined as

$$D(\theta) := -2 \log p(\mathbf{y}|\theta) = -2 \sum_{i=1}^{n} \log p(y_i|\theta), \tag{2.6}$$

and the contribution of each data point to the deviance is $D_i\left(\theta\right)=-2\log p\left(y_i|\theta\right)$. We will define and use $D(\hat{\theta})$ and $\widehat{D\left(\theta\right)}$ in the next section.

The *deviance residuals* are based on a standardized or "saturated" version of the deviance, defined as

$$D_S\left(\theta\right):=-2\sum_{i=1}^{n}\log p\left(y_i|\theta\right)+2\sum_{i=1}^{n}\log p\left(y_i\middle|\hat{\theta}_S\left(\boldsymbol{y}\right)\right),$$

where $\hat{\theta}_S\left(\boldsymbol{y}\right)$ are appropriate "saturated" estimates, e.g., we set $\hat{\theta}_S\left(\boldsymbol{y}\right)=\boldsymbol{y}$. The contribution of each data point to the standardized deviance is

$$D_{S_i}\left(\theta\right)=-2\log p\left(y_i|\theta\right)+2\log p\left(y_i\middle|\hat{\theta}_S\left(\boldsymbol{y}\right)\right).$$

The deviance residual is defined as

$$dr_i:=\text{sign}_i\sqrt{D_{S_i}\left(\theta\right)},$$

where sign_i is the sign of $y_i-\mathbb{E}(y_i'|\theta)$.

Example 2.7 (Three error structures for stack-loss data) The stack-loss data set contains 21 daily responses of stack loss $\boldsymbol{y}$, the amount of ammonia escaping, with covariates being air flow x_1, temperature x_2 and acid concentration x_3. We assume a linear regression on the expectation of y, i.e., $\mathbb{E}\left(y_i\right)=\mu_i=\beta_0+\beta_1 z_{i1}+\beta_2 z_{i2}+\beta_3 z_{i3}$, $i=1,\ldots,21$. We consider three error structures: normal, double exponential and t_4, as follows:

$$\begin{aligned} y_i &\sim \text{N}(\mu_i,\tau^{-1})\\ y_i &\sim \text{DoubleExp}(\mu_i,\tau^{-1})\\ y_i &\sim t_4(\mu_i,\tau^{-1}), \end{aligned}$$

where $z_{ij}=\left(x_{ij}-\overline{\boldsymbol{x}_j}\right)/\text{sd}\left(\boldsymbol{x}_j\right)$ for $j=1,2,3$ are covariates standardized to have zero mean and unit variance, and $\beta_0,\beta_1,\beta_2,\beta_3$ are given independent non-informative priors.

The deviance residuals of the three models have the following forms respectively:

$$\begin{aligned} D_{S_i} &= \sqrt{\tau}\left(y_i-\mu_i\right)\\ D_{S_i} &= \text{sign}_i\sqrt{2\tau\left|y_i-\mu_i\right|}\\ D_{S_i} &= \text{sign}_i\sqrt{5\log\left(1+\frac{\left(y_i-\mu_i\right)^2}{4}\right)}. \end{aligned}$$

We plot the posterior distributions of deviance residuals for each model in Fig. 2.7. The three residual plots agree on four outliers: 1, 3, 4 and 21.

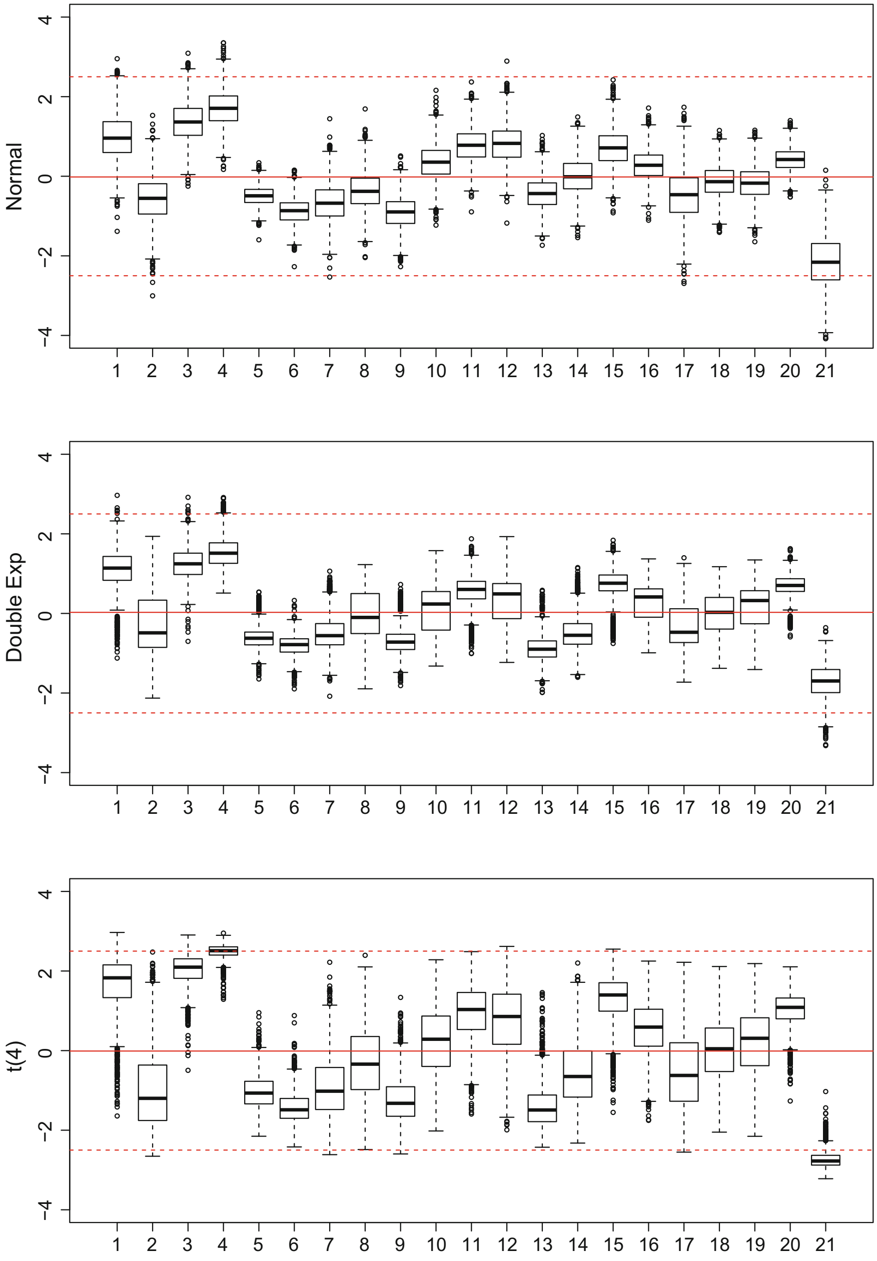


Fig. 2.7 The deviance residual plots of the three models

2.2.3 Bayesian Model Selection Methods

The model selection problem is a trade-off between a simple model and good fitting. Ideally, we want to choose the simplest model with best fitting. However good fitting models tend to be more complicated while simpler models tend to be *underfit*. The model selection methods used in frequentist statistics are typically *cross-validation* and *information criteria*, which are the modified *residual sum of squares* with respect to the model complexity and *overfitting*.

Cross-validation measures the fit of a model on the testing data set, which is not used to fit the model, while the information criteria adjust the measure of fit on the training data set by adding a penalty for model complexity.

2.2.3.1 The Predictive Accuracy of a Model

In the Bayesian framework, the fit of a model is sometimes called the *predictive accuracy* of a model (Gelman et al. 2014). We measure the predictive accuracy of a model to a data set $\boldsymbol{y}'$ by *log point wise predictive density* (lppd), calculated as follows:

$$\text{lppd:=}\log\prod_{i=1}^{n'}\mathbb{E}_{\theta|\mathbf{y}}p\left(y_i'|\theta\right)=\sum_{i=1}^{n'}\log\left(\mathbb{E}_{\theta|\mathbf{y}}p\left(y_i'|\theta\right)\right)=\sum_{i=1}^{n'}\log\left(\int p(y_i'|\theta)p(\theta|\mathbf{y})d\theta\right).$$

Ideally, $\boldsymbol{y}'$ should not be used to fit the model. If we choose $\boldsymbol{y}'=\boldsymbol{y}$, we get the *within-sample* lppd (denoted by $\text{lppd}_{\text{train}}$), which is typically larger than the *out-of-sample* lppd (denoted by $\text{lppd}_{\text{test}}$). To compute lppd in practice, we can evaluate the expectation using draws from the posterior distribution $p(\theta|\mathbf{y})$, which we label as $\theta^t,\ t=1,\ldots,T$. The computed lppd is defined as follows:

$$\text{computed lppd:=}\sum_{i=1}^{n'}\log\left(\frac{1}{T}\sum_{i=1}^{n'}p(y_i'|\theta^t)\right).$$

2.2.3.2 Cross-validation

In Bayesian cross-validation, the data are repeatedly partitioned into a training set y_{train} and a testing set y_{test}. For simplicity, we restrict our attention to *leave-one-out cross-validation* (LOO-CV), where y_{test} only contains a data point. The Bayesian LOO-CV estimate of out-of-sample lppd is defined as follows:

$$\text{lppd}_{\text{loo-cv}}:=\sum_{i=1}^{n}\log\left(\int p\left(y_i|\theta\right)p\left(\theta|\mathbf{y}_{-i}\right)d\theta\right), \tag{2.7}$$

where $\boldsymbol{y}_{-i}$ is the data set without the ith point. The $\mathrm{lppd}_{\text{loo-cv}}$ can be computed as

$$\text{computed } \mathrm{lppd}_{\text{loo-cv}} = \sum_{i=1}^{n} \log\left(\frac{1}{T}\sum_{t=1}^{T} p\left(y_i|\theta^{it}\right)\right),$$

where $\theta^{it},\ t = 1, \ldots, T$ are the simulations from the posterior distribution $p\left(\theta|\boldsymbol{y}_{-i}\right)$.

2.2.3.3 Deviance Information Criterion (DIC)

AIC and BIC

Before describing the DIC, we review another two information criteria employed in frequentist statistics. The Akaike information criterion (AIC) by Akaike (1973) is defined as

$$\mathrm{AIC} := -2\sum_{i=1}^{n} \log p\left(y_i|\theta_{\mathrm{MLE}}\right) + 2p.$$

The Bayesian information criterion (BIC) by Schwarz (1978) is defined as

$$\mathrm{BIC} := -2\sum_{i=1}^{n} \log p\left(y_i|\theta_{\mathrm{MLE}}\right) + p\log n,$$

where p is the number of parameters. The first common term $-2\sum_{i=1}^{n} \log p(y_i|\theta_{\mathrm{MLE}})$ measures the discrepancy between the fitted model and the data. The second term measures the model complexity.

DIC

In the Bayesian framework, we define a similar quantity to measure the discrepancy, $-2\sum_{i=1}^{n} \log p(y_i|\hat{\theta})$, where $\hat{\theta}$ is the posterior mean. Spiegelhalter et al. (2002) proposed a measure of number of effective parameters, which is defined as the difference between the posterior mean of the deviance and the deviance at the posterior means, as follows:

$$p_D := \widehat{D(\theta)} - D(\hat{\theta}) = -2\mathbb{E}_{\theta|\boldsymbol{y}}\left(\sum_{i=1}^{n} \log p\left(y_i|\theta\right)\right) + 2\sum_{i=1}^{n} \log p\left(y_i\middle|\hat{\theta}\right),$$

where D is the deviance defined in Eq. (2.6).

Furthermore, they proposed a deviance information criterion (DIC), defined as the deviance at the posterior means plus twice the effective number of parameters, to give

$$\mathrm{DIC} := D(\hat{\theta}) + 2p_D.$$

DIC is viewed as a Bayesian analogue of AIC. We prefer the model with smaller DIC. Note that DIC and p_{D} are sensitive to the level of a hierarchical model. They are appropriate when we are interested in the parameters directly related to the data. DIC and p_{D} can be calculated using OpenBUGS, which will be discussed in Sect. 3.3.

2.2.3.4 Watanabe-Akaike or widely available information criterion (WAIC)

Watanabe (2010) proposed another measure of number of effective parameters as follows:

$$p_{\mathrm{WAIC}} := \widehat{D(\theta)} + 2\mathrm{lppd}_{\mathrm{train}} = -2\mathbb{E}_{\theta|\boldsymbol{y}}\left(\sum_{i=1}^{n} \log p\left(y_i|\theta\right)\right) + 2\sum_{i=1}^{n} \log\left(\mathbb{E}_{\theta|\boldsymbol{y}}\, p\left(y_i|\theta\right)\right),$$

where $-2\mathrm{lppd}_{\mathrm{train}}$ plays a role as $D(\hat{\theta})$ as in p_{D}. As with AIC and DIC, the Watanabe-Akaike information criterion (WAIC) is defined as

$$\mathrm{WAIC} := -\, 2\mathrm{lppd}_{\mathrm{train}} + 2p_{\mathrm{WAIC}}.$$

2.2.3.5 Leave-One-Out Information Criterion (LOOIC)

Different from the definition of number of effective parameters in AIC, DIC, and WAIC, we define

$$p_{\mathrm{loo}} := \mathrm{lppd}_{\mathrm{train}} - \mathrm{lppd}_{\mathrm{loo\text{-}cv}},$$

where $\mathrm{lppd}_{\mathrm{loo\text{-}cv}}$ comes from Eq. (2.7). The *leave-one-out information criterion* (LOOIC) is defined as

$$\mathrm{LOOIC} := -\, 2\mathrm{lppd}_{\mathrm{train}} + 2p_{\mathrm{loo}} = -2\mathrm{lppd}_{\mathrm{loo\text{-}cv}},$$

which is reasonable since $\mathrm{lppd}_{\mathrm{loo\text{-}cv}}$ already penalizes the overfitting (or equivalently the model complexity).

Example 2.8 (*p_D in a random effects model)* This example follows Spiegelhalter et al. (2002). Consider the following random effects Bayesian model:

$$\begin{aligned} y_{ij} &\sim \mathrm{N}(\theta_i, \tau_i^{-1}),\ i = 1, \ldots, m,\ j = 1, \ldots, n \\ \theta_i &\sim \mathrm{N}(\mu, \lambda^{-1}) \\ \mu &\sim \mathrm{N}(0, \infty) \end{aligned}$$

where $\tau_i,\ i = 1, \ldots, m$, and λ are known precision parameters. τ_i is termed as the *within-group* precision, and λ is termed as the *between-group* precision. It can be shown that the posterior distribution of population mean is

$$\mu|\boldsymbol{y} \sim \mathrm{N}\left(\bar{\boldsymbol{y}}, \left(\lambda \sum_{i=1}^{m} \rho_i\right)^{-1}\right),$$

where

$$\bar{\boldsymbol{y}} = \frac{\sum_{i=1}^{m} \rho_i \bar{y}_i}{\sum_{i=1}^{m} \rho_i}, \; \rho_i = \frac{\tau_i}{\tau_i + \lambda}, \; \overline{\boldsymbol{y}_i} = \frac{\sum_{j=1}^{n} y_{ij}}{n}.$$

Assuming $\theta = (\theta_1, \ldots, \theta_m)$, we will have the following equations:

$$\begin{aligned}
\widehat{D(\theta)} &= \sum \rho_i + \lambda \sum \rho_i (1 - \rho_i) (\bar{y}_i - \bar{\boldsymbol{y}})^2 + \frac{\sum \rho_i (1 - \rho_i)}{\sum \rho_i} \\
D(\hat{\theta}) &= \lambda \sum \rho_i (1 - \rho_i) (\bar{y}_i - \bar{\boldsymbol{y}})^2 \\
p_D &= \sum \rho_i + \frac{\sum \rho_i (1 - \rho_i)}{\sum \rho_i}.
\end{aligned}$$

Consider the number of effective parameters p_D under the following three cases:

- If $\lambda \to \infty$, then $\rho_i \to 0$, and $p_D = 1$. All the groups have the same mean μ, which is the only effective parameter. The model is equivalent to:

$$\begin{aligned}
y_{ij} &\sim \mathrm{N}\left(\mu, \tau_i^{-1}\right), i = 1, \ldots, m, \, j = 1, \ldots, n \\
\mu &\sim \mathrm{N}(0, \infty).
\end{aligned}$$

- If $\lambda \to 0$, then $\rho_i \to 1$, and $p_D = m$. All the groups are independent and have different means. The model is equivalent to:

$$\begin{aligned}
y_{ij} &\sim \mathrm{N}\left(\theta_i, \tau_i^{-1}\right), i = 1, \ldots, m, \, j = 1, \ldots, n \\
\theta_i &\sim \mathrm{N}(0, \infty).
\end{aligned}$$

- If τ_i are equal, then $\rho_1 = \cdots = \rho_m = \rho$ and $p_D = 1 + (m - 1)\rho$.

In summary, if we assign the majority of variation in $\boldsymbol{y}$ to the within-group variation rather than between-group variation (i.e., λ is much larger than τ_i), then the group means θ_i tend to converge to the population mean μ, and we have only one parameter (i.e., θ_i cannot be effectively estimated distinguishably).

On the other hand, if we assign the majority of variation in $\boldsymbol{y}$ to the between-group variation (i.e., τ_i is much larger than λ), then there is no accurate estimate of μ, and every θ_i tends to "escape" from the "trap" distribution $\theta_i \sim \mathrm{N}\left(\mu, \lambda^{-1}\right)$. Every θ_i can be effectively estimated by its group mean, and there are m effective parameters.

Example 2.9 (Three error structures for stack-loss data) We continue with Example 2.7 and calculate $\mathrm{lppd}_{\text{loo-cv}}$, DIC, p_{D}, WAIC and p_{WAIC} for the three models discussed on page 26. As shown in Table 2.3, $\mathrm{lppd}_{\text{loo-cv}}$, DIC, and WAIC agree on the model with double exponential error distribution.

Table 2.3 $\text{lppd}_{\text{loo-cv}}$, DIC and WAIC for the three models

Error structures	$\text{lppd}_{\text{loo-cv}}$	DIC	p_D	WAIC	p_{WAIC}
Normal	−59.0	115.5	5.23	116.5	4.8
DoubleExp	−57.3	113.3	5.53	114.5	5.7
t_4	−57.8	114.2	5.53	115.6	5.8

2.2.4 Overfitting in the Bayesian Framework

Suppose that we have a sample of size n from a common normal distribution with unknown mean and known precision, $y_i \sim \text{N}\left(\mu, \tau^{-1}\right),\ i = 1, \ldots, m$. In the Bayesian framework we can assume m parameters, each of which is for one data value. Such a Bayesian model can be written as follows:

$$\begin{aligned} y_i &\sim \text{N}\left(\mu_i, \tau^{-1}\right),\ i = 1, \ldots, m \\ \mu_i &\sim \text{N}\left(\mu_0, \tau_0^{-1}\right), \end{aligned}$$

where τ is known and $p(\mu_0, \tau_0^{-1}) \propto \tau_0$ is a non-informative improper hyperprior. This is a special case when $n = 1$ in Example 2.8.

This random effects model can also be viewed as a *hierarchical model* with three levels. We refer to the top level distribution related to the data as the sampling distribution or likelihood, the second level distribution as the prior and the third level distribution as the hyperprior. Accordingly, μ_i, τ are called parameters and μ_0, τ_0 are called hyperparameters.

The model has m data values and $m + 2$ parameters, which presents an overfitting issue in the frequentist framework on account of parameters treated as unknown fixed constants. However, it is quite common for the number of parameters to be larger than the number of data values in the Bayesian framework, where the number of effective parameters would be smaller than m as shown in Example 2.8.

2.3 Bibliographic Notes

Bayesian statistics derives from Bayes' famous 1763 essay, which has been reprinted as Bayes (1763). For other early contributions, see also Laplace (1785, 1810). Gelman et al. (2014) contains most of the current developments in Bayesian statistics.

Jeffreys' priors and the invariance principles for non-informative priors are studied in Jeffreys (1961). The asymptotic normality of the posterior distribution was known by Laplace (1810).

The method of posterior predictive checking was proposed by Rubin (1981, 1984). The posterior predictive p-value was studied by Meng (1994). Akaike (1973) introduced the expected predictive deviance and AIC. Schwarz (1978) introduced BIC.

Spiegelhalter et al. (2002) introduced the DIC. Watanabe (2010, 2013) presented WAIC. RJMCMC was introduced by Green (1995). A recent work summarizing criteria for evaluation of Bayesian model selection procedures is Bayarri et al. (2012).

References

Akaike, H. (1973). Information theory and an extension of the maximum likelihood principle. In *Second International Symposium on Information Theory*, 267–281.

Bayarri, M. J., Berger, J. O., Forte, A., & Garcia-Donato, G. (2012). Criteria for Bayesian model choice with application to variable selection. *The Annals of Statistics*, *40*, 1550–1577.

Bayes, T. (1763). An essay towards solving a problem in the doctrine of chances. *Philosophical Transactions of the Royal Society,* 330–418.

Berger, J. O., Bernardo, J. M., & Sun, D. (2009). The formal definition of reference priors. *The Annals of Statistics*, *37*, 905–938.

Berry, D. A., & Stangl, D. (1996). *Bayesian biostatistics*. New York: Marcel Dekker.

Gelman, A., Carlin, J. B., Stern, H. S., & Rubin, D. B. (2014). *Bayesian data analysis* (3rd ed.). Boca Raton: Chapman & Hall.

Green, P. J. (1995). Reversible jump Markov chain Monte Carlo computation and Bayesian model determination. *Biometrika*, *82*, 711–732.

Jeffreys, H. (1961). *Theory of probability* (3rd ed.). London: Oxford University Press.

Kendall, M. G., & Stuart, A. (1961). *The advanced theory of statistics: Inference and relationship*. London: Charles Griffin.

Laplace, P. S. (1785). Memoire sur les approximations des formules qui sont fonctions de tres grands nombres. In *Memoires de l'Academie Royale des Sciences*.

Laplace, P. S. (1810). Memoire sur les approximations des formules qui sont fonctions de tres grands nombres, et sur leur application aux probabilites. In *Memoires de l'Academie des Science de Paris*.

Meng, X. L. (1994). Posterior predictive p-values. *The Annals of Statistics*, *22*, 1142–1160.

Robins, J. M., van der Vaart, A. W., & Ventura, V. (2000). Asymptotic distribution of p-values in composite null models. *Journal of the American Statistical Association*, *95*, 1143–1156.

Rubin, D. B. (1981). Estimation in parallel randomized experiments. *Journal of Educational and Behavioral Statistics*, *6*, 377–401.

Rubin, D. B. (1984). Bayesianly justifiable and relevant frequency calculations for the applied statistician. *The Annals of Statistics*, *12*, 1151–1172.

Schwarz, G. (1978). Estimating the dimension of a model. *The Annals of Statistics*, *6*, 461–464.

Spiegelhalter, D. J., Best, N. G., Carlin, B. R., & van der Linde, A. (2002). Bayesian measures of model complexity and fit. *Journal of the Royal Statistical Society B*, *64*, 583–616.

Watanabe, S. (2010). Asymptotic equivalence of Bayes cross validation and widely applicable information criterion in singular learning theory. *Journal of Machine Learning Research*, *11*, 3571–3594.

Watanabe, S. (2013). A widely applicable Bayesian information criterion. *Journal of Machine Learning Research*, *14*, 867–897.

Chapter 3
Advanced Bayesian Computation

Abstract The popularity of Bayesian statistics is largely due to advances in computing and developments in computational methods. Currently, there are two main types of Bayesian computational methods. The first type involves iterative Monte Carlo simulation and includes the Gibbs sampler, the Metropolis-Hastings algorithm, Hamiltonian sampling etc. These first type methods typically generate a Markov chain whose stationary distribution is the target distribution. The second type involves distributional approximation and includes Laplace approximation (Laplace 1785, 1810), variational Bayes (Jordan et al. 1999), etc. These second type methods try to find a distribution with the analytical form that best approximates the target distribution. In Sect. 3.1, we review Markov chain Monte Carlo (MCMC) methods including the general Metropolis-Hastings algorithm (M-H), Gibbs sampler with conjugacy, and Hamiltonian Monte Carlo (HMC) algorithm (Neal 1994). Section 3.2 discusses the convergence and efficiency of the above sampling methods. We then show how to specify a Bayesian model and draw model inferences using OpenBUGS and Stan in Sect. 3.3. Section 3.4 provides a brief summary on the mode-based approximation methods including Laplace approximation and Bayesian variational inference. Finally, in Sect. 3.5, a full Bayesian analysis is performed on a biological data set from Gelfand et al. (1990). The key concepts and the computational tools discussed in this chapter are demonstrated in this section.

3.1 Markov Chain Monte Carlo (MCMC) Methods

In Sect. 2.1, we discussed how to make inferences about parameters from the posterior distribution. When the posterior distribution is complicated, it is tedious to make any inferences analytically. We have seen that in Example 2.3 the marginal posterior distribution $p(\lambda|\boldsymbol{y})$ contains a complicated integral. Even if $p(\lambda|\boldsymbol{y})$ can be found analytically, it still requires some effort to get the exact posterior mean and the CPDR of λ. This motivates us to explore other methods.

Monte Carlo simulation is a sampling process from a target distribution. Once sufficient samples are obtained, the inferences of target distribution can be approximated by sample statistics, such as sample mean, sample standard error, sample

© Springer Nature Singapore Pte Ltd. 2018

G. Gao, *Bayesian Claims Reserving Methods in Non-life Insurance with Stan*,
https://doi.org/10.1007/978-981-13-3609-6_3

percentile etc. The traditional Monte Carlo simulation methods involve inversing the cumulative distribution function, the *rejection sampling* method, etc. These methods generate independent samples. In contrast, *Markov chain Monte Carlo* (MCMC) methods generate a Markov chain whose *stationary distribution* is equivalent to the target distribution. In MCMC, the next sampled value typically depends on the previous sampled value.

In this section, we first briefly state some properties of a Markov chain with a stationary distribution. Then the *Metropolis-Hastings (M-H) algorithm*, *Gibbs sampler* and *Hamiltonian Monte Carlo (HMC)* are reviewed. Throughout this section, we continue with Example 2.3. We compare the MC-based inferences to analytical inferences.

3.1.1 Markov Chain and Its Stationary Distribution

Let $\mathscr{S}$ be a finite set. A *Markov chain* is characterized by a *transition matrix* $\boldsymbol{K}(s, s')$ with $\boldsymbol{K}(s, s') \geq 0$ for any $s, s' \in \mathscr{S}$, and $\sum_{s' \in \mathscr{S}} \boldsymbol{K}(s, s') = 1$ for any $s' \in \mathscr{S}$. All of the Markov chains considered in this chapter have a *stationary distribution* $\pi(s)$ which satisfies the equation

$$\sum_{s \in \mathscr{S}} \pi(s)\boldsymbol{K}(s, s') = \pi(s').$$

The stationary theorem of Markov chains says, under a simple *connectedness condition*, π is unique, and high powers of $\boldsymbol{K}$ converge to a rank one matrix with all rows equal to π. That is

$$\boldsymbol{K}^n(s, s') \to \pi(s') \text{ for } s, s' \in \mathscr{S}.$$

The probabilistic content of the theorem is that from any starting state s, the nth step of a run of the Markov chain has a chance close to $\pi(s')$ of being at s' if n is large. In computational settings, when the cardinality of $\mathscr{S}$ is large, it is easy to move from s to s' according to $\boldsymbol{K}(s, s')$, but it is hard to sample from π directly.

Example 3.1 (The stationary distribution of a Markov chain process). Suppose a Markov chain with the sample space $\mathscr{S} = \{0, 1, 2, 3\}$, and a transition matrix as follows:

$$\boldsymbol{K} = \begin{pmatrix} 0.9 & 0.1 & 0 & 0 \\ 0.9 & 0 & 0.1 & 0 \\ 0.9 & 0 & 0 & 0.1 \\ 0.9 & 0 & 0 & 0.1 \end{pmatrix}.$$

A little more calculation shows that

$$\boldsymbol{K}^4 = \begin{pmatrix} 0.9 \; 0.09 \; 0.009 \; 0.001 \\ 0.9 \; 0.09 \; 0.009 \; 0.001 \\ 0.9 \; 0.09 \; 0.009 \; 0.001 \\ 0.9 \; 0.09 \; 0.009 \; 0.001 \end{pmatrix},$$

so that for some $m \geq 4$, $\boldsymbol{K}^m(s_1, s') = \boldsymbol{K}^m(s_2, s')$ for all $s_1, s_2 \in \mathscr{S}$. It follows that $\boldsymbol{K}^{m+1} = \boldsymbol{K}^m$, since

$$\boldsymbol{K}^{m+1}(s, s') = \sum_{v \in \mathscr{S}} \boldsymbol{K}(s, v)\, \boldsymbol{K}^m\left(v, s'\right) = \boldsymbol{K}^m(s, s') \sum_{v \in \mathscr{S}} \boldsymbol{K}(s, v) = \boldsymbol{K}^m(s, s').$$

Therefore, $\lim_{n\to\infty}\boldsymbol{K}^n(s, s') = \boldsymbol{K}^m(s, s') = \pi(s')$, where we write the final equality without reference to s since all the rows of $\boldsymbol{K}^m(s, s')$ are identical. $\pi(s')$ is the stationary distribution.

3.1.2 Single-Component Metropolis-Hastings (M-H) Algorithm

Suppose we want to simulate a sample of θ. When θ contains multiple variables, instead of sampling the whole θ at a time, it is often more convenient and computationally efficient to divide θ into components as $\{\theta_1, \theta_2, \ldots, \theta_h\}$, and sample these components one by one, i.e., using single-component *Metropolis-Hastings algorithm.*

An iteration of the single-component Metropolis-Hastings algorithm comprises h updating processes. Suppose θ is updated sequentially according to the component index and the target multivariate distribution as π. The ith updating process for θ_i at the tth iteration in the M-H algorithm works as follows:

1. Draw a value from a proposal distribution of θ_i, $g_i\left(\theta_i^* | \theta_i^{t-1}, \theta_{-i}^t\right)$, where $\theta_{-i}^t = \left\{\theta_1^{t+1}, \ldots, \theta_{i-1}^{t+1}, \theta_{i+1}^t, \ldots, \theta_h^t\right\}$, and θ_i^{t-1} denotes the value of θ_i at the end of iteration $t-1$ or denotes the initial value when $t = 1$.
2. Calculate the acceptance ratio

$$A_i\left(\theta_i^*, \theta_i^{t-1}\right) = \frac{\pi\left(\theta_i^* | \theta_{-i}^t\right) g_i\left(\theta_i^{t-1} | \theta_i^*, \theta_{-i}^t\right)}{\pi\left(\theta_i^{t-1} | \theta_{-i}^t\right) g_i\left(\theta_i^* | \theta_i^{t-1}, \theta_{-i}^t\right)},$$

   where $\pi\left(\cdot | \theta_{-i}^t\right)$ is the full conditional distribution of θ_i.
3. Accept θ_i^* and set $\theta_i^t = \theta_i^*$ with probability $A_i\left(\theta_i^*, \theta_i^{t-1}\right)$. Otherwise, reject θ_i^* and set $\theta_i^t = \theta_i^{t-1}$.

Note that the parameters in the proposal distribution g_i are called *tuning parameter*; these are specified in advance and will affect the acceptance rates and the convergence. In Sect. 3.3, we will see that OpenBUGS has a phase called "adapting", when the program automatically chooses the appropriate tuning parameters. In the M-H algorithm, we need to discard the first few iterations, which are called *burn-in*. We judge the length of burn-in by looking at trace plots, BGR plots (Gelman and Rubin 1992) or potential scale reduction factor (Gelman et al. 2014), which will be discussed in Sect. 3.2.1.

Example 3.2 (An autoregressive process of order one). We continue with Example 2.3 on page 15. Now we complete the following tasks:

1. Write a M-H algorithm to generate a sample of size $T = 1000$ from the joint posterior distribution $p(\alpha, \lambda|\boldsymbol{x})$, and produce trace plots for α and λ. Also calculate the acceptance rates for both parameters.
2. Draw histograms for the sampled values in (1) and superimpose density estimates of marginal posterior distributions, $p(\alpha|\boldsymbol{x})$ and $p(\lambda|\boldsymbol{x})$. Estimate the posterior means $\hat{\alpha}$ and $\hat{\lambda}$ and give the 95% confidence intervals for them. Also report the 95% CPDR estimates for α and λ.

Solutions to (1):

Instead of using $p(\alpha, \lambda|\boldsymbol{x})$ directly, we take the logarithm of it, denoted by $l(\alpha, \lambda|\boldsymbol{x})$, and calculate the acceptance ratio on a logarithm scale. The tth iteration in the M-H algorithm is as follows:

1. Draw a proposed value $\alpha^* \sim \mathrm{U}(\alpha - c, \alpha + c)$. If $\alpha^* \notin [-1, 1]$, reject it and redraw. Otherwise, calculate the acceptance ratio:

$$A_\alpha\left(\alpha^*, \alpha^{t-1}\right) = \exp\left[l\left(\alpha^*|\boldsymbol{x}, \lambda^{t-1}\right) - l\left(\alpha^{t-1}|\boldsymbol{x}, \lambda^{t-1}\right)\right],$$

   where α^{t-1} and λ^{t-1} are the values at the end of the $(t-1)$th iteration or the initial values when $t = 1$. If $A_\alpha\left(\alpha^*, \alpha^{t-1}\right) \geq 1$, accept α^* and set $\alpha^t = \alpha^*$. If $A_\alpha\left(\alpha^*, \alpha^{t-1}\right) \leq 1$, accept α^* and set $\alpha^t = \alpha^*$ with probability of $A_\alpha\left(\alpha^*, \alpha^{t-1}\right)$; otherwise, set $\alpha^t = \alpha^{t-1}$.
2. Draw a proposed value $\lambda^* \sim \mathrm{U}(\alpha - d, \alpha + d)$. If $\lambda^* < 0$, reject it and redraw. Otherwise, calculate the acceptance ratio:

$$A_\lambda\left(\lambda^*, \lambda^{t-1}\right) = \exp\left[l\left(\lambda^*|\boldsymbol{x}, \alpha^t\right) - l\left(\lambda^{t-1}|\boldsymbol{x}, \alpha^t\right)\right],$$

   where α^t comes from step 1. If $A_\lambda\left(\lambda^*, \lambda^{t-1}\right) \geq 1$, accept λ^* and set $\lambda^t = \lambda^*$. If $A_\lambda\left(\lambda^*, \lambda^{t-1}\right) \leq 1$, accept λ^* and set $\lambda^t = \lambda^*$ with probability of $A_\lambda\left(\lambda^*, \lambda^{t-1}\right)$; otherwise, set $\lambda^t = \lambda^{t-1}$.

With $c = 0.3,\ d = 0.2,\ \alpha^0 = 0,\ \lambda^0 = 1$, the M-H algorithm converges within 100 iterations with acceptance rate of 71% for α and 69% for λ over a total of 10,000 iterations. The trace plots for α and λ are shown in Fig. 3.1.

Solutions to (2):

The last 9,900 sampled values are used for inference. The MC estimate of posterior mean $\hat{\alpha}$ is $\bar{\alpha} = (\sum_{t=101}^{10000} \alpha^t)/9900 = 0.4721$, with the 95% CI

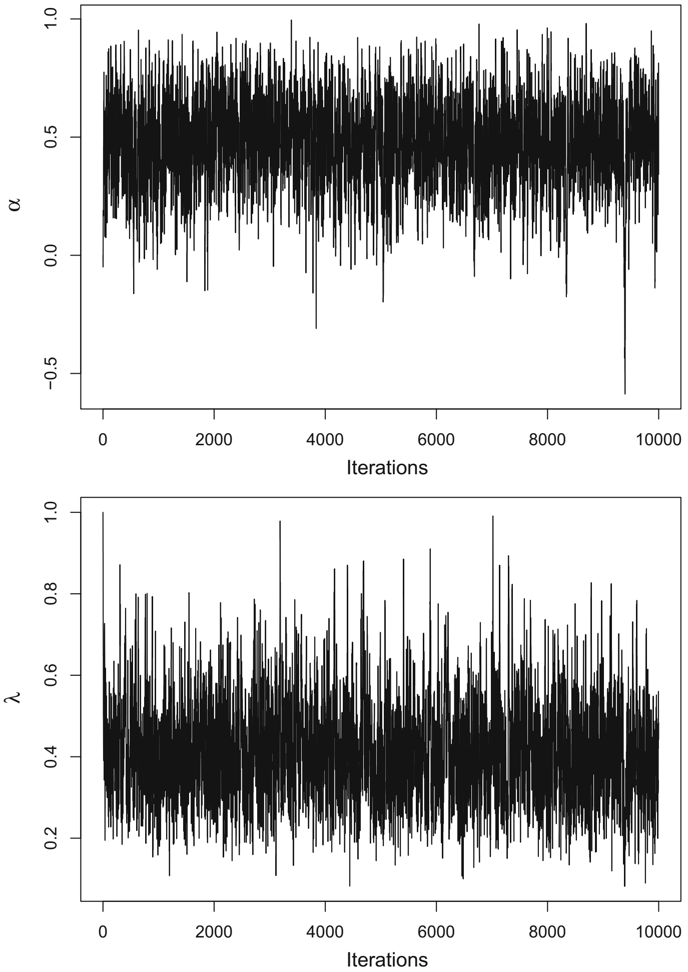


Fig. 3.1 The trace plots of α and λ

$$\left(\bar{\alpha} - 1.96\sqrt{\frac{\mathrm{Var}\,(\alpha)}{9900}},\ \ \bar{\alpha} + 1.96\sqrt{\frac{\mathrm{Var}\,(\alpha)}{9900}}\right) = (0.4683, 0.4759)\,,$$

where $\mathrm{Var}\,(\alpha)$ is the MC sample variance (i.e., the sample variance of α^t, $t = 101, \ldots, 10000$). The MC estimate of 95% CPDR for α is (0.0726, 0.8188).

Similarly, the MC estimate of posterior mean $\hat{\lambda}$ is $\bar{\lambda} = (\sum_{t=101}^{10000} \lambda^t)/9900 = 0.4101$, with the 95% CI $(0.4075, 0.4126)$. The MC estimate of 95% CPDR for λ is $(0.1947, 0.6959)$. We show the MC histograms and the MC density estimates comparing with the exact densities in Fig. 3.2. We can see the MC estimates are quite close to the exact values.

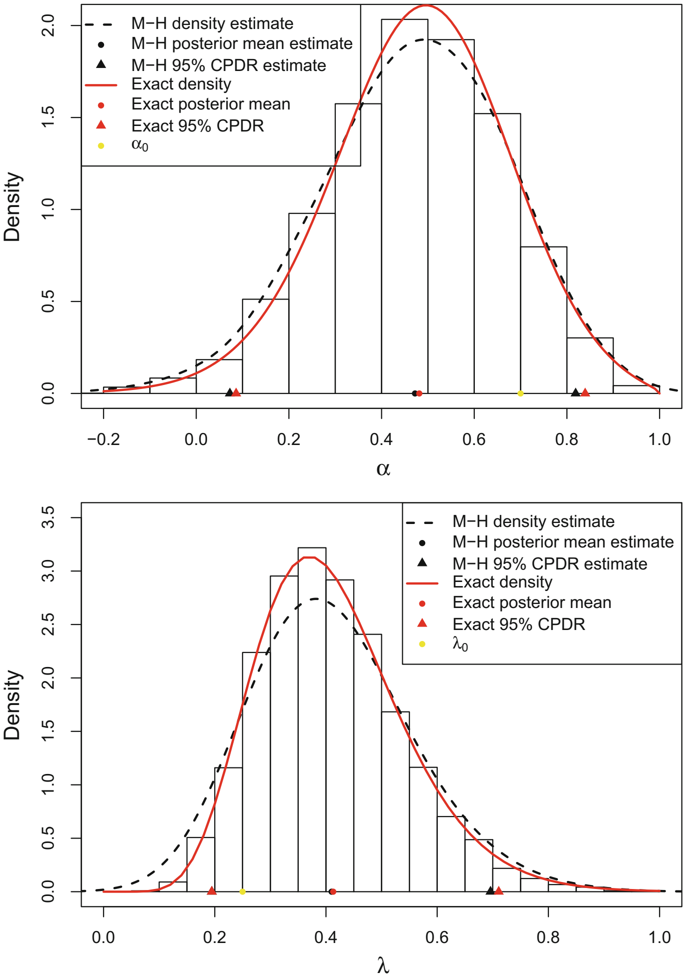


Fig. 3.2 The MC estimates of α and λ using M-H

Since there is strong series dependence in a Markov chain, it is not good to make inferences directly from the original MCMC sample. Two methods can be applied to reduce the dependence: the *batch means* (BM) method and the *thinning sample* (TS) method. We will discuss these two methods in more detail in Sect. 3.2.2. In the batch means method we place 20 bins and in the thinning sample method we extract one value from every 20 successive samples. Table 3.1 lists the inferences made from the two methods. Note that * indicates the exact posterior mean is in the 95% CI.

Table 3.1 The MC, BM, TS estimates of the posterior means and the associated 95% CIs using the M-H algorithm

	MC est.	MC CI	BM CI	TS CI	Exact
$\hat{\alpha}$	0.4721	(0.4683, 0.4759)	(0.4598, 0.4845)*	(0.4461, 0.4800)	0.4814
$\hat{\lambda}$	0.4101	(0.4075, 0.4126)	(0.4047, 0.4154)*	(0.3982, 0.4208)	0.4129

3.1.3 Gibbs Sampler

The *Gibbs sampler* is another MCMC method which simulates the joint distribution via full conditional distributions. In fact, if we choose the full conditional distribution of each component in single-component M-H algorithm as the proposal distribution for this component, i.e., $g_i\left(\theta_i^*|\theta_i^{t-1},\theta_{-i}^t\right)=\pi\left(\theta_i^*|\theta_{-i}^t\right)$, the acceptance ratio will be

$$A_i\left(\theta_i^*,\theta_i^{t-1}\right)=\frac{\pi\left(\theta_i^*|\theta_{-i}^t\right)\pi\left(\theta_i^{t-1}|\theta_{-i}^t\right)}{\pi\left(\theta_i^{t-1}|\theta_{-i}^t\right)\pi\left(\theta_i^*|\theta_{-i}^t\right)}=1,$$

which guarantees the proposed value θ_i^* being accepted. So the Gibbs sampler is a special case of the M-H algorithm.

Compared with the M-H algorithm, the Gibbs sampler does not have the accept-reject step and tuning parameters. However, the main difficulty with the Gibbs sampler is simulating from the full conditional distribution which sometimes does not have a recognizable form. In that case, we may turn to other sampling methods such as *adaptive rejection sampling* (Gilks and Wild 1992); see details in Appendix B on page 196.

Adaptive rejection sampling is a generalized rejection sampling method that can be used to simulate for any univariate log-concave probability density function. As sampling proceeds, the rejection envelope and the squeezing function converge to the target function. The adaptive rejection sampling and the M-H algorithm are both intended for the situation where there is non-conjugacy of the Gibbs sampler in a Bayesian model.

Example 3.3 (An autoregressive process of order one). We continue with Example 2.3. The Gibbs sampler is applied to the joint posterior distribution of α and λ. The full conditional distributions are

$$p\left(\alpha|\boldsymbol{x},\lambda\right)\propto\left(1-\alpha^2\right)^{\frac{1}{2}}\exp\left[-\frac{\lambda}{2}h\left(\boldsymbol{x},\alpha\right)\right]$$

$$\lambda|\boldsymbol{x},\alpha\sim\text{Gamma}\left(\frac{n}{2},\frac{h\left(\boldsymbol{x},\alpha\right)}{2}\right).$$

The full conditional distribution of α is unrecognisable. We can write a Gibbs sampler for λ and keep the M-H algorithm for α.

Table 3.2 The MC, BM, TS estimates of the posterior means and the associated 95% CIs using a Gibbs sampler

	MC est.	MC CI	BM CI	TS CI	Exact
$\hat{\alpha}$	0.477	(0.473, 0.480)	(0.466, 0.487)	(0.451, 0.484)	0.481
$\hat{\lambda}$	0.413	(0.411, 0.416)	(0.411, 0.416)	(0.398, 0.420)	0.413
$\hat{x}_{21}$	0.363	(0.331, 0.395)	(0.329, 0.396)	(0.210, 0.491)	0.352

To simulate x_{21}, we add an extra step to every iteration: draw a value from $\mathrm{N}\left(\alpha^t x_{20}, 1/\lambda_j{}^t\right)$, where $\alpha_j{}^t, \lambda_j{}^t$ are the ending values at the tth iteration. Similar to Table 3.1, we can obtain the new MC estimates based on the Gibbs sampler as shown in Table 3.2. Another way to find the posterior mean and the posterior marginal density is to apply the *Rao-Blackwell (RB) method*. We can estimate the marginal posterior distribution of λ as

$$\bar{p}\left(\lambda|\boldsymbol{x}\right) = \frac{1}{T}\sum_{t=1}^{T}\mathrm{Gamma}\left(\lambda\left|\frac{n}{2}, \frac{h\left(\boldsymbol{x}, \alpha^t\right)}{2}\right.\right),$$

where α^t is the tth sampled value from the posterior distribution $p\left(\alpha|\boldsymbol{x}\right)$. The posterior mean is estimated as

$$\bar{\lambda} = \frac{1}{T}\sum_{t=1}^{T}\frac{n}{h\left(\boldsymbol{x}, \alpha^t\right)}.$$

The 95% CI for posterior mean is calculated as $(\bar{\lambda} \pm 1.96s/\sqrt{T})$, where s is the sample standard deviation of $n/h\left(\boldsymbol{x}, \alpha^t\right),\ t = 1, \ldots, T$.

Similarly, we can estimate the posterior predictive distribution of x_{21} as

$$\bar{p}\left(x_{21}|\boldsymbol{x}\right) = \frac{1}{T}\sum_{t=1}^{T}p\left(x_{20}|\boldsymbol{x}, \alpha^t, \lambda^t\right) = \frac{1}{T}\sum_{t=1}^{T}N\left(x_{21}\left|\alpha^t x_{20}, \frac{1}{\lambda^t}\right.\right).$$

The posterior mean of x_{21} is estimated as

$$\bar{x}_{21} = \frac{1}{T}\sum_{t=1}^{T}\alpha^t x_{20}.$$

The 95% CI for the posterior mean is calculated as $\left(\bar{x}_{21} \pm 1.96s/\sqrt{T}\right)$, where s is the sample standard deviation of $\alpha^t x_{20}$, $t = 1, \ldots, T$. We summarize the Rao-Blackwell estimates in Fig. 3.3. We see that the 95% RB CIs cover the exact posterior means, and the RB density estimate of λ is almost equal to its exact density.

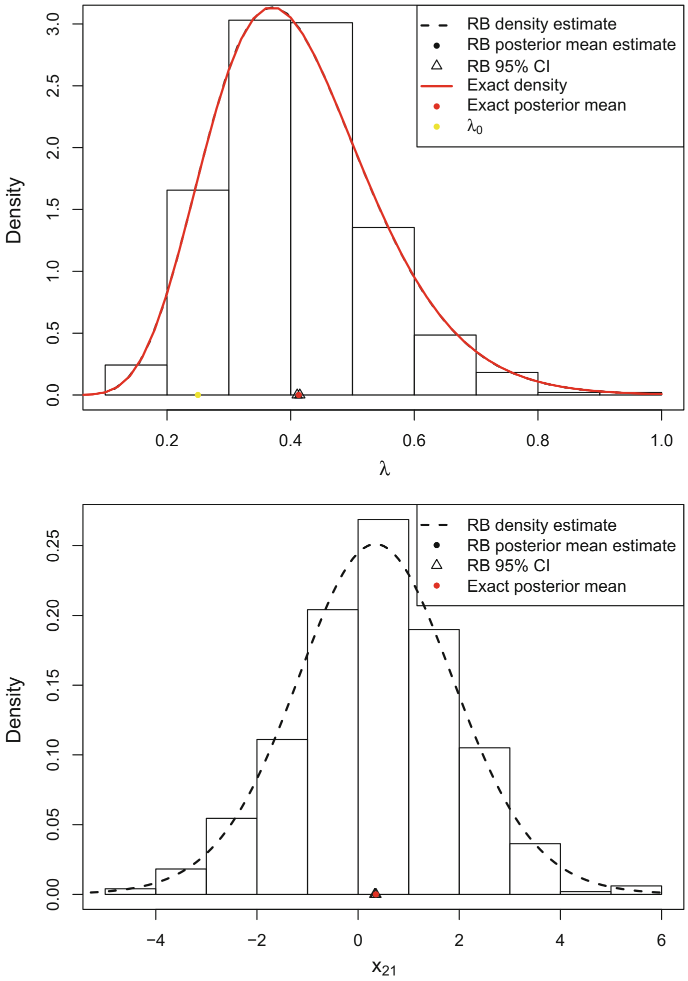


Fig. 3.3 The Rao-Blackwell estimates of λ and x_{21}

3.1.4 Hamiltonian Monte Carlo (HMC)

Hamiltonian Monte Carlo (HMC) was introduced to physics by Duane et al. (1987) and to statistical problems by Neal (1994, 2011). In contrast to the random-walk Metropolis algorithm where the proposed value is not related to the target distribution, HMC proposes a value by computing a *trajectory* according to Hamiltonian dynamics that takes account of the target distribution.

3.1.4.1 Hamiltonian Dynamics

Suppose we have a Hamiltonian dynamics scenario in which a frictionless ball slides over a surface of varying height. The state of the ball at any time consists of the *position* and the *momentum*. Denote the position by a h vector θ and the momentum by a same length vector ϕ.

Hamiltonian functions can be written as

$$H(\theta, \phi) = U(\theta) + K(\phi),$$

where $U(\theta)$ is called the *potential energy* and will be defined to be minus the log probability density of the distribution of θ we wish to simulate, $K(\phi)$ is called the *kinetic energy* and is usually defined as

$$K(\phi) = \phi^T \boldsymbol{\Sigma}^{-1} \phi/2,$$

where $\boldsymbol{\Sigma}^{-1}$ is a symmetric positive-definite "mass matrix" which is typically diagonal and is often a scalar multiple of the identity matrix. This form of $K(\phi)$ corresponds to the minus log probability density of the zero-mean Gaussian distribution with covariance matrix $\boldsymbol{\Sigma}$.

The state of the ball in the next *infinitesimal time* is determined by Hamilton's equations of motion:

$$\begin{aligned}\frac{d\theta_i}{dt} &= \frac{\partial H}{\partial \phi_i} = [\boldsymbol{\Sigma}^{-1}\phi]_i \\ \frac{d\phi_i}{dt} &= -\frac{\partial H}{\partial \theta_i} = -\frac{\partial U}{\partial \theta_i}.\end{aligned}$$

For computer implementation, Hamilton's equation must be approximated by discretizing time, using some small *step size*, ε. The most straightforward method is *Euler's method*. The solution to the above system of differential equations can be approximated by Euler's method as follows:

$$\begin{aligned}\phi_i(t+\varepsilon) &= \phi_i(t) + \varepsilon \frac{d\phi_i}{dt}(t) = \phi_i(t) - \varepsilon \frac{\partial U}{\partial \theta_i}(\theta_i(t)) \\ \theta_i(t+\varepsilon) &= \theta_i(t) + \varepsilon \frac{d\theta_i}{dt}(t) = \theta_i(t) + \varepsilon[\boldsymbol{\Sigma}^{-1}\phi]_i.\end{aligned}$$

However, Euler's method does not preserve the *volume* and the resulting trajectory would diverge from the exact trajectory to infinity. A better trajectory may be generated by using the *leapfrog method* as follows:

$$\begin{aligned}
\phi_i(t+\varepsilon/2) &= \phi_i(t) - (\varepsilon/2)\frac{\partial U}{\partial \theta_i}(\theta_i(t)) \\
\theta_i(t+\varepsilon) &= \theta_i(t) + \varepsilon[\boldsymbol{\Sigma}^{-1}\phi(t+\varepsilon/2)]_i \\
\phi_i(t+\varepsilon) &= \phi_i(t+\varepsilon/2) - (\varepsilon/2)\frac{\partial U}{\partial \theta_i}(\theta_i(t+\varepsilon)).
\end{aligned}$$

The leapfrog method preserves volume exactly.

3.1.4.2 MCMC from Hamiltonian Dynamics

Suppose we want to simulate a sample from the target density $p(\theta)$. HMC introduces *auxiliary* momentum variables ϕ and draws from a joint density $p(\theta, \phi)$. We assume the auxiliary density is a multivariate Gaussian distribution, independent of the parameter θ. The covariance matrix $\boldsymbol{\Sigma}$ acts as a *Euclidean metric* to rotate and scale the target distribution. The joint density $p(\theta, \phi)$ defines a Hamiltonian function as follows:

$$H(\theta, \phi) = -\log p(\theta, \phi) = -\log p(\theta) - \log p(\phi) = U(\theta) + K(\phi).$$

Starting from the value of the parameters at the end of the $t-1$th iteration, θ^{t-1}, a new value θ^* is proposed by two steps before being subjected to a Metropolis accept step.

First, a value ϕ^{t-1} for the momentum is drawn from the multivariate Gaussian distribution, $\mathrm{N}(0, \boldsymbol{\Sigma})$. Next the joint system $(\theta^{t-1}, \phi^{t-1})$ is evolved via the following leapfrog method for L steps to get the proposed value (θ^*, ϕ^*):

$$\begin{aligned}
\phi_i^{t-1+\varepsilon/2} &= \phi_i^{t-1} + (\varepsilon/2)\frac{\partial \log p(\theta^{t-1})}{\partial \theta_i} \\
\theta_i^{t-1+\varepsilon} &= \theta_i^{t-1} + \varepsilon[\boldsymbol{\Sigma}^{-1}\phi^{t-1+\varepsilon/2}]_i \\
\phi_i^{t-1+\varepsilon} &= \phi_i^{t-1+\varepsilon/2} + (\varepsilon/2)\frac{\partial \log p(\theta^{t-1+\varepsilon})}{\partial \theta_i}.
\end{aligned}$$

Note that $\theta^* = \theta^{t-1+\varepsilon L}, \phi^* = \phi^{t-1+\varepsilon L}$. If there were no numerical errors in the leapfrog step (i.e., the leapfrog trajectory followed the exact trajectory), we would accept (θ^*, ϕ^*) definitely. However, there are always errors given the non-zero step size. Hence, we conduct a *Metropolis accept step* with the acceptance rate as

$$\min\{1, \exp[H(\theta^{t-1}, \phi^{t-1}) - H(\theta^*, \phi^*)]\}.$$

Neal (1994) suggests that HMC is optimally efficient when its acceptance rate is approximately 65% while the multi-dimensional M-H algorithm is optimal at an acceptance rate of around 23%.

3.1.4.3 The No-U-Turn Sampler (NUTS)

There are three tuning parameters in HMC: the mass matrix $\boldsymbol{\Sigma}$, the step size ε and the number of steps L. If ε is too large, the resulting trajectory will be inaccurate and too many proposals will be rejected. If ε is too small, too many steps will be taken by the leapfrog method, leading to long simulation times per iteration. If L is too small, the trajectory traced out will be too short and sampling will devolve to a random walk. If L is too large, the algorithm will spend too much time in one iteration. The mass matrix $\boldsymbol{\Sigma}$ needs to be comparable with the covariance of the posterior.

In MCMC, all the tuning parameters should be fixed during the simulation that will be used for inference; otherwise the algorithm may converge to the wrong distribution. BUGS has an adaptive period during which suitable tuning parameters are selected.

NUTS (Homan and Gelman 2014) can dynamically adjust the number of leapfrog steps at each iteration to send the trajectory as far as it can go during that iteration. If such a rule is applied alone, the simulation will not converge to the desired target distribution. The full NUTS is more complicated, going backward and forward along the trajectory in a way that satisfies *detailed balance* (Gelman et al. 2014).

The programming of NUTS is much more complicated than a M-H algorithm. We rely on *Stan* to implement NUTS inferential engine. More details of Stan are provided in Sect. 3.3. Along with this algorithm, Stan can automatically optimize ε to match an acceptance rate target and estimate $\boldsymbol{\Sigma}$ based on warm up iterations. Hence we do not need to specify any tuning parameters in Stan.

3.2 Convergence and Efficiency

Two concerns in MCMC methods are checking the convergence of sampled values and designing an efficient algorithm.

3.2.1 Convergence

We can detect the convergence by eye, relying on the trace plots such as Fig. 3.1. Informally speaking, a “fat hairy caterpillar” appearance indicates the convergence. For numerical diagnosis, we use the *Brooks-Gelman-Rubin (BGR) ratio* and *potential scale reduction factor*, both of which are based on the mixture and stationarity of simulated multiple chains starting from diversified initial values.

3.2.1.1 The Brooks-Gelman-Rubin (BGR) Ratio

The numerical diagnosis for convergence in OpenBUGS is based on comparing within- and between- chain variability (Gelman and Rubin 1992). Suppose we simulate I chains, each of length J, with a view to assessing the degree of stationarity in the final $J/2$ iterations. We take the width of $100\,(1-\alpha)\,\%$ credible interval for the parameter of interest as a measure of *posterior variability*.

From the final $J/2$ iterations we calculate the width of empirical $100\,(1-\alpha)\,\%$ credible interval for each chain as $W_i, \quad i=1,\ldots,I$, then find the average width across these chains as $W=\sum_{i=1}^{I} W_i/m$. We also pool IJ iterations together and find the pooled width B.

The BGR ratio is defined as the ratio of pooled interval widths to average interval widths, $\hat{R}_{\text{BGR}}=B/W$. It should be larger than 1 if the starting values are suitably diversified and will tend to be 1 as convergence is approached. So we can assume convergence for practical purposes if $\hat{R}_{\text{BGR}}<1.05$.

Brooks and Gelman (1998) further suggested splitting the total iteration range of each chain into M batches of length $a=J/M$ and calculating $B(m)$, $W(m)$ and $\hat{R}_{\text{BGR}}(m)$ based on the latter halves of iterations $(1,\ldots,ma)$ for $m=1,\ldots,M$.

3.2.1.2 The Potential Scale Reduction Factor

Gelman et al. (2014) propose a similar quantity to monitor the convergence, namely *potential scale reduction factor*. This factor is automatically monitored in Stan. Again, suppose we simulate I chains, each of length J (this is all after discarding the burn-in iterations). We split each chain into two parts to get $2I$ batches, each of length $J/2$. We label the simulations as $\theta_{i,j}, i=1,\ldots,2I, j=1,\ldots,J/2$ and calculate the between- and within- batch variances as a measure of posterior variability rather than the width of credible interval as in BGR.

The average within-batch variance is $W_{\text{Var}}=\sum_{i=1}^{2I} s_i^2$, where s_i^2 is the sample variance of the ith batch. The between-batch variance is

$$B_{\text{Var}}=\frac{J/2}{2I-1}\sum_{i=1}^{2I}\left(\bar{\theta}_{i\cdot}-\bar{\theta}\right)^2,$$

where $\bar{\theta}_{i\cdot}$ is the sample mean of ith batch and $\bar{\theta}$ is the pooled sample mean. The reason for containing a factor of $J/2$ is that B_{Var} is based on the sample variance of batch means $\bar{\theta}_{i\cdot}$. Note that W_{Var} and B_{Var} are both estimates of the posterior variance $\text{Var}\,(\theta|\boldsymbol{y})$. Later we will show that $\sqrt{B_{\text{Var}}/IJ}$ is the MC standard error of the posterior mean estimate using batch-mean method.

Gelman et al. (2014) proposes an estimate of $\text{Var}\,(\theta|\boldsymbol{y})$ as a weighted average of W_{Var} and B_{Var}:

$$\widetilde{\text{Var}}\,(\theta|\boldsymbol{y})=\frac{J/2-1}{J}W_{\text{Var}}+\frac{1}{J}B_{\text{Var}},$$

which is also an unbiased estimate under stationarity, but an overestimate if involving the burn-in iterations. On the other hand, W_{Var} always underestimates Var $(\theta|\boldsymbol{y})$ due to limited sample size $J/2$ and dependent iterations. So we monitor convergence by estimating the potential scale reduction factor by

$$\hat{R} = \sqrt{\frac{\widetilde{\text{Var}}\,(\theta|y)}{W_{\text{Var}}}}, \tag{3.1}$$

which declines to 1 as $J \to \infty$. If $\hat{R}$ is high, we believe that more iterations are needed to guarantee the convergence.

3.2.2 *Efficiency*

For a given sample size, the accuracy of our inferences is dependent on the efficiency of the posterior sample, which decreases with an increasing level of autocorrelation. We can improve the efficiency by refining the algorithm or resampling from the MC sample to reduce the correlation.

3.2.2.1 Reparameterization, Thinning and Adding Auxiliary Variables

One way of increasing efficiency is to reparameterize the model so that the posterior correlation among parameters is reduced, as shown in Example 3.4 and Sect. 3.5.4.

Another way to improve efficiency is to perform a process known as *thinning*, whereby only every vth value from the MC sample is actually retained for inference. In Sect. 3.3, we will see there is an option of “thin” in the OpenBUGS Update Tool window.

Finally, the Gibbs sampler can often be simplified or the convergence can be accelerated by adding an *auxiliary* variable (Gelman et al. 2014).

3.2.2.2 The Batch Means Method

In Example 2.3 we are interested in the 95% CI of the posterior mean. The standard error of the posterior mean estimate (i.e., the MC sample mean) is calculated by the sample standard deviation over the squared root of sample size. This follows the *central limit theorem* (CLT) under the condition of independent samples. However, the MC sample is from a Markov chain and each sampled value depends on the previous sampled value. The MC sample variance is not an accurate estimate of the posterior variance Var $(\theta|\boldsymbol{y})$. We will turn to the batch means method to get a more accurate estimate.

Suppose we have I chains, each of length J, and split every chain into M batches, each of length J/M, and J/M is sufficiently large that CLT holds for each batch. We label the simulations as $\theta_{ij}, i = 1, \ldots, IM, j = 1, \ldots, J/M$.

We calculate the batch means $\bar{\theta}_{i\cdot}$, which are roughly independent and identically distributed with a mean of posterior mean and a variance of posterior variance over J/M. Then we can use the sample variance of batch means to estimate the posterior variance as

$$\widehat{\text{Var}\,(\theta|\boldsymbol{y})} = \frac{J/M}{IM-1}\sum_{i=1}^{IM}\left(\bar{\theta}_{i\cdot} - \bar{\theta}\right)^2.$$

The standard error of the posterior mean estimate $\bar{\theta} = \sum_{ij}\theta_{ij}/(IJ)$ can be approximated more accurately by

$$\sqrt{\frac{\widehat{\text{Var}\,(\theta|\boldsymbol{y})}}{IJ}} = \sqrt{\frac{1}{IM\,(IM-1)}\sum_{i=1}^{IM}\left(\bar{\theta}_{i\cdot} - \bar{\theta}\right)^2}, \tag{3.2}$$

which is also called the Monte Carlo standard error given in the “MC_error” column in OpenBUGS output and the “se_mean” column in Stan. Using the batch means method, the 95% CI of $\hat{\theta}$ is modified as $\left(\bar{\theta} \pm 1.96\sqrt{\widehat{\text{Var}\,(\theta|\boldsymbol{y})}/(IJ)}\right)$.

3.2.2.3 Effective Sample Size

Gelman et al. (2014) defined an estimate of *effective sample size* as

$$n_{\text{eff}} = \frac{IJ}{1 + 2\sum_{t=1}^{\infty}\rho_t}, \tag{3.3}$$

where I, J follow the notation in batch means method and ρ_t is the autocorrelation of the MC sample at lag t. Stan automatically monitors n_{eff} for each parameter of interest and gives them in the column of `n_eff`.

Example 3.4 (Reparameterize a simple linear regression model). Consider the simple linear regression model: $y_i \sim \text{N}\left(a + bx_i, \sigma^2\right), i = 1, \ldots, 30,$ with true parameters $a = 17, b = 3, \sigma^2 = 16$. Assume $\boldsymbol{x} = (0.5, 1.0, \ldots, 15)$ and generate a response vector $\boldsymbol{y} = (y_1, y_2, \ldots, y_{30})$. We assume a non-informative prior, i.e., $p\left(a, b, \sigma^2\right) \propto 1/\sigma^2$.

Gibbs sampler (1):

A Gibbs sampler which could be applied here is based on the following full conditional distributions:

$$
\begin{aligned}
a|\cdot &\sim \mathrm{N}\left(\frac{1}{n}\sum_{i=1}^{n}\left(y_i - bx_i\right), \frac{\sigma^2}{n}\right)\\
b|\cdot &\sim \mathrm{N}\left(\frac{\sum_{i=1}^{n} x_i\left(y_i - a\right)}{\sum_{i=1}^{n} {x_i}^2}, \frac{\sigma^2}{\sum_{i=1}^{n} {x_i}^2}\right)\\
\sigma^2|\cdot &\sim \text{ Inv-Gamma}\left(\frac{n}{2}, \frac{\sum_{i=1}^{n}\left(y_i - a - bx_i\right)^2}{2}\right).
\end{aligned}
$$

The dependence of $p\left(a|\cdot\right)$ on b makes the Gibbs sampler (1) ineffective, especially for σ^2. We reparameterize the simple linear regression model as

$$
y_i \sim \mathrm{N}\left(c + b\left(x_i - \bar{\boldsymbol{x}}\right), \sigma^2\right),
$$

where $c = a + b\bar{\boldsymbol{x}}$. The prior for c can be shown as $\mathrm{N}\left(a + b\bar{\boldsymbol{x}}, \infty\right)$. So c also has a non-informative flat prior.

Gibbs sampler (2):

An alternative Gibbs sampler is based on the following full conditional distributions:

$$
\begin{aligned}
c|\cdot &\sim \mathrm{N}\left(\frac{1}{n}\sum_{i=1}^{n} y_i, \frac{\sigma^2}{n}\right)\\
b|\cdot &\sim \mathrm{N}\left(\frac{\sum_{i=1}^{n}\left(x_i - \bar{\boldsymbol{x}}\right) y_i}{\sum_{i=1}^{n}\left(x_i - \bar{\boldsymbol{x}}\right)^2}, \frac{\sigma^2}{\sum_{i=1}^{n}\left(x_i - \bar{\boldsymbol{x}}\right)^2}\right)\\
\sigma^2|\cdot &\sim \text{Inv-Gamma}\left(\frac{n}{2}, \frac{\sum_{i=1}^{n}\left(y_i - c - bx_i + b\bar{\boldsymbol{x}}\right)^2}{2}\right),
\end{aligned}
$$

where $p\left(c|\cdot\right)$ does not depend on b and $p(b|\cdot)$ is not dependent on c. The independence between full conditional distributions will make Gibbs sampler (2) more effective than Gibbs sampler (1).

We compare the MC estimates and the least-squares estimates in Table 3.3. Gibbs sampler (2) improves the MC estimates of posterior means $\hat{\sigma}^2$, $\hat{y}'$, while performing equally well for $\hat{a}$ and $\hat{b}$ as Gibbs sampler (1).

Table 3.3 Comparison of the least-squared estimates with the MC estimates using different Gibbs samplers

Estimation method	$\hat{\sigma}^2$	95% CPDR	$\hat{y}'$	95% CI/CPDR
L-S estimates	22.81	NA	32.83	(22.82, 42.84)
Gibbs sampler (1)	34.35	(15.26, 86.18)	32.95	(20.56, 45.44)
Gibbs sampler (2)	24.56	(14.30, 41.55)	32.81	(22.67, 43.10)

3.3 OpenBUGS and Stan

The MCMC methods are useful Bayesian model computation tools, especially when the posterior distribution does not have a closed form. The programming of MCMC requires a lot of effort, even for a simple linear regression model as in Example 3.4. Moreover, we need to customize a MCMC algorithm for every model. To relieve the burden of programming MCMC, several packages have been developed. The two main statistical packages are BUGS and Stan. We will see in this section how to use these two packages to do a Bayesian analysis.

3.3.1 OpenBUGS

BUGS stands for Bayesian inference Using Gibbs Sampler. The BUGS project began in 1989 and has developed into two versions: WinBUGS and OpenBUGS. Currently all development is focused on OpenBUGS. As its name suggests, OpenBUGS uses a Gibbs sampler which updates unknown quantities one by one, based on their full conditional distribution.

The MCMC building blocks include the conjugacy Gibbs sampler, the M-H algorithm, various types of *rejection sampling* and *slice sampling*; see details in Appendix B on page 197. Such methods are used only as a means of updating full conditional distributions within a Gibbs sampler. OpenBUGS has an "expert system", which determines an appropriate MCMC method for analysing a specified model.

3.3.1.1 Directed Graphical Models

Suppose we have a set of quantities $\mathscr{G}$ arranged as a *direct acyclic* graph, in which each quantity $\upsilon \in \mathscr{G}$ represents a node in the graph. The "intermediate" nodes always have "parents" and "descendants". The relationship between parent and child can be logical or stochastic. If it is a logical relationship, the value of the node is determined exactly by its parents. If it is a stochastic relationship, the value of the node is generated by a distribution which is determined only by its parents.

Conditional on its parents, denoted by $\mathrm{pa}\,[\upsilon]$, υ is independent of all the other nodes except its descendants, denoted by $\mathrm{ch}\,[\upsilon]$. This conditional independence assumption implies the joint distribution of all the quantities $\mathscr{G}$ has a simple factorization in terms of the conditional distribution $p(\upsilon|\mathrm{pa}\,[\upsilon])$, as follows:

$$p\,(\mathscr{G}) = \prod_{\upsilon\in\mathscr{G}} p(\upsilon|\mathrm{pa}\,[\upsilon]),$$

where the conditional distribution may degenerate to a logical function of its parents if the relationship is logical. The full joint distribution $p(\mathscr{G})$ can be fully specified by the parent-child relationships.

The crucial idea behind BUGS is that this factorization forms the basis for both the model description and the computational methods. The Gibbs sampler for each unknown quantity θ_i, is based on the following full conditional distribution:

$$p(\theta_i|\theta_{-i}, \boldsymbol{y}) \propto p(\theta_i|\text{pa}[\theta_i]) \times \prod_{\upsilon \in \text{ch}[\theta_i]} p(\upsilon|\text{pa}[\upsilon]) .$$

Note that θ_i can be any unknown quantities, not just unknown parameters. An important implication of directed graphical models is that every node should appear in the left side of an assignment sign only once. This implication can be used as a debugging tool of the BUGS language.

3.3.1.2 The BUGS Language

For a complex model, it is better to use the BUGS language to specify the model rather than using a graphical model. It takes time for R users to get used to BUGS. The fundamental difference is the *declarative* language in BUGS, so it does not matter in which order the statements come in BUGS.

3.3.2 Stan

Stan stands for Sampling Through Adaptive Neighbourhoods, which applies the no-U-turn sampler (NUTS). Besides the no-U-turn sampler, Stan can also approximate Bayesian inference using *variational Bayes*, which will be discussed in Sect. 3.4.2, and do penalized maximum likelihood estimation if we specify the priors as the penalized term.

The key steps of the algorithm include data and model input, computation of the log posterior density (up to an arbitrary constant that cannot depend on the parameters in the model) and its gradients, a warm-up phase in which the tuning parameters, ε and M, are set, an implementation of NUTS to move through the parameter space, convergence monitoring, and inferential summaries at the end.

Compared with OpenBUGS, Stan works seamlessly with R. Stan is installed as a package in R. The output from Stan is stored in R automatically and can be analyzed and plotted in R directly. Instead, BUGS works by itself. BUGS has its own graph tools and output form. The output from BUGS needs to be transferred into another package such as R before it can be used for further analysis.

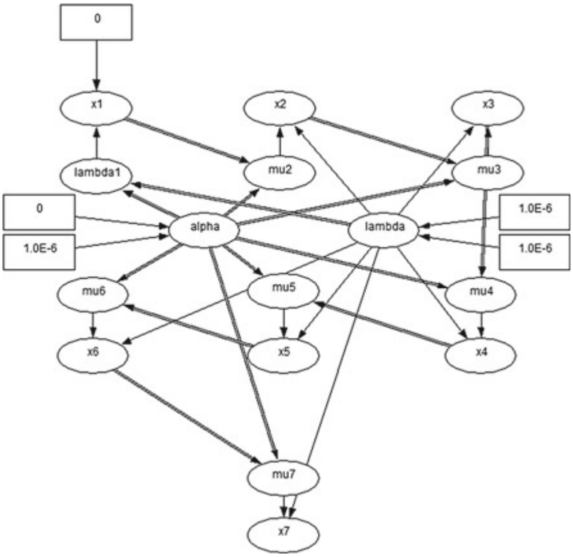


Fig. 3.4 The graphical model for AR(1)

Stan can analyze all the BUGS examples. It provides more instructive error messages than BUGS. This is particularly helpful when we work with a "black box" inferential engine. Stan can solve the multi-level models with unknown covariance matrices which BUGS can not easily deal with. Moreover, it is easier to specify the constraints of parameters in Stan.

Example 3.5 (An autoregressive process of order one). We continue with Example 3.2. Rather than programming the MCMC, we rely on BUGS and Stan to make inference.

BUG:

A graphical model (also called a *Doodle*) representation is shown in Fig. 3.4. For the simplicity, we only assume 6 observations. The single arrows imply stochastic relationship, while double arrows imply logical relationship. A "parent" constant is denoted by a squared plate, while other nodes are denoted by ellipse plates. The BUGS can generate codes from graphical model by using "pretty print" under "model" menu.

The modelling procedure using BUGS language typically includes the following steps:

1. Check the syntax of the model specification by "Specification Tool"; if the model is correctly specified, the message "model is syntactically correct" will appear on the bottom left of the screen.

2. Read in the following data by clicking “load data”:

```
list(K=20,x=
c(-0.58196581,-1.70339058,-4.29434356,-2.00495593,
-0.09234224,-1.56433489,-0.49151508,-1.55912920,
-0.90546327,-1.31576285,-1.12240668, 0.50931757,
 0.54899741,-1.87582922,-4.54187225,-0.41553845,
 0.31656492,-0.32832899, 1.69457825, 0.73050020)).
```

   The message “data loaded” will appear.
3. Specify the number of chains as 2 and compile the model. The message “model compiled” will appear.
4. Load the following initial values:

```
list(alpha=-0.99,lambda=100)
list(alpha=0.99, lambda=0.001).
```

   The message “model initialised” or “initial values loaded but chain contains uninitialised variables” will appear. In the second case, we need to click “gen inits”, which will generate initial values from priors.
   After compiling and loading data, BUGS will choose an appropriate MCMC method for each unknown quantity, which is shown under the menu “Info/Updater types”.
5. Start the simulation using “Update Tool”. We have the following options:
   - Thin: Every *k*th iteration will be used for inference.
   - Adapting: This will be ticked while the M-H or slice sampling is in its initial tuning phase where some optimization parameters are tuned.
   - Over relax: This generates multiple samples at each iteration and then selects one that is negatively correlated with the current value. The within-chain correlations should be reduced.
6. Monitor the interested unknown quantities using “Sample Monitor Tool”. Typing * into the node box means all monitored nodes.
7. Diagnose the convergence via “bgr diag” plots and trace plots shown in Fig. 3.5. MCMC converges after 750 iterations, so we can rely on the subsequent iterations to make inferences.
8. Report the inferences. We can get the inference by clicking “stats” in “Sample Monitor Tool” window. OpenBUGS also automatically reports DIC, p_D, $\widehat{D(\theta)}$ (shown as “Dbar”), and $D(\hat{\theta})$ (shown as “Dhat”). In this example, p_D is close to the number of parameters. See the following output:

```
      Dbar    Dhat    DIC     pD
x     75.83   73.86   77.81   1.975
total 75.83   73.86   77.81   1.975
```

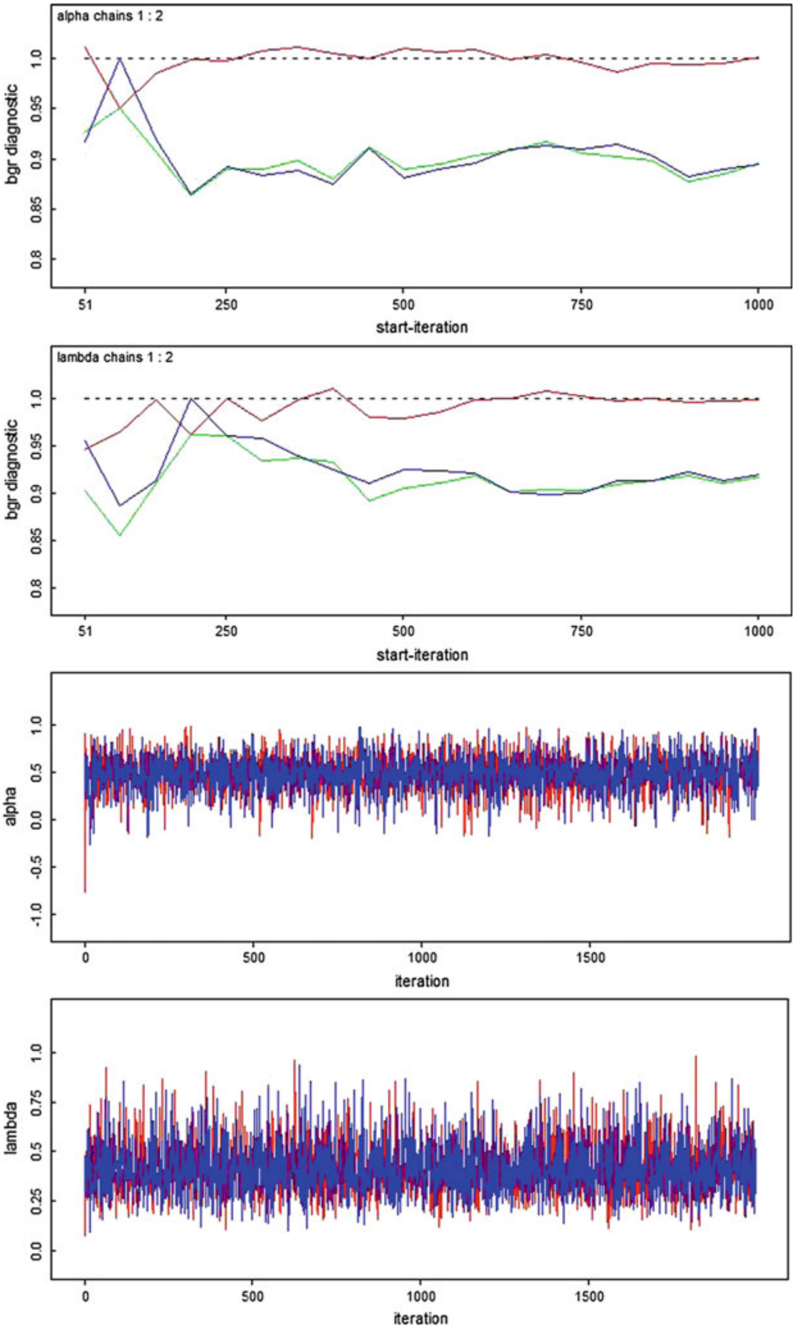


Fig. 3.5 The BGR plots and the trace plots of α and λ from OpenBUGS

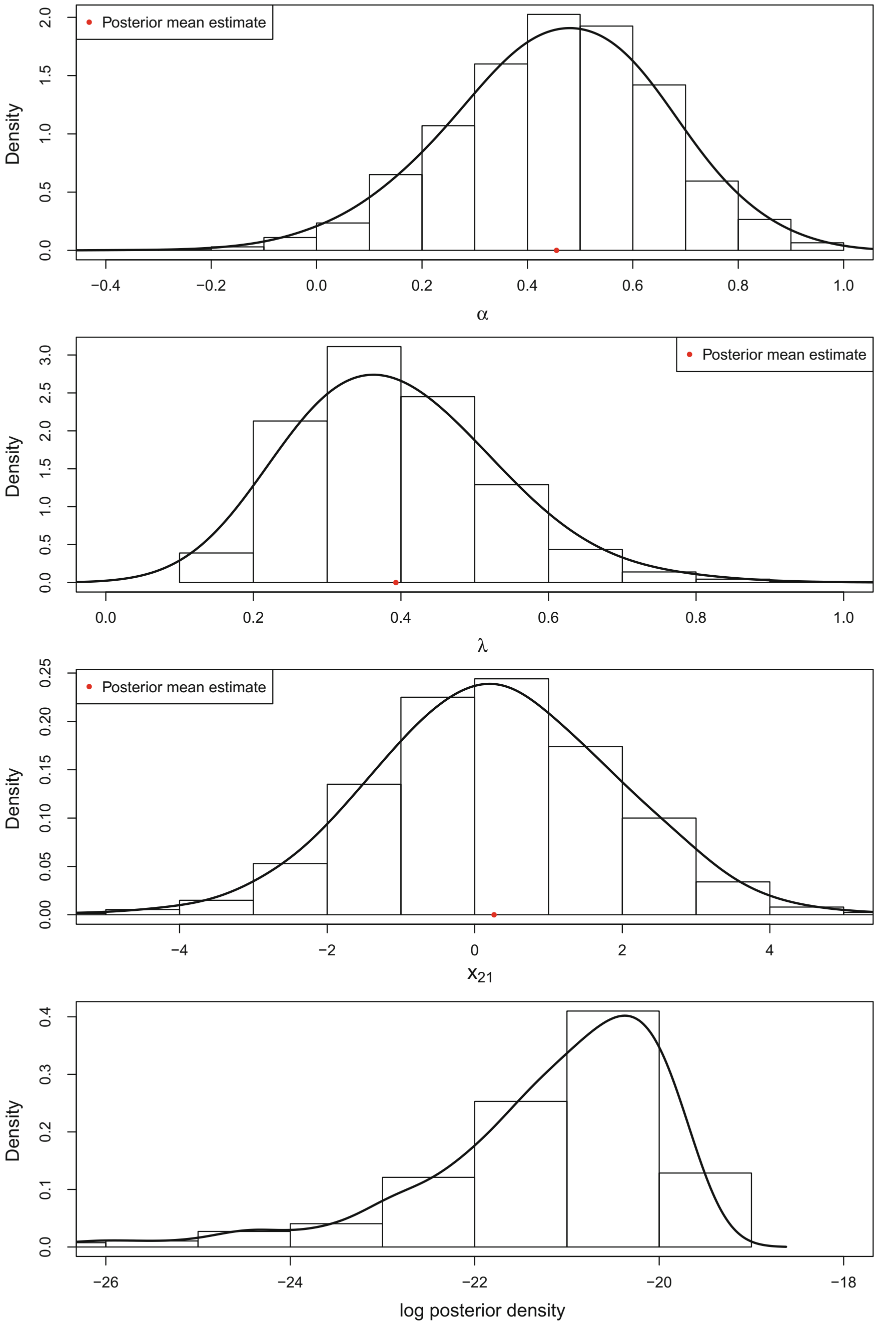


Fig. 3.6 The MC estimates of α, λ and log posterior density from Stan

Stan:

Programming in Stan is more flexible and easier than in BUGS. For example, there is no need to specify flat priors, logical operators are allowed in stochastic expressions, constraints are easily incorporated, and there are more instructive error messages. The Stan code is as follows:

```
modelcode<-"
data{
  int<lower=0> J;
  real x[J];
}
parameters{
  real<lower=-1,upper=1> alpha;
  real<lower=0> sigma;
  real x21;
}
transformed parameters{
  real<lower=0> sigma1;
  sigma1<-sigma/(1-alpha^2);
}
model{
  x[1] ~ normal(0,sigma1);
  for (j in 2:J) x[j] ~ normal (alpha*x[j-1],sigma);
  x21 ~ normal(alpha*x[J],sigma);
}
generated quantities{
  real <lower=0> lambda;
  lambda<-1/sigma^2;
}
"
stanmodel<-stan_model(model_code=modelcode)
J=length(x20)
dat<-list(J=J,x=x20)
fit<-stan(model_code=modelcode, data=dat, iter=1000, chains=4)
print(fit,par=c("alpha","lambda","x21"))
## Inference for Stan model: modelcode.
## 4 chains, each with iter=1000; warmup=500; thin=1;
## post-warmup draws per chain=500, total post-warmup draws=2000.
##
##          mean se_mean   sd   2.5%    25%    50%    75%  97.5%
    n_eff Rhat
## alpha    0.45    0.01 0.19   0.05   0.33   0.47   0.59   0.80
      853    1
## lambda   0.39    0.00 0.13   0.18   0.30   0.38   0.47   0.70
     1079    1
## x21      0.39    0.05 1.70  -2.96  -0.71   0.35   1.53   3.73
      962    1
##
## Samples were drawn using NUTS(diag_e) at Thu Sep 10 23:23:28
    2018.
## For each parameter, n_eff is a crude measure of effective
    sample size,
## and Rhat is the potential scale reduction factor on split
    chains (at
## convergence, Rhat=1).
```

Note that `x20` is the data. We run `iter=1000` iterations for each of four chains. By default, Stan discards the first half of each chain as burn-in. In the output, the last row is normalized log posterior density. The `se_mean` column contains the MC errors defined in Eq. (3.2). The last two columns correspond to `n_eff` and `Rhat`, which we defined in Eqs. (3.3) and (3.1). The posterior densities of α, λ, x_{21} and log posterior density are shown in Fig. 3.6, which are similar to Figs. 3.2 and 3.3.

3.4 Modal and Distributional Approximations

The joint posterior modes can be found using the `optimizing( )` function in Stan, which applies the Broyden-Fletcher-Goldfarb-Shanno (BFGS) algorithm (Nocedal and Wright 2006). Conditional maximization Newton's method can also find the posterior joint modes. For the marginal modes, a well-known method is the expectation-maximization (EM) algorithm.

3.4.1 Laplace Approximation

Once the posterior mode is found, we can approximate the target distribution by a multivariate Gaussian distribution with the same mode and covariance matrix as the inverse of the *log posterior density curvature* at the mode. This approximation works well for large sample sizes following the asymptotic theory discussed in Sect. 2.1.4.

3.4.2 Variational Inference

When facing a difficult problem for which we cannot give an exact solution, we typically have two alternatives. One is to stick to this problem and give an approximation to the exact answer. That is what the MCMC methods do. We approximate the exact posterior distribution using a Markov chain. The other is to introduce a closely similar problem for which we can give an exact answer. That is what the variational inference tries to do.

We introduce an approximate distribution family q that is easier to deal with than $p(\theta|\boldsymbol{y})$. The log model evidence $\log p(\boldsymbol{y})$ can be written as follows:

$$\begin{aligned}
\log p(\boldsymbol{y}) &= \log \frac{p(\boldsymbol{y},\theta)}{p(\theta|\boldsymbol{y})} \\
&= \int q(\theta) \log \frac{p(\boldsymbol{y},\theta)}{p(\theta|\boldsymbol{y})} d\theta \\
&= \int q(\theta) \left(\log \frac{q(\theta)}{p(\theta|\boldsymbol{y})} + \log \frac{p(\boldsymbol{y},\theta)}{q(\theta)} \right) d\theta \\
&= \int q(\theta) \log \frac{q(\theta)}{p(\theta|\boldsymbol{y})} d\theta + \int q(\theta) \log \frac{p(\boldsymbol{y},\theta)}{q(\theta)} d\theta \\
&= \mathrm{KL}[q||p] + F(q,\boldsymbol{y}),
\end{aligned}$$

where the first term in the last line is called the *Kullback-Leibler divergence* between $q(\theta)$ and $p(\theta|\boldsymbol{y})$, and the second term is called free energy. If we want to find an approximate distribution q to minimize $\mathrm{KL}[q||p]$, we can just maximize the free energy since the model evidence is a constant given the sample.

3.4.2.1 Mean Field Variational Inference

A common choice of $q(\theta)$ is to assume it can be factorized into independent partitions:

$$q(\theta) = \prod_{i=1}^{h} q_i(\theta_i).$$

This assumption is called *mean field assumption*. Under this assumption, if we dissect out the dependence on $q_k(\theta_k)$, then the free energy can be written as

$$\begin{aligned}
F(q, \boldsymbol{y}) &= \int q(\theta) \log \frac{p(\boldsymbol{y}, \theta)}{q(\theta)} d\theta \\
&= \int \prod_{i=1}^{h} q_i(\theta_i) \times \left(\log p(\boldsymbol{y}, \theta) - \sum_{i=1}^{h} \log q_i(\theta_i) \right) d\theta \\
&= \int q_k(\theta_k) \prod_{i \neq k} q_i(\theta_i) \times \left[\log p(\boldsymbol{y}, \theta) - \log q_k(\theta_k) \right] d\theta \\
&\quad - \int q_k(\theta_k) \prod_{i \neq k} q_i(\theta_i) \sum_{i \neq k} \log q_i(\theta_i) d\theta \\
&= \int q_k(\theta_k) \left(\int \prod_{i \neq k} q_i(\theta_i) \log p(\boldsymbol{y}, \theta) d\theta_{-k} - \log q_k(\theta_k) \right) d\theta_k \\
&\quad - \int q_k(\theta_k) \left(\int \prod_{i \neq k} q_i(\theta_i) \sum_{i \neq k} \log q_i(\theta_i) d\theta_{-k} \right) d\theta_k \\
&= \int q_k(\theta_k) \log \frac{\exp\{\mathrm{E}_{\theta_{-k}} \log p(\boldsymbol{y}, \theta)\}}{q_k(\theta_k)} d\theta_k + C \\
&= -\mathrm{KL}\left[q_k(\theta_k) || \exp\{\mathrm{E}_{\theta_{-k}} \log p(\boldsymbol{y}, \theta)\} \right] + C.
\end{aligned}$$

Then the approximate distribution $q_k(\theta_k)$ that maximizes the free energy is given by

$$q_k^* = \underset{q_k}{\operatorname{argmax}}\ F(q, \boldsymbol{y}) = \frac{\exp\{\mathrm{E}_{\theta_{-k}} \log p(\boldsymbol{y}, \theta)\}}{Z}.$$

This implies a straightforward algorithm for variational inference. Assume the parameters in distribution q_k are ϕ_k. The algorithm consists of the following two steps:

- Determine the form of the approximating distribution. Average $\log p(\boldsymbol{y}, \theta)$ over $q_{-k}(\theta_{-k})$ to find the marginal approximate distribution q_k^*, whose parameters are some function of parameters in q_{-k}, ϕ_{-k}.
- Iterative update ϕ. The first step establishes the a circular dependence among ϕ_i. We iterate ϕ until there are no more visible changes and use the last update $q(\theta|\phi)$ as an approximation to $p(\theta|\boldsymbol{y})$.

3.5 A Bayesian Hierarchical Model for Rats Data

We have seen a hierarchical model in Example 2.8. A hierarchical model is often used when considering the variations on different levels. For most hierarchical models, the posterior distribution does not have a closed form. We compute Bayesian inference via programming a MCMC algorithm or using BUGS/Stan.

In this section, we reanalyze the rats' weights data set shown in Table 3.4, and extend the work by Gelfand et al. (1990) and Lunn et al. (2000). The data set contains the weights of 60 rats measured weekly for 5 weeks. The first 30 rats are under control while the rest are under treatment. Our interest is the effect of treatment on the growth rates and on the growth volatility.

In Sect. 3.5.1, a classical fixed effects model and a random effects model are considered. In Sects. 3.5.2 and 3.5.3, two Bayesian hierarchical models are used. The advantages of Bayesian models are the accommodation of parameters uncertainties and the inherent hierarchical structure. We turn to Stan to do model inference in this section. In Sect. 3.5.4, we reparameterize the univariate normal hierarchical model to propose a more efficient Gibbs sampler as we did in Example 3.4.

3.5.1 Classical Regression Models

We first fit a fixed effects model, then a random effects model with rat IDs as group levels. We will see that the random effects model is better at capturing the two levels of variation: between-rat variation and within-rat variation.

Table 3.4 The rats' weights measured at the end of each week (Gelfand et al. 1990)

Rat id.	8 days	15 days	22 days	29 days	36 days
1	151	199	246	283	320
...	...	...	...	...	...
60	136	177	223	256	287

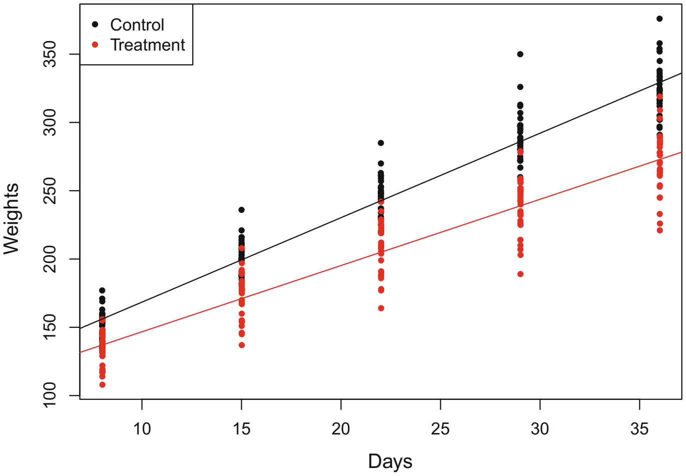


Fig. 3.7 Two regression lines for the control and treatment groups

3.5.1.1 Two Regression Lines

We fit one regression line to each of the control group and the treatment group respectively. Figure 3.7 roughly shows the negative effect of the treatment on the weights.

3.5.1.2 A Random Effects Model

As we saw in Fig. 3.7, after considering the effect of treatment, there still remains variation between different rats, so we may fit the following *random effects model*:

$$y_{ij} = \alpha_0 + \beta_0 x_j + \alpha_1 1_{\text{treat}}(i) + \beta_1 x_j 1_{\text{treat}}(i) + a_i + b_i x_j + \varepsilon_{ij}$$

$$a_i \sim \mathrm{N}\left(0, \sigma_\alpha^2\right), b_i \sim \mathrm{N}\left(0, \sigma_\beta^2\right), \varepsilon_{ij} \sim \mathrm{N}\left(0, \sigma^2\right),$$

where i indicates the ith rat, j indicates the jth week; α_0, β_0 are the population intercept and slope for the control group; α_1, β_1 are the incremental population intercept and the slope for the treatment group; a_i, b_i are the random intercepts and the slopes for the ith rats; and x_j is the days until the jth week (i.e., $x_1 = 8, \ldots, x_5 = 36$).

In the random effects model, we effectively separate the residual variation from the fixed effects model into two parts: the variation in random effects, measured by σ_α^2, σ_β^2, and the variation in residuals, measured by σ^2. We compare the residuals from the fixed effects model and the random effects model in Fig. 3.8. In the random effects model, the variation of residuals is largely reduced, and the residuals for each rat are closer to a normal distribution. Note that red dots indicate the means of residuals for each rat.

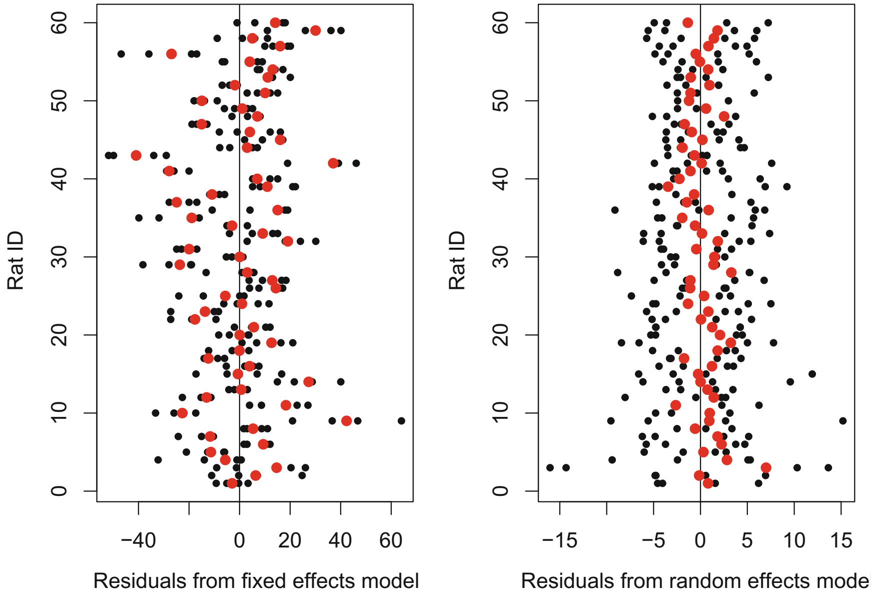


Fig. 3.8 Residuals from the fixed effects model and the random effects model

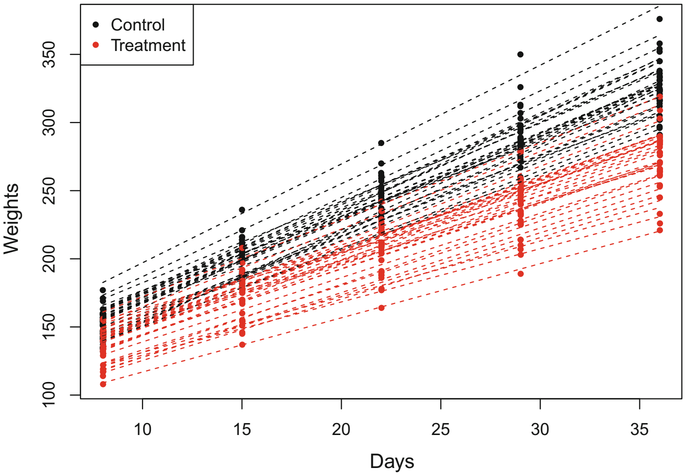


Fig. 3.9 Fitted lines in the random effects model

We draw the fitted lines for each rat in Fig. 3.9. In the random effects model, the fitted values for the ith rat are obtained by adding the population fitted values (based only on the fixed effects estimates) and the estimated contributions of the random effects to the fitted values. The resulting values estimate the *best linear unbiased predictions (BLUPs)* for the ith rat.

One interest is the effect of treatment on the growth rate, measured by β_1. The summary output shows the significant negative effect of treatment on the growth rate. Another interest is whether a rat with higher birth weight will grow faster. A Pearson correlation test of intercepts and slopes shows there is no significant relationship between birth weights and growth rates.

3.5.2 A Bayesian Bivariate Normal Hierarchical Model

A Bayesian bivariate normal hierarchical model is used to fit both control and treatment groups as follows:

$$\begin{aligned}
y_{ij} &\sim \mathrm{N}\left(\alpha_i + \beta_i x_j, \sigma_c^2\right), i = 1, \ldots, 30 \\
y_{ij} &\sim \mathrm{N}\left(\alpha_i + \beta_i x_j, \sigma_t^2\right), i = 31, \ldots, 60 \\
\begin{pmatrix}\alpha_i \\ \beta_i\end{pmatrix} &\sim \mathrm{N}\left(\begin{pmatrix}\alpha \\ \beta\end{pmatrix}, \Sigma_{\boldsymbol{c}}\right), i = 1, \ldots, 30 \\
\begin{pmatrix}\alpha_i \\ \beta_i\end{pmatrix} &\sim \mathrm{N}\left(\begin{pmatrix}\alpha + \Delta\alpha \\ \beta + \Delta\beta\end{pmatrix}, \boldsymbol{\Sigma_t}\right), i = 31, \ldots, 60 \\
\boldsymbol{\Sigma_c}, \boldsymbol{\Sigma_t} &\sim \text{Inv-Wishart}\left(\begin{pmatrix}200 & 0 \\ 0 & 0.2\end{pmatrix}^{-1}, 2\right),
\end{aligned} \tag{3.4}$$

where α, β, $\Delta\alpha$, $\Delta\beta$, σ_c^2, σ_t^2 have non-informative priors.

We are interested in the effect of treatment on the growth rate, $\Delta\beta$, the variation ratio of treatment group to control group, σ_t/σ_c, and the correlation between growth rates and born weights (for either control group, i.e., ρ_c, or treatment group, i.e., ρ_t), $\rho_{c/t} = \boldsymbol{\Sigma}_{c/t}[1,2] / \sqrt{\boldsymbol{\Sigma}_{c/t}[1,1]\,\boldsymbol{\Sigma}_{c/t}[2,2]}$. The Stan code is as follows:

```
rats_code1<-"
data{
  int   <lower=8, upper=36> day[5];
  real  <lower=0>           weights[60,5];
}
parameters{
  vector[2]       ab[60];
  vector[2]       ab_ave;
  vector[2]       ab_treat;
  real <lower=0>  sigmaC;
  real <lower=0>  sigmaT;
  cov_matrix[2]   cov_ave;
  cov_matrix[2]   cov_treat;
}
model{
  for (j in 1:30)   ab[j] ~ multi_normal(ab_ave, cov_ave);
  for (j in 31:60)  ab[j] ~ multi_normal(ab_ave+ab_treat, cov_
      treat);
  for (j in 1:30)
```

```
    for (t in 1:5)  weights[j,t] ~ normal(ab[j,1]+ab[j,2]*day[t], sigmaC);
  for (j in 31:60)
    for (t in 1:5)  weights[j,t] ~ normal(ab[j,1]+ab[j,2]*day[t], sigmaT);
}
generated quantities{
  real           TC_sigma;
  matrix[2,2]    TC_ab;
  vector[300]    log_lik;
  vector[300]    dev_res;
  vector[300]    fitted;
  real           rhoC;
  real           rhoT;
  real           D;
  TC_sigma   <- sigmaT/sigmaC;
  TC_ab      <- cov_treat ./ cov_ave;
  rhoC       <- cov_ave[1,2]/sqrt(cov_ave[1,1]*cov_ave[2,2]);
  rhoT       <- cov_treat[1,2]/sqrt(cov_treat[1,1]*cov_treat[2,2]);
  for (j in 1:30){
    for (t in 1:5) {
      log_lik[5*(j-1)+t] <- normal_log(weights[j,t], ab[j,1] + ab[j,2] * day[t], sigmaC);
      dev_res[5*(j-1)+t] <- (weights[j,t] - ab[j,1] - ab[j,2] * day[t]) / sigmaC;
      fitted[5*(j-1)+t]  <- ab[j,1] + ab[j,2] * day[t];
    }
  }
  for (j in 31:60){
    for (t in 1:5) {
      log_lik[5*(j-1)+t] <- normal_log(weights[j,t], ab[j,1] + ab[j,2] * day[t], sigmaT);
      dev_res[5*(j-1)+t] <- (weights[j,t] - ab[j,1] - ab[j,2] * day[t]) / sigmaC;
      fitted[5*(j-1)+t]  <- ab[j,1] + ab[j,2] * day[t];
    }
  }
  D <- sum(-2*log_lik);
}
"
```

In Stan, we simulate four chains, each of 400 iterations, and discard the first halves. The MC estimates are shown in Table 3.5. The MC estimated posterior densities of interested quantities are drawn in Fig. 3.10. According to Table 3.5 and Fig. 3.10, we make the following conclusion: the effect of treatment on the growth rates is negative, i.e., the CPDR of $\Delta\beta = \beta_t - \beta_c$ is negative; the treatment group is less volatile, i.e., the CPDR of σ_t/σ_c is less than 1; there is no significant relationship between born weights and growth rates for either group, i.e., the CPDRs of ρ_c and ρ_t contain 0. Finally, Fig. 3.11 validates the assumption of normal error distribution.

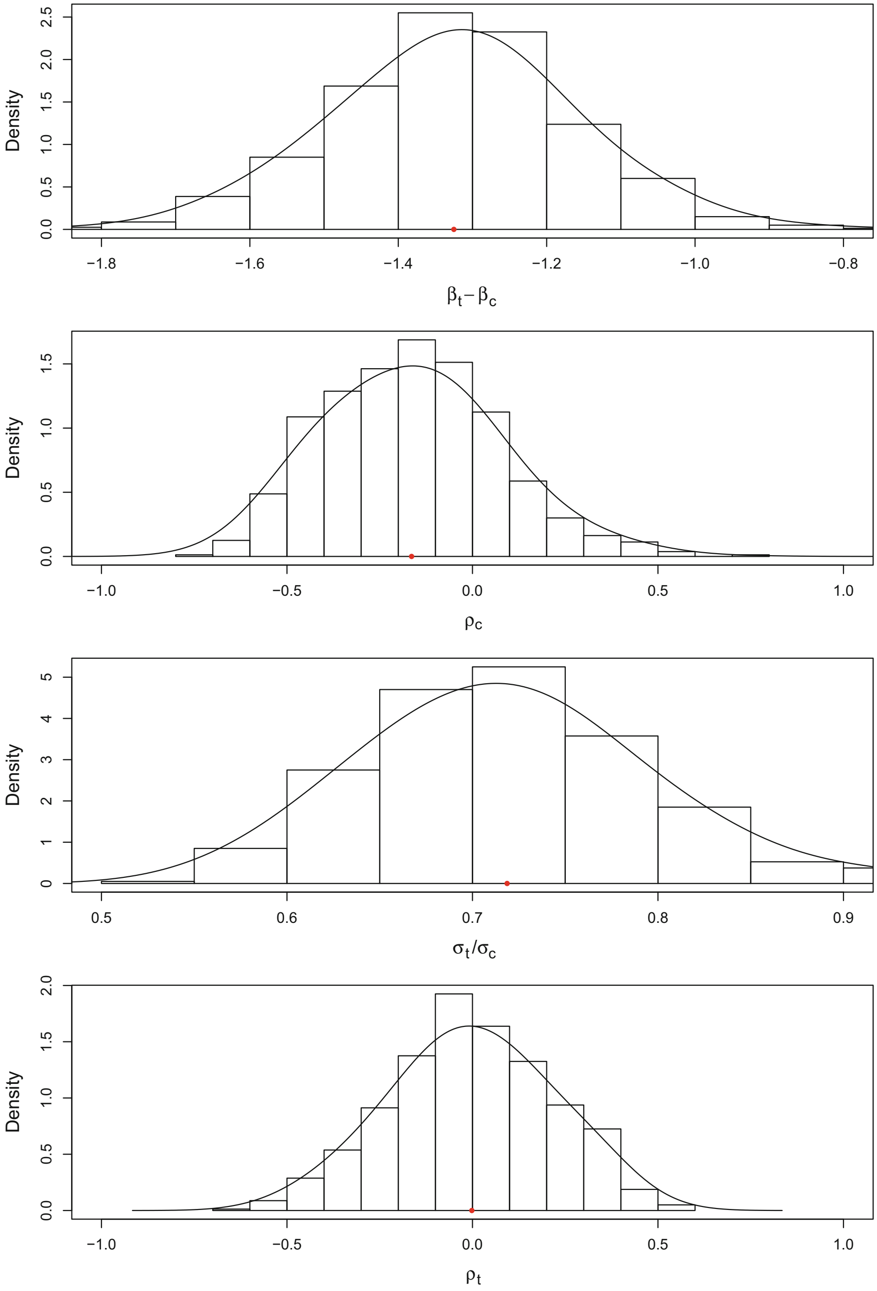


Fig. 3.10 The posterior density plots of interested parameters in the Bayesian bivariate model

Table 3.5 The MC estimates made by Stan

Parameter	Post. mean	Mean err.	2.5%	Median	97.5%	Eff. size	$\hat{R}$
$\Delta\beta$	-1.33	0.01	-1.64	-1.32	-1.01	800	1.00
σ_t/σ_c	0.72	0.00	0.58	0.72	0.89	661	1.00
ρ_c	-0.17	0.01	-0.59	-0.19	0.32	428	1.01
ρ_t	0.00	0.01	-0.43	-0.01	0.40	800	1.00

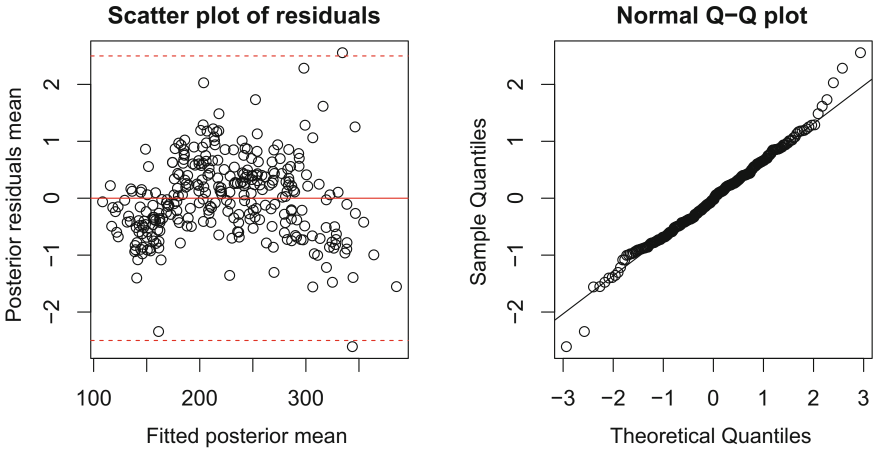


Fig. 3.11 The deviance residual plots of the Bayesian bivariate model

3.5.3 A Bayesian Univariate Normal Hierarchical Model

In the previous section, ρ is not significantly different from 0. If we can assume that $\rho = 0$, the bivariate normal hierarchical model (3.4) can be simplified to a univariate normal hierarchical model, as follows:

$$\begin{aligned} y_{ij} &\sim \mathrm{N}\left(\alpha_i + \beta_i x_j, \sigma_c^2\right), i = 1, \ldots, 30 \\ y_{ij} &\sim \mathrm{N}\left(\alpha_i + \beta_i x_j, \sigma_t^2\right), i = 31, \ldots, 60 \\ \alpha_i &\sim \mathrm{N}\left(\alpha, \sigma_{\alpha c}^2\right), i = 1, \ldots, 30 \\ \beta_i &\sim \mathrm{N}\left(\beta, \sigma_{\beta c}^2\right), i = 1, \ldots, 30 \\ \alpha_i &\sim \mathrm{N}\left(\alpha + \Delta\alpha, \sigma_{\alpha t}^2\right), i = 31, \ldots, 60 \\ \beta_i &\sim \mathrm{N}\left(\beta + \Delta\beta, \sigma_{\beta t}^2\right), i = 31, \ldots, 60, \end{aligned} \tag{3.5}$$

where α, β, $\Delta\alpha$, $\Delta\beta$, $\sigma^2_{\alpha c}$, $\sigma^2_{\alpha t}$, $\sigma^2_{\beta c}$, $\sigma^2_{\beta t}$, σ^2_c, σ^2_t are assumed to have non-informative priors. The Stan code is as follows:

```
rats.code2<-"
data{
  int <lower=8, upper=36> day[5];
  real <lower=0>          weights[60,5];
}
parameters{
  real            alpha[60];
  real            beta[60];
  real            alpha_ave;
  real            alpha_treat;
  real            beta_ave;
  real            beta_treat;
  real <lower=0>  sigmaC;
  real <lower=0>  sigmaT;
  real <lower=0>  sigma_alphaC;
  real <lower=0>  sigma_alphaT;
  real <lower=0>  sigma_betaC;
  real <lower=0>  sigma_betaT;
}
model{
  for (j in 1:30)   alpha[j]  ~ normal(alpha_ave,
      sigma_alphaC);
  for (j in 31:60)  alpha[j]  ~ normal(alpha_ave+alpha_treat,
      sigma_alphaT);
  for (j in 1:30)   beta[j]   ~ normal(beta_ave,
      sigma_betaC);
  for (j in 31:60)  beta[j]   ~ normal(beta_ave+beta_treat,
      sigma_betaT);
  for (j in 1:30)
    for (t in 1:5)  weights[j,t] ~ normal(alpha[j]+beta[j]*day[t
        ],sigmaC);
  for (j in 31:60)
    for (t in 1:5)  weights[j,t] ~ normal(alpha[j]+beta[j]*day[t
        ],sigmaT);
}
generated quantities{
  real TC_sigma;
  real TC_bsigma;
  real TC_asigma;
  vector[300]   log_lik;
  vector[300]   dev_res;
  vector[300]   fitted;
  real          D;
  TC_sigma<-sigmaT/sigmaC;
  TC_asigma<-sigma_alphaT/sigma_alphaC;
  TC_bsigma<-sigma_betaT/sigma_betaC;
  for (j in 1:30){
    for (t in 1:5) {
      log_lik[5*(j-1)+t] <- normal_log(weights[j,t], alpha[j] +
          beta[j] * day[t], sigmaC);
      dev_res[5*(j-1)+t] <- (weights[j,t] - alpha[j] - beta[j] *
          day[t]) / sigmaC;
      fitted[5*(j-1)+t]  <- alpha[j] + beta[j] * day[t];
    }
  }
  for (j in 31:60){
    for (t in 1:5) {
      log_lik[5*(j-1)+t] <- normal_log(weights[j,t], alpha[j] +
          beta[j] * day[t], sigmaT);
      dev_res[5*(j-1)+t] <- (weights[j,t] - alpha[j] - beta[j] *
          day[t]) / sigmaT;
      fitted[5*(j-1)+t]  <- alpha[j] + beta[j] * day[t];
    }
  }
  D <- sum(-2*log_lik);
}
"
```

We get similar estimates of $\Delta\beta$ and σ_t/σ_c as in model (3.4). We display the model selection criteria in Table 3.6. Both DIC and WAIC agree on the best model (3.5). The Stan code for information criteria is as follows:

```
rats.stanfit1<-stan(model_code=rats.code1, data=c("weights","day
    "),iter=1000,chains=4,seed=20) #or model_code=rats.code2
rats.sim1<-extract(rats.stanfit1,permuted =T)
# loo and WAIC
loo(extract_log_lik(rats.stanfit1,"log_lik"))
# pD and DIC
Dbar1<-mean(rats.sim1$D)
Dhat1<-0
for (j in 1:30){
  for (t in 1:5)
  Dhat1<-Dhat1-2*dnorm(weights[j,t], mean(rats.sim1$alpha[,j]) +
      mean(rats.sim1$beta[,j]) * day[t], mean(rats.sim1$sigmaC),
      log=T);
}
for (j in 31:60){
  for (t in 1:5)
  Dhat1<-Dhat1-2*dnorm(weights[j,t], mean(rats.sim1$alpha[,j]) +
      mean(rats.sim1$beta[,j]) * day[t], mean(rats.sim1$sigmaT),
      log=T);;
}
```

3.5.4 Reparameterization in the Gibbs Sampler

An issue arises when R is used to reproduce the results from the Stan analysis. Table 3.7 shows the posterior mean estimates of scale parameters in model (3.5) using a Gibbs sampler coded in R, compared with the estimates from Stan. The estimates of $\widehat{\sigma_c}$ and $\widehat{\sigma_t}$ using the R Gibbs sampler are unduly large.

Table 3.6 Information criteria of models (3.4) and (3.5)

Model	$\text{lppd}_{\text{loo-cv}}$	DIC	p_D	WAIC	p_{WAIC}
(3.4)	−988.6	1938.7	107.3	1948.0	91.9
(3.5)	−988.6	1937.2	103.2	1946.2	88.8

Table 3.7 Comparison of the MC estimates of scale parameters via different sampling methods

Estimation method	$\hat{\sigma}_c$	$\hat{\sigma}_{\alpha c}$	$\hat{\sigma}_{\beta c}$	$\hat{\sigma}_t$	$\hat{\sigma}_{\alpha t}$	$\hat{\sigma}_{\beta t}$
Stan	6.2	10.7	0.52	4.3	13.8	0.55
Gibbs sampler	13.2	11.1	0.5	14.2	13.6	0.56
New Gibbs sampler	5.6	12.7	0.46	3.9	14.5	0.52

The effectiveness of the Gibbs sampler crucially depends on the choice of parameters to be simulated. Gelman et al. (2014) suggested parameterization in terms of independent components as an approach to constructing an efficient simulation algorithm. Following the suggestion, model (3.5) is reparameterized as follows:

$$\begin{aligned}
y_{ij} &\sim \mathrm{N}\left(\gamma_i + \beta_i\left(x_{ij} - \bar{\boldsymbol{x}}_i\right), \sigma_c^2\right), i = 1, \ldots, 30\\
y_{ij} &\sim \mathrm{N}\left(\gamma_i + \beta_i\left(x_{ij} - \bar{\boldsymbol{x}}_i\right), \sigma_t^2\right), i = 31, \ldots, 60\\
\gamma_i &\sim \mathrm{N}\left(\alpha + \beta\bar{\boldsymbol{x}}_i, \sigma_{\alpha c}^2 + \sigma_{\beta c}^2\bar{\boldsymbol{x}}_i^2\right), i = 1, \ldots, 30\\
\beta_i &\sim \mathrm{N}\left(\beta, \sigma_{\beta c}^2\right), i = 1, \ldots, 30\\
\gamma_i &\sim \mathrm{N}\left(\alpha + \Delta\alpha + (\beta + \Delta\beta)\bar{\boldsymbol{x}}_i, \sigma_{\alpha t}^2 + \sigma_{\beta t}^2\bar{\boldsymbol{x}}_i^2\right), i = 31, \ldots, 60\\
\beta_i &\sim \mathrm{N}\left(\beta + \Delta\beta, \sigma_{\beta t}^2\right), i = 31, \ldots, 60,
\end{aligned}$$

where the prior of γ_i is derived based on the relationship $\gamma_i = \alpha_i + \beta_i\bar{x}_i$.

For $i = 1, \ldots, 30$, the full conditional distributions of γ_i and β_i are

$$\gamma_i|\cdot \sim \mathrm{N}\left(\frac{\sum_{j=1}^5 y_{ij}\left(\sigma_{\alpha c}^2 + \sigma_{\beta c}^2\bar{\boldsymbol{x}}_i^2\right) + (\alpha + \beta\bar{\boldsymbol{x}}_i)\,\sigma_c^2}{5\left(\sigma_{\alpha c}^2 + \sigma_{\beta c}^2\bar{\boldsymbol{x}}_i^2\right) + \sigma_c^2}, \frac{\left(\sigma_{\alpha c}^2 + \sigma_{\beta c}^2\bar{\boldsymbol{x}}_i^2\right)\sigma_c^2}{5\left(\sigma_{\alpha c}^2 + \sigma_{\beta c}^2\bar{\boldsymbol{x}}_i^2\right) + \sigma_c^2}\right)$$

$$\beta_i|\cdot \sim \mathrm{N}\left(\frac{\sum_{j=1}^5 y_{ij}\left(x_{ij} - \bar{\boldsymbol{x}}_i\right)\sigma_{\beta c}^2 + \beta\sigma_c^2}{\sum_{j=1}^5\left(x_{ij} - \bar{\boldsymbol{x}}_i\right)^2\sigma_{\beta c}^2 + \sigma_c^2}, \frac{\sigma_{\beta c}^2\sigma_c^2}{\sum_{j=1}^5\left(x_{ij} - \bar{\boldsymbol{x}}_i\right)^2\sigma_{\beta c}^2 + \sigma_c^2}\right),$$

where $p(\gamma_i\,|\,\cdot)$ does not depend on β_i and $p(\beta_i\,|\cdot)$ does not depend on γ_i. We use these full conditional distributions to update γ_i, β_i, and then recover α_i as $\gamma_i - \beta_i\bar{\boldsymbol{x}}_i$. This new Gibbs sampler gives more accurate posterior mean estimates of scale parameters, as shown in Table 3.7.

3.6 Bibliographic Notes

Metropolis et al. (1953) were the first to describe the Metropolis algorithm. This was generalized by Hastings (1970). The Gibbs sampler was first so-named by Geman and Geman (1984). HMC was introduced by Duane et al. (1987) in the physics literature and Neal (1994) for statistics problems.

Gelman and Rubin (1992) and Brooks and Gelman (1998) provided a theoretical justification of the convergence checking methods presented in Sect. 3.2.1 and 3.2.2. For improving the efficiency of MCMC, Tanner and Wong (1987) discussed data augmentation and auxiliary variables. Hills and Smith (1992) and Roberts and Sahu (1997) discussed different parameterizations for the Gibbs sampler.

Lunn et al. (2012) is the first book about the BUGS project. Other references to BUGS include Lunn et al. (2000) and Spiegelhalter et al. (2003). The references to Stan include Stan Development Team (2014), Carpenter et al. (2017), Gelman et al. (2015), Homan and Gelman (2014) and Kucukelbir et al. (2015). Vehtari et al. (2015) demonstrated the calculation of WAIC and LOO cross-validation in Stan.

The EM algorithm was first presented in full generality by Dempster et al. (1977). Some references on variational Bayes include Jordan et al. (1999), Jaakkola and Jordan (2000), Blei et al. (2003) and Gershman et al. (2012). Hoffman et al. (2013) presented a stochastic variational algorithm that is computable for large datasets.

Gilks et al. (1996) is a book full of examples and applications of MCMC methods. The data and model investigated in Sect. 3.5 are from Gelfand et al. (1990).

For other sampling methods, Neal (2003) discussed slice sampling, and Gilks and Wild (1992) introduced adaptive rejection sampling.

References

Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent Dirichlet allocation. *Journal of Machine Learning Research*, *3*, 993–1022.

Brooks, S. P., & Gelman, A. (1998). General methods for monitoring convergence of iterative simulations. *Journal of Computational and Graphical Statistics*, *7*, 434–455.

Carpenter, B., Gelman, A., Hoffman, M., Lee, D., Goodrich, B., Betancourt, M., et al. (2017). Stan: A probabilistic programming language. *Journal of Statistical Software*, *76*, 1–32.

Dempster, A. P., Laird, N. M., & Rubin, D. B. (1977). Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society B*, *39*, 1–38.

Duane, S., Kennedy, A. D., Pendleton, B. J., & Roweth, D. (1987). Hybrid Monte Carlo. *Physics Letters B*, *195*, 216–222.

Gelfand, A. E., Hills, S. E., Racinepoon, A., & Smith, A. F. M. (1990). Illustration of Bayesian-inference in normal data models using Gibbs sampling. *Journal of the American Statistical Association*, *85*, 972–985.

Gelman, A., Lee, D., & Guo, J. (2015). Stan: A probabilistic programming language for Bayesian inference and optimization. *Journal of Educational and Behavioral Statistics*, *40*, 530–543 .

Gelman, A., & Rubin, D. B. (1992). Inference from iterative simulation using multiple sequences. *Statistical Science*, *7*, 457–472.

Gelman, A., Carlin, J. B., Stern, H. S., & Rubin, D. B. (2014). *Bayesian data analysis* (3rd ed.). Boca Raton: Chapman & Hall.

Geman, S., & Geman, D. (1984). Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *6*, 721–741.

Gershman, S., Hoffman, M., & Blei, D. (2012). Nonparametric variational inference. In *29th International Conference on Machine Learning*.

Gilks, W. R., & Wild, P. (1992). Adaptive rejection sampling for Gibbs sampling. *Journal of the Royal Statistical Society C*, *41*, 337–348.

Gilks, W. R., Richardson, S., & Spiegelhalter, D. J. (1996). *Monte Carlo Markov chain in practice*. New York: Chapman & Hall.

Hastings, W. K. (1970). Monte Carlo sampling methods using Markov chains and their applications. *Biometrika*, *57*, 97–109.

Hills, S. E., & Smith, A. F. M. (1992). *Parameterization issues in Bayesian inference*. London: Oxford University Press.

Hoffman, M. D., Blei, D. M., Wang, C., & Paisley, J. (2013). Stochastic variational inference. *Journal of Machine Learning Research*, *14*, 1303–1347.

Homan, M. D., & Gelman, A. (2014). The no-u-turn sampler: Adaptively setting path lengths in Hamiltonian Monte Carlo. *The Journal of Machine Learning Research*, *15*, 1593–1623.

Jaakkola, T. S., & Jordan, M. I. (2000). Bayesian parameter estimation via variational methods. *Statistics and Computing*, *10*, 25–37.

Jordan, M. I., Ghahramani, Z., Jaakkola, T. S., & Saul, L. K. (1999). An introduction to variational methods for graphical models. *Machine Learning*, *37*, 183–233.

Kucukelbir, A., Ranganath, R., Gelman, A., & Blei, D. M. (2015). Automatic variational inference in Stan. arXiv:1506.03431.

Laplace, P. S. (1785). Memoire sur les approximations des formules qui sont fonctions de tres grands nombres. In *Memoires de l'Academie Royale des Sciences*.

Laplace, P. S. (1810). Memoire sur les approximations des formules qui sont fonctions de tres grands nombres, et sur leur application aux probabilites. In *Memoires de l'Academie des Science de Paris*.

Lunn, D. J., Thomas, A., Best, N., & Spiegelhalter, D. (2000). WinBUGS–A Bayesian modelling framework: Concepts, structure, and extensibility. *Statistics and Computing*, *10*, 325–337.

Lunn, D., Jackson, C., Best, N., Thomas, A., & Spiegelhalter, D. (2012). *The BUGS book: A practical introduction to Bayesian analysis*. Boca Raton: Chapman & Hall.

Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H., & Teller, E. (1953). Equation of state calculations by fast computing machines. *Journal of Chemical Physics*, *21*, 1087–1092.

Neal, R. M. (2011). MCMC using Hamiltonian dynamics. In *Handbook of Markov chain Monte Carlo*.

Neal, R. M. (1994). An improved acceptance procedure for the hybrid Monte Carlo algorithm. *Journal of Computational Physics*, *111*, 194–203.

Neal, R. M. (2003). Slice sampling. *The Annals of Statistics*, *31*, 705–741.

Nocedal, J., & Wright, S. (2006). *Numerical optimization*. New York: Springer Science & Business Media.

Roberts, G. O., & Sahu, S. K. (1997). Updating schemes, correlation structure, blocking and parameterization for the Gibbs sampler. *Journal of the Royal Statistical Society B*, *59*, 291–317.

Spiegelhalter, D., Thomas, A., Best, N., and Lunn, D. (2003). WinBUGS user manual. http://www.mrc-bsu.cam.ac.uk/wp-content/uploads/manual14.pdf.

Stan Development Team (2014). *Stan modeling language: User's guide and reference manual*. http://mc-stan.org/users/documentation/

Tanner, M. A., & Wong, W. H. (1987). The calculation of posterior distributions by data augmentation. *Journal of the American statistical Association*, *82*, 528–540.

Vehtari, A., Gelman, A., & Gabry, J. (2015). Efficient implementation of leave-one-out cross-validation and WAIC for evaluating fitted Bayesian models. arXiv:1507.04544.

Chapter 4
Bayesian Chain Ladder Models

Abstract We study the Bayesian chain ladder models and their extensions in this chapter. In Sect. 4.1, the non-life insurance claims reserving background is reviewed. There are two parts in this section. The first part reviews claims reserving terminology. The second part summarizes widely used traditional reserving methods, including the chain ladder (CL) method and the Bornhuetter-Ferguson (BF) method. Stochastic models are discussed in Sects. 4.2 and 4.3. We focus on a Bayesian over-dispersed Poisson (ODP) model with an exponential decay curve component (Verrall et al. 2012). Reversible jump MCMC is used to simulate a sample from this model. In Sect. 4.4, we propose a compound model based on the payments per claim incurred (PPCI) method. A fully Bayesian analysis blending with preliminary classical model checking is performed on the weekly benefit data set and the doctor benefit data set from WorkSafe Victoria, a workers compensation scheme in Victoria state of Australia. We compare our results with the PwC evaluation (Simpson and McCourt 2012).

4.1 Non-life Insurance Claims Reserving Background

Non-life insurance is also known as *property and casualty (P&C) insurance* in the United States and *general insurance* in Australia. There have been much stochastic claims reserving literature proposed in recent decades. England and Verrall (2002) is a good summary of stochastic models up to 2002. Wüthrich and Merz (2008, 2015) are very much mathematically driven. The literature using Bayesian methods include Taylor (2000), England et al. (2012), Verrall and Wüthrich (2012), Zhang et al. (2012), Meyers (2015), etc. We follow Taylor (2000) to review the claims reserving terminology and the traditional claims reserving methods.

© Springer Nature Singapore Pte Ltd. 2018

G. Gao, *Bayesian Claims Reserving Methods in Non-life Insurance with Stan*,
https://doi.org/10.1007/978-981-13-3609-6_4

4.1.1 Terminology

A non-life insurance *policy* is a contract between two parties, the insurer and the insured, providing for the insurer to pay an amount of money to the insured on the occurrence of specified events.

A *claim* is the right of the insured to these amounts and the aggregate of facts establishing that right and the insurer's fulfillment of it. These facts are also called *trigger events*. For a personal automobile policy, the trigger event is usually a car accident. For a workers compensation policy, the trigger event is usually a work-place accident. For a homeowners policy, it can be a fire or storm.

The date on which the events generating the claim took place is called *date of occurrence*. Most non-life insurance policies are *occurrence* policies, which limit the insurer's liability to the trigger events within the policy period. In contrast, *claims-made* policies cover the claims made during the policy period even if these claims arise from an event that happened before policy inception. Most *malpractice insurance* policies are belong to this type. *Claim amount* is the amount which the insurer is obliged to pay with respect to a claim. It is also called loss amount, claim payment, loss payment, paid claim, paid loss etc.

4.1.1.1 The Claims Process

Figure 4.1 shows the time line of a claim. The period A to B is the policy effective period, during which accidents fulfilling other policy conditions will be covered. t_1 is the date of occurrence. The claim is not notified to the insurer until t_2, when the policy is already expired.

Typically, the claim will not be paid immediately. At the very least there will be administrative delays. For more complicated claims, investigation, dispute, litigation or other processes are needed before determination of any payments. It may be in the nature of the policy that the payments extend over years, e.g., when the benefits are income replacement under workers compensation. At time t_5, the insurer considered the action on the claim was complete and closed it. At time t_6, the early closure decision was found to be wrong and claim was reopened, further payments made, and it was closed again at t_8.

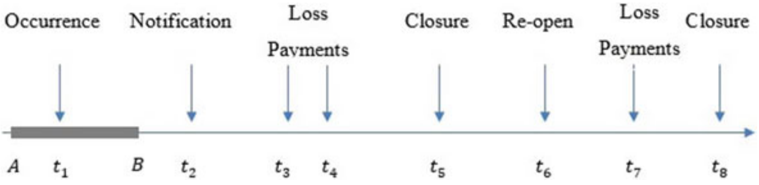


Fig. 4.1 Time line of a claim

4.1.1.2 The Components of Unpaid Claims

Unpaid claims as of a particular time are defined as the *outstanding loss liability* with regarding to the past exposure period. For the claim in Fig. 4.1, the unpaid claims as of time B are called the *incurred but not reported claim* (IBNR), since there is no notification of the claim.

At t_2, when the claim is notified, the unpaid claims consist of case estimates, future development of case estimates and estimates for reopened claim. Case estimate is established by the claim department or independent adjusters. The sum of future development of case estimates and estimates of the re-opened claim are called *incurred but not enough reported* (IBNER).

Aggregately, at any particular time point, the unpaid claims of an insurer consist of IBNR, case estimates for reported claims, and IBNER. The case estimates and IBNER are set up individually according to the characteristics of a particular claim, while IBNR must be estimated aggregately since it comes from the existing claims not yet reported to the insurer. Actuaries rely on the historical aggregate claims data to estimate IBNR, which is also **one of the main tasks of this monograph**.

4.1.1.3 Loss Reserving

The outstanding loss liability is distinct from loss reserve. The outstanding loss liability is an unknown random variable which would be recognized after all the claims are paid. Before all the claims are closed, an unbiased estimate of unpaid claims liability as of a valuation date is called *expected outstanding loss liability*.

A reserve set at this level would have a roughly 50% chance of ultimate adequacy. Often an insurer will wish to reserve more strongly than this and will add a *margin* to the expected liability. This margin is also referred to as the *prudential margin* or *provision for adverse deviation*. To quantify the margin, the uncertainty of outstanding loss liability or, ideally, its predictive distribution needs to be estimated.

4.1.2 Run-Off Triangles

As mentioned before, the estimation of IBNR is impossible for a single claim. So we need to rely on the aggregate claims history. The claims are usually cross-aggregated by two factors: *period of occurrence* and *period of development*. We treat all the claims with the same occurrence period as a group and track the group's development in the future. This structure is analogous to the rats growth data in Sect. 3.5. The only difference is that the claims groups have varying development periods at a particular time.

Table 4.1 An incremental claims run-off triangle

Occurrence period	Development period			
	1	2	...	I
1	$y_{1,1}$	$y_{1,2}$	...	$y_{1,I}$
2	$y_{2,1}$	$y_{2,2}$	...	
$\vdots$	$\vdots$			
I	$y_{I,1}$			

4.1.2.1 Notation for a Run-Off Triangle

We denote the occurrence periods (or accident periods) by $i = 1, \ldots, I$, and the development periods by $j = 1, \ldots, J$. The unit can be a quarter, half or full year, but the occurrence periods and development periods should use the same units and the intervals should be equal. The experience periods (or calendar periods) are denoted by $k = i + j$, which contains a cross-section of experience from various periods of occurrence lying on a diagonal line, and the incremental claims of occurrence period i during the development period j as $y_{i,j}$.

In the case of $I > J$, the run-off triangle becomes a *trapezoid* where the early occurrence periods $i = 1, \ldots, J - I$ are assumed fully run-off by the development period J. A trapezoid can be converted to a triangle by adding $J - I$ development periods and assuming $y_{i,j} = 0$ for $J < j \leq I$. So we always consider the case when $I = J$. Table 4.1 shows a typical structure of incremental claims run-off triangle, where the upper triangle $\left\{y_{i,j} : i + j \leq I + 1\right\}$ is available by the end of most recent accident year I (or by the end of most recent experience period $I + 1$). The loss reserving problem is to predict the lower triangle $\{y_{i,j} : i + j > I + 1, j \leq I\}$, and tail development $\{y_{i,j} : j > I\}$ if not fully run-off by the end of development period I. The final reserve is not equal to the summation of predicted lower triangle and possible tails development but depends on the uncertainty around them.

We define the *cumulative* claims for occurrence year i as of development period j as $c_{i,j} = \sum_{l=1}^{j} y_{i,l}$, and the *ultimate* claims of occurrence year i as $c_{i,\infty}$ or u_i, which is equal to $c_{i,I}$ when the claims are fully run-off by the development period I. The unpaid claims of accident year i are defined as $R_i = \sum_{j=I-i+2}^{\infty} y_{i,j}$. In the case of no development after I, $R_i = c_{i,I} - c_{i,I-i+1}$. The total unpaid claims are defined as $R = \sum_{i=1}^{I} R_i$.

4.1.3 *Widely-Used Claims Reserving Methods*

Here we list two methods: the *chain ladder (CL) method* and the *Bornhuetter-Ferguson (BF) method*. Friedland (2010) discusses other popular methods such as the Cape Cod method, frequency-severity method, case development method etc. But the CL and BF methods are the building blocks of all the other methods.

Table 4.2 An age-to-age factors triangle

Occurrence period	Age-to-age factor			
	1 to 2	2 to 3	...	$I-1$ to I
1	$f_{1,1} = c_{1,2}/c_{1,1}$	$f_{1,2} = c_{1,3}/c_{1,2}$	...	$f_{1,I-1} = c_{1,I}/c_{1,I-1}$
2	$f_{2,1} = c_{2,2}/c_{2,1}$	$f_{2,2} = c_{2,3}/c_{2,2}$		
$\vdots$	$\vdots$			
$I-1$	$f_{I-1,1} = c_{I-1,2}/c_{I-1,1}$			

4.1.3.1 The Chain Ladder Method

The CL method is the most popular and basic technique. The key assumption is that the future claims development is similar to prior years' development. An implicit assumption is that, for an immature accident year, the claims observed so far tell something about the claims yet to be observed. This is in contrast to the assumption underlying the BF method. Other important assumptions include a consistent claim processing and a stable mix of claim types.

The CL method first calculates the observed age-to-age factor (also called the development factor) triangle as in Table 4.2. The CL method requires the judgementally selected age-to-age factors among the candidates including all-year average, last three-year average, *volume-weighted* average etc. We define the CL estimate of development factor of j to $j+1$ as the volume-weighted average:

$$\hat{f}_j = \frac{\sum_{i=1}^{I-j} c_{i,j+1}}{\sum_{i=1}^{I-j} c_{i,j}} \text{ for } j = 1, \ldots, I-1.$$

Assume the tail factor as f_I. In the case of no development after I, $f_I = 1$. The CL estimate of ultimate claim of occurrence period i is

$$\hat{u}_i = \hat{c}_{i,\infty} = c_{i,I+1-i}\hat{f}_{I+1-i} \ldots \hat{f}_I.$$

The expected outstanding liability of occurrence period i is

$$\hat{R}_i = c_{i,I+1-i}\left(\hat{f}_{I+1-i} \cdots \hat{f}_I - 1\right).$$

4.1.3.2 The Bornhuetter-Ferguson Method

The Bornhuetter-Ferguson (BF) method (Bornhuetter and Ferguson 1972) assumes that unpaid claims will develop based on a prior ultimate claim estimate. In other words, the claims reported to date contain no informational value as to the amount of claim yet to be reported. The BF method is rather robust against the unreliable immature claim in the recent accident years.

The BF method applies the same estimate of development pattern as the CL method, but uses a prior estimate of ultimate claims $\tilde{u}_i$. The BF reserve is $\tilde{R}_i = \tilde{u}_i \left(1 - \hat{z}_{I+1-i}\right)$, where $\hat{z}_{I+1-i}$ is the estimated percentage of the ultimate claims amount that is expected to be known by the end of the most recent development period $I + 1 - i$ for the occurrence period i (i.e. by the end of the most recent experience period $I + 1$). The BF method simply uses the CL estimates $\hat{f}_j$ to estimate z as follows:

$$\hat{z}_1 = \left(\hat{f}_1 \ldots \hat{f}_{I-1}\hat{f}_I\right)^{-1}, \ldots, \hat{z}_{I-1} = \left(\hat{f}_{I-1}\hat{f}_I\right)^{-1}, \hat{z}_I = \hat{f}_I^{-1}.$$

4.2 Stochastic Chain Ladder Models

Wüthrich and Merz (2008) commented on the development of claim reserving methods that:

> Reserving actuaries now have to not only estimate reserves for the outstanding loss liabilities but also to quantify possible shortfalls in these reserves that may lead to potential losses. Such an analysis requires stochastic modelling of loss liability cash flows and it can only be done within a stochastic framework.

This section summarizes the recent literature on stochastic claims reserving models. They can be divided into two categories according to the mean functions: multiplicative (cross-classifed) structure using occurrence period and development period as factor covariates; parametric curve using development period as a continuous variable.

The first type of models can give the CL estimates when using over-dispersed error structure but they cannot accommodate the tail development. The second type of models have far fewer parameters and can accommodate the tail development. We will turn to the bootstrap or the MCMC methods to get the predictive distribution of unpaid claims. RJMCMC is discussed in this section as a way of combining the MCMC methods with the model selection.

4.2.1 Frequentist Chain Ladder Models

The *distribution-free model* by Mack (1993) and the *over-dispersed Poisson (ODP) model* by Renshaw and Verrall (1998) use the same mean function to fit the incremental claims. The mean function is the multiplication of two parameters, which correspond to the occurrence periods and the development periods respectively. Besides having the same response variable and mean function, they both assume the variance of the response variable is proportional to its mean. It is not surprising that both of them give the CL estimates.

The distribution-free model does not assume a distribution family and relies on the unbiased estimators, while the ODP model assumes a Poisson distribution and relies on the MLE. They have different prediction errors and predictive distributions which can be estimated via the bootstrap.

4.2.1.1 The Distribution-Free Model

Mack (1993) proposed a distribution-free model assuming only the first two moments, as follows:

$$\begin{aligned} \mathbb{E}\left(c_{i,j}|c_{i,j-1}\right) &= f_{j-1}c_{i,j-1}, \quad i=1,\ldots,I, \ \ j=2,\ldots,I \\ \mathrm{Var}\left(c_{i,j}|c_{i,j-1}\right) &= \sigma^2_{j-1}c_{i,j-1}, \quad i=1,\ldots,I, \ \ j=2,\ldots,I. \end{aligned} \tag{4.1}$$

It can be shown that the CL estimators $\hat{f}_j$ are the unbiased estimator of f_j. Using the CL estimators $\hat{f}_j$, the unpaid claims estimate is the same as the CL estimate. Furthermore, an unbiased estimator for σ^2_j is

$$\begin{aligned} \hat{\sigma}^2_j &= \frac{1}{I-j-1}\sum_{i=1}^{I-j} c_{i,j}\left(\frac{c_{i,j+1}}{c_{i,j}} - \hat{f}_j\right)^2, \quad j=1,\ldots,I-2 \\ \hat{\sigma}^2_{I-1} &= \min\left(\hat{\sigma}^2_{I-2}/\hat{\sigma}^2_{I-3}, \min\left(\hat{\sigma}^2_{I-3}, \hat{\sigma}^2_{I-2}\right)\right). \end{aligned}$$

The conditional mean squared error of prediction (MSEP) for $\hat{R}_i$ is

$$\mathrm{MSEP}_{\mathrm{c}}\left(\hat{R}_i\middle|\boldsymbol{y}\right) = \mathbb{E}\left(\left(R_i - \hat{R}_i\right)^2\middle|\boldsymbol{y}\right) = \mathrm{Var}\left(R_i\right) + \left(\mathbb{E}\left(R_i\right) - \hat{R}_i\right)^2,$$

where $\boldsymbol{y} = \left\{y_{ij} : i=1,\ldots I, \ \ j=1,\ldots,I-i+1\right\}$ is the upper triangle. In words, the conditional prediction variance is equal to the sum of process variance and estimation bias squared. Note that $\mathbb{E}\left(R_i\right) \neq \hat{R}$; see Mack (1993). The analytical results of conditional MSEP of individual occurrence period reserve and total reserve are available. As a final remark, Mack (1999) extends this model to involve the tail factor.

4.2.1.2 The Over-Dispersed Poisson (ODP) Model

One of the most popular generalized linear models in the claims reserving problem is the ODP model which has the following form:

$$\frac{y_{i,j}}{\varphi} \sim \text{Poisson}\left(\frac{\mu_i\gamma_j}{\varphi}\right), \ i=1,\ldots,I, \ \ j=1,\ldots,I, \tag{4.2}$$

with the constraint $\sum_{j=1}^{J}\gamma_j = 1$. Here μ_i is interpreted as the expected ultimate claims of occurrence period i and γ_j as the expected proportion of incremental claims to the ultimate claims during development period j. This model has been intensively studied, including by Renshaw and Verrall (1998), Verrall (2000, 2004), England and Verrall (2002, 2006), England et al. (2012), Verrall et al. (2012) and Wüthrich (2013b).

An implicit assumption of this model is that the variance of the response variable is proportional to its mean. We can check this assumption by inspecting the residual plots. When it fails, other error structures such as a Tweedie distribution can be used.

It can be shown that the MLEs for μ_i and γ_j are equal to the CL estimates using the weighted averages of age-to-age factors. The ODP model can be extended to non-integer, and negative data (i.e., when recoveries are possible) via the quasi-likelihood method (Faraway 2015). The quasi-likelihood method is easily applied in R by specifying the argument `family` as `quasi` in the function `glm()`.

We define the unscaled Pearson residuals as

$$r_{i,j} = \frac{y_{i,j} - \hat{m}_{i,j}}{\sqrt{\hat{m}_{i,j}}},$$

where $\hat{m}_{i,j}$ is the MLE for $\mathbb{E}\left(y_{i,j}\right)$ (i.e., the fitted value). The dispersion parameter φ is estimated by

$$\hat{\varphi} = \frac{\sum_{i+j\le I+1} {r_{i,j}}^2}{N-p},$$

where $N = (I+1)I/2$ is the number of observations, and $p = 2I - 1$ is the number of parameters. Fortunately, R can calculate all of these estimates in a second. England and Verrall (2006) also consider the non-constant dispersion for development periods, which is the assumption of the distribution-free model (4.1).

The mean squared error of prediction for $\hat{R}_i$ is

$$\text{MSEP}(\hat{R}_i) = \mathbb{E}\Big(R_i - \hat{R}_i\Big)^2 = \text{Var}\left(R_i - \hat{R}_i\right) + \left(\mathbb{E}\left(R_i\right) - \mathbb{E}(\hat{R}_i)\right)^2.$$

The second term is approximately zero. Hence,

$$\text{MSEP}(\hat{R}_i) \approx \text{Var}\left(R_i - \hat{R}_i\right) = \text{Var}\left(R_i\right) + \text{Var}(\hat{R}_i). \tag{4.3}$$

In words, the prediction variance is roughly equal to the sum of process variance and estimation variance. R cannot provide the MSEP($\hat{R}_i$) directly since it is a complicated function of parameters. From Renshaw and Verrall (1998), MSEP($\hat{R}_i$) is estimated as

$$\sum_{j=I-i+2}^{I} \hat{\varphi}\hat{m}_{i,j} + \sum_{j=I-i+2}^{I} \hat{m}_{i,j}^2 \text{Var}\left(\hat{\eta}_{i,j}\right) + 2\sum_{k>l} \hat{m}_{i,k}\hat{m}_{i,l}\text{Cov}\left(\hat{\eta}_{i,k}, \hat{\eta}_{i,l}\right),$$

where η is the linear predictor and its covariance matrix is available directly from R. England and Verrall (2002) also give the MSEP of total reserve with an additional covariance term for different occurrence periods (i.e., Cov $(\hat{\eta}_{m,k}, \hat{\eta}_{n,l})$). We can rely on `ChainLadder` package to get MSEP$(\hat{R}_i)$. Later, we will use the bootstrap or MCMC to simulate R_i and estimate its MSEP based on the simulated sample.

4.2.1.3 The Predictive Distribution via the Bootstrap

Bootstrapping (Efron and Tibshirani 1994) is a powerful, yet simple, technique for obtaining information from a single sample of data. In a standard application of the bootstrap, where data are assumed to be independent and identically distributed, resampling with replacement takes place of the data themselves.

In regression problems the data are usually assumed to be independent but not identically distributed due to the existence of covariates. Therefore, with regression problems it is common to bootstrap residuals, rather than data themselves, since the residuals are approximately independent and identically distributed. For model (4.1) and model (4.2), we use the scaled *Pearson residuals* for bootstrapping.

The bootstrap for model (4.1).

Model (4.1) is in a recursive structure. England and Verrall (2002) showed that an equivalent model can be obtained using the observed factors $f_{i,j}$ as a response variable with the following mean and variance:

$$\mathbb{E}\left(f_{i,j}|c_{i,j}\right) = f_j$$
$$\text{Var}\left(f_{i,j}|c_{i,j}\right) = \frac{\sigma_j^2}{c_{i,j}}.$$

The scaled Pearson residuals are defined as

$$r_{i,j}^s = \frac{f_{i,j} - \hat{f}_j}{\hat{\sigma}_j / \sqrt{c_{i,j}}}.$$

The bootstrap algorithm for model (4.1) is as follows:

1. Sample with replacement, from the set of scaled Pearson residuals, to get a sample of residuals for a single bootstrap iteration $\left\{r_{i,j}^B : i + j \le I\right\}$.
2. Back out the residual definition to obtain a pseudo run-off triangle of development factor as follows:
$$f_{i,j}^B = \frac{r_{i,j}^B \hat{\sigma}_j}{\sqrt{c_{i,j}}} + \hat{f}_j.$$
3. Obtain the new volume-weighted development factor

$$\tilde{f}_j = \frac{\sum_{i=1}^{I-j} f_{i,j}^B \, c_{i,j}}{\sum_{i=1}^{I-j} c_{i,j}}.$$

4. Simulate the future claims. Starting from the latest cumulative claims $c_{i,I+1-i}$, forecast the next cumulative claims by sampling a value from a gamma distribution:

$$\tilde{c}_{i,I+2-i}|c_{i,I+1-i} \sim \text{Gamma}\left(\frac{\tilde{f}_{I+1-i}^2 \; c_{i,I+1-i}}{\hat{\sigma}_{I+1-i}^2}, \; \frac{\tilde{f}_{I+1-i}}{\hat{\sigma}_{I+1-i}^2}\right) \text{ for } i = 2, \ldots, I.$$

5. Recursively predict the future cumulative claims by sampling from

$$\tilde{c}_{i,j+1}|\tilde{c}_{i,j} \sim \text{Gamma}\left(\frac{\tilde{f}_j^2 \tilde{c}_{i,j}}{\hat{\sigma}_j^2}, \frac{\tilde{f}_j}{\hat{\sigma}_j^2}\right) \text{for } i = 3, \ldots, I \text{ and } j = I - i + 3, \ldots, I.$$

6. Calculate each accident year future claims and total future claims as

$$\tilde{R}_i = \tilde{c}_{i,I} - c_{i,I+1-i}, \text{ for } i = 2, \ldots, I$$

$$\tilde{R} = \tilde{R}_2 + \tilde{R}_3 + \cdots + \tilde{R}_I.$$

7. Repeat steps 1–6 to get a sample of $\tilde{R}_i$ and $\tilde{R}$.

The empirical distribution of the bootstrap sample approximates the predictive distribution. The prediction variance of total liability can be estimated by the sample variance of the bootstrap sample of total liability. Note that the bootstrap sample variation consists of variation due to bootstrapping in step 1 (i.e., estimation variance) and variation due to forecasting in step 4 and 5 (i.e., process variance), which correspond to the two terms on the right side of Eq. (4.3).

The bootstrap for model (4.2).

The scaled Pearson residuals of model (4.2) are

$$r_{i,j}^s = \frac{y_{i,j} - \hat{m}_{i,j}}{\sqrt{\hat{\varphi}\hat{m}_{i,j}}}.$$

The bootstrap algorithm for model (4.2) is as follows:

1. Sample with replacement from the set of scaled Pearson residuals to get a sample of residuals for a single bootstrap iteration $\left\{r_{i,j}^B : i + j \leq I\right\}$.
2. Back out the residual definition to obtain a pseudo run-off triangle of incremental claims as follows:

$$y_{i,j}^B = r_{i,j}^B \sqrt{\hat{\varphi}\hat{m}_{i,j}} + \hat{m}_{i,j}.$$

Table 4.3 The total outstanding liability estimates from models (4.1) and (4.2)

Model	Estimate	No tail factor	With tail factor
(4.1)	$\hat{R}$	1,463,076	1,599,558
	$\sqrt{\text{MSEP}_c(\hat{R})}$	55,300	58,528
(4.2)	$\hat{R}$	1,463,076 (1,471,906)	NA
	$\sqrt{\text{MSEP}(\hat{R})}$	60,444 (60,087)	NA

3. Use the CL method to get the new estimate $\tilde{\mu}_i$, $\tilde{\gamma}_j$ based on the pseudo incremental claims run-off triangle from step 2.
4. Simulate the future claims from the following ODP model:

$$\tilde{R}_i \sim \hat{\varphi}\text{Poisson}\left(\frac{\tilde{\mu}_i \sum_{j=I-i+2}^{I} \tilde{\gamma}_j}{\hat{\varphi}}\right) \text{ for } i = 2, 3, \ldots, I.$$

   Calculate the total future claims as $\tilde{R} = \tilde{R}_2 + \tilde{R}_3 + \cdots + \tilde{R}_I$.
5. Repeat steps 1–4 to get a sample of $\tilde{R}_i$ and $\tilde{R}$.

In the case when $\hat{\varphi}$ is large, e.g., $\hat{\varphi} = 1000$, $\tilde{R}_i$ will be sampled from $\{0, 1000, \ldots\}$, which is undesirable. We can use an alternative gamma distribution with the target mean and variance in step 4.

Example 4.1 (*Liability insurance claims data*) We use the liability claims run-off data with 22 accident years and 22 development years from Verrall and Wüthrich (2012). The R package `ChainLadder` by Gesmann et al. (2015) can estimate all the quantities we have previously mentioned.The residual plots are needed to validate the model assumptions.

Table 4.3 shows that models (4.1) and (4.2) both give the same point estimate of total liability, which are also equal to the CL estimate. The numbers in parentheses are from the bootstrap method. The distribution-free model (4.1) can accommodate tail development, which consists of nearly 10% of total liability. The conditional mean squared error is smaller than the unconditional mean squared error since the latter involves the extra uncertainty induced by the historical claims data (i.e., estimation error).

The function `BootChainLadder` in the R package `ChainLadder` performs the bootstrap for model (4.2). Here we bootstrap $1,000$ times. We show the histogram of the bootstrap sample of total outstanding liability in Fig. 4.2, and we get the bootstrap estimate of total outstanding liability and the standard error in Table 4.3 (stated in parentheses).

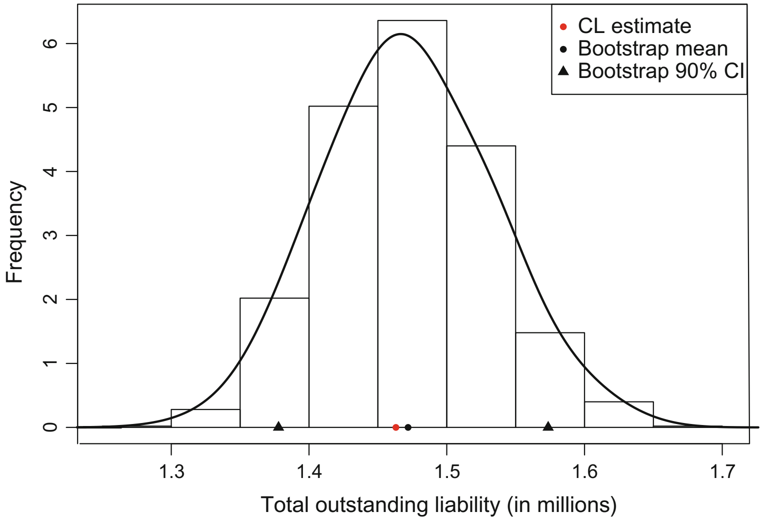


Fig. 4.2 The histogram of the total outstanding claims liability via the bootstrap

4.2.2 A Bayesian Over-Dispersed Poisson (ODP) Model

The model (4.2) in a Bayesian framework has the following form:

$$\begin{aligned} \frac{y_{i,j}}{\varphi} &\sim \text{Poisson}\left(\frac{\mu_i \gamma_j}{\varphi}\right) \\ \mu_i &\sim \text{Gamma}\left(a_i, b_i\right) \\ \gamma_j &\sim \text{Gamma}\left(c_j, d_j\right), \end{aligned} \tag{4.4}$$

where μ_i is related to the ultimate claim of accident year i, γ_j is related to the incremental claims percentage during development year j, and $a_i,\ b_i, c_j, d_j$ are constant hyperparameters whose values are adjusted according to prior knowledge. In the case where there is no prior knowledge, we assume μ_i and γ_j follow the same non-informative prior. φ is a plug-in estimate via GLM (see Sect. 4.2.1.2). We can assume a prior for φ, see Example 3.17 of Wüthrich and Merz (2015). Note that the uncertainty around φ doesn't have a significant influence on the predictive uncertainty of unpaid claims. Hence, it is reasonable to assume a constant φ as we did here.

The joint posterior distribution of $\mu = (\mu_1, \ldots, \mu_I)$ and $\gamma = (\gamma_1, \ldots, \gamma_I)$ is

$$p\left(\mu, \gamma|\mathbf{y}\right) = \frac{p\left(\mathbf{y}|\mu, \gamma\right) p\left(\mu, \gamma\right)}{\int_{\mu,\gamma} p\left(\mathbf{y}|\mu, \gamma\right) p\left(\mu, \gamma\right) d\mu d\gamma} \propto p\left(\mathbf{y}|\mu, \gamma\right) p\left(\mu, \gamma\right)$$

$$\propto \prod_{i+j\leq I+1} \exp\left(-\frac{\mu_i\gamma_j}{\varphi}\right)\left(\frac{\mu_i\gamma_j}{\varphi}\right)^{\frac{y_{i,j}}{\varphi}} \prod_{i=1}^{I} {\mu_i}^{a_i-1} \exp\left(-b_i\mu_i\right) \prod_{j=1}^{I} {\gamma_j}^{c_j-1} \exp\left(-d_j\gamma_j\right).$$

Our interest is in not only the parameter μ, γ but also the future claims. We have the following posterior predictive distribution of future claims:

$$p(y'|\boldsymbol{y}) = \int_{\mu,\gamma} p\left(y'|\mu,\gamma\right) p\left(\mu,\gamma|\boldsymbol{y}\right) d\mu d\gamma,$$

where y' is the set of lower triangle. It is hard to solve $p\left(y'|\boldsymbol{y}\right)$ analytically. The conditional mean squared error of prediction for a predictor $\hat{R}$ is

$$\mathrm{MSEP_c}(\hat{R}) = \mathbb{E}\left(\left(\hat{R}-R\right)^2\bigg|\boldsymbol{y}\right) = \mathrm{Var}\left(R|\boldsymbol{y}\right) + \left(\hat{R}-\mathbb{E}\left(R|\boldsymbol{y}\right)\right)^2.$$

We prefer the predictor $\hat{R} = \mathbb{E}\left(R|\boldsymbol{y}\right)$ (i.e., the posterior mean) which minimizes $\mathrm{MSEP_c}(\hat{R})$. The MSEP of the posterior mean is $\mathrm{Var}\left(R|\boldsymbol{y}\right)$, which can be estimated from a MC sample.

4.2.2.1 A Gibbs Sampler for Model (4.4)

The Gibbs sampler is a special case of the Metropolis-Hastings (M-H) algorithm. In the M-H algorithm if we choose the full conditional distribution as the proposed distribution, the acceptance rate will be 1. The use of the Gibbs sampler implicitly requires that the full conditional distribution is recognisable; otherwise, we need to turn to the general M-H algorithm or adaptive rejection sampling (Gilks and Wild 1992).

The full conditional distribution of μ_i is obtained from $p\left(\mu,\gamma|\boldsymbol{y}\right)$, assuming all the other parameters constant, as follows:

$$p\left(\mu_i|\boldsymbol{y},\gamma,\mu_{-i}\right) \propto \exp\left(-\frac{\mu_i\sum_{j=1}^{I+1-i}\gamma_j}{\varphi}\right)\mu_i^{\frac{\sum_{j=1}^{I+1-i}y_{i,j}}{\varphi}} {\mu_i}^{a_i-1}\exp\left(-b_i\mu_i\right),$$

where μ_{-i} is the vector μ excluding μ_i. It can be recognized as a gamma distribution

$$\mu_i|\boldsymbol{y},\gamma \sim \mathrm{Gamma}\left(a_i + \frac{\sum_{j=1}^{I+1-i}y_{i,j}}{\varphi}, b_i + \frac{\sum_{j=1}^{I+1-i}\gamma_j}{\varphi}\right). \tag{4.5}$$

Symmetrically, the full conditional distribution of γ_j for $j = 1,\ldots,I$ is

$$\gamma_j|\boldsymbol{y},\mu \sim \mathrm{Gamma}\left(c_j + \frac{\sum_{i=1}^{I+1-j}y_{i,j}}{\varphi}, d_j + \frac{\sum_{i=1}^{I+1-j}\mu_i}{\varphi}\right). \tag{4.6}$$

A Gibbs sampler based on the above full conditional distributions has the following steps:

1. Initialize $\mu^0,\ \gamma^0$. For $t \geq 1$, repeat the steps 2–4.
2. For $1 \leq i \leq I$, draw a value μ_i^t from distribution (4.5) with $\gamma = \gamma^{t-1}$, and set $\mu^t = \left(\mu_1^t, \ldots, \mu_I^t\right).$
3. For $1 \leq j \leq I$, draw a value γ_j^t from distribution (4.6) with $\mu = \mu^t$, and set $\gamma^t = \left(\gamma_1^t, \ldots, \gamma_I^t\right).$
4. For $1 \leq i \leq I$, draw a value R_i^t from the distribution

$$\varphi\text{Poisson}\left(\frac{\mu_i^t \sum_{j=I-i+1}^{I} \gamma_j^t}{\varphi}\right),$$

and set $R^t = R_2^t + \cdots + R_I^t$.

Steps 2 and 3 provide a Markov chain $\left(\mu^t, \gamma^t\right)_{t \geq 0}$ whose stationary distribution is $p\left(\mu, \gamma | \mathbf{y}\right)$. Step 4 provides a sample of the total outstanding liability. The prediction error of future claims consists of estimation error via steps 2 and 3 and process error via step 4, which correspond to the bootstrap resampling step and forecasting step respectively.

Note that parameters μ and γ are not uniquely defined. In Example 4.2, we will see that the multiplication $\mu_i \gamma_j$ is converged rather than μ_i,γ_j by themselves. In other words, μ_i, γ_j cannot be estimated accurately individually. For interpretation purposes, we define the normalized μ_i, γ_j as

$$\mu_i^* = \mu_i \sum_{j=1}^{I} \gamma_j, \quad \gamma_j^* = \frac{\gamma_j}{\sum_{k=1}^{I} \gamma_k}.$$

Inferences under non-informative priors

Under the non-informative priors, i.e., $a \to 0, b \to 0$, c $\to 0, d \to 0$, distributions (4.5) and (4.6) define the following conditional expectations:

$$\mathbb{E}\left(\mu_i | \mathbf{y}, \gamma\right) = \frac{\sum_{j=1}^{I+1-i} y_{i,j}}{\sum_{j=1}^{I+1-i} \gamma_j}, \quad \mathbb{E}\left(\gamma_j | \mathbf{y}, \mu\right) = \frac{\sum_{i=1}^{I+1-j} y_{i,j}}{\sum_{i=1}^{I+1-j} \mu_i}.$$

If we substitute the left sides with μ_i and γ_j, the above equations define a system of equations whose solutions will be consistent with the CL estimates. Strictly, the posterior mean of outstanding liability is close but not exactly equal to the CL estimate.

In Example 4.1, we use the plug-in estimate $\hat{\varphi} = 631.8$, and non-informative prior for μ, γ, (i.e., $a, b, c, d \to 0$). We iterate for $T = 1000$ times and get the MC estimate of posterior mean of total outstanding liability as 1,461,958 dollars, with the standard error of 60,902 dollars. These values are quite close to the result in Table 4.3.

Inferences under strong priors for μ

Assume the prior knowledge of μ is some value around m with small variation, i.e., $b/m_i \to \infty$ and $a_i = m_i b$. Distributions (4.5) and (4.6) define the following conditional expectations:

$$\mathbb{E}\left(\mu_i | \mathbf{y}, \gamma\right) \approx m_i, \quad \mathbb{E}\left(\gamma_j | \mathbf{y}, \mu\right) \approx \frac{\sum_{i=1}^{I+1-j} y_{i,j}}{\sum_{i=1}^{I+1-j} m_i},$$

which follows the BF predictor proposed by Mack (2008). The estimation error of μ is close to 0, and the standard error of claims liability will be largely reduced.

Example 4.2 (*A Monte Carlo study of model* (4.4) *using simulated data*) We assume the parameters in model (4.4) as $\mu = \left(10^7, 1.02 \times 10^7, \ldots, 1.02^9 \times 10^7\right)$, $\gamma = (0.30,\ 0.21,\ 0.15,\ 0.10,\ 0.08,\ 0.06,\ 0.04,\ 0.03,\ 0.02,\ 0.01)$, $\varphi = 25000$, where the sum of γ is 1 implying no claims development beyond age 10. We simulate a sample of incremental claims in the upper triangle.

Inferences under non-informative priors

We use the plug-in estimate $\hat{\varphi} = 23{,}488$, and choose $a = 0, b = 0$. We iterate for $T = 1000$ times. The trace plots in Fig. 4.3 show that μ_6^*, γ_6^* converge rather than μ_6, γ_6. The MC estimates of posterior means of μ^*, γ^* are close to the CL estimates as shown in Fig. 4.4. The predictive distributions of outstanding liability are shown in Fig. 4.5. We check whether the 95% CPDRs have 95% chances to cover the true parameters if we replicate the above process (i.e., simulate the data then estimate the 95% CPDR) for 100 times. Table 4.4 confirms our expectation except for the last accident year and the last development period, due to the sparse data for these two periods.

Table 4.4 The proportions of the 95% CPDRs containing the true values

μ_1	μ_2	μ_3	μ_4	μ_5	μ_6	μ_7	μ_8	μ_9	μ_{10}
0.93	0.91	0.95	0.92	0.93	0.89	0.91	0.96	0.95	0.89
γ_1	γ_2	γ_3	γ_4	γ_5	γ_6	γ_7	γ_8	γ_9	γ_{10}
0.94	0.94	0.94	0.94	0.95	0.92	0.97	0.92	0.96	0.78
R	R_2	R_3	R_4	R_5	R_6	R_7	R_8	R_9	R_{10}
0.94	0.90	0.94	0.98	0.95	0.98	0.97	0.97	0.95	0.88

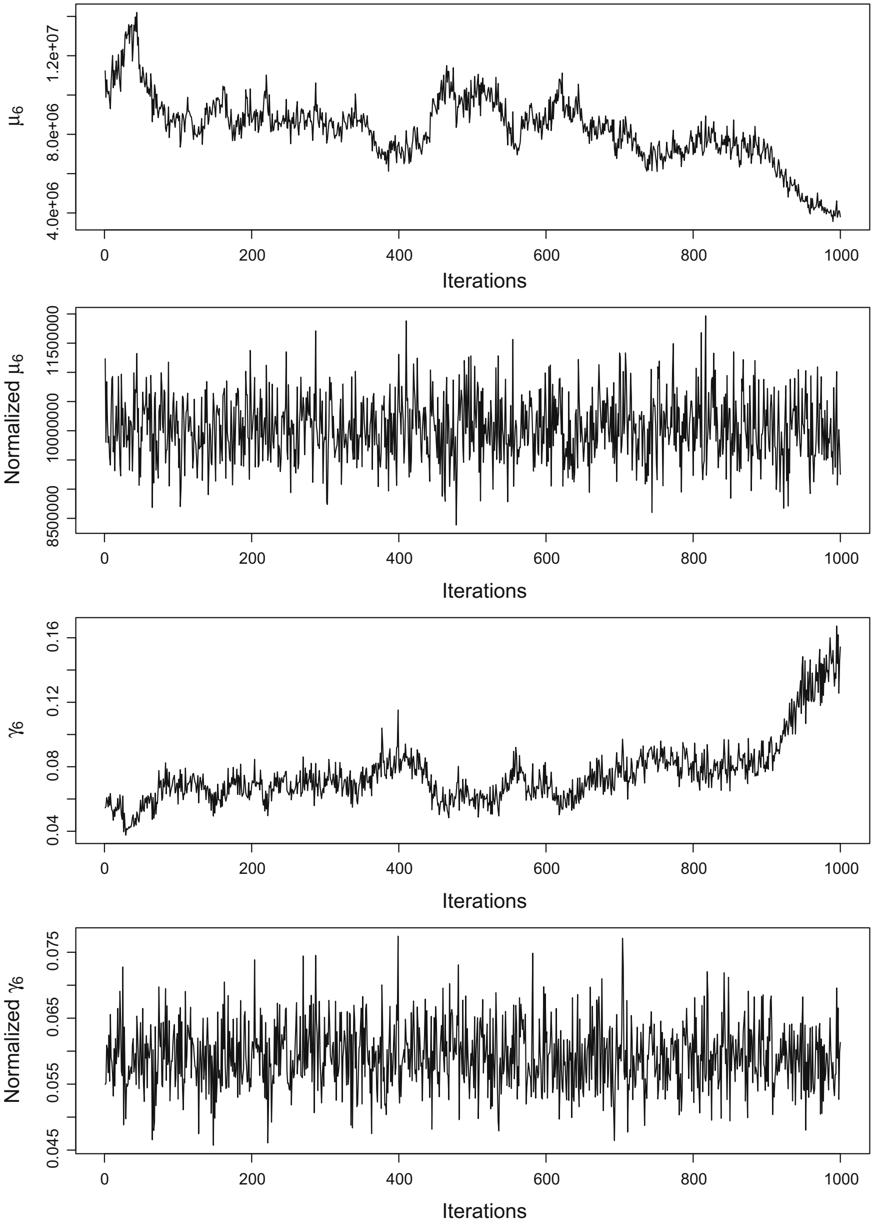


Fig. 4.3 The trace plots of the first 10,000 iterations

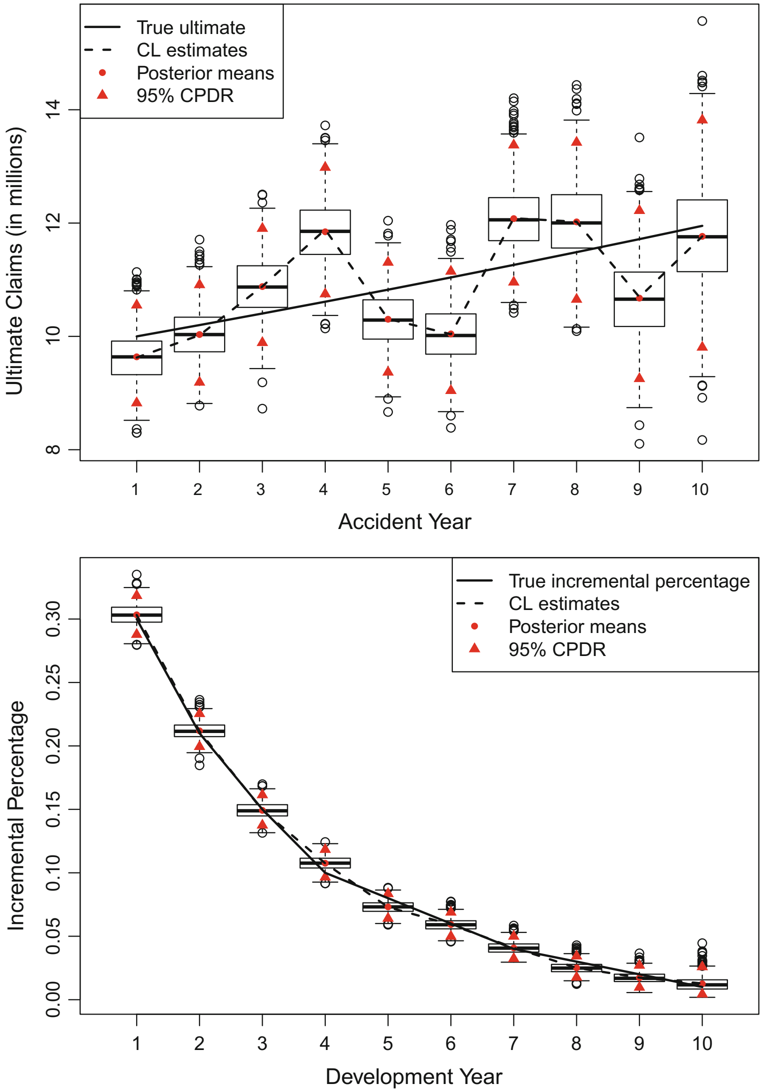


Fig. 4.4 The MC estimates of the ultimate claims μ^* and the incremental claims percentages γ^*

Inferences under strong priors for μ

We choose the following strong priors for μ: $a = 10^{12}$, $m = \left(10^7,\ 1.02 \times 10^7, \ldots\right)$, $b = 0$. We iterate for $T = 1{,}000$ times and get the MC estimates as in Table 4.5. As we expected, the variations of outstanding liability under strong priors are substantially smaller than those under a non-informative prior.

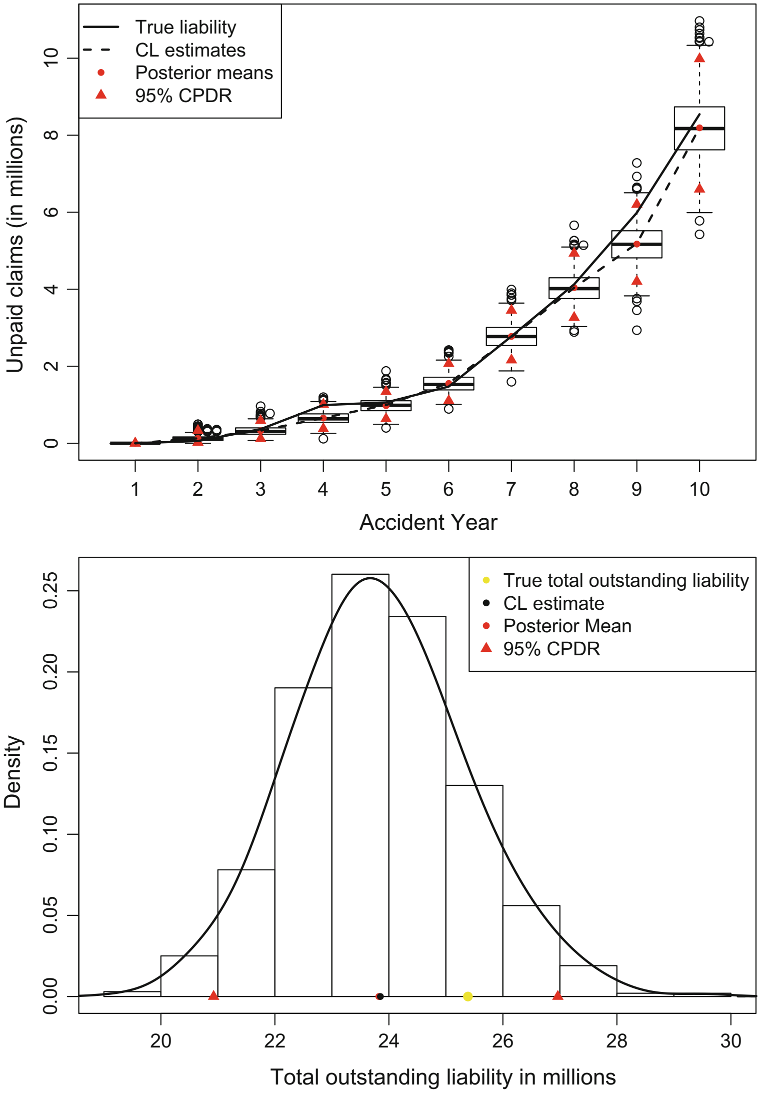


Fig. 4.5 The predictive distributions of outstanding claims liability for each accident year and the predictive distribution of the total outstanding claims liability

4.3 A Bayesian ODP Model with Tail Factor

In this section we will focus on the following model:

$$\begin{aligned}
\frac{y_{i,j}}{\varphi} &\sim \text{ Poisson}\left(\frac{\mu_i \gamma_j}{\varphi}\right),\; i = 1, .., I,\; j = 1, \ldots, I \\
\mu_i &\sim \text{Gamma}\left(a_i, b_i\right) \\
\gamma_j &\sim \text{Gamma}\left(c_j, d_j\right),\;\; j = 1, \ldots, k-1 \\
\gamma_j &= \exp\left(\alpha - j\beta\right),\;\; j = k, \ldots, I \\
\alpha &\sim \text{ N}\left(e, \sigma_1^2\right) \\
\beta &\sim \text{ N}\left(f, \sigma_2^2\right) \\
\Pr\left(k = i\right) &= \frac{1}{I-1},\; i = 2, \ldots, I,
\end{aligned} \tag{4.7}$$

where $a, b, c, d, e, f, \sigma_1^2, \sigma_2^2$ are the specified hyperparameters and φ is a plug-in estimate. This is the same Bayesian ODP model as model (4.4) but extended to include a suitable tail factor.

To illustrate this model, we specify $a_i = 100, b_i = a_i/m_i, c_j = 1, d_j = c_j/h_j$, $e = 0, f = 0, \sigma_1^2 = 100, \sigma_2^2 = 100$, where m_i and h_j are the CL ultimate claims estimates and the CL incremental claims proportion estimates. The choice of these hyperparameters ensures the convergence of the RJMCMC algorithm while allowing sufficient flexibility. Denote $\theta_k = \{\alpha, \beta, \mu, \gamma_1, \ldots, \gamma_{k-1}\}$. This model reduces the number of parameters from $2I$ in model (4.4) to $k + 2$. Note that k is usually much smaller than I.

Model (4.7) implicitly includes a tail factor

$$\gamma_J = \sum_{j=I+1}^{J} \exp\left(\alpha - j\beta\right),$$

where J is chosen judgementally. $J \to \infty$ leads

$$\gamma_\infty = \frac{\exp\left(\alpha - \left(I+1\right)\beta\right)}{1 - \exp\left(-\beta\right)}.$$

The main task of this section is to determine which k leads to the optimal model fit. Since different ks will lead to different parameter dimensions, this problem is

Table 4.5 The outstanding liability estimates under different priors

Estimate	Strong prior case			Non-informative prior case		
	Post. mean	Sd. error	CV (%)	Post. mean	Sd. error	CV (%)
R	24,244,540	1,006,232	4.2	23,867,671	1,524,567	6.4
R_{10}	8,340,955	453,191	5.4	8,206,132	857,982	10.5
R_9	5,706,840	383,309	6.7	5,189,284	528,886	10.2
R_8	3,862,495	321,367	8.3	4,040,881	422,256	10.5

equivalent to model selection. Here we investigate two methods: deviance information criteria (DIC) (Spiegelhalter et al. 2002) and reversible jump Markov chain Monte Carlo (RJMCMC) method (Green 1995). There are other methods to compare and evaluate Bayesian models such as BIC, cross-validation and posterior predictive checking (see Sect. 2.2).

4.3.1 Reversible Jump Markov Chain Monte Carlo

RJMCMC generalizes the Metropolis-Hastings (M-H) algorithm to include a model indicator. The joint state space (θ_l, l) is defined by both model parameters θ_l and the model index l, where $l \in \{1, 2, \ldots, L\}$. The joint posterior distribution of θ_l, l can be factorized as

$$p\left(l, \theta_l|\mathbf{y}\right) \propto p\left(\theta_l|\mathbf{y}\right) p\left(l|\mathbf{y}\right) \propto p\left(\mathbf{y}|\theta_l, l\right) p\left(\theta_l\right) p\left(l\right),$$

which is the product of the likelihood, the prior of θ_l and the prior of l.

Before turning to the RJMCMC algorithm, we review the M-H algorithm. In the M-H algorithm, a proposal distribution from θ to θ^* is $q(\theta^*|\theta)$, and the acceptance rate is

$$\min\left(1, \frac{p\left(\theta^*|\mathbf{y}\right) q(\theta|\theta^*)}{p\left(\theta|\mathbf{y}\right) q(\theta^*|\theta)}\right).$$

For RJMCMC, we need a model index proposal distribution from l to l^*, $q(l^*|l)$, and a parameter proposal distribution from θ_l to θ_{l^*}. Since θ_l and θ_{l^*} may have different dimensions, the parameter proposal process involves two steps: generate $u \sim q_{l \to l^*}$, and then set $\left(\theta_{l^*}, u^*\right) := T_{l \to l^*}\left(\theta_l, u\right)$, where $T_{l \to l^*}$ is a one-to-one mapping with $T_{l \to l^*} = T^{-1}_{l^* \to l}$.

Note that (θ_l, u) must have the same dimension as $\left(\theta_{l^*}, u^*\right)$. It is possible that u is zero-dimensional, e.g., θ_l has more parameters than θ_{l^*}. Similar to the M-H algorithm, the acceptance rate is calculated as

$$\min\left(1, \frac{p\left(l^*, \theta_{l^*}|\mathbf{y}\right)}{p\left(l, \theta_l|\mathbf{y}\right)} \frac{q(l|l^*)}{q(l^*|l)} \frac{q_{l^* \to l}\left(u^*\right)}{q_{l \to l^*}\left(u\right)} \left|\frac{\partial T_{l \to l^*}\left(\theta_l, u\right)}{\partial\left(\theta_l, u\right)}\right|\right),$$

where the final term is the determinant of the Jacobian matrix.

The RJMCMC algorithm typically has the following steps:

1. Initialize l^0 and $\theta^0_{l^0}$. In the following we use the shortened notation $\theta^t_{l^t}$ for θ^t. For $t \geq 1$, repeat the following steps.
2. Propose a new model index l^* from the distribution $q\left(l^*|l^t\right)$.
3. If $l^* = l^t$, do the following *within-model* update. Otherwise, jump to step 4.

   a. Update the current model l^t by one iteration (i.e., via normal MCMC).

  b. Set $l^{t+1} = l^*$ and θ^{t+1} as the updated parameters.
  c. Go to step 2.

4. If $l^* \neq l^t$, do the following *between-model* update.

  a. Generate $u^t \sim q_{l^t \to l^*}$.
  b. Set $\left(\theta^*, u^*\right) := T_{l^t \to l^*}\left(\theta^t, u^t\right)$.
  c. Compute the acceptance rate as

$$\min\left(1, \frac{p\left(l^*, \theta^* | \mathbf{y}\right)}{p\left(l^t, \theta^t | \mathbf{y}\right)} \frac{q\left(l^t | l^*\right)}{q\left(l^* | l^t\right)} \frac{q_{l^* \to l^t}\left(u^*\right)}{q_{l^t \to l^*}\left(u^t\right)} \left| \frac{\partial T_{l^t \to l^*}\left(\theta^t, u^t\right)}{\partial\left(\theta^t, u^t\right)} \right| \right).$$

  d. With this acceptance rate, set $l^{t+1} = l^*$ and $\theta^{t+1} = \theta^*$. Otherwise keep $l^{t+1} = l^t$ and $\theta^{t+1} = \theta^t$.
  e. Go to step 2.

The RJMCMC algorithm provides a Markov chain $\left(l^t, \theta^t\right)_{t \geq 0}$ whose stationary distribution is $p(l, \theta_l | \mathbf{y})$. We can either choose the model l_0 which has the highest posterior probability $p(l|\mathbf{y})$, or perform model averaging over $p(l, \theta_l | \mathbf{y})$.

4.3.2 RJMCMC for Model (4.7)

In model (4.7), k is a model index variable whose value determines the parameter dimension. The joint posterior of k and θ_k is simplified as $p\left(k, \theta_k | \mathbf{y}\right) \propto p\left(\mathbf{y} | \theta_k\right) p\left(\theta_k\right)$. We use the following model index proposal distributions:

$$\begin{cases} q\left(k^* = k | k\right) = q\left(k^* = k+1 | k\right) = q\left(k^* = k-1 | k\right) = \frac{1}{3} \text{ for } k = 3, 4, \ldots, I-1 \\ q\left(k^* = k | k\right) = \frac{2}{3} \text{ and } q\left(k^* = k+1 | k\right) = \frac{1}{3} \text{ for } k = 2 \\ q\left(k^* = k | k\right) = \frac{2}{3} \text{ and } q\left(k^* = k-1 | k\right) = \frac{1}{3} \text{ for } k = I \end{cases} \tag{4.8}$$

which implies that k can equally jump to the nearest neighbourhood or stay in the current state. The RJMCMC algorithm for model (4.7) consists of a within-model update and a between-model update.

4.3.2.1 Within-Model Update

Suppose at the $t+1$th iteration we propose $k^* = k^t$ from (4.8). The parameters at the end of tth iteration are denoted by $\theta^t = \left\{\alpha^t, \beta^t, \mu^t, \gamma_1^t, \ldots, \gamma_{k^t-1}^t\right\}$. The following steps update θ^t to θ^{t+1}:

1. For $\mu^{t+1}, \gamma_1^{t+1}, \ldots, \gamma_{k^t-1}^{t+1}$, we apply the Gibbs sampler algorithm from Sect. 4.2.2.
2. For $\alpha^{t+1}, \beta^{t+1}$, we apply the following M-H algorithm:

a. Propose $\alpha^* \sim \mathrm{N}\left(\alpha^t, 0.02^2\right), \beta^* \sim \mathrm{N}\left(\beta^t, 0.02^2\right)$.
b. Set $\theta^* = \left\{\alpha^*, \beta^*, \mu^{t+1}, \gamma_1^{t+1}, \ldots, \gamma_{k^t-1}^{t+1}\right\}$.
c. Calculate the acceptance as

$$\min\left(1, \frac{p\left(\boldsymbol{y}|\theta^*\right)\mathrm{N}\left(\alpha^t|\alpha^*, 0.02^2\right)\mathrm{N}\left(\beta^t|\beta^*, 0.02^2\right)}{p\left(\boldsymbol{y}|\theta^t\right)\mathrm{N}\left(\alpha^*|\alpha^t, 0.02^2\right)\mathrm{N}\left(\beta^*|\beta^t, 0.02^2\right)}\right),$$

where $\mathrm{N}\left(x|a, b\right)$ is the normal density at x with mean a and variance b.
d. With this acceptance rate, set $\alpha^{t+1} = \alpha^*, \beta^{t+1} = \beta^*$. Otherwise keep $\alpha^{t+1} = \alpha^t, \beta^{t+1} = \beta^t$.

3. Set $k^{t+1} = k^*, \theta^{t+1} = \left\{\alpha^{t+1}, \beta^{\mathrm{t}+1}, \mu^{t+1}, \gamma_1^{t+1}, \ldots, \gamma_{k^t-1}^{t+1}\right\}$. Note that the within-model acceptance rate of k^* is always 1.

4.3.2.2 Between-Model Update

Between-model update case 1:

Suppose at the $t+1$th iteration, we propose $k^* = k^t + 1$ from (4.8). The parameters at the end of the tth iteration are denoted by $\theta^t = \left\{\alpha^t, \beta^t, \mu^t, \gamma_1^t, \ldots, \gamma_{k^t-1}^t\right\}$. The following steps update θ^t to θ^{t+1}:

1. Propose a value u^t from a gamma distribution with shape of 100 and mean of $\exp\left(\alpha^t - k^t\beta^t\right)$, as follows:

$$u^t \sim q_{k^t \to k^*} = \mathrm{Gamma}\left(100, \frac{100}{\exp\left(\alpha^t - k^t\beta^t\right)}\right).$$

2. Set $\left(\theta^*, u^*\right) := T_{k^t \to k^*}\left(\theta^t, u^t\right) = \left(\theta^t, u^t\right)$, where u^* has zero-dimension. $T_{k^t \to k^*}$ is an identity mapping matrix with the Jacobian of 1.
3. Calculate the acceptance rate as

$$\min\left(1, \frac{p\left(\boldsymbol{y}|\theta^*\right)p\left(\theta^*\right)}{p\left(\boldsymbol{y}|\theta^t\right)p(\theta^t)\mathrm{Gamma}\left(u^t \middle| 100, \frac{100}{\exp(\alpha^t - k^t\beta^t)}\right)}\right).$$

4. With this acceptance rate, set $\left(k^{t+1}, \theta^{t+1}\right) = \left(k^*, \theta^*\right)$. Otherwise keep $\left(k^{t+1}, \theta^{t+1}\right) = \left(k^t, \theta^t\right)$.

Between-model update case 2:

Suppose at the $t+1$th iteration, we propose $k^* = k^t - 1$ from (4.8). The parameters at the end of the tth iteration are denoted by $\theta^t = \left\{\alpha^t, \beta^t, \mu^t, \gamma_1^t, \ldots, \gamma_{k^t-1}^t\right\}$. The following steps update θ^t to θ^{t+1}:

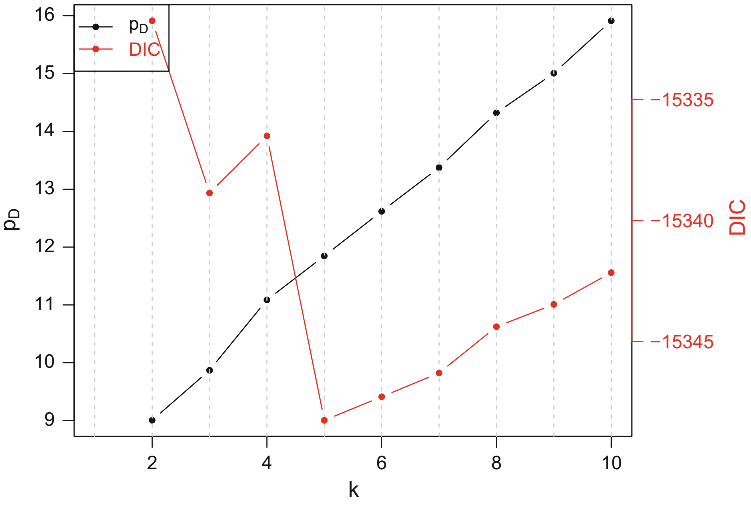


Fig. 4.6 DIC's and p_D's for the simulated data with respect to k

1. Set $\left(\theta^*, u^*\right) := T_{k^t \to k^*}\left(\theta^t, u^t\right) = \left(\theta^t, u^t\right)$, where u^t has zero-dimension, $u^* = \gamma^t_{k^t-1}$. $T_{k^t \to k^*}$ is an identity mapping matrix with the Jacobian of 1.
2. Calculate the acceptance rate as

$$\min\left(1, \frac{p\left(\mathbf{y}|\theta^*\right) p\left(\theta^*\right) \text{Gamma}\left(u^* \middle| 100, \frac{100}{\exp(\alpha^t-(k^t-1)\beta^t)}\right)}{p\left(\mathbf{y}|\theta^t\right) p(\theta^t)}\right).$$

3. With this acceptance rate, set $\left(k^{t+1}, \theta^{t+1}\right) = \left(k^*, \theta^*\right)$. Otherwise keep $\left(k^{t+1}, \theta^{t+1}\right) = \left(k^t, \theta^t\right)$.

Example 4.3 (*A Monte Carlo study of model* (4.7)) We specify the true parameters as follows:

$$I = 10, k = 5, \alpha = -1.4, \beta = 0.2, \varphi = 25{,}000$$
$$\mu = \left(10^7, 1.02 \times 10^7, \ldots, 1.02^9 \times 10^7\right)$$
$$\gamma = (0.159,\ 0.179,\ 0.179,\ 0.139)\,,$$

and simulate a sample from model (4.7).

DIC method:

We want to determine which k leads to the optimal model fit. Applying MCMC to different models indexed by k gives the corresponding DIC and p_D. We prefer the model with smaller DIC, thus $k = 5$ is preferred as shown in Fig. 4.6. Also note that p_D is always less than the length of θ_k, since p_D depends on the strength of priors, the structure of the Bayesian model and the data (Spiegelhalter et al. 2002).

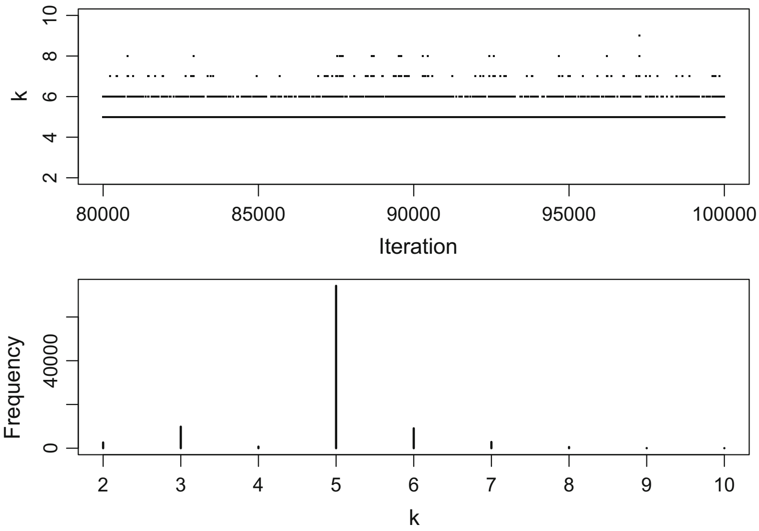


Fig. 4.7 The trace plot and the histogram of k

RJMCMC method:

We iterate for 10^5 times. The within-model acceptance rate is 0.37 and the between-model acceptance rate is 0.11. We plot the trace plot and the histogram of k in Fig. 4.7. In this example, DIC and RJMCMC suggest the same best model, $k = 5$. However, the DIC method takes a much longer time than RJMCMC. The reason is that the DIC method spends equal time on every model while RJMCMC always tends to jump to a more "accepted" model. Hence, in term of running time, RJMCMC is more efficient.

Example 4.4 (*Liability insurance claims data*) We continue with Example 4.1. DIC method suggests that the models with k larger than 7 perform equally well as shown in Fig. 4.8. We choose $k = 8$ to keep p_D as small as possible. RJMCMC is then applied starting from $k^0 = 3$ and iterating for 10^5 times. The trace plot and histogram of k are shown in Fig. 4.9. Again, the model with $k = 8$ is preferred. RJMCMC outperforms DIC in terms of distinguishing the "best" model from the other candidates.

We set $k = 8$ and estimate the posterior mean and the 95% CPDR of γ, comparing with the CL estimates (in logarithm scale) shown in Fig. 4.10. The development pattern after age 8 is smoothed to a straight line due to an exponential decay curve being used. The big jump at development period 23 represents a large proportion of tail development to the ultimate claims. In fact, the last point is valued as

$$\log\left(\sum_{j=I+1}^{J} \exp\left(\alpha - j\beta\right)\right).$$

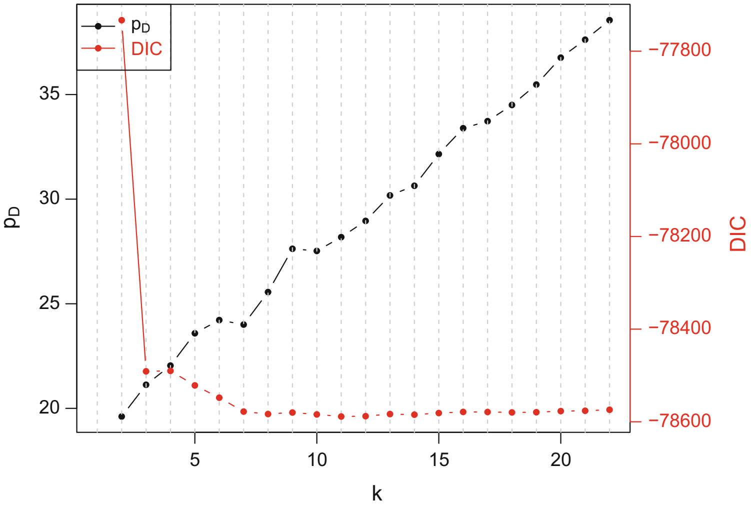


Fig. 4.8 DIC's and p_D's for Verrall and Wüthrich (2012) data with respect to k

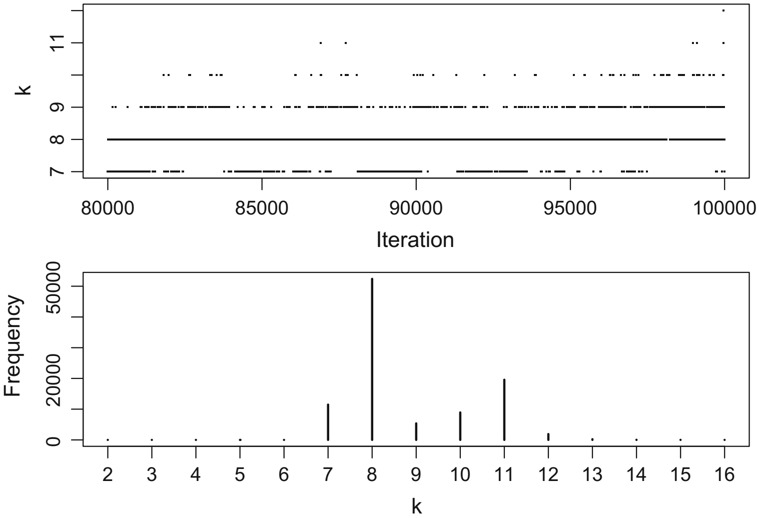


Fig. 4.9 The trace plot and the histogram of k for Verrall and Wüthrich (2012) data

We close this subsection by summarizing the total outstanding liability estimates from different models in Table 4.6. For model (4.1) and (4.2), $\hat{R}$ is an unbiased estimate and equal to the CL estimate. For model (4.4) and (4.7), $\hat{R}$ is an estimate of posterior mean.

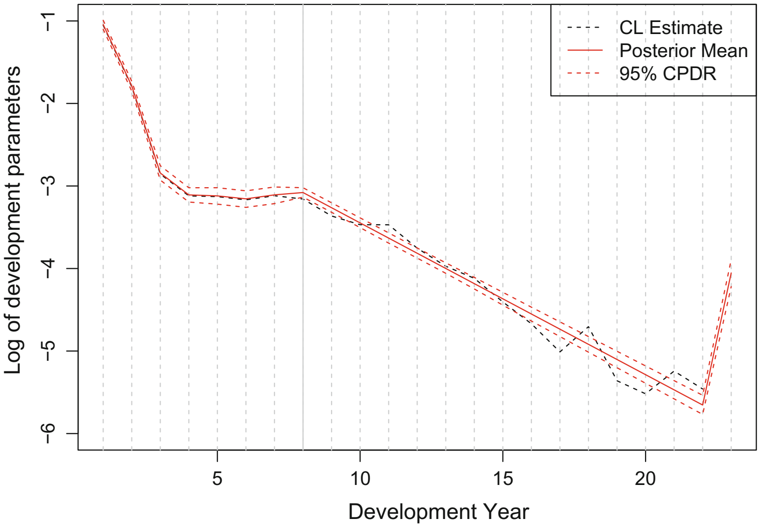


Fig. 4.10 The Logarithm of development parameters γ's including the tail factor

Table 4.6 Comparison of the total outstanding liability estimates from four different models

Model	Estimate	No tail	With tail
(4.1)	$\hat{R}$	1,463,076	1,599,558
	se(R)	55,300	58,528
(4.2)	$\hat{R}$	1,463,076	NA
	se(R)	60,444	NA
(4.4)	$\hat{R}$	1,463,312	NA
	se(R)	60,428	NA
(4.7)	$\hat{R}$	1,475,336	1,610,734
	se(R)	54,060	56,746

4.4 Estimation of Claims Liability in WorkSafe VIC

In this section, we analyze WorkSafe Victoria claims data to estimate the claims liabilities of the weekly benefit and doctor benefit. The data are from the actuarial valuation reports of outstanding claims liability for the scheme as of 30 June 2012 by Pricewaterhouse Coopers (PwC) Actuarial Pty Ltd (Simpson and McCourt 2012).

4.4.1 Background of WorkSafe Victoria

A company operating in Victoria must take out WorkSafe insurance if it pays more than $7,500 a year in rateable remuneration. WorkSafe insurance covers employee's

Table 4.7 Summary of the PwC report

Benefit	Sub-benefit	Method	Key note
Weekly	Weekly	PPAC	34% of the total liability
	Occupational rehabilitation	Relate to income	Help workers back to work
Medical and like	Doctor	PPCI	Shorter tail than weekly benefit
	Hospital	PPCI	Correlated with doctor
	Paramedical	PPAC	Generally ceases one year after weekly benefit
	Hearing aids	PPCI	Missing data before experience year 1994
	Personal and household services	PPAC	Including attendant care, personal services, home care, case management, home and vehicle modification payments
	Community integration program	CL on amounts	Personal & household services for catastrophically injured workers
	Medical reports	PPCI	Refers to independent medical examinations and treating health practitioners' reports
Common law	Common law damages and legal costs	PPCR	Relates to damages and costs arising from common law claims with respect to injuries occurring on or after 20 Oct 1999
	Old common law	PPCR	Date of injury prior to 12 Nov 1997
Impairment and death benefits	Impairment	PPCR	Injured workers can access impairment benefit if their whole person impairment is assessed as being 10% or more
	Maim	PPCR	The maim benefit is in run-off, being applicable only for injuries occurring prior to 12 Nov 1997
	Death lump sum	PPCR	Includes payments of statutory lump sum and interest payments on it
	Death pension	PPAC	Payment pattern determines the method used
Disputes, recoveries and others	Statutory legal	PPCR	All legal costs, other than those associated with common law cases, arising from workers and employers appealing decisions relating to eligibility of payments or continuance of benefits
	Investigation costs	PPCI	Can be incurred before any claims payments
	Recoveries	PPCI	Relates to recoveries from negligent third parties or recoveries of amounts where agents have paid injured workers in excess of the required amount
	Other	PPCI	Travel and accommodation costs

work related claims, such as back-injury during work. The benefits include income replacement, medical costs, rehabilitation etc. The premiums depend on the remuneration, the industry classification, industry claims history or its own business claims history, capping etc. Most of the functions associated with premium and claims management are performed by WorkSafe agents appointed by WorkSafe, including Allianz Australia Workers' Compensation Ltd., CGU Workers Compensation Ltd. etc.

4.4.1.1 Benefits

Depending on the features of a claim, one benefit or several benefits may be paid. A benefit can be a stream of payments extending for years or a lump sum. In the claims reserving problem, it is desirable to distinguish benefits in terms of payment period, settlement rate, average size etc. The PwC report divides claims payments into five benefits shown in Table 4.7, each of which has several sub-benefits. The reserving method is chosen for each sub-benefit depending on the benefit features and the data available. The last column in Table 4.7 provides some key information about each sub-benefit.

4.4.1.2 Reserving Methods Used by the PwC Report

The methods used in the PwC report mainly include payments per active claim (PPAC), payments per claim incurred (PPCI) and payments per claim resolved.

For example, it is suitable to use PPAC to model the weekly benefit. The weekly benefit is to compensate the loss of salary. So PPAC during a development year should be stably proportional to average weekly salary for that period. In contrast, PPCI is not suitable for the weekly benefit since PPCI does not take account of the duration of a claim, a main factor determining the weekly benefit.

4.4.2 Estimation of the Weekly Benefit Liability Using Models (4.1) and (4.7)

We analyze the weekly benefit using the distribution-free model with tail factor (4.1) and the Bayesian ODP model with tail factor (4.7). We will show that the tail development consists of a large percentage of total outstanding liability.

Table 4.8 The outstanding claims liability estimates of the weekly benefit from different models

Model	Expected value	Standard deviation	95% PI/CPDR
(4.1)	2,902,875,000	172,396,900	(2,558,081,200, 3,247,668,800)
(4.7)	3,127,649,615	145,385,671	(2,849,161,960, 3,417,721,458)
PwC	2,831,072,753	NA	NA

4.4.2.1 The Distribution-Free Model (4.1)

We apply this model to the incremental payments run-off triangle. The total outstanding liability is estimated as 2,902,875,000 dollars with the standard error of 172,396,900 dollars (CV = 6.0%). The PwC estimate of 2,831,072,753 dollars is within the 95% prediction interval (2,558,081,200, 3,247,668,800).

From the diagnostic plots in Fig. 4.11, we can see an obvious pattern in the standardized residuals vs. original years plot, which implies that the distribution-free model does not fit the data well (i.e. the model assumptions do not hold). The PwC report mentioned that the scheme structure changed in 2010, 2006, 1999 and 1997. These changes affected the weekly benefit, which more or less explains the pattern observed.

4.4.2.2 The Bayesian Over-Dispersed Poisson Model with Tail Factor (4.7)

First we apply the RJMCMC algorithm. The trace plot and the histogram of k are plotted in Fig. 4.12. Then we apply the M-H algorithm with $k = 8$ to estimate the outstanding liability. The tail factor is considered and J is assumed to be 37. The posterior mean of total outstanding liability is estimated as 3,127,649,615 dollars with the standard error of 145,385,671 dollars (CV = 4.6%). The 95% CPDR is (2,849,161,960, 3,417,721,458) as shown in Table 4.8.

4.4.2.3 Limitations

The above analysis demonstrates that real world problems are always more complex than our models. In the actuarial area, we typically use a statistical model to identify and quantify the *independent risk*. Other risks, such as event risk, strategic risk, operational risk, legal risk etc, are difficult to be quantified by a statistical model.

The models discussed in this chapter all assume that historical experience can predict the future. When the assumption does not hold, actuarial judgement is necessary to adjust the prediction inferred from the model. Nevertheless, a comprehensive understanding of model assumptions, historical events and possible future events is required before making any judgements.

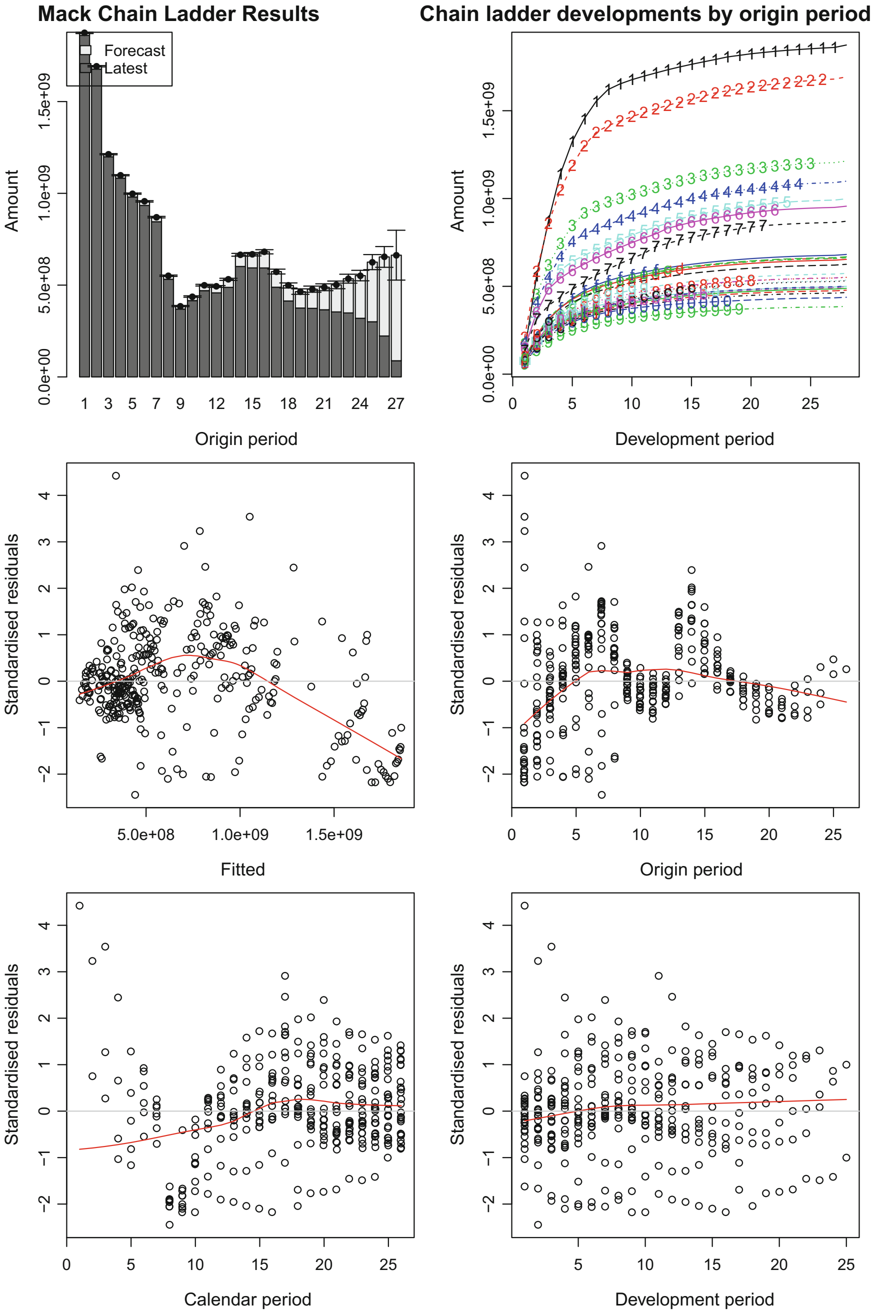


Fig. 4.11 The diagnostic plots for the distribution-free model applied to the weekly benefit

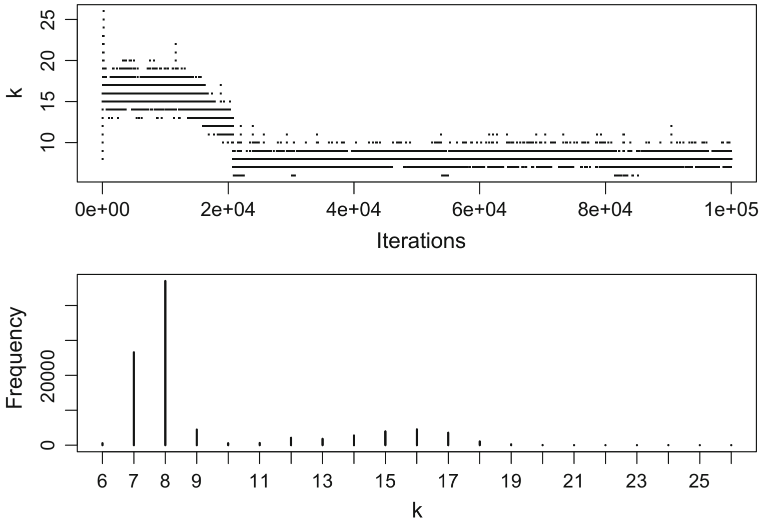


Fig. 4.12 The trace plot and the histogram of k for the weekly benefit data

4.4.3 *Estimation of the Doctor Benefit Liability Using a Compound Model*

The doctor benefit is not subject to changes in legislation as frequently as the weekly benefit, hence the historical claims data are much more instructive for the future claims. The PPCI method is used to analyze the doctor benefit. Compared to the CL method applied to the claims amounts directly, the PPCI method provides more information, such as the total incurred claims number and the average claim size. There are three steps in the PPCI method:

1. Project the ultimate incurred claims number for each accident year.
2. Divide the incremental claims amounts by the ultimate claims number to get the PPCI triangle, and project the PPCI triangle to get the outstanding PPCI.
3. Combine the ultimate claims number with the outstanding PPCI to get the outstanding liability.

Here we apply the Bayesian ODP model without tail factor model (4.4) to both the claims number and PPCI triangles, since the doctor benefit is not a long-tailed benefit. We then aggregate them using a compound model.

4.4.3.1 Preliminary GLM Analysis

Before going to the Bayesian analysis, we apply a quasi-likelihood GLM to the incremental claims number, in which a log link function and variance proportional to mean are specified. It is equivalent to fitting an ODP model (4.2). We get the scaled Pearson residual plot in Fig. 4.13.

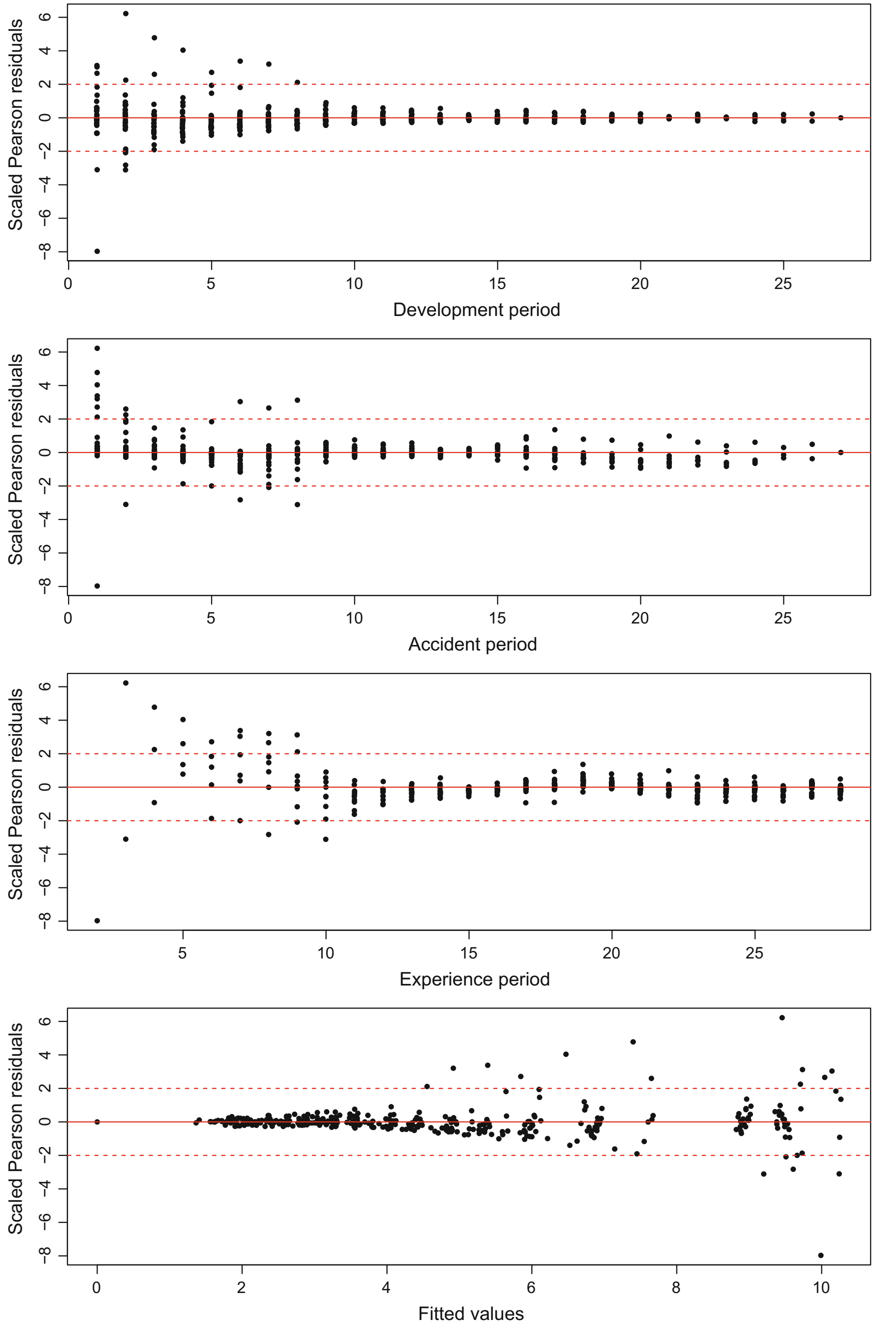


Fig. 4.13 The scaled Pearson residuals of the ODP model

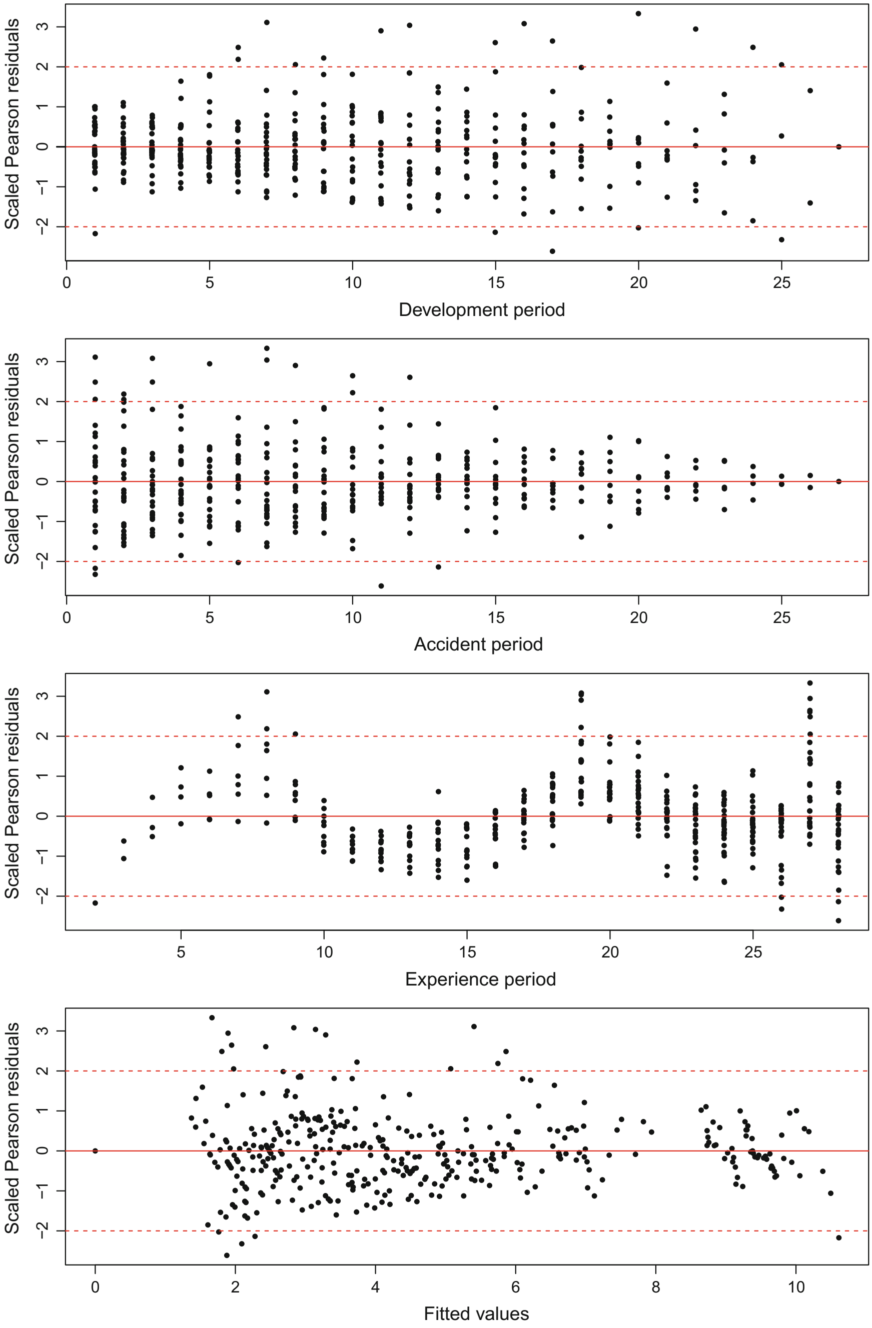


Fig. 4.14 The scaled Pearson residuals of the GLM with a gamma error and a log link function

It displays heteroscedasticity, implying the variance is proportional to the mean powered to more than one. We then try a GLM with the same link function but with variance proportional to mean squared, as follows:

$$n_{i,j} \sim \text{Gamma}\left(\alpha, \frac{\alpha}{\mu_i \gamma_j}\right), i = 1, \ldots, 27, j = 1, \ldots, 27.$$

A better residual plot is obtained as in Fig. 4.14. The scaled Pearson residual in this model is defined as

$$r_{ij} = \frac{e_{ij}}{\sqrt{\hat{\phi} V\left(\hat{n}_{ij}\right)}} = \frac{n_{ij} - \hat{n}_{ij}}{\hat{n}_{ij}} \sqrt{\hat{\alpha}}.$$

By dividing the incremental payments triangle by the ultimate claims number predicted from the above model, we get the PPCI triangle. The same process is applied to the PPCI triangle as to the claims number. Similarly, a gamma error distribution does a better fit than an ODP error structure.

This preliminary GLM fitting provides valuable information about the further Bayesian analysis. In the following, we will use the gamma error distribution for both claims numbers and PPCI.

4.4.3.2 A Bayesian Gamma Model for the Claims Numbers and PPCI

According to the preliminary GLM analysis, a Bayesian gamma model (similar to model (4.4)) is used here, as follows:

$$\begin{aligned} n_{i,j} &\sim \text{Gamma}\left(\alpha, \frac{\alpha}{\mu_i \gamma_j}\right) \\ \mu_i &\sim \text{Gamma}\left(a_i, b_i\right) \\ \gamma_j &\sim \text{Gamma}\left(c_j, d_j\right). \end{aligned}$$

The prior N (20000, 1000) is assumed for the ultimate claims numbers of the three most recent accident years, $\mu_i, i = 25,26,27$. The strong prior works as the BF method to reduce the leverage effect of the immature claims numbers. The Stan code is as follows:

```
number.code<-"
data{
  int N;                  // Number of observations
  int K;                  // Number of accident years
  int M;                  // Number of development years
  int acc[N];             // Accident years in upper triange
  int dev[N];             // Development years in upper triangle
  real first_inc[N];      // Number of claims in upper triangle
  int n;                  // Number of future prediction
  int acc_p[n];           // Accident years in lower triangle
  int dev_p[n];           // Development years in lower triangle
}
```

```
parameters{
  vector<lower=0,upper=50000>[K] ult;
  vector[M] dev_raw;
  real<lower=0> alpha;
}
transformed parameters{
  vector[N] means;
  vector[M] dev_norm;  //Normalized development pattern
  dev_norm<-exp(dev_raw)/sum(exp(dev_raw));
  for (i in 1:N){
    means[i]<-dev_norm[dev[i]]*ult[acc[i]];
  }
}
model{
  for (i in 1:N)
    first_inc[i] ~ gamma(alpha,alpha/means[i]);
  for (i in 25:27)
    ult[i] ~ normal(20000,1000);
}
generated quantities
{
  real pearson_res[N];
  real means_p[n];
  for (i in 1:N)
    pearson_res[i]<-(first_inc[i]-means[i])/means[i]*sqrt(alpha);
  for (i in 1:n)
    means_p[i]<-dev_norm[dev_p[i]]*ult[acc_p[i]];
}
"
```

The posterior mean of residuals vs. linear predictors is plotted in Fig. 4.15, which shows a similar pattern to Fig. 4.14. It seems that the variance is proportional to the mean powered to some value between 1 and 2. We could use a Tweedie family in `glm( )` function in R, but Stan does not have such a distribution. The predictive distribution of outstanding claims numbers is positively skewed. The posterior mean of outstanding claims number is estimated as 13,923, which is higher than the PwC estimate of 12,811. It takes one minute to run 1,600 iterations. We use the posterior means of ultimate claims numbers to derive the PPCI triangle and fit the same model as for the claims numbers. The residual plot and the histogram of total outstanding PPCI are shown in Fig. 4.16. The predictive distribution of outstanding PPCI is roughly symmetric with the posterior mean of 18,012 dollars, compared with the PwC estimate of 17,827 dollars.

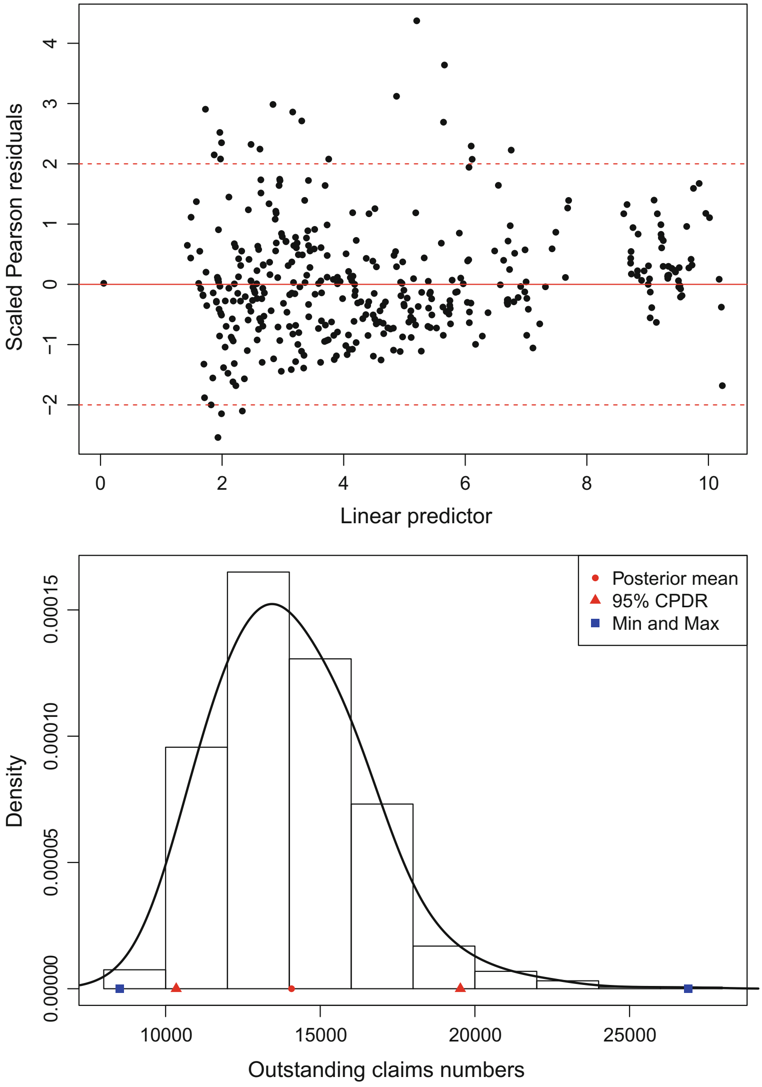


Fig. 4.15 The residual plot and the histogram of total outstanding claims number

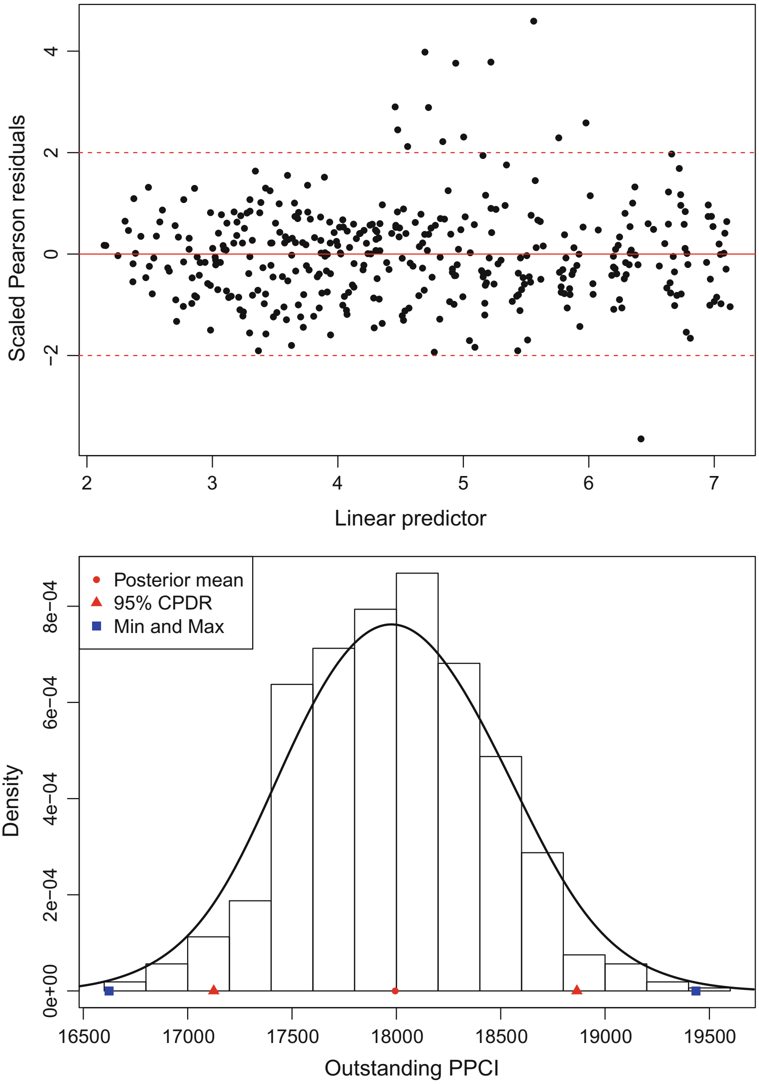


Fig. 4.16 The residual plot and the histogram of total outstanding PPCI

4.4.3.3 A Compound Model to Combine the Ultimate Claims Numbers and the Outstanding PPCI

Ideally, we should use the predictive distribution of ultimate claims numbers to derive the PPCI triangle, then combine the predictive distribution of the outstanding PPCI with the corresponding ultimate claims numbers to get the predictive distribution of outstanding liability. This method requires a large amount of computing time.

Here we propose a compound model to get the predictive distribution of outstanding liability. The model is specified as follows:

$$\begin{aligned} y_{ij} &= \sum_{k=1}^{\mu_i} x_{ijk},\ i = 1, \ldots, 27,\ j = 1, \ldots, 27 \\ \mu_i &\sim \text{Distribution}_i \\ x_{ijk} &\sim \text{Gamma}\left(\alpha_{ij}, \beta_{ij}\right),\ k = 1, \ldots, \mu_i, \end{aligned}$$

where μ_i is the ultimate claims number of accident year i whose distribution is approximated by a Bayesian model, and x_{ijk} is the payment for the kth claim during the development year j, with the distribution depending on both accident year and development year.

The payments per claim incurred (PPCI) during the development period j of accident year i is defined as

$$\text{PPCI}_{ij} := y_{ij}/\mathbb{E}\left(\mu_i\right).$$

Note that $\mathbb{E}(\text{PPCI}_{ij}) = \mathbb{E}(x_{ijk})$. The posterior mean of μ_i is an estimate of $\mathbb{E}\left(\mu_i\right)$.

The relationship between the variance of PPCI_{ij} and the variance of x_{ijk} is as follows:

$$\begin{aligned} \text{Var}\left(\text{PPCI}_{ij}\right) &= \text{Var}\left(\frac{\sum_{k=1}^{\mu_i} x_{ijk}}{\mathbb{E}\left(\mu_i\right)}\right) \\ &= \frac{\text{Var}\left(x_{ijk}\right)\mathbb{E}\left(\mu_i\right) + \left(\mathbb{E}\left(x_{ijk}\right)\right)^2 \text{Var}\left(\mu_i\right)}{\left(\mathbb{E}\left(\mu_i\right)\right)^2} \\ &= \frac{\text{Var}\left(x_{ijk}\right)\mathbb{E}\left(\mu_i\right) + \left(\mathbb{E}\left(\text{PPCI}_{ij}\right)\right)^2 \text{Var}\left(\mu_i\right)}{\left(\mathbb{E}\left(\mu_i\right)\right)^2}. \end{aligned}$$

We can solve $\text{Var}\left(x_{ijk}\right)$ as

$$\text{Var}\left(x_{ijk}\right) = \frac{\left(\mathbb{E}\left(\mu_i\right)\right)^2 \text{Var}\left(\text{PPCI}_{ij}\right) - \text{Var}\left(\mu_i\right)\left(\mathbb{E}\left(\text{PPCI}_{ij}\right)\right)^2}{\mathbb{E}\left(\mu_i\right)}, \tag{4.9}$$

where all the quantities on the right hand side can be estimated by a MC sample. The distribution of y_{ij} conditional on μ_i is $\text{Gamma}\left(\mu_i\alpha_{ij}, \beta_{ij}\right)$, where $\alpha_{ij} = \mathbb{E}\left(x_{ijk}\right)^2/\text{Var}\left(x_{ijk}\right)$, $\beta_{ij} = \alpha_{ij}/\mathbb{E}\left(x_{ijk}\right)$.

The outstanding claims liability of accident year i is $R_i|\mu_i = \sum_{j=I-i+1}^{I} y_{ij}$. The predictive distribution of total claims liability is shown in Fig. 4.17. The posterior mean of total claims liability is estimated as 391,761,803 dollars with the standard deviation of 10,195,111 (CV = 2.6%), compared with 396,827,792 dollars estimated by PwC. The 95% CPDR of total claims liability is estimated as (373,902,941, 414,549,267). We summarize the predictions made from the compound model in Table 4.9.

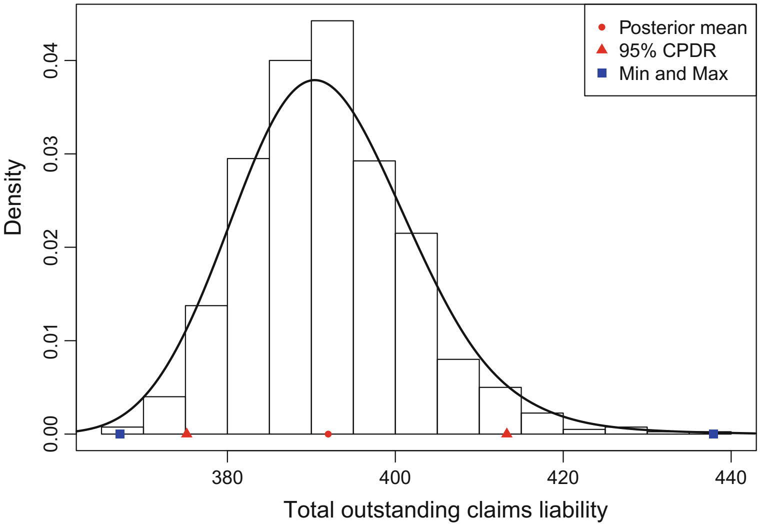


Fig. 4.17 The predictive distribution of total outstanding liability of the doctor benefit

Table 4.9 Summary of the predictions made from the compound model

	Post. mean	Std. deviation	95% CPDR	PwC estimate
O/S claims no.	13,923	2,407	(9,742, 19,117)	12,811
O/S PPCI	18,012	474	(17,056, 18,901)	17,827
O/S liability	391,761,803	10,195,111	(373,902,941, 414,549,267)	396,827,792

4.4.3.4 Other Ways to Combine the Ultimate Claims Numbers with the Outstanding PPCI

As a final remark, we point out that the PPCI triangle is conditional on the posterior mean of ultimate claims number, i.e., $\mathbb{E}(\mu_i|\boldsymbol{y})$. If we only consider the variation in PPCI and keep the ultimate claims numbers fixed at the posterior mean, we would underestimate the variation of outstanding liability, i.e., we ignore the estimation error in $\mathbb{E}(\mu_i|\boldsymbol{y})$.

The key point of the compound model is Eq. (4.9), which recovers the variation in a single claim payment x_{ijk}, which is assumed to be independent of the ultimate claims number μ_i.

4.5 Discussion

Occasionally, we see some abnormal values in a particular diagonal line or some pattern in the residuals vs. experience periods plot. This is called the *experience period effect* or *calendar period effect*. It can be due to the uncommon inflation rates in a particular calendar year. The straightforward way to address this problem is to involve an experience period covariate. This covariate effectively isolate the outliers in the diagonal lines, so the estimation of accident period parameters and development period parameters are not affected.

For the run-off triangle data, the experience period parameters are not used in the prediction of future claims since all future claims correspond to new experience periods. So the main purpose of introducing the experience period covariate is to remove the discontinuous abnormal calendar year effect.

An innovative contribution made in this chapter is using a compound model to quantify the uncertainty associated with the estimates from the PPCI method. The distributional assumption of x_{ijk} has not been checked. To check this assumption, we need the payments data during the whole life of individual claims.

We also stress the importance of preliminary GLM fitting. Bayesian modelling needs time-consuming inferential tools. We normally cannot get the inference and do the goodness-of-fit check of a Bayesian model as easily as a GLM. So a preliminary GLM fitting can help us set up the Bayesian model with regards to the error distribution, the mean function, the priors for parameters etc.

Finally, we point out that it is hard to program RJMCMC and there are no statistical packages available to do RJMCMC directly. To avoid RJMCMC but still incorporate a tail factor, a non-linear curve mean function, such as log-logistic curve and Hoerl curve (Taylor 2000), can be used. If these non-linear curves are used, GLM will not work, which demonstrates an advantage of Bayesian models. In the next chapter, rather than using curves, we go a step further to use a basis expansion model, which is a non-parametric approach.

4.6 Bibliographic Notes

The Bornhuetter-Ferguson method derives from Bornhuetter and Ferguson (1972). Friedland (2010) is the reading material of the CAS Exam 5 and provides an overview of basic techniques of estimating unpaid claims. For the stochastic claims reserving methods, Mack (1993, 1999, 2008) established the Mack's models. Australian actuaries are largely influenced by Taylor (2000). England and Verrall (2002, 2006) and Wüthrich and Merz (2008, 2015) are summaries of stochastic reserving models.

An excellent GLM reference is McCullagh and Nelder (1989). The references of ODP model in claims reserving problem include Renshaw and Verrall (1998), Verrall (2000, 2004), Alai et al. (2009), Saluz et al. (2011), England et al. (2012), Verrall and Wüthrich (2012) and Wüthrich (2013a).

Other papers using a Bayesian approach include Scollnik (2001), De Alba (2002), Ntzoufras and Dellaportas (2002) and Meyers (2009, 2015).

Clark (2003) and Zhang et al. (2012) used the stochastic curve models. Brydon and Verrall (2009) and Wüthrich (2013a) considered the calendar year effect. Piwcewicz (2008) and Beens et al. (2010) are two presentations about Bayesian claims reserving method in IAA's non-life insurance seminars.

Verrall et al. (2012) and Verrall and Wüthrich (2012) used RJMCMC. RJMCMC is proposed by Green (1995). The collective risk model (or aggregate risk model) have been much studied in the standard risk modelling text books such as Klugman et al. (2012) and Gray and Pitts (2012). Gao et al. (2018) investigates the uncertainty associated with the PPCI method from a different perspective.

References

Alai, D. H., Merz, M., & Wüthrich, M. V. (2009). Mean square error of prediction in the Bornhuetter-Ferguson claims reserving method. *Annals of Actuarial Science*, *4*, 7–31.

Beens, F., Bui, L., Collings, S., & Gill, A. (2010). Stochastic reserving using Bayesian models: Can it add value? In *Institute of Actuaries of Australia 17th General Insurance Seminar*.

Bornhuetter, R. L., & Ferguson, R. (1972). The actuary and IBNR. *Proceedings of the Casualty Actuarial Society*, *59*, 181–195.

Brydon, D., & Verrall, R. J. (2009). Calendar year effects, claims inflation and the chain-ladder technique. *Annals of Actuarial Science*, *4*, 287–301.

Clark, D. R. (2003). LDF curve-fitting and stochastic reserving: A maximum likelihood approach. *Casualty Actuarial Society Forum (Fall 2003)*, 41–92.

De Alba, E. (2002). Bayesian estimation of outstanding claim reserves. *North American Actuarial Journal*, *6*, 1–20.

Efron, B., & Tibshirani, R. J. (1994). *An Introduction to the Bootstrap*. New York: Chapman & Hall.

England, P. D., & Verrall, R. J. (2002). Stochastic claims reserving in general insurance. *British Actuarial Journal*, *8*, 443–518.

England, P. D., & Verrall, R. J. (2006). Predictive distributions of outstanding liabilities in general insurance. *Annals of Actuarial Science*, *1*, 221–270.

England, P. D., Verrall, R. J., & Wüthrich, M. V. (2012). Bayesian over-dispersed poisson model and the Bornhuetter-Ferguson claims reserving method. *Annals of Actuarial Science*, *6*, 258–283.

Faraway, J. J. (2015). *Linear models with R* (2nd ed.). Boca Raton: Chapman & Hall.

Friedland, J. (2010). Estimating unpaid claims using basic techniques. *Casualty Actuarial Society Study Notes*.

Gao, G., Meng, S., & Shi, Y. (2018). Stochastic payments per claim incurred. *North American Actuarial Journal*, (forthcoming).

Gesmann, M., Murphy, D., Zhang, W., Carrato, A., Crupi, G., Wüthrich, M. V., et al. (2015). *Chain ladder: statistical methods and models for claims reserving in general insurance*. R package version 0.2.1.

Gilks, W. R., & Wild, P. (1992). Adaptive rejection sampling for Gibbs sampling. *Journal of the Royal Statistical Society C*, *41*, 337–348.

Gray, R. J., & Pitts, S. M. (2012). *Risk modelling in general insurance: From principles to practice*. Cambridge: Cambridge University Press.

Green, P. J. (1995). Reversible jump Markov chain Monte Carlo computation and Bayesian model determination. *Biometrika*, *82*, 711–732.

Klugman, S. A., Panjer, H. H., & Willmot, G. E. (2012). *Loss models: From data to decisions* (4th ed.). New York: Wiley.

Mack, T. (1993). Distribution-free calculation of the standard error of chain ladder reserve estimates. *ASTIN Bulletin*, *23*, 213–225.

Mack, T. (1999). The standard error of chain-ladder reserve estimates, recursive calculation and inclusion of a tail factor. *ASTIN Bulletin*, *29*, 361–366.

Mack, T. (2008). The prediction error of Bornhuetter-Ferguson. *ASTIN Bulletin*, *38*, 87.

McCullagh, P., & Nelder, J. A. (1989). *Generalized linear models* (2nd ed.). New York: Chapman & Hall.

Meyers, G. (2009). Stochastic loss reserving with the collective risk model. *Variance*, *3*, 239–269.

Meyers, G. (2015). Stochastic loss reserving using Bayesian MCMC models. *CAS Monograph Series*, *1*, 1–64.

Ntzoufras, I., & Dellaportas, P. (2002). Bayesian modelling of outstanding liabilities incorporating claim count uncertainty. *North American Actuarial Journal*, *6*, 113–125.

Piwcewicz, B. (2008). Stochastic reserving: case study using a Bayesian approach. In *Institute of Actuaries of Australia 16th General Insurance Seminar*.

Renshaw, A. E., & Verrall, R. J. (1998). A stochastic model underlying the chain-ladder technique. *British Actuarial Journal*, *4*, 903–923.

Saluz, A., Gisler, A., & Wüthrich, M. V. (2011). Development pattern and prediction error for the stochastic Bornhuetter-Ferguson claims reserving method. *ASTIN Bulletin*, *41*, 279–313.

Scollnik, D. P. (2001). Actuarial modeling with MCMC and BUGS. *North American Actuarial Journal*, *5*, 96–124.

Simpson, L., & McCourt, P. (2012). *Worksafe Victoria actuarial valuation of outstanding claims liability for the scheme as at 30 June 2012*, Technical report. PricewaterhouseCoopers Actuarial Pty Ltd.

Spiegelhalter, D. J., Best, N. G., Carlin, B. R., & van der Linde, A. (2002). Bayesian measures of model complexity and fit. *Journal of the Royal Statistical Society B*, *64*, 583–616.

Taylor, G. (2000). *Loss reserving: An actuarial perspective*. Boston: Kluwer Academic Publishers.

Verrall, R. J. (2000). An investigation into stochastic claims reserving models and the chain-ladder technique. *Insurance: Mathematics and Economics 26*, 91–99.

Verrall, R. J. (2004). A Bayesian generalized linear model for the Bornhuetter-Ferguson method of claims reserving. *North American Actuarial Journal*, *8*, 67–89.

Verrall, R. J., & Wüthrich, M. V. (2012). Reversible jump Markov chain Monte Carlo method for parameter reduction in claims reserving. *North American Actuarial Journal*, *16*, 240–259.

Verrall, R. J., Hössjer, O., & Björkwall, S. (2012). Modelling claims run-off with reversible jump Markov chain Monte Carlo methods. *ASTIN Bulletin*, *42*, 35–58.

Wüthrich, M. V. (2013a). Calendar year dependence modeling in run-off triangles. In *ASTIN Colloquium*, 21–24.

Wüthrich, M. V. (2013b). Challenges with non-informative gamma priors in the Bayesian over-dispersed Poisson reserving model. *Insurance: Mathematics and Economics, 52*, 352–358.

Wüthrich, M. V., & Merz, M. (2008). *Stochastic claims reserving methods in insurance*. Chichester: Wiley.

Wüthrich, M. V., & Merz, M. (2015). Stochastic claims reserving manual: Advances in dynamic modelling. *SSRN*, ID 2649057.

Zhang, Y., Dukic, V., & Guszcza, J. (2012). A Bayesian non-linear model for forecasting insurance loss payments. *Journal of the Royal Statistical Society A*, *175*, 637–656.

Chapter 5
Bayesian Basis Expansion Models

Abstract In this chapter, Bayesian basis expansion models are used to fit various development patterns and accommodate the tail factor. A parametric model is typically characterized by a parametric mean function and an error distribution. The shape of the mean function is restricted by the space of parameters. Non-parametric models such as basis expansion models are able to automatically adjust to fit any shape of data. In Sect. 5.1, the aspects of splines are reviewed, including spline basis functions, smoothing splines, low rank smoothing splines and Bayesian shrinkage splines. In Sect. 5.2, we study two simulated examples. The first simulated example is based on a trigonometric mean function, while the second simulated example is based on the claims payments process. Both examples illustrate the usefulness of natural cubic spline basis in the extrapolation beyond the range of data. Section 5.3 is the application of above methodology to the doctor benefit in WorkSafe Victoria. The basis expansion model used to fit the PPCI triangle induces a tail development.

5.1 Aspects of Splines

There is a trade-off between flexibility and simplicity in model fitting. Basis expansion models on one hand are more flexible, able to be adjusted to fit various shapes of data, while on the other hand, they are more complicated (i.e., involve more parameters). Before using a non-parametric model, we should consider whether there is a capable parametric model. The *log-logistic curve* and *Hoerl curve* together with the models in the previous chapter can tackle many claims reserving problems.

Consider the following underlying true model:

$$y_i \sim f(x_i) + \varepsilon_i, i = 1, \dots, n,$$

where ε_i are i.i.d $\mathrm{N}\left(0,\ \sigma_\varepsilon^2\right)$. A non-parametric approach is to approximate f by a non-parametric function m. Basis expansion is a way to express the form of m. The core idea of basis expansion is to expand the input x with additional variables, which are transformations of x, and then to apply linear models to this newly expanded space of input x. In basis expansion models, m is written as a linear combination of basis functions, as follows:

© Springer Nature Singapore Pte Ltd. 2018

G. Gao, *Bayesian Claims Reserving Methods in Non-life Insurance with Stan*,
https://doi.org/10.1007/978-981-13-3609-6_5

$$m\left(x\right)=\sum_{h=1}^{H}\beta_h b_h\left(x\right),$$

where b_h is called a *basis function*. A common choice of b_h is a polynomial. The mechanism of defining b_h determines the behaviour of m. Here we consider m as *splines*, which use polynomials as basis functions with some constraints. Splines are a combination of polynomials and *step functions*. In polynomial models, the basis functions have the form of $b_h\left(x\right)=x^h$. Polynomial models tend to capture the shape of the data as long as there are high-degree polynomials. A disadvantage of polynomial models is the global representation of basis functions, which means all the data points can affect parameter estimation and every parameter can affect the mean function. A step function model partitions the data into H parts and fits the hth part using a basis function $b_h\left(x\right)$ whose value is zero for the remaining parts of data. Step function models have a disadvantage of discontinuity at the boundaries of partition. Spline models are a combination of polynomial models and step function models. For example, a cubic spline is a series of piecewise-cubic polynomials joined continuously up to the second derivatives. The properties of continuity and being piecewise are realised by using a particular set of basis functions.

5.1.1 Basis Functions of Splines

5.1.1.1 Truncated Power Basis

One intuitive choice is truncated power basis of degree p, which contains $K+p+1$ basis functions as follows:

$$1, x, \ldots, x^p, \left(x-\kappa_1\right)_+^p, \ldots, \left(x-\kappa_K\right)_+^p,$$

where $\left(x-\kappa_i\right)_+^p=\left(x-\kappa_i\right)^p$ for $x>\kappa_i$ and 0 elsewhere, κ_i, $i=1,\ldots,K$ are called *knots*. The basis functions consist of two parts: the global polynomials up to degree p, and the truncated degree p polynomials which have the local representation property. It can be shown that any linear combination of these basis functions has continuous derivatives up to order $p-1$ at every knot.

The degrees of freedom of a spline is the number of parameters in the mean function. Truncated power basis of degree p has $K+p+1$ degrees of freedom, which is intuitive to join $K+1$ pieces of degree p polynomials smoothly (up to $p-1$th derivatives at knots), Kp degrees of freedom are lost, leaving $K+p+1$ degrees of freedom, i.e., $K+p+1=\left(K+1\right)\left(p+1\right)-Kp$. In the GLM setting, we write the design matrix as

$$\boldsymbol{X}=\begin{bmatrix}1 & x_1 & \cdots & x_1^p & \left(x_1-\kappa_1\right)_+^p & \cdots & \left(x_1-\kappa_K\right)_+^p\\ 1 & \cdots\cdots\cdots & & \cdots & \cdots & & \cdots\\ 1 & x_n & \cdots & x_n^p & \left(x_n-\kappa_1\right)_+^p & \cdots & \left(x_n-\kappa_K\right)_+^p\end{bmatrix}.$$

At first glance, it seems that spline models are more complicated than either polynomial models or step function models. This is not true. Compared with polynomial models, we do not need the higher degree polynomials to capture all the curvatures of the data, since we have the local basis functions. Compared with step function models, we overcome the problem of discontinuity via the mechanism of basis functions. Spline models combine the advantages of both polynomial and step functions models, and get rid of the flaws of both models when they are used alone.

A truncated power basis has a practical disadvantage in that it is far from orthogonal, i.e., the columns of design matrix $\boldsymbol{X}$ are not orthogonal. It is better to work with an equivalent basis with more stable numerical properties. Note that two bases are equivalent if they span the same set of functions.

5.1.1.2 B-Spline Basis

The most common choice for spline basis is the B-spline basis of degree p, which consists of piecewise continuous functions only non-zero over the intervals between $p + 2$ adjacent knots. The degrees of freedom of a K-knot degree p B-spline basis is $K - p + 1$, since the spline is to be evaluated only over the interval $\left[\kappa_{p+1}, \kappa_{K-p}\right]$.

To span the same function space as truncated power basis of degree p with K knots, we need to add p arbitrary knots to the ends of $[\kappa_1, \kappa_K]$, i.e., we usually choose the knots $\{\kappa_1, \kappa_1, \kappa_1, \kappa_1, \kappa_2, \ldots, \kappa_{K-1}, \kappa_K, \kappa_K, \kappa_K, \kappa_K\}$ in a cubic B-spline basis. A B-spline basis is an orthogonal set.

5.1.1.3 Radial Basis

Another set of basis functions equivalent to a truncated power basis of degree p (odd) is a *radial basis*, as follows:

$$1, x, \ldots, x^p, |x - \kappa_1|^p, \ldots, |x - \kappa_K|^p.$$

We will come back to radial basis functions in smoothing splines and Bayesian shrinkage splines.

5.1.2 Smoothing Splines

Smoothing splines come from the solution to the optimal problem of finding a function g to minimizes the residual sum of squares (RSS) plus a penalty on the integral of the squared second derivatives of g. This penalized residual sum of squares is

$$\text{RSS}\,(g, \lambda) = \sum_{i=1}^{n} [y_i - g\,(x_i)]^2 + \lambda \int \left[g''\,(t)\right]^2 dt, \tag{5.1}$$

where λ is a fixed smoothing parameter. The first term measures closeness to the data, while the second term penalizes curvature in the function and λ establishes a trade-off between the two. Two special cases are: $\lambda \to 0$, $\hat{g}$ can be any function that interpolates the data (i.e., RSS $= 0$); $\lambda \to \infty$, $\hat{g}$ is the simple linear regression fit since no second derivative can be tolerated. Note that a smoothing spline is a one-dimensional thin plate spline.

Remarkably, even without the constraint of g as splines, it can be shown, for $0 < \lambda < \infty$, that $\hat{g}$ is a *natural cubic spline* with knots placed at the unique values of $x_i, i = 1, \ldots, n$ (Hastie and Tibshirani 1990). Natural cubic splines are cubic splines with the constraint that they are linear beyond the boundary knots. Hence, the degrees of freedom of a smoothing spline $\hat{g}$ are n (i.e., $n = n + 3 + 1 - 2 - 2$), since 4 degrees of freedom are lost due to the linear constraints at two boundary knots.

We can write this natural cubic spline as

$$\hat{g}(x) = \sum_{h=1}^{n} \hat{\beta}_h b_h(x),$$

where $\{b_h : h = 1, \ldots, n\}$ is a set of n basis functions for representing this natural cubic spline. We write the design matrix as

$$\boldsymbol{X} = \begin{bmatrix} b_1(x_1) & \cdots & b_n(x_1) \\ \cdots & \cdots & \cdots \\ b_1(x_n) & \cdots & b_n(x_n) \end{bmatrix},$$

which is an $n \times n$ matrix. RSS in (5.1) can be written as

$$\text{RSS}(\beta, \lambda) = \boldsymbol{y} - \boldsymbol{X}\beta + \lambda\beta^T \boldsymbol{\Omega}\beta, \tag{5.2}$$

where $\boldsymbol{\Omega}[i, j] = \int b_i''(t) b_j''(t) dt$. The solution is $\hat{\beta} = \left(\boldsymbol{X}^T\boldsymbol{X} + \lambda\boldsymbol{\Omega}\right)^{-1}\boldsymbol{X}^T\boldsymbol{y}$, which has a additional penalty term $\lambda\boldsymbol{\Omega}$ compared with the ordinary least squares solution.

5.1.2.1 Rank of a Smoother and Effective Degrees of Freedom

The fitted values of smoothing splines are

$$\hat{\boldsymbol{y}} = \boldsymbol{X}\left(\boldsymbol{X}^T\boldsymbol{X} + \lambda\boldsymbol{\Omega}\right)^{-1}\boldsymbol{X}^T\boldsymbol{y} = \boldsymbol{S}_\lambda\boldsymbol{y},$$

where $\boldsymbol{S}_\lambda$ is known as the *smoother matrix* or *hat matrix*. We list some features of $\boldsymbol{S}_\lambda$ as follows:

- $\boldsymbol{S}_\lambda$ is a symmetric positive semi-definite matrix with rank n.
- $\boldsymbol{S}_\lambda$ has n eigenvectors and n non-zero eigenvalues.
- λ cannot affect the eigenvectors of $\boldsymbol{S}_\lambda$.
- λ affects the eigenvalues of $\boldsymbol{S}_\lambda$ negatively, except the first two which are always 1 corresponding to the two-dimensional eigenspace of functions linear in x. Other eigenvalues are between 0 and 1 depending on λ.
- The degree of freedom of $\boldsymbol{S}_\lambda$ is $\mathrm{df}_\lambda = \mathrm{trace}\,(\boldsymbol{S}_\lambda) =$ sum of eigenvalues, which is always between 2 and n.
- When $\lambda \to 0$, all the eigenvalues are 1. $\mathrm{df}_\lambda = \mathrm{trace}\,(\boldsymbol{S}_\lambda) =$ sum of eigenvalues $= n$, corresponding to any functions interpolating the data.
- When $\lambda \to \infty$, all the eigenvalues are 0 except the first two. $\mathrm{df}_\lambda = \mathrm{trace}\,(\boldsymbol{S}_\lambda) =$ sum of eigenvalues $= 2$, corresponding to a straight line.

5.1.2.2 Radial Basis Functions for Smoothing Splines

Smoothing splines have a natural representation in terms of radial basis functions. For a given λ, a smoothing spline can be written as

$$\hat{g}\,(x) = \hat{\gamma}_0 + \hat{\gamma}_1 x + \sum_{k=1}^{n} \hat{\delta}_k |x - x_k|^3,$$

where $\hat{\theta} = \left(\hat{\gamma}_0, \hat{\gamma}_1, \hat{\delta}_1, \ldots, \hat{\delta}_n\right)$ minimizes the penalized residual sum of squares

$$\sum_{i=1}^{n} \left(y_i - \hat{\gamma}_0 - \hat{\gamma}_1 x_i - \sum_{k=1}^{n} \hat{\delta}_k |x_i - x_k|^3 \right)^2 + \lambda \sum_{i=1}^{n} \hat{\delta}_i \sum_{k=1}^{n} \hat{\delta}_k |x_i - x_k|^3, \tag{5.3}$$

subject to the constraints $\sum_{k=1}^{n} \hat{\delta}_k = \sum_{k=1}^{n} \hat{\delta}_k x_k = 0$. The constraints make the number of parameters n rather than $n + 2$ which is consistent with the degrees of freedom of a smoothing spline.

The criterion (5.3) is connected with the criterion of *best linear unbiased prediction* (BLUP) in a mixed effects model, which opens a gate for a Bayesian mixed effects model representing a smoothing spline.

5.1.2.3 Choice of λ

The above discussion is based on a given λ. We can treat λ as a *tuning parameter* which indexes different smoothing models. The choice of λ can be thought as a model selection problem. The selection criterion relates to a model's prediction capability on an independent test data set. Typically, we use test error as a measure of prediction capability, defined as the prediction squared error over an independent test sample.

The most widely used method for estimating the test error is cross-validation (see Sect. 2.2). λ is chosen by minimizing CV or generalized CV (Hastie et al. 2009).

5.1.3 *Low Rank Thin Plate Splines*

The rank of smoother $\boldsymbol{S}_\lambda$ is the number of distinct x. Sometimes it is called a *full rank smoother*. Wood (2003, 2006) uses the truncated eigen-decomposition of $\boldsymbol{X}$ to achieve a low rank smoother approximating the full rank smoother. A simpler approximation is to set up a new natural cubic spline basis with specified knots $\kappa_i,\ i = 1, \ldots, K$, rather than at every distinctive x.

It can be shown that a natural cubic spline with specified knots fitted by minimizing (5.1) can approximate the full rank smoothing spline well (Ruppert et al. 2003). A spline with fixed knots is called a *spline regression*. If it is fitted by minimizing (5.1), it is called a *penalized spline regression*, or more generally a *low rank thin plate spline*.

5.1.3.1 Rank of a Fixed-Knot Thin Plate Spline and Effective Degrees of Freedom

Some features of a K-knot thin plate spline smoother $\boldsymbol{S}_\lambda$ are as follows:

- $\boldsymbol{S}_\lambda$ is a symmetric positive semi-definite matrix with rank of K.
- $\boldsymbol{S}_\lambda$ has K eigenvectors and K non-zero eigenvalues.
- λ cannot affect the eigenvectors of $\boldsymbol{S}_\lambda$.
- λ affects the eigenvalues of $\boldsymbol{S}_\lambda$ negatively, except the first two which are always 1 corresponding to the two-dimensional eigenspace of functions linear in x. Other eigenvalues are between 0 and 1 depending on λ.
- The degrees of freedom of $\boldsymbol{S}_\lambda$ is $\text{df}_\lambda = \text{trace}\,(\boldsymbol{S}_\lambda) =$ sum of eigenvalues, which is always between 2 and K.
- When $\lambda \to 0$, all the eigenvalues are 1, $\boldsymbol{S}_\lambda \to I$. $\text{df}_\lambda = \text{trace}\,(\boldsymbol{S}_\lambda) =$ sum of eigenvalues $= K$, corresponding to any functions interpolating the K knots.
- When $\lambda \to \infty$, all the eigenvalues are 0 except the first two. $\text{df}_\lambda = \text{trace}\,(\boldsymbol{S}_\lambda) =$ sum of eigenvalues $= 2$, corresponding to a straight line.

5.1.3.2 Radial Basis Functions for a Fixed-Knot Thin Plate Spline

For a given λ and fixed knots $\kappa_i,\ i = 1, \ldots, K$, the fixed-knot thin plate spline can be written as

$$\hat{g}\,(x) = \hat{\gamma}_0 + \hat{\gamma}_1 x + \sum_{k=1}^{K} \hat{\delta}_k |x - \kappa_k|^3,$$

where $\hat{\theta} = \left(\hat{\gamma}_0, \hat{\gamma}_1, \hat{\delta}_1, \ldots, \hat{\delta}_K\right)$ minimizes the following penalized residual sum of squares,

$$\sum_{i=1}^{n}\left(y_i - \hat{\gamma}_0 - \hat{\gamma}_1 x_i - \sum_{k=1}^{K}\hat{\delta}_k |x_i - \kappa_k|^3\right)^2 + \lambda \sum_{l=1}^{K}\hat{\delta}_l \sum_{k=1}^{K}\hat{\delta}_k |\kappa_l - \kappa_k|^3,$$

subject to the constraints $\sum_{k=1}^{K}\hat{\delta}_k = \sum_{k=1}^{K}\hat{\delta}_k\kappa_k = 0$. The constraint makes the number of parameters K rather than $K+2$ which is consistent with the degrees of freedom of a natural cubic spline with K knots. For compact notation and programming, we can write the above equation in terms of matrices, as follows:

$$\mathrm{RSS} = \|\boldsymbol{y} - \boldsymbol{X}\hat{\gamma} - \boldsymbol{Z}\hat{\delta}\| + \lambda\hat{\delta}^T\boldsymbol{K}\hat{\delta}, \tag{5.4}$$

where $\boldsymbol{X}[i,] = (1, x_i)^T$, $\boldsymbol{Z}[i,k] = |x_i - \kappa_k|^3$, $\hat{\gamma} = \left(\hat{\gamma}_0, \hat{\gamma}_1\right)$, $\hat{\delta} = \left(\hat{\delta}_1, \ldots, \hat{\delta}_K\right)$ and $\boldsymbol{K}[l,k] = |\kappa_l - \kappa_k|^3$, $l = 1, \ldots, K$, $k = 1, \ldots, K$.

5.1.3.3 Linkage to a Mixed Effects Model

As already mentioned at the end of Sect. 5.1.2, the criterion of minimizing (5.4) is related to the criterion for calculating the best linear unbiased prediction (BLUP) in a mixed effects model. Suppose we have the following mixed effects model:

$$\begin{aligned} y_i &= \gamma_0 + \gamma_1 x_i + \sum_{k=1}^{K}\delta_k |x_i - \kappa_k|^3 + \varepsilon_i \\ \mathbb{E}(\delta_k) &= 0; \quad \mathrm{Var}(\delta) = \sigma_\delta^2\boldsymbol{K}^{-1} \\ \mathbb{E}(\varepsilon_i) &= 0; \quad \mathrm{Var}(\boldsymbol{\varepsilon}) = \sigma_\varepsilon^2\boldsymbol{I}. \end{aligned}$$

BLUP of γ and δ is defined as follows:

$$\tilde{\gamma}, \tilde{\delta} = \underset{\gamma', \delta'}{\mathrm{argmin}}\, \mathbb{E}\left\{\left(s^T\boldsymbol{X}\gamma' + t^T\boldsymbol{Z}\delta'\right) - \left(s^T\boldsymbol{X}\gamma + t^T\boldsymbol{Z}\delta\right)\right\}^2,$$

for any arbitrary s and t, and subject to the unbiasedness constraint

$$\mathbb{E}\left(s^T\boldsymbol{X}\gamma' + t^T\boldsymbol{Z}\delta'\right) = \mathbb{E}\left(s^T\boldsymbol{X}\gamma + t^T\boldsymbol{Z}\delta\right).$$

It can be shown that $\tilde{\gamma}$ and $\tilde{\delta}$ also minimize the following penalized RSS:

$$\left(\boldsymbol{y} - \boldsymbol{X}\gamma' - \boldsymbol{Z}\delta'\right)^T\left(\sigma_\varepsilon^2\boldsymbol{I}\right)^{-1}\left(\boldsymbol{y} - \boldsymbol{X}\gamma' - \boldsymbol{Z}\delta'\right) + \hat{\delta}^T\left(\sigma_\delta^2\boldsymbol{K}^{-1}\right)^{-1}\hat{\delta},$$

which is equivalent to minimizing (5.4) with $\lambda = \sigma_{\varepsilon}^2/\sigma_{\delta}^2$. $\tilde{\gamma}$ and $\tilde{\delta}$ have the following expression:

$$\begin{bmatrix} \tilde{\gamma} \\ \tilde{\delta} \end{bmatrix} = \left(\boldsymbol{C}^T\boldsymbol{C} + \lambda\boldsymbol{B}\right)^{-1}\boldsymbol{C}^T\boldsymbol{y},$$

where

$$\boldsymbol{C} = [\boldsymbol{X}, \boldsymbol{Z}]\,,\ \boldsymbol{B} = \begin{bmatrix} 0 & 0 \\ 0 & \boldsymbol{K} \end{bmatrix}.$$

The fitted values are $\hat{y} = \boldsymbol{C}\left(\boldsymbol{C}^T\boldsymbol{C} + \lambda\boldsymbol{B}\right)^{-1}\boldsymbol{C}^T\boldsymbol{y}$. Note that $\tilde{\gamma}$ and $\tilde{\delta}$ depend on the variance parameters σ_{δ}^2 and σ_{ε}^2, which can be estimated via maximum likelihood or restricted maximum likelihood (REML).

The connection of a fixed-knot thin plate spline with a mixed effects linear model makes it possible to analyze a smoothing spline in the framework of a Bayesian mixed effects linear model. Bayesian mixed effects linear models can quantify the estimation uncertainties of variance parameters which are ignored in the REML approach.

5.1.4 Bayesian Splines

Rather than using the equivalence of a smoothing spline to a mixed effects linear model, we can set up a mixed effects model structure directly on the basis expansion functions. The core idea of a smoothing spline is to shrink the parameters $\delta_i,\ i = 1, \ldots, n$ towards 0 in Eq. (5.3), where the shrinkage force and style are controlled by the smoothing parameter λ.

In the Bayesian framework, we can assume shrinkage priors, which perform the role of the smoothing parameter. Generally, we use the following Bayesian shrinkage spline model:

$$\begin{aligned} y_i &= \sum_{h=1}^{H} \beta_h b_h\left(x_i\right) + \varepsilon_i \\ \varepsilon_i &\sim \mathrm{N}\left(0, \sigma^2\right) \\ \beta_h &\sim G_h, h = 1, \ldots, H, \end{aligned}$$

where G_h is a *shrinkage prior* having high density at zero and heavy tails to avoid over-shrinking. G_h can be a t distribution with small degrees of freedom, or a double exponential distribution (Laplace distribution) which is related to the lasso method. The Laplace prior induces sparsity in the posterior mode, in that the posterior mode $\tilde{\beta}_h$ can be exactly equal to zero. The Laplace prior is the prior having heaviest tails which still produces a computationally convenient uni-modal posterior density.

An alternative is to use a *generalized double Pareto* prior distribution (Gelman et al. 2014), which resembles the double exponential near the origin while having arbitrarily heavy tails.

One can sample from a generalized double Pareto with scale parameter of ξ and shape parameter of α by instead drawing $\beta_h \sim \mathrm{N}\left(0, \sigma^2\tau_h\right)$, with $\tau_h \sim \mathrm{Exp}\left(\lambda_h^2/2\right)$ and $\lambda_h \sim \mathrm{Gamma}\,(\alpha, \alpha\xi)$. Placing the prior $p\,(\sigma) \propto 1/\sigma$, we then obtain a simple block Gibbs sampler having the following full conditional posterior distributions:

$$\begin{aligned}
\beta|\cdot &\sim \mathrm{N}\left(\left(\boldsymbol{X}^T\boldsymbol{X} + \boldsymbol{T}^{-1}\right)^{-1}\boldsymbol{X}^T\boldsymbol{y}, \sigma^2\left(\boldsymbol{X}^T\boldsymbol{X} + \boldsymbol{T}^{-1}\right)^{-1}\right)\\
\sigma^2|\cdot &\sim \text{Inv-gamma}\left(\frac{n+k}{2}, \frac{(\boldsymbol{y} - \boldsymbol{X}\beta)^T\,(\boldsymbol{y} - \boldsymbol{X}\beta)}{2} + \frac{\beta^T\boldsymbol{T}^{-1}\beta}{2}\right)\\
\lambda_h|\cdot &\sim \mathrm{Gamma}\left(\alpha + 1, \frac{|\beta_h|}{\sigma} + \eta\right), h = 1, \ldots, H\\
\tau_h^{-1}|\cdot &\sim \text{Inv-Gaussian}\left(\mu = \frac{\lambda_h\sigma}{\beta_h}, \rho = \lambda_h^2\right), h = 1, \ldots, H,
\end{aligned}$$

where

$$\boldsymbol{X} = \begin{bmatrix} b_1\,(x_1) & \cdots & b_H\,(x_1) \\ \cdots & \cdots & \cdots \\ b_1\,(x_n) & \cdots & b_H\,(x_n) \end{bmatrix}, \boldsymbol{T} = \mathrm{Diag}\,(\tau_1, \cdots, \tau_H)\,.$$

5.2 Two Simulated Examples

Now we turn to two simulated examples. It is always good to first check our methods using some simulated data to see whether these methods work before we go into the more complicated application. In these two examples, even though we know the underlying true mean function, estimating the coefficients in the mean function is not straightforward. We use smoothing splines, low rank smoothing splines and Bayesian shrinkage splines to estimate the mean function.

The first simulated example uses a trigonometric mean function with normal errors. It is an example used by Faraway (2015). Here we are more interested in prediction beyond the boundary. Besides the methods used by Faraway (2015), we also study this example in the Bayesian framework. The second simulated example assumes the response variable following a gamma distribution with a log-logistic curve mean function. We design the second example to mimic the claims payment process in general insurance. This prepares for application to real claims data in Sect. 5.3.

5.2.1 A Model with a Trigonometric Mean Function and Normal Errors

We generate the data from the following model:

$$
\begin{aligned}
y_i &= \sin^3(2\pi x_i^3) + \varepsilon_i, i = 1, \dots, 100 \\
x_i &\sim \mathrm{U}\,(0, 1) \\
\varepsilon_i &\sim \mathrm{N}\,(0, 0.01)\,.
\end{aligned}
$$

5.2.1.1 Polynomial Basis Expansion Regression Models

The R function `poly( )` generates an orthogonal polynomial basis matrix of specified degree at specified values. In Fig. 5.1, the first plot shows the raw polynomial basis of degree 4 at values from 0 to 1, where each line corresponds to a polynomial. The second plot shows the orthogonal polynomial basis of degree 4 at values from 0 to 1, where each line corresponds to a linear combination of polynomials, x, x^2, x^3, x^4. The third plot shows the orthogonal polynomial basis of degree 11 at values from 0 to 1.

We use the orthogonal polynomial basis of degrees 4, 7, and 11 to fit the simulated data. The fitted lines are shown in Fig. 5.2. Note that the degrees of freedom (df) shown in the legend box include the intercept term. None of the fitted lines can capture the shape of data adequately.

5.2.1.2 Spline Regression Models

The R function `bs( )` works similarly to `poly( )`. It generates the B-spline basis matrix of specified degree and knots. The number of rows of the B-spline basis matrix is equal to the number of values to be calculated. The number of columns of B-spline basis matrix is equal to the degrees of freedom of this spline. Here we use a cubic B-spline with 8 knots at (0.2, 0.4, 0.5, 0.6, 0.7, 0.8, 0.85, 0.9). So the degrees of freedom (or equivalently the number of columns) is 12, including the intercept term.

Using the R function `ns( )`, we generate a natural cubic B-spline with 8 interior knots at (0.2, 0.4, 0.5, 0.6, 0.7, 0.8, 0.85, 0.9) and the boundary knots at the ending points of x. A natural cubic B-spline has the property of orthogonality and linearity beyond the boundary knots, so the degrees of freedom of this natural cubic B-spline are 10 (i.e., 8 + 2 + 3 + 1 − 4), including the intercept term.

The comparison of a normal cubic B-spline basis with a natural cubic B-spline basis is shown in Fig. 5.3. There are 12 lines in the first plot corresponding to 12 columns of a cubic B-spline basis matrix. Except the marginal lines, all the lines are non-zero over the interval between 5 adjacent knots. There are 10 lines in the second plot, corresponding to 10 columns of the natural cubic B-spline basis matrix.

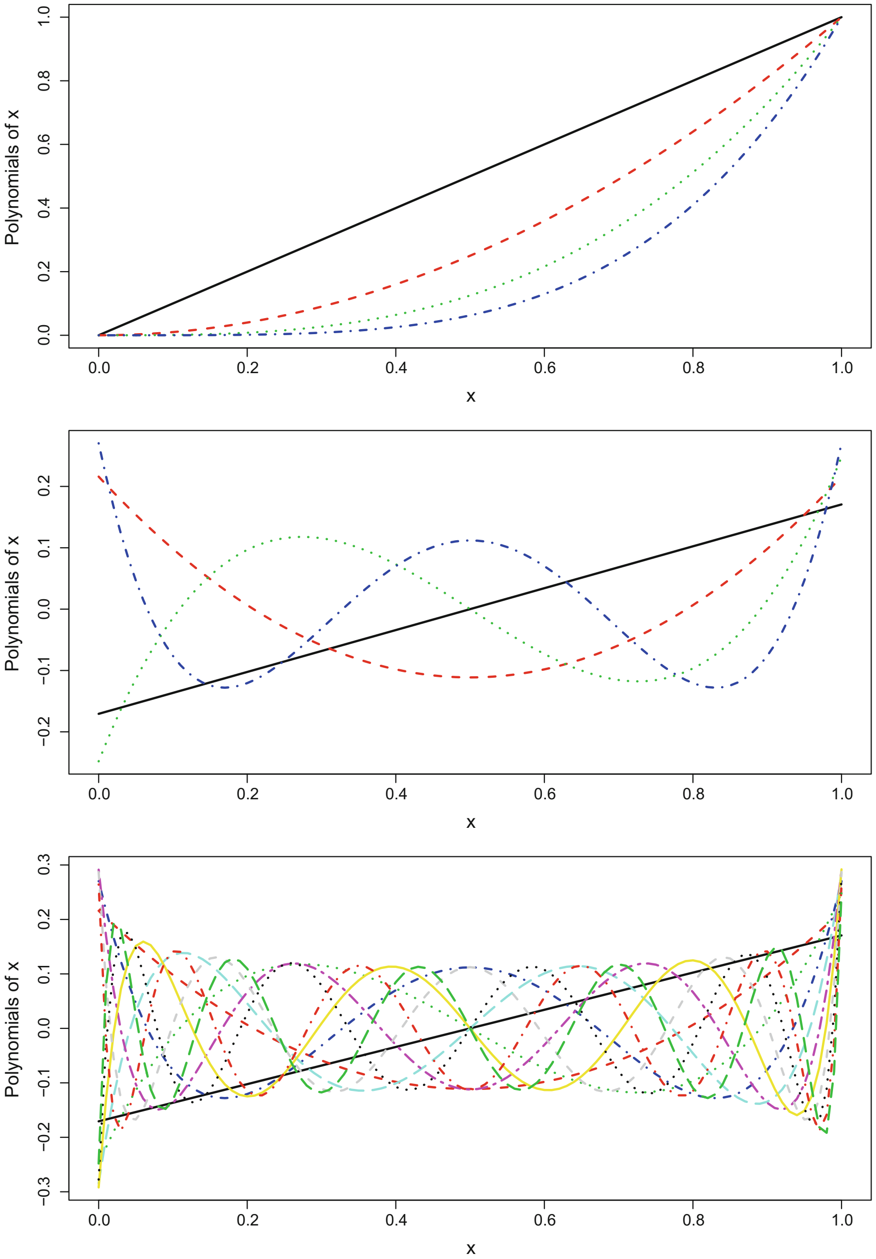


Fig. 5.1 Three polynomial basis functions in the interval [0, 1]: a raw polynomial basis of 4°, an orthogonal polynomial basis of 4° and an orthogonal polynomial basis of 11°

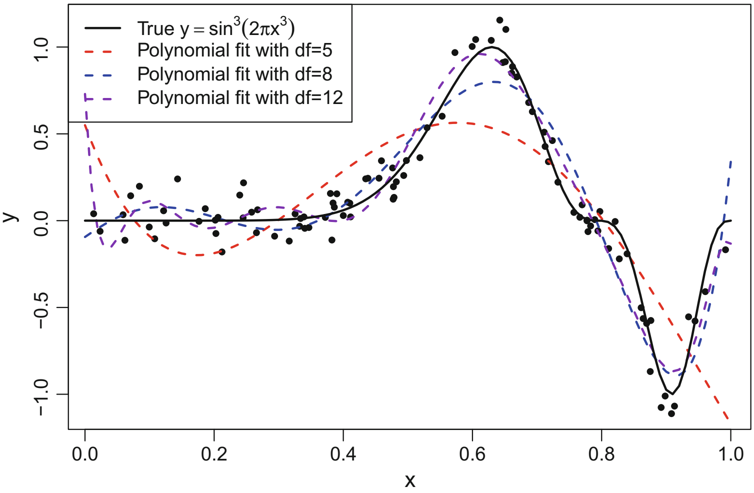


Fig. 5.2 The fitted lines of three polynomial models with df = 5, 8, 12

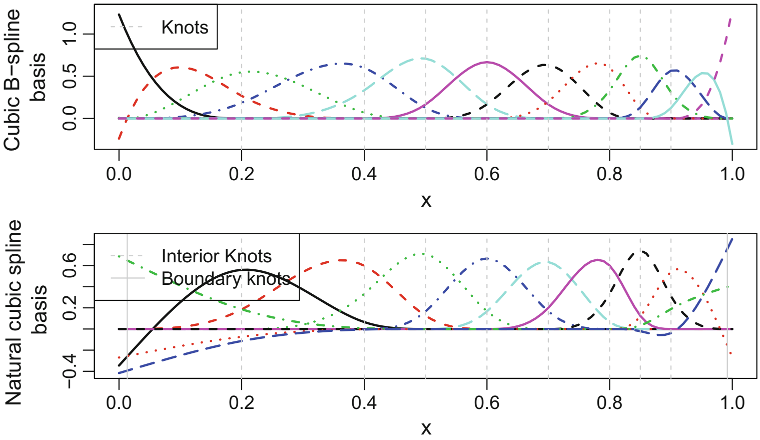


Fig. 5.3 A cubic B-spline basis and a natural cubic B-spline basis

Figure 5.4 shows the fitted lines from spline basis expansion regressions. The cubic spline with 12 degrees of freedom is less wiggly than the polynomial regression with the same degrees of freedom. This is mainly due to the local representation of spline basis functions. However, the cubic spline spreads weirdly outside the range of data, especially for $x > 1$.

The natural cubic spline regression with 10 degrees of freedom has similar performance to the cubic spline within the range of data. Moreover, it has a better extrapolation outside the range of data due to the linear constraints.

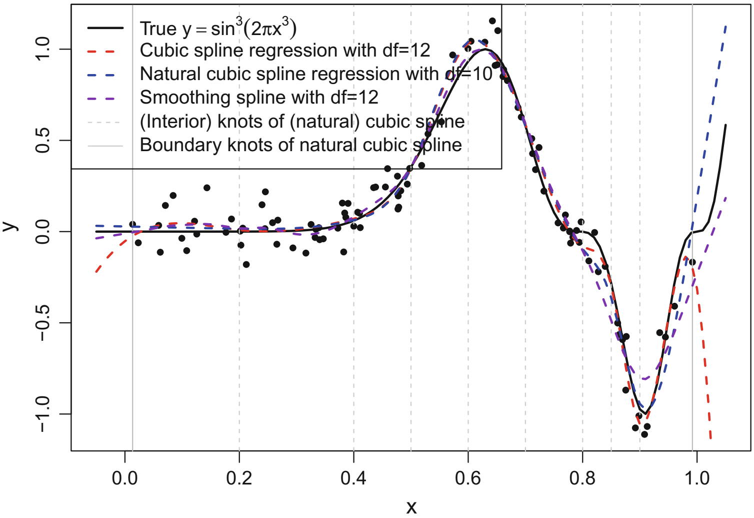


Fig. 5.4 The fitted lines of two spline regressions and the smoothing spline

5.2.1.3 A Full Rank Thin Plate Spline

The full rank smoothing spline is as good as the natural cubic spline since the smoothing spline also puts linear constraints beyond the range of data. However, the fitting process of a smoothing spline is quite different.

A smoothing spline uses the natural basis functions with knots at every unique x and shrink the coefficients by a penalty matrix based on (5.2), while the natural cubic spline regression does not shrink the coefficients but uses the least squares estimates.

5.2.1.4 Low Rank Thin Plate Splines

Rather than using the full rank basis matrix as in a smoothing spline, Wood (2003, 2006) uses the truncated eigen-decomposition of a full rank basis matrix to achieve a low rank smoother approximating the full rank smoother. Package `mgcv` can fit a low rank smoothing spline by smoothing function `s`. A disadvantage of this package is that we cannot specify the degrees of freedom or the location of knots. They are chosen automatically by generalized cross-validation criteria.

Another approach to solving the low rank smoothing spline is using a set of radial basis functions with specified knots as in Sect. 5.1.3. Due to the equivalence of a low rank thin plate spline to a mixed effects model, we can set up a low rank thin plate spline model as a Bayesian mixed effects model:

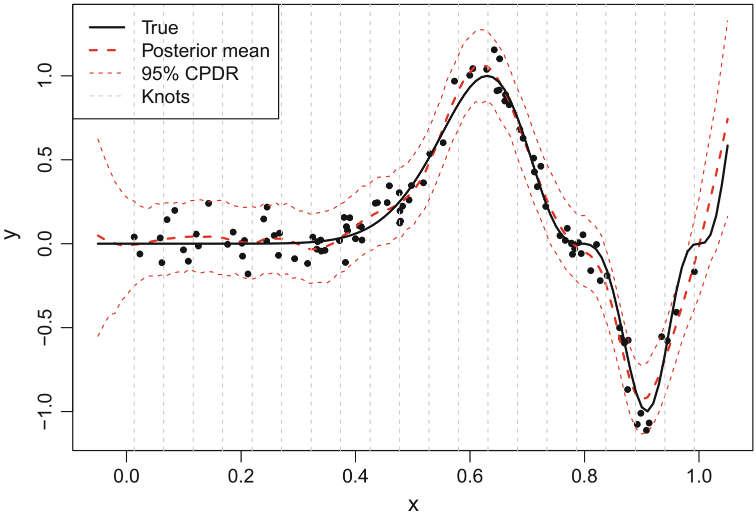


Fig. 5.5 A Bayesian mixed effects model using radial basis functions

$$\begin{aligned} \boldsymbol{y} &= \boldsymbol{X}\gamma + \boldsymbol{Z}\delta + \varepsilon \\ \delta &\sim \mathrm{N}\left(0, \sigma_\delta^2 \boldsymbol{K}^{-1}\right) \\ \varepsilon &\sim \mathrm{N}\left(0, \sigma_\varepsilon^2 \boldsymbol{I}\right), \end{aligned} \tag{5.5}$$

where $\boldsymbol{X}[i,] = (1, x_i)^T, \boldsymbol{Z}[i,k] = |x_i - \kappa_k|^3, \gamma = (\gamma_0, \gamma_1), \delta = (\delta_1, \ldots, \delta_K)$ and $\boldsymbol{K}[l,k] = |\kappa_l - \kappa_k|^3, l = 1, \ldots, K, k = 1, \ldots, K$. Here we specify a set of 20 equally located knots spreading the range of x. We give non-informative priors for $\gamma, \sigma_\delta^2, \sigma_\varepsilon^2$. The smoothing parameter $\lambda = \sigma_\varepsilon^2 / \sigma_\delta^2$ is not fixed, and we can get the posterior distribution of it.

We use Stan to simulate from the posterior distribution. It takes approximately 5 min to generate 1,600 iterations of which the first half are discarded as burn-in.

The posterior mean of λ is estimated as 0.000121 compared with 0.000102 from smoothing spline fit. We also plot the posterior predictive distribution for $x \in (-0.05, 1.05)$ in Fig. 5.5. The estimated number of effective parameters is $p_{\mathrm{D}} = 17.9$, $p_{\mathrm{WAIC}} = 16.5$, $p_{\mathrm{loo}} = 17.5$, which indicates around 16 degrees of freedom for the smoothing line (i.e., 18 minus two scale parameters, σ_δ^2 and σ_ε^2).

5.2.1.5 A Bayesian Spline Model

We apply the method in Sect. 5.1.4. A natural cubic spline basis is used with 20 equally located interior knots spreading the range of x and the boundary knots at the ending points of $\boldsymbol{x}$. We compare the goodness-of-fit of three shrinkage priors: generalized double Pareto (gdP) prior, Laplace prior (double exponential prior) and Cauchy prior. The Stan code when using gdP prior is as follows:

```
E1.code<-"
data{
  int H;        // Number of basis functions
  int N;        // Number of observations
  int n;        // Number of predicted values
  vector[N] y;  // Observations
  matrix[N,H] basis;    // Basis functions at observed x
  matrix[n,H] basis_hat;// Basis functions  at some fixed points
     x_hat
}
parameters{
  vector[H] b;  // Parameters of basis functions
  real<lower=0> sigmaE; // Standard deviation of observations y
  vector<lower=0>[H] tau;       // Hyperparameter in gdP prior
  vector<lower=0>[H] lambda;    // Hyperparameter in gdP prior
  }
transformed parameters{
  vector[N] means;
  means<-basis*b;
}
model{
  for (i in 1:H){
  b[i] ~ normal (0, sigmaE*tau[i]^0.5);
  tau[i] ~ exponential (lambda[i]^2/2);
  lambda[i] ~ gamma (1,1);
  }
  y ~ normal (means,sigmaE);
}
generated quantities{
  vector[n] means_hat;
  vector[N] log_lik;
  real D;
  means_hat<-basis_hat*b;
  for (i in 1:N)
    log_lik[i]<-normal_log(y[i],means[i],sigmaE);
  D<-sum(-2*log_lik);
}"
funky <- function(x) sin(2*pi*x^3)^3
set.seed(1); x <- sort(runif(100,0,1))
y <- funky(x) + 0.1*rnorm(length(x))
knots<-seq(min(x),max(x),length.out=20)[-c(1,20)]
basis<-ns(x,knots=knots,intercept = T) # using the default
    boundary knots of range of data
H<-ncol(basis); N<-nrow(basis)
x_hat<-seq(-0.05,1.05,0.01)
basis_hat<-ns(x_hat,knots=knots,intercept=T,Boundary.knots =
    range(x)) # make sure to use the same knots as design matrix
n<-length(x_hat)
E1.stanfit<-stan(model_code = E1.code, data=c("H","N","basis","y"
    ,"basis_hat","n"), iter=800,chains=4,seed=10)
```

The smoothness depends on the hyperparameters in the shrinkage priors, which can be specified as fixed constants or left to be estimated from the data. We list several information criteria in Table 5.1. Generally, all the shrinkage priors perform

Table 5.1 Comparison of Bayesian spline models using different shrinkage priors in the first simulated example. The computing time for 4 × 800 iterations is on a PC of 6G RAM with 2.8 GHz dual CPU. We assume the scale and shape parameters for gdP prior, and assume the mean and standard variance parameters for the Laplace prior and the Cauchy prior

Shrinkage prior		Computing time	p_D	p_{WAIC}	p_{loo}	DIC	WAIC	LOOIC
gdP	(1, 1)	13 s	17.5	15.5	16.5	−168.5	−168.1	−166.0
	(?, ?)	5 min	17.6	16.1	22.6	−171.7	−171.0	−158.0
Laplace	(0, 0.1)	1 s	16.6	15.6	16.5	−168.7	−167.1	−165.3
	(0, ?)	1 s	20.0	18.0	19.3	−167.9	−166.7	−164.2
Cauchy	(0, 0.1)	1 s	17.6	16.0	16.9	−169.2	−168.2	−166.5
	(0, ?)	1 s	19.1	17.3	18.6	−168.3	−167.2	−164.5
Model (5.5)		5 min	17.9	16.5	17.5	−165.4	−164.1	−162.0

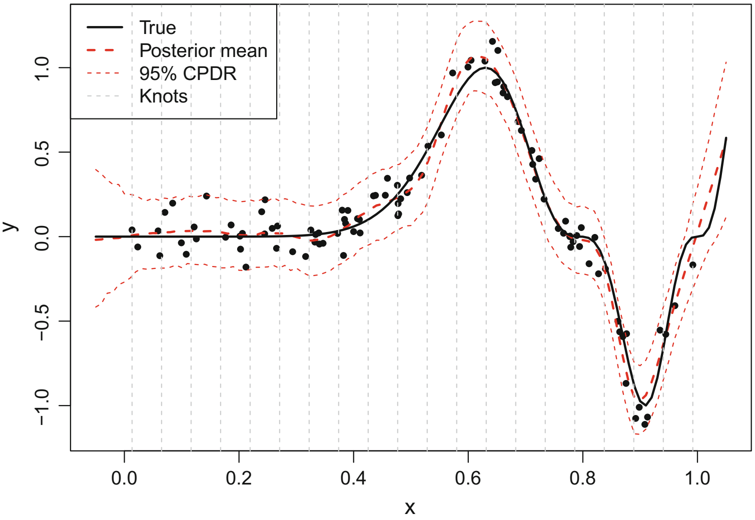


Fig. 5.6 A Bayesian natural cubic spline model using Cauchy (0, 0.01) prior

equally well. The fitted line is not sensitive to shrinkage priors. Hence we only give the posterior mean of the fitted line with the 95% CPDR under Cauchy (0, 0.1) shrinkage prior in Fig. 5.6. Note that the information criteria can be calculated using the following code:

```
E1.sim<-extract(E1.stanfit,permuted=T)
# loo and WAIC
loo(extract_log_lik(E1.stanfit,"log_lik"))
waic(extract_log_lik(E1.stanfit,"log_lik"))
# pD and DIC
Dbar<-mean(E1.sim$D)
Dhat<-0
for (i in 1:N){
  Dhat<-Dhat-2*dnorm(y[i], basis[i,]%*%apply(E1.sim$b,2,mean),
       mean(E1.sim$sigmaE),log=T);
}
list(Dhat=Dhat,Dbar=Dbar,pD=Dbar-Dhat,DIC=2*Dbar-Dhat)
```

5.2.2 A Gamma Response Variable with a Log-Logistic Growth Curve Mean Function

We assume the cumulative claims following a log-logistic growth curve, and generate the incremental claims from a gamma distribution. More specifically, we use the following model to generate the incremental claims:

$$
\begin{aligned}
y_{ij} &\sim \text{Gamma}\left(100, \frac{100}{\mu_{ij}}\right), i = 1, \dots, 30, j = 1, \dots, 40 \\
\mu_{ij} &= P_i \times LR_i \times \left(G\left(j; \theta_i, \omega_i\right) - G\left(j-1; \theta_i, \omega_i\right)\right) \\
P_i &= (1.00 + i \times 0.01) \times 10^6 \\
LR_i &\sim \text{N}\left(0.8, 0.1^2\right) \\
\theta_i &\sim \text{N}\left(7.5, 0.05^2\right) \\
\omega_i &\sim \text{N}\left(2.5, 0.03^2\right) \\
G\left(l; \theta, \omega\right) &= \frac{l^\omega}{l^\omega + \theta^\omega}, l = 0, \dots, 40,
\end{aligned}
$$

where P_i is the earned premium of accident year i, LR_i is the loss ratio of accident year i and G is a log-logistic function. Note that the earned premiums are always available and are used as the offset later. We choose the shape parameter of the gamma distribution to be 100, implying the coefficient of variation of y_{ij} as 0.1.

We define the cumulative claims at the end of development year j for the accident year i as $c_{ij} = \sum_{l=1}^{j} y_{il}$. We assume that there is no development after 40 years since $G\,(40; 7.5, 2.5) = 0.985$.

Suppose the evaluation time of outstanding liability is at the end of first development year of accident year 30. We have the triangle data set $\boldsymbol{y} = \left\{y_{i,j} : i + j \leq 31, i = 1, \dots, 30\right\}$ available. The task is to predict the future claims up to the development year 40, $y' = \left\{y_{i,j} : i + j > 31, i = 1, \dots, 30, j \leq 40\right\}$. The simulated data is plotted in Fig. 5.7.

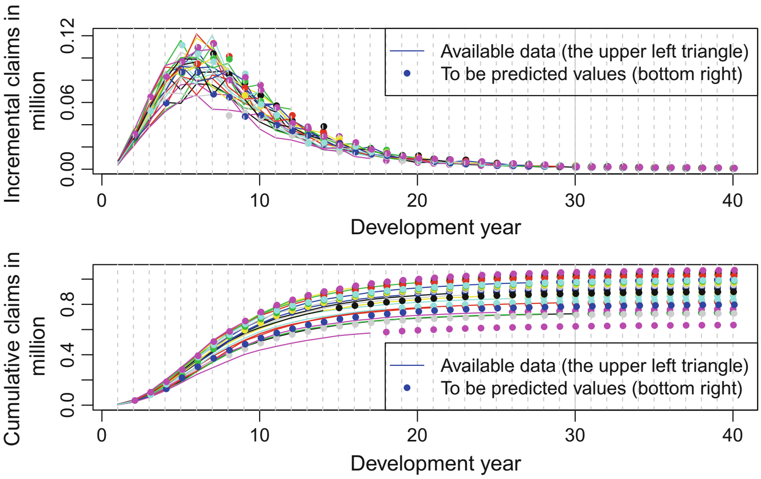


Fig. 5.7 Simulated incremental and cumulative claims

In the following, we fit four models: a polynomial basis expansion regression model, a natural cubic spline regression model, a low rank smoothing spline model, and a Bayesian shrinkage natural cubic spline model. All the models have the following common structure:

$$\begin{aligned} y_{ij} &\sim \text{Gamma}\left(\alpha, \frac{\alpha}{\mu_{ij}}\right), i = 1, \ldots, 30, j = 1, \ldots, 40 \\ \mu_{ij} &= P_i \times LR_i \times \exp\left(\sum_{h=1}^{H} \beta_h b_h(j)\right) \\ P_i &= (1.00 + i \times 0.01) \times 10^6. \end{aligned} \tag{5.6}$$

5.2.2.1 A Polynomial Basis Expansion Regression Model

We fit a GLM with a gamma family and a logarithm link function. The offset term is $\log P_i$. The number of parameters is $31 + H'$, where H' is the degrees of freedom of polynomial basis without intercept. H' is chosen according to AIC. Figure 5.8 shows that $H' = 10$ is optimal.

For this model, we make the prediction of the lower triangle and tail development during development years 31–40. The predicted values are shown as lines and simulated data of the same accident year are shown as dots in the same colour. We separate the prediction of the lower triangle from the prediction of the tail development in Fig. 5.9.

As in the first simulated example, the polynomial basis expansion model cannot make good prediction beyond the range of data.

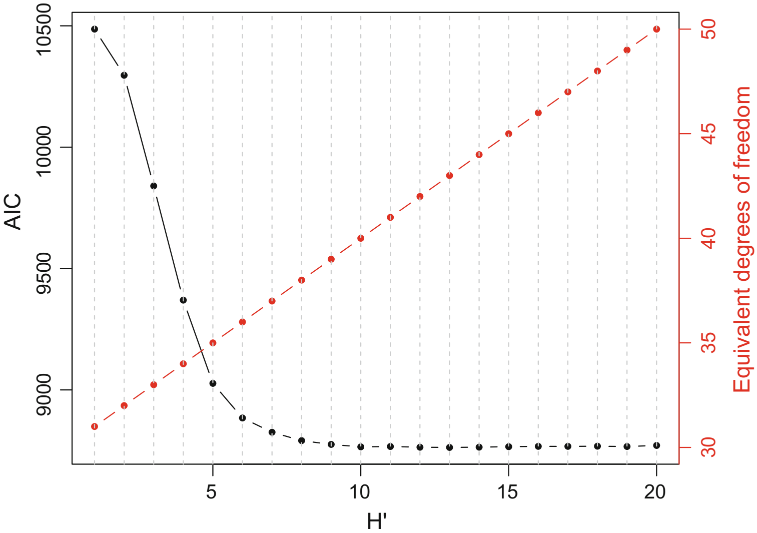


Fig. 5.8 AIC versus H' of polynomial basis expansion models

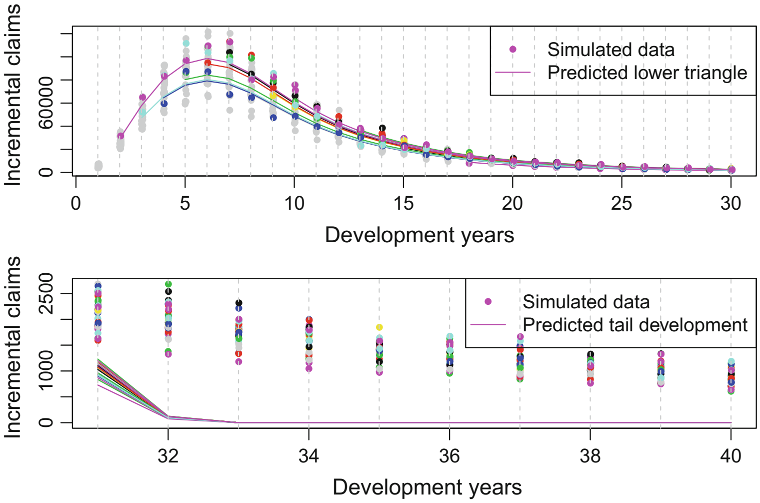


Fig. 5.9 Prediction of future claims from a polynomial basis expansion model

5.2.2.2 A Natural Cubic Spline Regression Model

We choose 5 interior knots at 2, 3, 5, 10, 20 and 2 boundary knots at 1 and 30. This induces a 7 degrees of freedom smoothing development curve. The prediction of future claims is shown in Fig. 5.10. The model can predict the tail development more accurately compared with the polynomial basis expansion model.

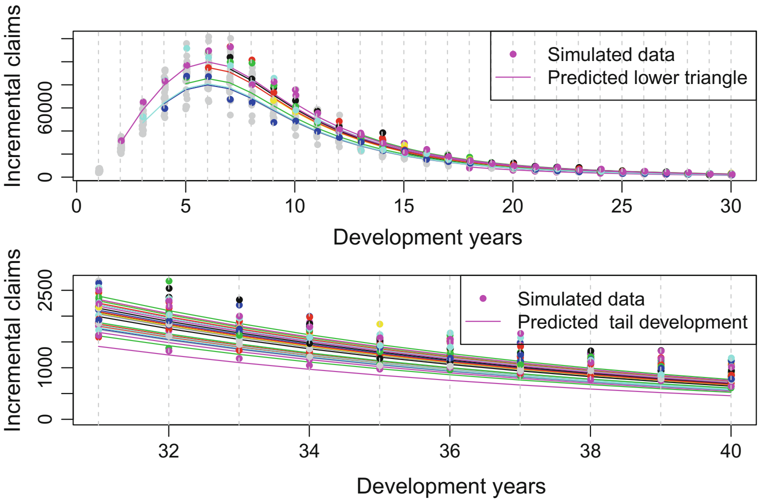


Fig. 5.10 Prediction of future claims from a natural cubic spline regression model

5.2.2.3 A Low Rank Thin Plate Spline

We rely on the `mgcv` package by Wood (2006) to fit a generalized additive model (GAM) with a low rank smoothing spline for the development year covariate. The degrees of freedom of the smoothing spline cannot be specified but are chosen using the criterion of generalized cross-validation. The rank reduction is achieved by a truncated eigen-decomposition rather than the choice of knots.

The predicted lower triangle is quite close to those predicted by the previous two models. Here we compare the tail development predictions made by the three models: the polynomial basis expansion model, the natural cubic spline basis expansion model and the low rank smoothing spline model. As shown in Fig. 5.11, the natural cubic spline regression model can best capture the tail development. Next we will set up a natural cubic spline basis expansion model in the Bayesian framework.

5.2.2.4 A Bayesian Natural Cubic Spline

In the previous simulated example, we saw that a Bayesian mixed effects model is more computationally expensive but no better fit than a Bayesian shrinkage spline model (see Table 5.1). Here we consider only a Bayesian full rank natural cubic spline model with shrinkage priors. An alternative is to use a fixed-knot natural cubic spline model which leads to similar prediction given the knots are chosen properly.

The Bayesian shrinkage spline model we will focus on is as follows:

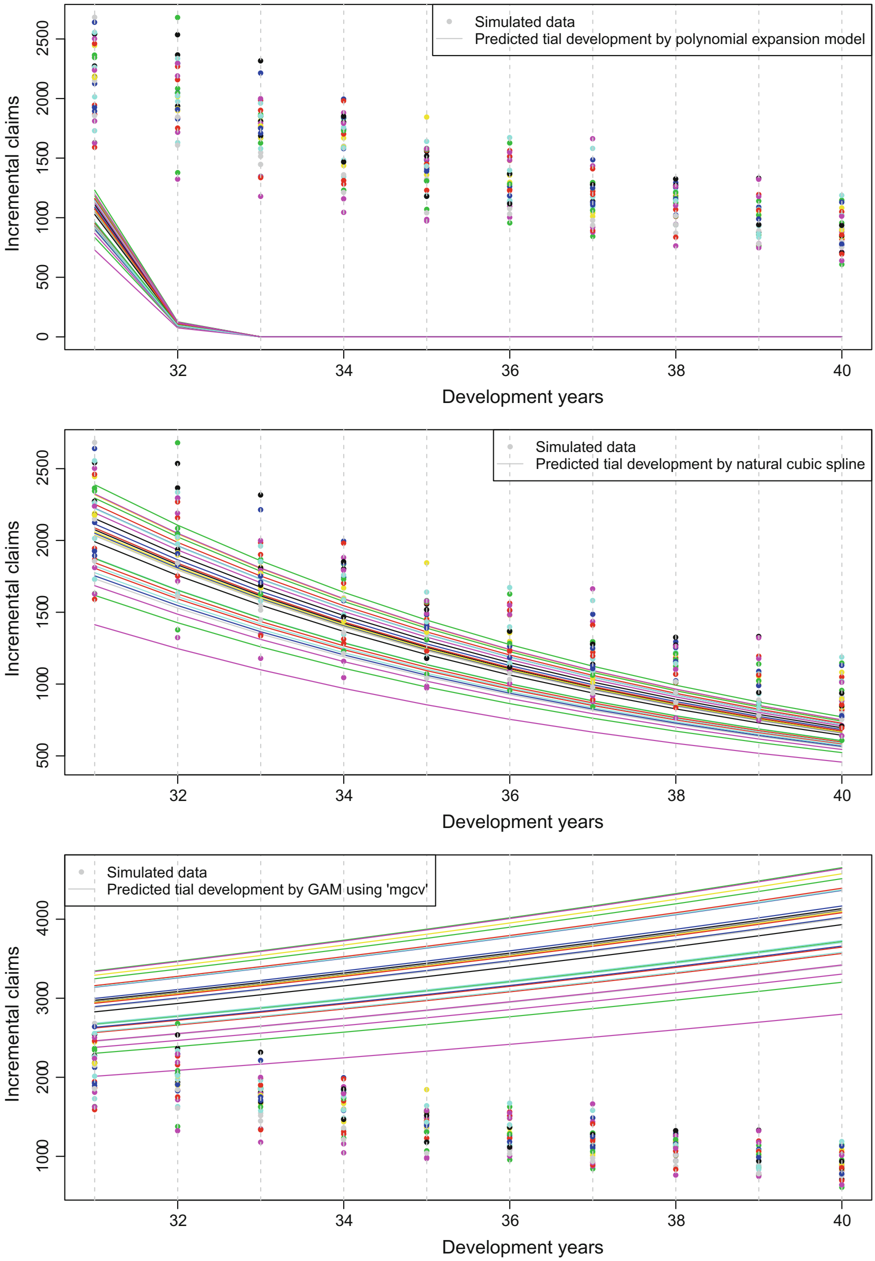


Fig. 5.11 Comparison of tail development predictions by three models: a polynomial regression, a natural cubic spline regression and a GAM

$$
\begin{aligned}
y_{ij} &\sim \text{Gamma}\left(\alpha, \frac{\alpha}{\mu_{ij}}\right), i = 1, \ldots, 30, j = 1, \ldots, 40 \\
\mu_{ij} &= P_i \times LR_i \times \exp\left(\sum_{h=1}^{30} \beta_h b_h(j)\right) \\
\beta_h &\sim \text{DoubleExp}(0, 1), h = 1, \ldots, 30 \\
P_i &= (1.00 + i \times 0.01) \times 10^6,
\end{aligned}
\tag{5.7}
$$

where $\{b_h : h = 1, \ldots, 30\}$ is a set of natural cubic spline basis functions with interior knots placed at $2, \ldots, 29$ and boundary knots placed at 1 and 30.

Denote the natural cubic spline basis matrix (40×30) by $\boldsymbol{B}$. Hence

$$
\sum_{h=1}^{30} \beta_h b_h(j) = (\boldsymbol{B}\beta)[, j],
$$

where $\beta = (\beta_1, \ldots, \beta_{30})$. We use the double exponential (Laplace) shrinkage priors with mean zero and variance 1. We assume non-informative priors for LR_i, α.

Model inference

We use Stan to estimate parameters and predict future claims. The code is as follows:

```
E2.code<-"
data{
  int   N;              // Number of obs.
  int   H;              // Number of basis functions
  int   n;              // Number of future values
  int   K;              // Number of accident year
  int   M;              // Number of develop year
  vector[N]     inc;    // Incremental claims in upper triangle
  matrix[M,H]   dev_basis;      // Basis expansion matrix
  int   acc[N];         // Accident years in upper triangle
  int   dev[N];         // Development years in upper triangle
  vector[N]    pre;     // Premiums in upper triangle
  int   acc_p[n];       //Accident years in lower triangle
  int   dev_p[n];       //Development years in lower triangle
  vector[n]      pre_p; //Premiums in lower triangle
}
parameters{
  vector[H]                     b;
  vector<lower=0.6,upper=1>[K]  ratio;
  real<lower=0>                 alpha;
}
transformed parameters{
  vector[N] means;
  vector[M] dev_raw;
  dev_raw<-exp(dev_basis * b);
  for (i in 1:N){
    means[i]<-pre[i]*ratio[acc[i]]*dev_raw[dev[i]];
  }
}
```

```
model{
  b ~ double_exponential(0,1);
  for (i in 1:N){
    inc[i] ~ gamma(alpha, alpha/means[i]);
  }
}
generated quantities{
  vector[n] means_p;
  vector[N] residuals;
  vector[N] log_lik;
  real      D;
  for (i in 1:n){
    means_p[i]<-pre_p[i]*ratio[acc_p[i]]*dev_raw[dev_p[i]];
  }
  for (i in 1:N){
    residuals[i]<-(inc[i]-means[i])/means[i]*sqrt(alpha);
    log_lik[i]<-gamma_log(inc[i],alpha,alpha/means[i]);
  }
  D<-sum(-2*log_lik);
}
"
knots<-c(2:29)
dev_basis<-ns(1:40,knots=knots,Boundary.knots = c(1,30),intercept
     =T)
E2.stanfit<-stan(model_code = E2.code,data=c("N","n","H","K","M",
     "inc","acc","dev","dev_basis","pre","acc_p","dev_p","pre_p"),
     iter=400,chains=4,seed=1)
```

It takes 40 s for 1600 iterations. After checking convergence, we plot the posterior mean of Pearson residuals versus the posterior mean of fitted values in Fig. 5.12. Not surprisingly, it shows a randomly spread, since the gamma distribution assumption is the same as the underlying error structure generating the data.

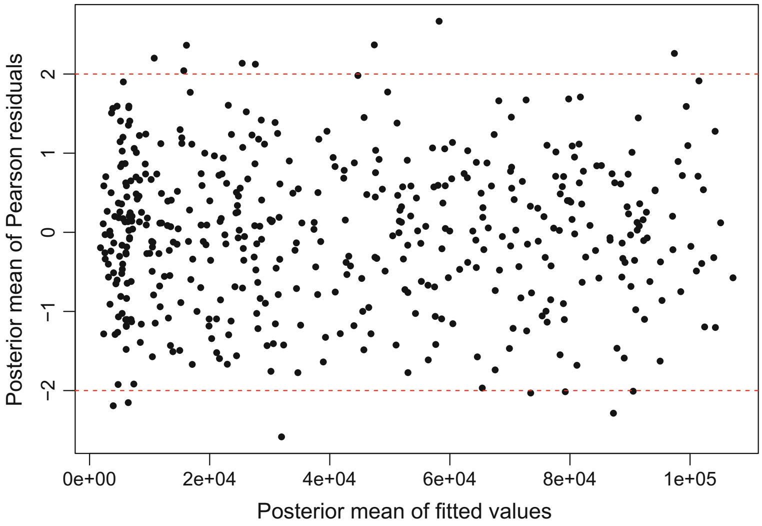


Fig. 5.12 The residual plot of a Bayesian natural cubic spline model

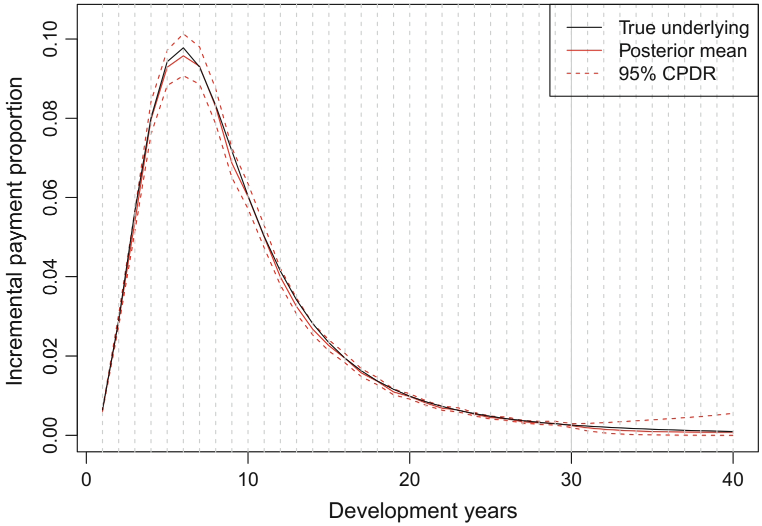


Fig. 5.13 Proportions of the incremental claims to the ultimate claims

The posterior mean and the 95% CPDR of the proportion of the incremental claims to the ultimate claims (i.e., the term $\exp\left(\sum_{h=1}^{30}\beta_h b_h\left(j\right)\right)$ in Eq. (5.7)) is shown in Fig. 5.13. The posterior mean is close to the true underlying log-logistic curve. The 95% CPDR covers most of the true underlying curve. As we expected, the 95% CPDR spreads out after development year 30, since there are no data after development year 30.

We plot the posterior distribution of cumulative claims up to development year 40 for 9 accident years shown in Fig. 5.14. The ultimate claims distributions are positively skewed due to the assumption of gamma likelihood. The posterior distribution of total outstanding unpaid claims liability is plotted in Fig. 5.15. We also plot the result using a Cauchy shrinkage prior for comparison in Fig. 5.15. Both models lead to similar posterior distributions that are positively skewed.

Advantages of using a Bayesian model

An important advantage of Bayesian modelling is the ability to evaluate the uncertainty via simulation from the posterior distribution. Frequentist models typically use the asymptotic property of parameters under resampling to estimate the uncertainty associated with parameters and future values. This method becomes problematic for some complicated functions of direct predictions.

For the claims reserving problem, the response variable is the incremental claims, but our interest is the cumulative claims whose uncertainty is difficult to estimate. The bootstrap method can tackle this task through resampling residuals and generating the pseudo-data. In the Bayesian framework, we use MCMC or HMC to simulate the joint posterior distribution of parameters and perform a further step to generate the future claims. Essentially, the distribution of any functions of response variable can be simulated through this process.

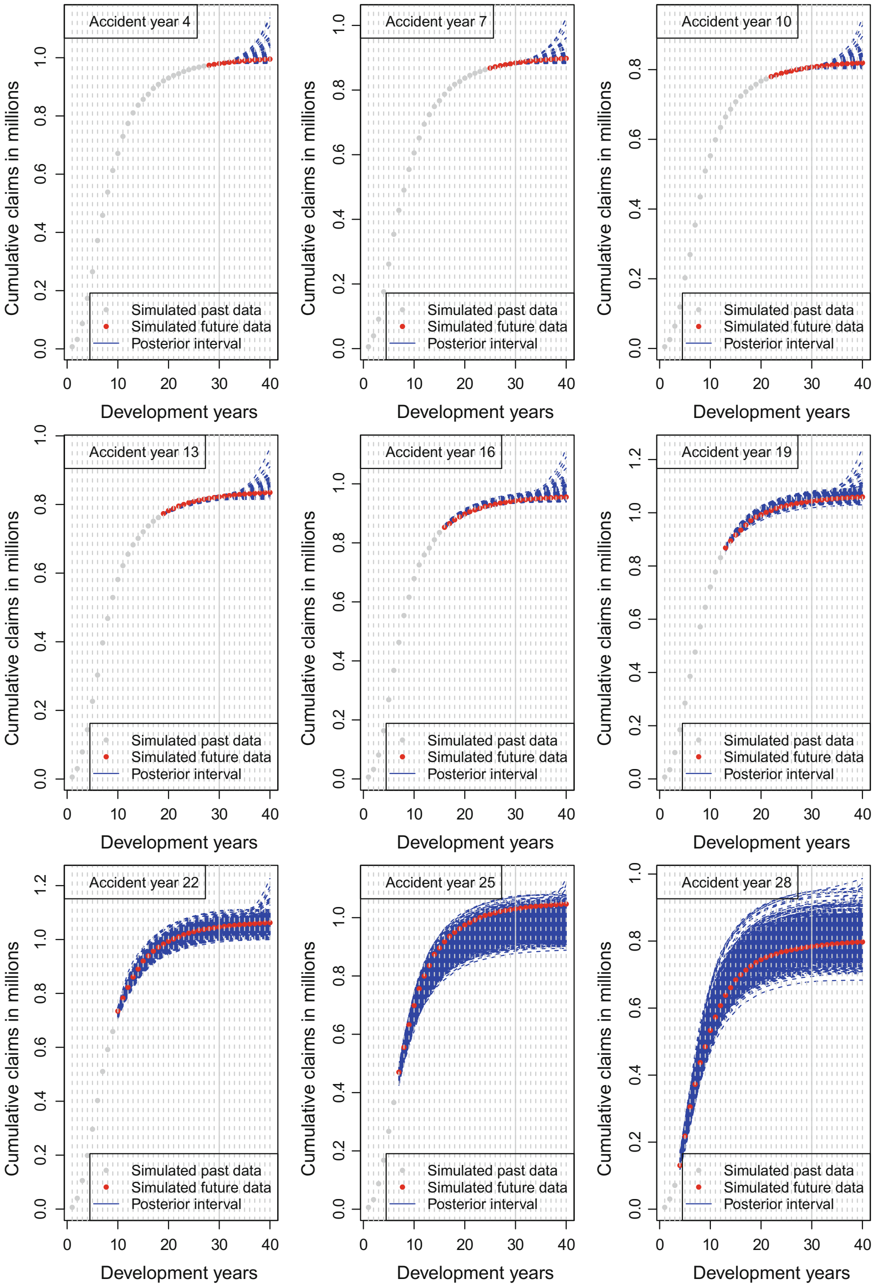


Fig. 5.14 The predictive distributions of cumulative claims for 9 accident years

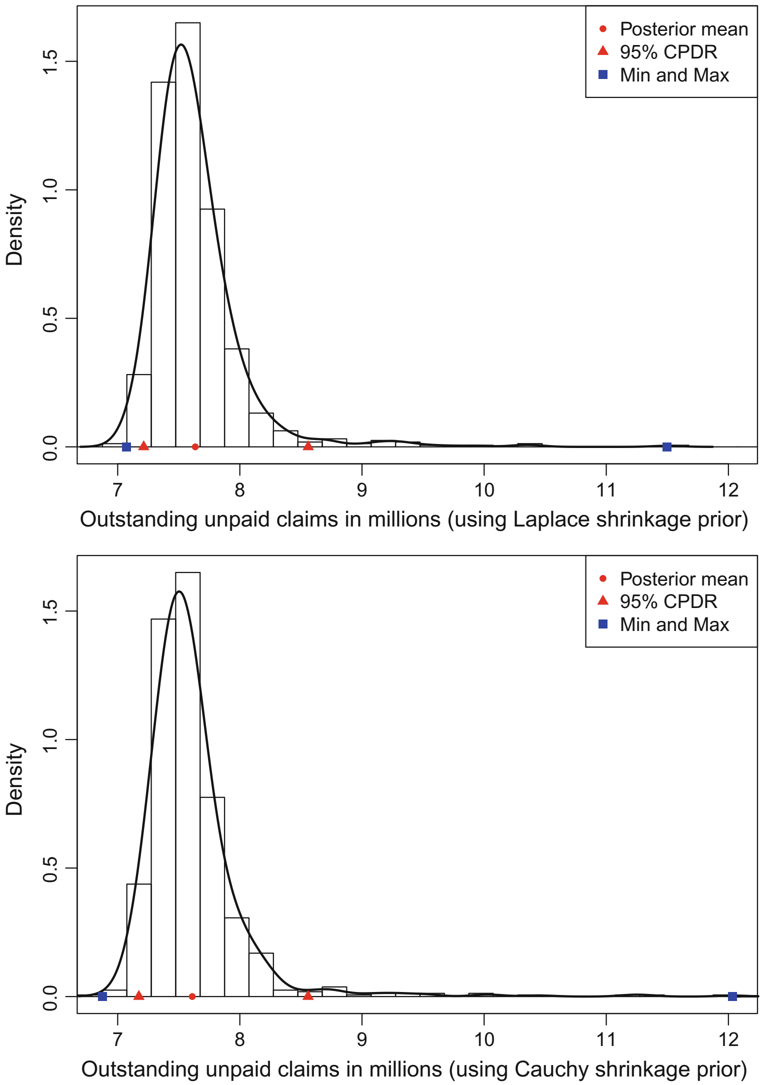


Fig. 5.15 The predictive distribution of the total outstanding liability using different shrinkage priors

Table 5.2 Comparison of Bayesian spline models using different shrinkage priors in the second simulated example. The computing time for 4 × 800 iterations is on a PC of 6G RAM with 2.8 GHz dual CPU. We assume the scale and shape parameters for gdP prior, and assume the mean and standard variance parameters for the Laplace prior and the Cauchy prior

Shrinkage prior		Computing time (s)	p_D	p_{WAIC}	p_{loo}	DIC	WAIC	LOOIC
Laplace	(0, 1)	35	57.1	51.6	52.8	8783.8	8784.4	8786.8
	(0, ?)	35	55.7	49.8	51.1	8780.5	8780.3	8782.9
Cauchy	(0, 1)	34	58.2	51.9	53.4	8786.0	8785.9	8789.0
	(0, ?)	34	57.3	51.2	52.7	8784.1	8784.1	8787.1

Model selection

Finally, we compare four models in terms of the three information criteria discussed in Sect. 2.2. As shown in Table 5.2, these four models have similar goodness-of-fit values. The differences are mainly due to the randomness.

5.3 Application to the Doctor Benefit

In the previous chapter, the analysis of doctor benefit did not accommodate the tail development. While all the claims seem to be reported by the development year 27, the benefit payments seem to continue beyond the development year 27. So we need to consider the tail development of PPCI.

A basis expansion model is applied to extrapolate the tail development. The natural cubic spline is at the top of our option list, since it comes from an optimal problem and has the linear constraint beyond the boundary knots.

As in the previous chapter, we have three steps to fit a compound model. The first step is to fit a Bayesian natural cubic spline model to the claims numbers. The posterior mean of ultimate claims number is used to calculate the PPCI triangle. Next, we fit a Bayesian natural cubic spline model to the PPCI triangle to get the posterior distribution of outstanding PPCI. The payments are assumed to continue up to the development year 30. Finally, we apply a compound model to combine the ultimate claims numbers with the outstanding PPCI to get the claims liability.

5.3.1 Claims Numbers

A Bayesian natural cubic spline model with Cauchy shrinkage priors and a gamma distribution is fitted to the claims numbers triangle. The boundary knots are placed at the first and last available development years, i.e., the development years 1 and 27. The development years 2–26 are interior knots. The basis matrix for prediction must use the same knots. The Stan code is as follows:

```
number.code<-"
data{
  int N;                     // Number of observations
  int n;                     // Number of future values
  int K;                     // Number of accident years
  int M;                     // Number of develop years (27)
  int H;                     // Number of basis functions
  vector[N]     first_inc;  // Number of claims in upper triangle
  matrix[M,H]   dev_basis;  // The basis functions
  int           acc[N];     // Accident years in upper triangle
  int           dev[N];     // Development years in lower triangle
  int           acc_p[n];   // Accident years in lower triangle
  int           dev_p[n];   // Development years in lower triangle
}
parameters{
  vector[H]                                     b;
  vector<lower=0,upper=55000>[K]                ult;
  real<lower=0>                                 alpha;
  real<lower=0>                                 sigma;
}
transformed parameters{
  vector[N] means;
  vector[M] dev_raw;
  vector[M] dev_norm;
  dev_raw<-exp(dev_basis * b);
  dev_norm<-dev_raw/sum(dev_raw);
  for (i in 1:N){
    means[i]<-ult[acc[i]]*dev_norm[dev[i]];
  }
}
model{
  b ~ cauchy(0,sigma);
  for (i in 1:N){
    first_inc[i] ~ gamma(alpha, alpha/means[i]);
  }
  for (i in 25:27)
    ult[i] ~ normal(20000,2000);
}
generated quantities{
  vector[n] means_p;
  vector[N] residuals;
  vector[N] log_lik;
  real      D;
  for (i in 1:n){
    means_p[i]<-ult[acc_p[i]]*dev_norm[dev_p[i]];
  }
  for (i in 1:N){
    residuals[i]<-(first_inc[i]-means[i])/means[i]*sqrt(alpha);
    log_lik[i]<-gamma_log(first_inc[i],alpha,alpha/means[i]);
  }
  D<-sum(-2*log_lik);
}"
M<-27; knots<-c(2:26)
dev_basis<-ns(c(1:M),knots=knots,Boundary.knots = c(1,27),
      intercept = T)
number.stanfit<-stan(model_code = number.code, data=c("first_inc"
,"N","n","K","M","H","acc","dev","acc_p","dev_p","dev_basis"),
     iter=800,chains=4,seed=1)
```

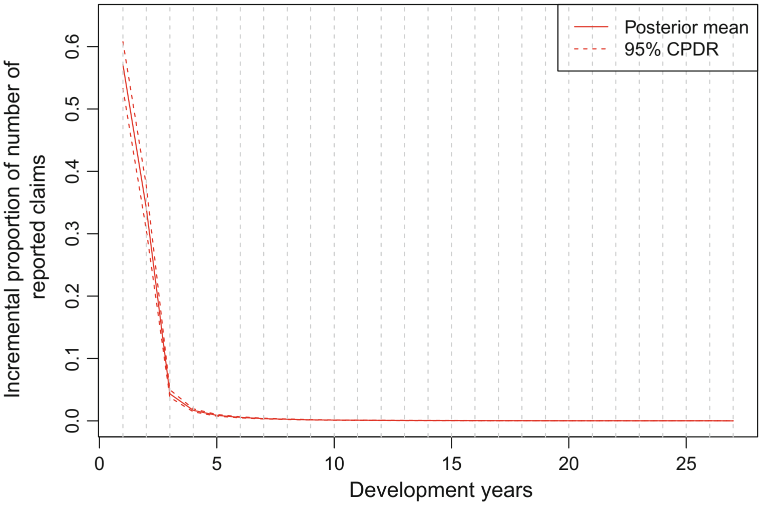


Fig. 5.16 Proportions of incremental claims numbers to ultimate claims numbers

The residual plot shows a quite similar pattern to Fig. 4.17 so we did not present it here. The posterior mean and the 95% CPDR for the proportion of incremental reported claims to the ultimate claims numbers are plotted in Fig. 5.16. It shows that nearly all the claims are reported by the development year 3, hence the assumption of no tail development after development year 27 is reasonable.

We plot the posterior distributions of cumulative claims numbers for the accident years 8, 10, 12, 14, 16, 18, 20, 22 and 24 in Fig. 5.17. It shows that the ultimate claims numbers for the older accident years can be estimated more accurately. For the recent accident years, the large uncertainties in the first three development years are carried forward to the ultimate claims numbers. We use the posterior mean of the ultimate claims number as a proxy to derive the PPCI triangle.

5.3.2 PPCI

Similar to the claims numbers, we fit a natural cubic spline model with Cauchy shrinkage priors to the PPCI triangle. The choice of knots is the same as for the claims numbers, and we assume the payments are finalized by the development year 30. The Stan code is similar to `number.code`, but with the following changes:

```
M<-30; knots<-c(2:26)
dev_basis<-ns(c(1:M),knots=knots,Boundary.knots = c(1,27),
       intercept = T)
```

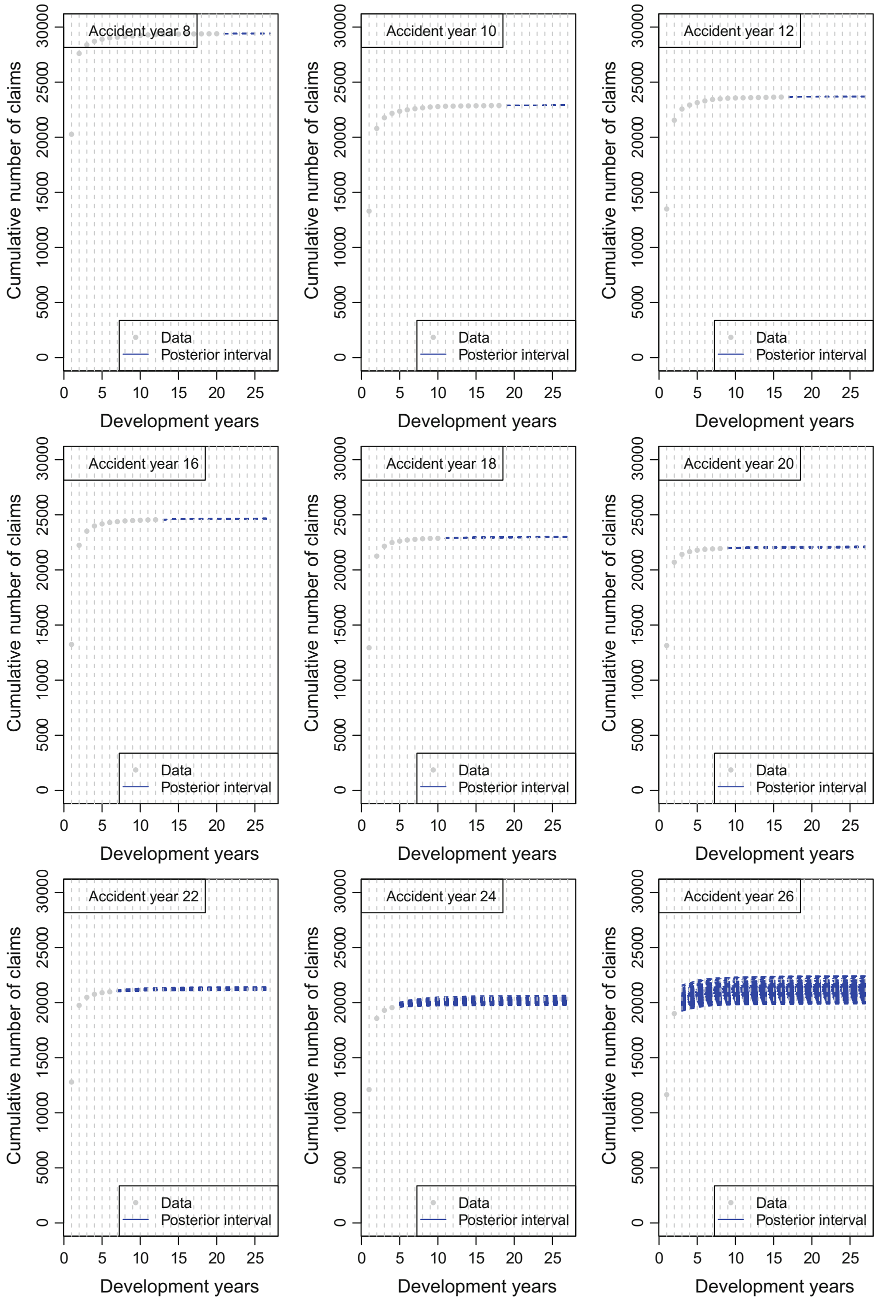


Fig. 5.17 The predictive distributions of cumulative claims numbers for 9 accident years

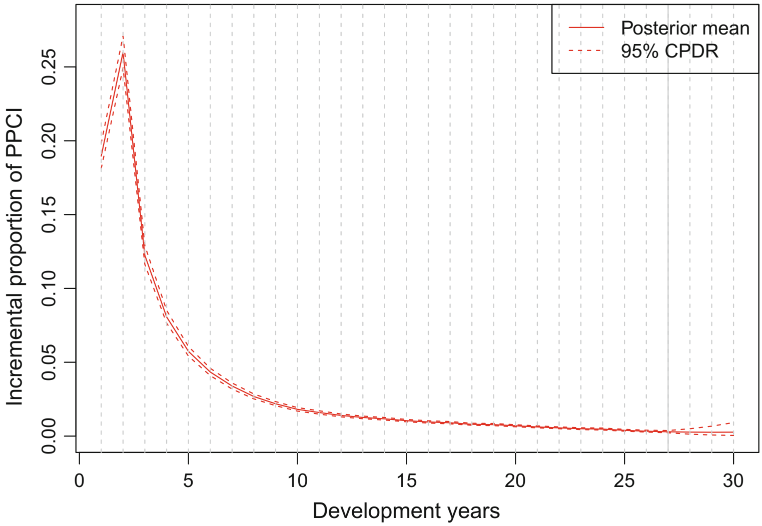


Fig. 5.18 Proportions of the incremental PPCI's to the ultimate PPCI's

The posterior inference of the proportion of incremental PPCI to the ultimate PPCI is shown in Fig. 5.18. The 95% CPDR spreads out in the tail area due to the lack of data. The development of PPCI for accident years 8, 10, 12, 14, 16, 18, 20, 22 and 24 is plotted in Fig. 5.19. As expected, less developed accident years show more variation.

Here we saw the advantage of the basis expansion model compared with model (4.7). Model (4.7) separates the development curve into two parts: the first few development years, characterized by a factor covariate, and the last mature development years, characterized by an exponential curve. The RJMCMC method is used to simulate from the posterior distribution, which is a joint distribution of the model index and parameters. By using a basis expansion model, only one model is focused and non-significant coefficients are shrunk to zero.

5.3.3 Combining the Ultimate Claims Numbers with the Outstanding PPCI

A compound model discussed in the previous chapter (see Sect. 4.4) is applied to calculate the posterior distribution of total outstanding claims liability as shown in Fig. 5.20. Table 5.3 lists the predictions made from the compound model. The posterior mean of total outstanding liability is 419,770,032 dollars (7% higher than in the previous chapter) with standard variance of 10,492,327 dollars. The 95% CPDR

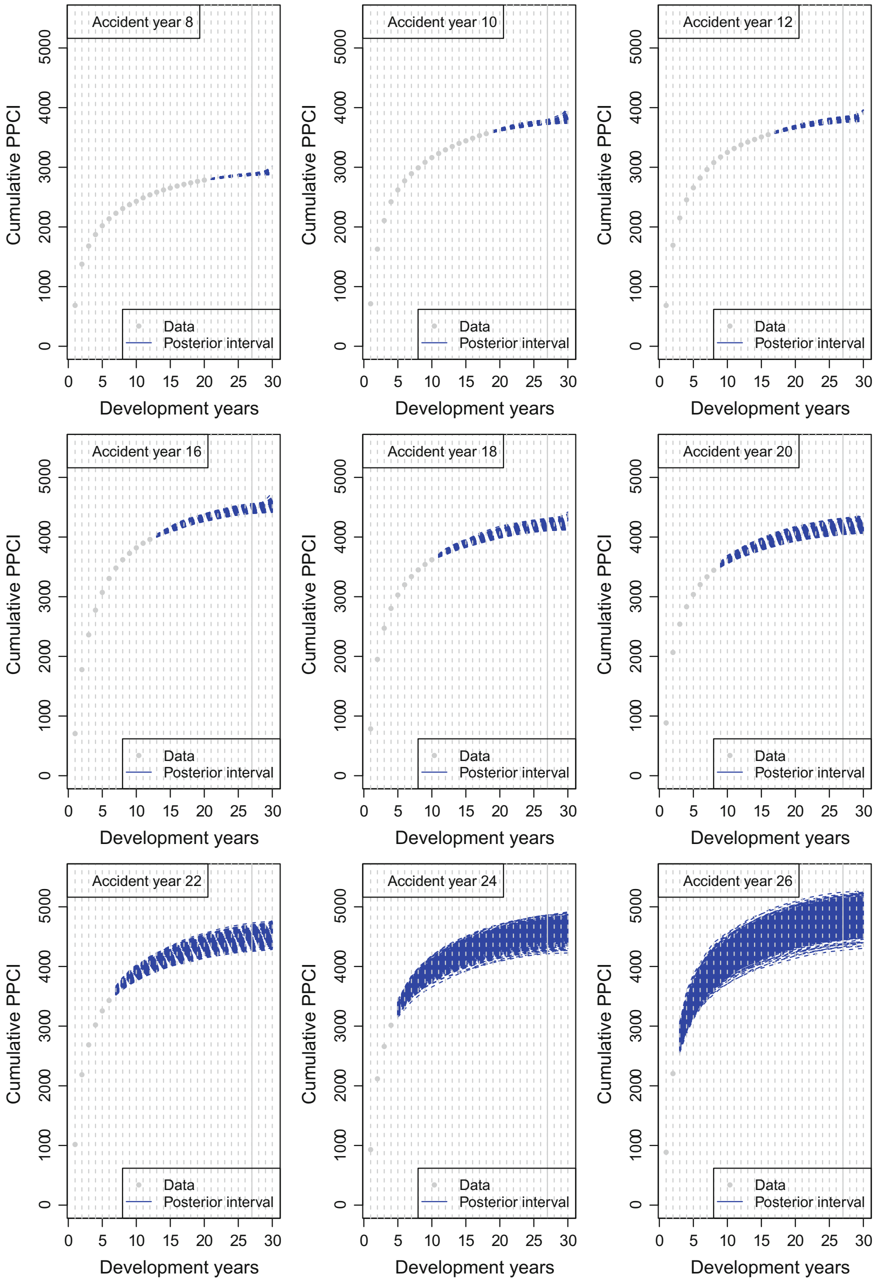


Fig. 5.19 The predictive distributions of cumulative PPCI's for 9 accident years

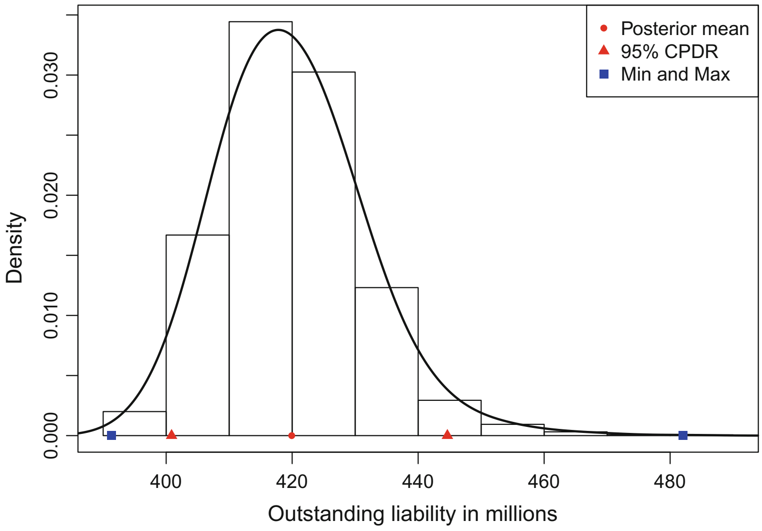


Fig. 5.20 The predictive distribution of total outstanding claims liability of the doctor benefit

Table 5.3 The predictions made from the compound model for the doctor benefit

	Post. mean	Std. deviation	95% CPDR
O/S claims no.	13,693	2,397	(9,846, 19,060)
O/S PPCI	18,320	386	(17,548, 19,059)
O/S liability	419,770,032	10,492,327	(401,778,990, 442,281,893)

Table 5.4 The outstanding claims liability estimates of the doctor benefit from different models

Model	Post. mean	Std. deviation	95% CPDR
Previous chapter	391,761,803	10,195,111	(373,902,941, 414,549,267)
This chapter	419,770,032	10,492,327	(401,778,990, 442,281,893)
PwC	396,827,792	NA	NA

is (401,778,990, 442,281,893). These estimates should be compared with those from the previous chapter in Table 5.4.

5.3.4 Computing Time

Finally, we point out that the computing time for the Bayesian basis expansion model is much less than for the Bayesian chain ladder model in the previous chapter, since

Table 5.5 Comparison of the computing times for the Bayesian chain ladder model and the Bayesian spline model. The computing time is on a Mac of 4 GB 1600 MHz DDR3 with 1.3 GHz Intel Core i5

Model	Response variable	Iterations	Computing time (s)
Bayesian chain ladder	Claims no.	4 × 400	86
	PPCI	4 × 400	364
Bayesian basis expansion	Claims no.	4 × 800	73
	PPCI	4 × 800	65

we use the orthogonal basis functions in the basis expansion model. The computing times for the models used in this section and for those used in Sect. 4.4.3 are displayed in Table 5.5.

5.4 Discussion

To the best of our knowledge of the actuarial science literature, the contribution of this chapter is to introduce a Bayesian basis expansion model to the claims reserving problem. Compared with a stochastic chain ladder model, a Bayesian basis expansion model has the advantages of reducing the number of parameters via shrinkage priors and incorporating the tail factor via interpolation. Due to the orthogonality of basis functions, the running time of MCMC is largely reduced. Unlike a non-linear curve model, a Bayesian basis expansion model can accommodate all the shapes of data. Hence, the Bayesian basis expansion model is one of the most powerful tools according to our research.

This chapter considers the basis expansion of the development year covariate, and it is typically enough for the claims reserving problem. We can address the non-linear effect of both accident years and development years simultaneously (and their interaction) using the multivariate adaptive regression splines (MARS), see Hastie et al. (2009) and Section 3.2 of Wüthrich and Buser (2018). Another related work is Gabrielli et al. (2018) which embeds the MLEs from GLM into a neural network. Further research can consider the basis expansion of two or more covariates, which is more common in the insurance rating problem.

Finally, we point out a problem in Fig. 5.18. From a statistical point of view, since there is no data in the tail development, more variability is expected. However, from an actuarial point of view, the claims paid in the tail development period should be subjected to less variability since almost all the claims have been closed by this period. We do expect less variation associated with the tail development. To realize this expectation, a strong prior for the tail development can be assumed to limit its posterior variability. This method will be applied in the next chapter (see Figs. 6.12 and 6.13). This is a situation when the actuarial judgements override the data.

5.5 Bibliographic Notes

There are several books covering the topic of spline models: Hastie and Tibshirani (1990), Ruppert et al. (2003), Wood (2006), Hastie et al. (2009) and James et al. (2013).

Wood (2003) discusses low rank thin plate splines. Ruppert (2012) discusses selecting the number of knots. DiMatteo et al. (2001) apply RJMCMC to allocate the knots. Crainiceanu et al. (2005) fit a penalized spline model via WinBUGS and give several examples. Hall and Opsomer (2005) give some theoretical properties of penalized spline regression. Lay (2012) is an excellent reference book for matrix concepts such as orthogonality, rank, basis etc.

Bishop (2006) provides a useful review of basis function models. Park and Casella (2008) discuss inference using the Laplace prior distribution. References on generalized double Pareto shrinkage include Armagan et al. (2013). Komaki (2006) investigates the shrinkage predictive distributions based on vague priors.

There is little literature about non-parametric claims reserving models. England and Verrall (2001) apply the generalized additive model. Zhang and Dukic (2013) apply a semi-parametric Bayesian model proposed by Crainiceanu et al. (2005). Gao and Meng (2018) propose the Bayesian basis expansion claims reserving model.

References

Armagan, A., Dunson, D. B., & Lee, J. (2013). Generalized double Pareto shrinkage. *Statistica Sinica*, *23*, 119–143.

Bishop, C. M. (2006). *Pattern recognition and machine learning*. New York: Springer.

Crainiceanu, C. M., Ruppert, D., & Wand, M. P. (2005). Bayesian analysis for penalized spline regression using WinBUGS. *Journal of Statistical Software*, *14*, 1–14.

DiMatteo, I., Genovese, C. R., & Kass, R. E. (2001). Bayesian curve-fitting with free-knot splines. *Biometrika*, *88*, 1055–1071.

England, P. D., & Verrall, R. J. (2001). A flexible framework for stochastic claims reserving. *Proceedings of the Casualty Actuarial Society*, *88*, 1–38.

Faraway, J. J. (2015). *Linear models with R* (2nd ed.). Boca Raton: Chapman & Hall.

Gao, G., & Meng, S. (2018). Stochastic claims reserving via a bayesian spline model with random loss ratio effects. *ASTIN Bulletin*, *48*, 55–88.

Gabrielli, A., Richman, R., & Wüthrich, M. V. (2018). Neural network embedding of the over-dispersed Poisson reserving model. *SSRN*, ID 3288454.

Gelman, A., Carlin, J. B., Stern, H. S., & Rubin, D. B. (2014). *Bayesian data analysis* (3rd ed.). Boca Raton: Chapman & Hall.

Hall, P., & Opsomer, J. D. (2005). Theory for penalised spline regression. *Biometrika*, *92*, 105–118.

Hastie, T. J., Tibshirani, R. J., & Friedman, J. H. (2009). *The elements of statistical learning: Data mining, inference, and prediction* (2nd ed.). New York: Springer.

Hastie, T. J., & Tibshirani, R. J. (1990). *Generalized additive models*. New York: Chapman & Hall.

James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). *An introduction to statistical learning*. New York: Springer.

Komaki, F. (2006). Shrinkage priors for Bayesian prediction. *The Annals of Statistics*, *34*, 808–819.

Lay, D. C. (2012). *Linear algebra and its application* (4th ed.). Boston: Addison Wesley.

Park, T., & Casella, G. (2008). The Bayesian lasso. *Journal of the American Statistical Association*, *103*, 681–686.

Ruppert, D. (2012). Selecting the number of knots for penalized splines. *Journal of Computational and Graphical Statistics*, *11*, 735–757.

Ruppert, D., Wand, M. P., & Carroll, R. J. (2003). *Semiparametric regression*. New York: Cambridge University Press.

Wood, S. N. (2003). Thin plate regression splines. *Journal of the Royal Statistical Society B*, *65*, 95–114.

Wood, S. (2006). *Generalized additive models: An introduction with R*. New York: Chapman & Hall.

Wüthrich, M. V., & Buser, C. (2018). Data analytics for non-life insurance pricing. *SSRN*, ID 2870308.

Zhang, Y. W., & Dukic, V. (2013). Predicting multivariate insurance loss payments under the Bayesian copula framework. *Journal of Risk and Insurance*, *80*, 891–919.

Chapter 6
Multivariate Modelling Using Copulas

Abstract Copulas are a family of multivariate distributions whose marginal distributions are uniform. At the end of reserving problems, we need to aggregate the outstanding liability distribution of each line of business or each type of benefit to get the total outstanding liability distribution. The dependence between them must be considered. In the Bayesian copulas framework, all the uncertainties and correlations are considered during the inferential process which is an advantage compared with the likelihood-based frequentist inference. In Sect. 6.1, the elements of copulas are reviewed, including Sklar's theorem, parametric copulas, inference methods, etc. In Sect. 6.2, we discuss the usefulness of copulas in risk modelling generally. The copula is used to model the empirical dependence between risks while the marginal regression model is used to model the structural dependence. In Sect. 6.3, a bivariate Gaussian copula is used to aggregate the liabilities of the doctor benefit and the hospital benefit in WorkSafe Victoria. These two benefits are correlated positively even after removing the structural effects of the development periods.

6.1 Overview of Copulas

All the models we discussed before are univariate models, i.e., there is one response. However, for many applications, it is more appropriate to apply a multivariate model which captures important relationships. Property damage lines could be positively correlated, e.g., homeowners property damage insurance and personal auto damage insurance could be hit at the same time in *catastrophic events*. Liability lines could be positively correlated due to changes in litigation. It is important to consider the impacts of correlation between lines or benefits on the distribution of aggregated liability.

Typical multivariate distributions include multivariate Gaussian distribution, multivariate t-distribution, Wishart distribution etc. These multivariate distributions also determine the marginal distributions. Copulas are a family of multivariate distributions whose marginal distributions are uniform. In this section, we summarize the elements of copulas in four parts: the mechanism of copulas to join arbitrary marginal distributions, two copula families, measures of bivariate association, and the inferential methods.

© Springer Nature Singapore Pte Ltd. 2018

G. Gao, *Bayesian Claims Reserving Methods in Non-life Insurance with Stan*,
https://doi.org/10.1007/978-981-13-3609-6_6

6.1.1 Sklar's Theorem

Sklar's theorem (Sklar 1959) is perhaps the most important result regarding copulas. It establishes the general connection between any multivariate distribution and copulas and is used essentially in all copula applications. Sklar's theorem states that for an m-dimensional multivariate distribution function F with marginal distributions, $F_1, \ldots, F_m$, there always exists an m-dimensional copula C such that

$$F\left(y_1, \ldots, y_m\right) = C\left[F_1\left(y_1\right), \ldots, F_m\left(y_m\right)\right].$$

Conversely, if C is an m-dimensional copula and $F_1, \ldots, F_m$ are distribution functions, then the function F defined above is an m-dimensional multivariate distribution function with marginal distribution functions, $F_1, \ldots, F_m$.

From Sklar's theorem, we see that for any multivariate distributions, the marginal distributions can be separated from the multivariate dependence which can then be represented by a copula. A direct implication of Sklar's theorem is deriving a copula from a multivariate distribution as follows:

$$C\left(u_1, \ldots, u_m\right) = F\left[F_1^{-1}\left(u_1\right), \ldots, F_m^{-1}\left(u_m\right)\right],$$

where $u_1, \ldots, u_m$ follow marginal uniform distributions on the interval [0, 1].

6.1.1.1 Invariance to Monotone Transformation

While a joint distribution is affected by the monotone transformation of variables, a copula is invariant to the monotone transformation of variables. Let $(y_1, \ldots, y_m)$ be a vector of continuous random variables with a copula C. Define $x_1 = h_1\left(y_1\right), \ldots, x_m = h_m\left(y_m\right)$. If $h_1, \ldots, h_m$ are strictly increasing functions, then $(x_1, \ldots, x_m)$ also has the same copula C.

6.1.1.2 The Fréchet-Hoeffding Bounds for Bivariate Copulas

Fréchet (1935) found that any bivariate copula C is bounded by the Fréchet-Hoeffding lower bound W and the Fréchet-Hoeffding upper bound M, as follows:

$$W\left(u_1, u_2\right) \leq C\left(u_1, u_2\right) \leq M\left(u_1, u_2\right),$$

where $W\left(u_1, u_2\right) = \max\left(u_1 + u_2 - 1, 0\right), M\left(u_1, u_2\right) = \min\left(u_1, u_2\right)$. Figure 6.1 shows the surfaces and contours of W and M compared with the *independent copula* whose variables are independent with each other.

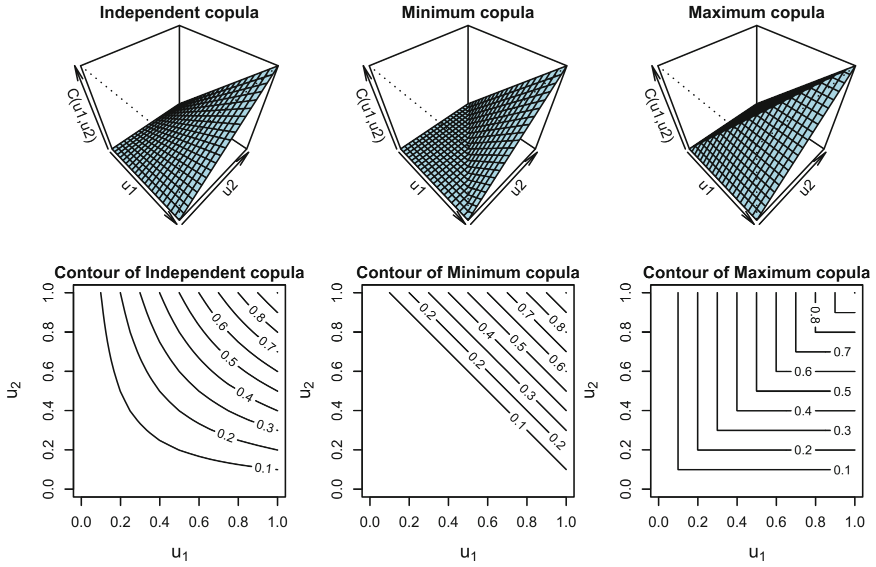


Fig. 6.1 The surfaces and contour plots of the independent, minimum, and maximum copulas

6.1.2 Parametric Copulas

We will investigate two parametric copula families: *elliptical copulas* and *Archimedean copulas*. Elliptical copulas are simply the copulas of elliptical distributions such as multivariate Gaussian distribution and multivariate t-distribution.

Rather than deriving from multivariate distributions, Archimedean copulas are functions of a convex generator and the dependence strength is governed by only one parameter. Archimedean copulas include the Clayton, Gumbel, Frank, and others.

6.1.2.1 Elliptical Copulas

The copula of an m-dimensional normal distributed random vector z with mean zero and correlation matrix $\mathbf{\Sigma}$ is

$$C(u) = \Phi_m \left[\Phi^{-1}(u_1), \ldots, \Phi^{-1}(u_m); \mathbf{\Sigma} \right],$$

where Φ^{-1} is the inverse of the standard normal distribution function and Φ_m is the joint distribution function of z. The connection between elliptical copulas and elliptical distributions provides an easy way to simulate from elliptical copulas: first simulate $z \sim \Phi_m$, then let $u_i = \Phi^{-1}(z_i)$ for $i = 1, \ldots, m$.

The copula of an m-dimensional t-distributed random vector x with mean zero, degrees of freedom ν and correlation matrix $\mathbf{\Sigma}$ is

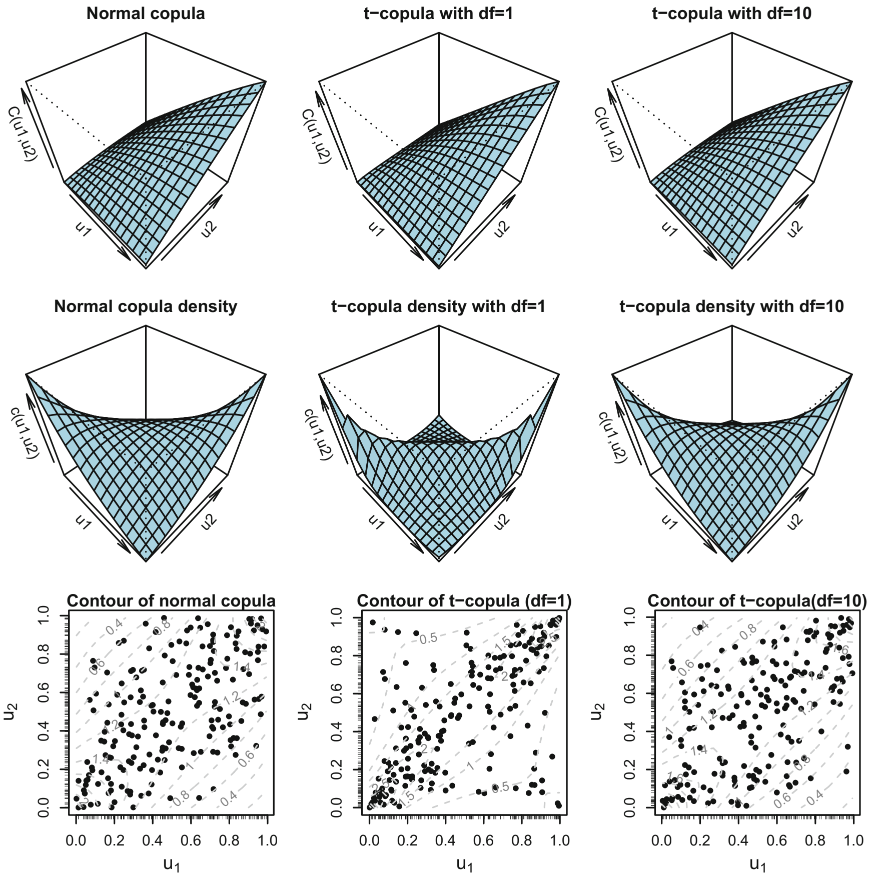


Fig. 6.2 A bivariate Gaussian copula and t-copulas with df = 1, 10, which have the same Pearson correlation of 0.8 and Kendall's tau of 0.5903

$$C\left(u\right) = t_{m,\nu}\left[t_{\nu}^{-1}\left(u_1\right), \ldots, t_{\nu}^{-1}\left(u_m\right); \mathbf{\Sigma}\right],$$

where t_{ν}^{-1} is the inverse of the t-distribution function with ν degrees of freedom and $t_{m,\nu}$ is the joint distribution function of x.

Figure 6.2 shows a bivariate Gaussian copula and a bivariate t-copula, both of which have the same Pearson correlation of 0.8 and Kendall's tau of 0.5903 (defined in Sect. 6.1.3). Kendall's tau of a t-copula does not depend on the degrees of freedom. With the degrees of freedom increasing, a t-copula approaches a normal copula.

Table 6.1 The generators, Kendall's tau and tail dependence for two elliptical copulas and three Archimedean copulas

Copula	Generator	Kendall's tau	Upper	Lower
Gaussian	NA	$2\arcsin\left(\mathbf{\Sigma}_{12}\right)/\pi$	0	0
t	NA	$2\arcsin\left(\mathbf{\Sigma}_{12}\right)/\pi$	0	0
Clayton	$\frac{1}{\theta}\left(u^{-\theta}-1\right)$	$\theta/(\theta+2)$	0	$2^{-1/\theta}$
Gumbel	$(-\log u)^{\theta}$	$1-1/\theta$	$2-2^{1/\theta}$	0
Frank	$-\log\left(\frac{\exp(-\theta u)-1}{\exp(-\theta)-1}\right)$	$1-\frac{4\theta-4\int_0^a t/(e^t-1)dt}{\theta^2}$	0	0

6.1.2.2 Archimedean Copulas

A general definition of Archimedean copulas can be found in Nelsen (2013). An Archimedean m-dimensional copula has the following form:

$$C\left(u\right)=\varphi^{[-1]}\left[\varphi\left(u_1;\theta\right)+\cdots+\varphi\left(u_m;\theta\right);\theta\right],$$

where φ is called the *generator* of copula C and $\varphi^{[-1]}$ is the pseudo-inverse of φ. The function φ is a continuous, strictly decreasing convex function mapping from $[0, 1]$ to $[0, \infty]$, such that $\varphi\left(1\right)=0$.

Table 6.1 shows the generators of three popular Archimedean copulas. We plot the cumulative distribution functions, the probability density functions and the contours of probability density functions for the three Archimedean copulas in Fig. 6.3.

6.1.3 Measures of Bivariate Association

Copulas are invariant under monotone transformation, so we want the measures of association to also be invariant to monotone transformation. Pearson correlation (or linear correlation) is invariant under linear transformation but not invariant under non-linear transformation.

In the following we will review two measures of association known as Kendall's tau and Spearman's rho, both of which depend on the variable ranks rather than their values (and hence are invariant under monotone transformations).

Moreover, we will discuss the tail dependence relating to the amount of dependence in the upper-right-quadrant tail or lower-left-quadrant tail of a bivariate distribution. It turns out that tail dependence is also a copula-based association measure that is invariant under monotone transformations.

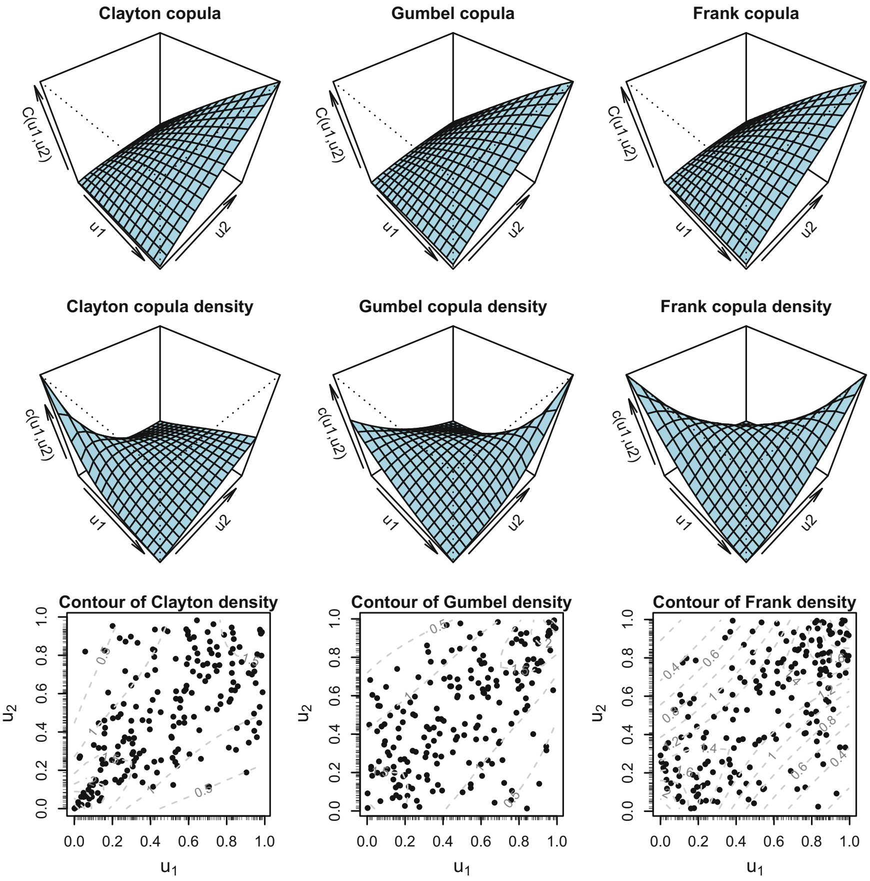


Fig. 6.3 Clayton, Gumbel and Frank copulas with the same Kendall's tau of 0.5903

6.1.3.1 Kendall's Tau and Spearman's Rho

Kendall's tau for two random variables is defined as the probability of *concordance* minus the probability of *discordance*. Assuming the two variables y_1, y_2 have a copula C, then Kendall's tau for y_1, y_2 is given by

$$\tau\left(y_1, y_2\right) := 4 \iint_{[0,1]^2} C\left(u_1, u_2\right) dC\left(u_1, u_2\right) - 1 = 4\mathbb{E}\left[C\left(u_1, u_2\right)\right] - 1.$$

Spearman's rho for y_1, y_2 is given by

$$\rho_S\left(y_1, y_2\right) = 12 \iint_{[0,1]^2} u_1 u_2 dC\left(u_1, u_2\right) - 3 = 12\mathbb{E}\left(u_1 u_2\right) - 3.$$

If the marginal distributions are F_1 and F_2, and $u_1 = F_1(y_1)$ and $u_2 = F_2(y_2)$, then

$$\rho_S(y_1, y_2) = \frac{\mathbb{E}(u_1 u_2) - 1/4}{1/12} = \frac{\text{Cov}\ (u_1, u_2)}{\sqrt{\text{Var}(u_1)}\sqrt{\text{Var}(u_2)}} = \rho(F_1(y_1), F_2(y_2)).$$

Table 6.1 lists Kendall's tau for two elliptical copulas and three Archimedean copulas discussed before. Figure 6.3 shows three Archimedean copulas, all of which have the same Kendall's tau of 0.5903.

6.1.3.2 Tail Dependence

The coefficient of upper tail dependence of the two variables y_1, y_2 with the copula C is defined as

$$\lambda_U := \lim_{u \to 1} \Pr\left[y_2 > F_2^{-1}(u) \,|\, y_1 > F_1^{-1}(u)\right].$$

It can be shown that λ_U is a copula property which has the following equivalent form:

$$\lambda_U = \lim_{u \to 1} \frac{1 - 2u + C(u, u)}{1 - u}.$$

The coefficient of lower tail dependence λ_L is defined in a similar way:

$$\lambda_L := \lim_{u \to 0} \frac{C(u, u)}{u}.$$

Table 6.1 lists the coefficients of upper and lower tail dependence for bivariate copulas. None of the copulas exhibit tail dependence except the Clayton copula and the Gumbel copula. The Clayton copula has a lower tail dependence while the Gumbel copula has an upper tail dependence.

6.1.4 Inference Methods for Copulas

In this section, we follow the model specification as in Pitt et al. (2006). Consider an m-element response variable $y = (y_1, \ldots, y_m)$. It is observed for n times, so the data is

$$\boldsymbol{y} = (y_1, \ldots, y_n) = \left((y_{11}, \ldots, y_{1m})^T, \ldots, (y_{n1}, \ldots, y_{nm})^T\right) = (\boldsymbol{y}_1, \ldots, \boldsymbol{y}_m)^T,$$

where y_i is an m-row-vector of the ith observation, $\boldsymbol{y}_j$ is an n-column-vector of the jth response variable.

For the jth element in the ith observation y_{ij}, we have a k-vector covariate x_{ij}. Marginally, we fit a generalized linear model F_j to y_j. We denote the associated parameters as $\theta_j = \left(\beta_j, \varphi_j\right)$, where β_j is a k-vector of coefficients of x_{ij} and φ_j is a vector of all the other parameters in F_j.

The joint distribution of the ith observation $y_i = (y_{i1}, \ldots, y_{im})$ is modelled by a copula with parameters θ_c as follows:

$$F\left(y_i\right) = C\left[F_1\left(y_{i1}\right), \ldots, F_m\left(y_{im}\right); \theta_c\right], \tag{6.1}$$

which can be seen as a joint distribution of residual ranks of response variables (after removing the systematic effects of covariates). In a Gaussian copula setting, we can write the above copula as

$$\Phi_m\{\Phi^{-1}\left[F_1\left(y_{i1}\right)\right], \ldots, \Phi^{-1}\left[F_m\left(y_{im}\right)\right]; \mathbf{\Sigma}\},$$

where Φ^{-1} is the inverse of a standard normal distribution function and Φ_m is an m-multivariate Gaussian distribution function with mean zero.

In the following, we discuss two likelihood-based estimations: the maximum likelihood estimation (MLE) and the inference functions for margins estimator (IFME). Bootstrap methods and MCMC methods can be applied to estimate the estimation error and the prediction error in IFME.

6.1.4.1 Maximum Likelihood Estimation (MLE)

The density function of y_i is the derivative of Eq. (6.1), as follows:

$$\begin{aligned} f\left(y_i\right) &= \frac{\partial^m C\left[F_1\left(y_{i1}\right), \ldots, F_m\left(y_{im}\right)\right]}{\partial y_{i1} \ldots \partial y_{im}} \\ &= c\left[F_1\left(y_{i1}\right), \ldots, F_m\left(y_{im}\right)\right] f_1\left(y_{i1}\right) \cdots f_m\left(y_{im}\right), \end{aligned}$$

where c is the density function of C and f_i is the density function of y_i. The likelihood function of $\mathbf{y} = (y_1, \ldots, y_n)$ is

$$L\left(\theta; \mathbf{y}\right) = \prod_{i=1}^{n} c\left[F_1\left(y_{i1}\right), \ldots, F_m\left(y_{im}\right)\right] \prod_{j=1}^{m} \prod_{i=1}^{n} f_j\left(y_{ij}\right).$$

The MLE is then defined as

$$\hat{\theta}_{\mathrm{MLE}} = \underset{\theta}{\operatorname{argmax}}\, L\left(\theta; \mathbf{y}\right).$$

Note that the optimization of global likelihood can be quite demanding since the copula likelihood part also contains marginal regression parameters θ_j, $j = 1, \ldots, m$.

6.1.4.2 Inference Functions for Margins Estimator (IFME)

Joe (2014) suggested first estimating θ_j for each jth marginal regression model, and then estimating the copula parameter θ_c via

$$\hat{\theta}_c^{\text{IFME}} = \underset{\theta_c}{\text{argmax}} \prod_{i=1}^{n} c\left(F_1\left(y_{i1}; \hat{\theta}_1\right), \ldots, F_m\left(y_{im}; \hat{\theta}_m\right); \theta_c\right),$$

where $\hat{\theta}_j$, $j = 1, \ldots, m$ are the MLEs of the marginal models. IFME is always easier to compute than the global MLE.

Predictive distributions via parametric bootstrap

Suppose we want to get the predictive distribution of $R = g\left(y_{n+1,1}, \ldots, y_{n+1,m}\right)$ given the covariates $x_{n+1} = (x_{n+1,1}, \ldots, x_{n+1,m})$, where g is a generic function. The bootstrap algorithm is as follows:

1. Fit a marginal regression to $\boldsymbol{y}_j$ to get the estimated parameters $\hat{\theta}_j$ for $j = 1, \ldots m$.
2. Calculate the cdfs given the estimated parameters in step 1 as

$$\hat{u}_{ij} = F_j\left(y_{ij}; \hat{\theta}_j\right) \text{ for } i = 1, \ldots n,\ j = 1 \ldots, m.$$

3. Calculate the IFME of θ_c:

$$\hat{\theta}_c^{\text{IFME}} = \underset{\theta_c}{\text{argmax}} \prod_{i=1}^{n} c\left(F_1\left(y_{i1}; \hat{\theta}_1\right), \ldots, F_m\left(y_{im}; \hat{\theta}_m\right); \theta_c\right).$$

4. Generate a bootstrap sample $u_{ij}^s, i = 1, \ldots, n,\ j = 1, \ldots, m$ from the copula $C(u; \hat{\theta}_c)$.
5. Inverse the cdfs to get a bootstrap data sample $y_{ij}^s = F_j^{-1}\left(u_{ij}^s; \hat{\theta}_j\right), i = 1, \ldots, n,$ $j = 1, \ldots, m$, where $\hat{\theta}_j$ is from step 1.
6. Fit a marginal regression to $\boldsymbol{y}_j^s$ to get the estimated parameters $\hat{\theta}_j^s$, $j = 1, \ldots m$.
7. Calculate the prediction as $R^s = g\left(y_{n+1,1}^s, \ldots, y_{n+1,m}^s\right)$, where $y_{n+1,j}^s = F_j^{-1}\left(u_{n+1,j}^s; \hat{\theta}_j^s\right)$. u_{n+1}^s is a realized sample from $C(u; \hat{\theta}_c)$.
8. Redo steps 4 to 7 for S times to get a bootstrap sample R^s, $s = 1, \ldots, S$.

The key steps are 4 and 7 which establish the correlation between the estimated parameters and the correlation between the predicted values.

Predictive distributions via MCMC

Again, suppose we want to get the predictive distribution of $R = g\left(y_{n+1,1}, \ldots, y_{n+1,m}\right)$ given the covariates $x_{n+1} = (x_{n+1,1}, \ldots, x_{n+1,m})$, where g is a generic function. The MCMC algorithm is as follows:

1. Apply the MCMC methods to each marginal model to generate a Markov chain whose stationary distribution is the posterior distribution of θ_j for $j = 1, \ldots, m$.
2. For the tth MC sampled parameters θ_j^t, calculate the corresponding cumulative probabilities $u_{ij}^t = F_j\left(y_{ij}; \theta_j^t\right)$, which will be used as the "observed" data of the copula.
3. Calculate the MLE of copula parameter θ_c^t, and generate a sample $u_{n+1}^t \sim C(u|\theta_c^t)$.
4. Calculate the prediction values as

$$R^t = g\left(F_1^{-1}\left(u_{n+1,1}^t; \theta_1^t\right), \ldots, F_m^{-1}\left(u_{n+1,m}^t; \theta_m^t\right)\right).$$

5. Repeat steps 2 to 4 for T times to get a MC sample $R^t, t = 1, \ldots, T$.

Example 6.1 (*A simulated example using a Gumbel copula*) Suppose the joint distribution of two response variables have a Gumbel copula and each variable is marginally modelled by a linear regression model:

$$\begin{aligned} y_{i1}, y_{i2} &\sim C\left(F_1\left(y_{i1}; \alpha, \beta_{01}, \beta_{11}\right), F_2\left(y_{i2}; \sigma^2, \beta_{02}, \beta_{12}\right); \theta_c\right) \\ y_{i1} &\sim \text{Gamma}\left(\alpha, \frac{\alpha}{\beta_{01} + \beta_{11} x_{i1}}\right) \\ \log y_{i2} &\sim \text{N}\left(\beta_{02} + \beta_{12} x_{i2}, \sigma^2\right). \end{aligned}$$

The following true parameters are specified: $n = 100$, $\beta_{01} = 1$, $\beta_{11} = 2$, $\alpha = 10$, $\beta_{02} = 0.1$, $\beta_{12} = 0.3$, $\sigma^2 = 0.5$, $\theta_c = 2$ (Kendall's tau is 0.5). x_{i1}, x_{i2} are generated independently from a uniform distribution U $[0, 10]$. y_{i1} and y_{i2} are associated via the same index i which can indicate the same time, the same place or other common features. Figure 6.4 shows the relationships between the variables. Due to the effects of covariates, there is no significant relationship between the two response variables.

Inference functions for margins estimator (IFME)

Two linear regression models are fitted to two response variables respectively. The estimated parameters of two models are shown in Table 6.2. We then calculate the cdfs of the response variables given the estimated regression parameters as

$$F_1\left(y_{i1}; \hat{\beta}_{01}, \hat{\beta}_{11}, \hat{\alpha}\right), F_2\left(y_{i2}; \hat{\beta}_{02}, \hat{\beta}_{12}, \hat{\sigma}\right),$$

which are denoted by $\hat{u}_{i1}, \hat{u}_{i2}, i = 1, \ldots, 100$.

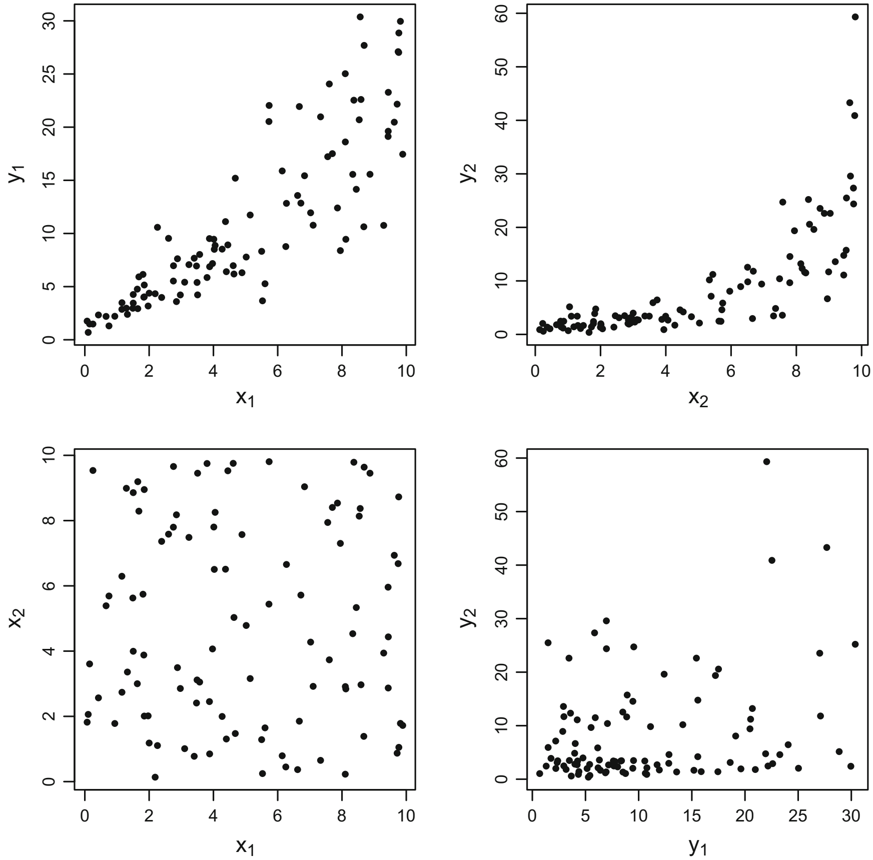


Fig. 6.4 The scatter plots of the simulated data

Table 6.2 The inferences made for two marginal linear regressions in Example 6.1.

Model	$\hat{\beta}_0$	$\hat{\beta}_1$	$1/\hat{\alpha}$ or $\hat{\sigma}$
$y_{i1} \sim \text{Gamma}\left(\alpha, \frac{\alpha}{\beta_{01}+\beta_{11}x_{i1}}\right)$	0.96	1.95	0.11
$\log y_{i2} \sim \text{N}\left(\beta_{02}+\beta_{12}x_{i2}, \sigma^2\right)$	0.07	0.31	0.53

The scatter plot of $\left(\hat{u}_{i1}, \hat{u}_{i2}\right), i = 1, \ldots, 100$ is shown in Fig. 6.5, indicating a significant positive relationship with an empirical Kendall's tau of 0.51. The rugs indicate that the marginal distributions of $\hat{u}_{i1}, \hat{u}_{i2}$ are close to a uniform distribution as expected.

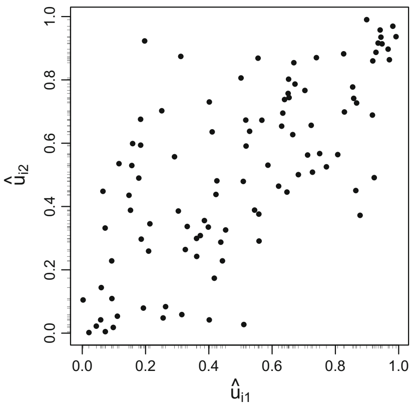


Fig. 6.5 $\hat{u}_{i1}$ versus $\hat{u}_{i2}$

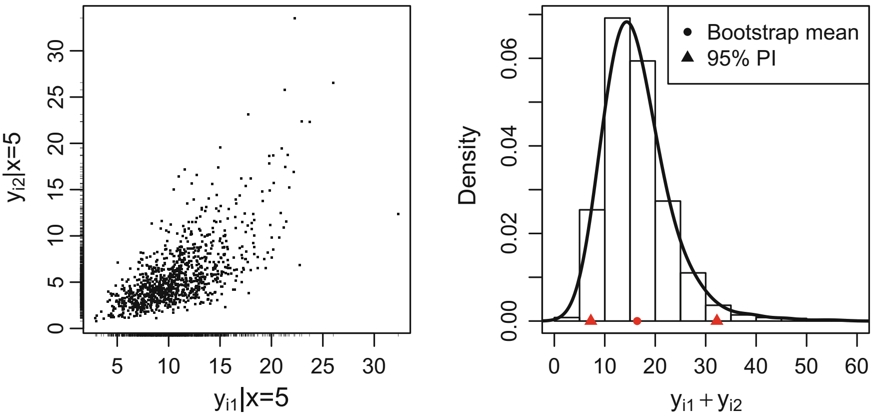


Fig. 6.6 $y_{101,1}$ versus $y_{101,2}$ and the predictive distribution of $y_{101,1} + y_{101,2}$ via the bootstrap methods

The predictive distribution via bootstrap methods

Suppose we want to predict the sum of $y_{101,1}$ and $y_{101,2}$, both of which have the same covariate of 5. The bootstrap algorithm discussed before is used to simulate the predictive distribution of $y_{101,1} + y_{101,2}$. Figure 6.6 shows a significant positive correlation between $y_{101,1}$ and $y_{101,2}$. The bootstrap estimate is 16.42 with the 95% PI of (7.24, 32.23).

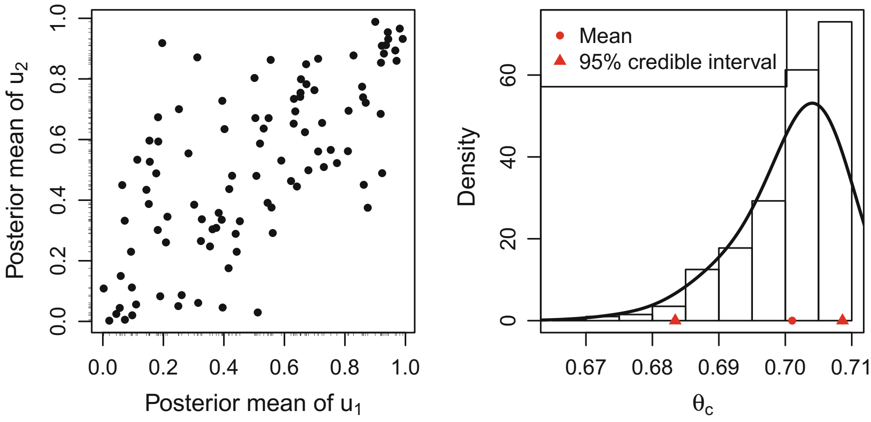


Fig. 6.7 $\bar{u}_{i1}$ versus $\bar{u}_{i2}$ and the posterior distribution of θ_c via the MCMC

The predictive distribution via MCMC methods

We fit two Bayesian linear models separately to the two response variables. HMC is applied to simulate from the posterior distribution. At the end of Bayesian inferential simulation, a sample of parameters is obtained. Assuming the tth sampled parameters as $\beta^t_{01}, \beta^t_{11}, \alpha^t, \beta^t_{02}, \beta^t_{12}, \sigma^t$, we can calculate the corresponding $u^t_{i1}, u^t_{i2}, i = 1, \ldots, n$. The Stan code for sampling u^t_{i1}, u^t_{i2} is as follows

```
E1.code<-"
data{
  int n;
  real x1[n];
  real x2[n];
  real y1[n];
  real y2[n];
}
parameters{
  real b01;
  real b11;
  real b02;
  real b12;
  real<lower=0> alpha;
  real<lower=0> sigma;
}
transformed parameters{
  real mu1[n];
  real mu2[n];
  for (i in 1:n){
    mu1[i]<-b01+b11*x1[i];
    mu2[i]<-b02+b12*x2[i];
  }
}
model{
  for (i in 1:n){
    y1[i] ~ gamma (alpha, alpha/mu1[i]);
    log(y2[i]) ~ normal (mu2[i], sigma);
  }
}
generated quantities{
  real F1[n];
  real F2[n];
  real res1[n];
  real res2[n];
  for (i in 1:n){
    F1[i]<-gamma_cdf(y1[i],alpha,alpha/mu1[i]);
    F2[i]<-normal_cdf(log(y2[i]),mu2[i], sigma);
    res1[i]<-(y1[i]-mu1[i])/mu1[i]*sqrt(alpha);
    res2[i]<-(log(y2[i])-mu2[i])/sigma;
  }
}
"
E1.stanfit<-stan(model_code=E1.code,data=c("n","x1","x2","y1","y2
"),iter=400,chains=4,seed=2)
E1.sim<-extract(E1.stanfit,permuted=T)
```

For the ease of copula parameter estimation, a bivariate Gaussian copula is chosen. The MLE of a bivariate normal copula parameter is just the sample correlation of $\Phi^{-1}(u^t_{i1})$ and $\Phi^{-1}(u^t_{i2})$, denoted by θ^t_c. Figure 6.7 shows the scatter plot of posterior means, $\bar{u}_{i1}$ versus $\bar{u}_{i2}$, which is quite similar to Fig. 6.5 indicating the suitability of using a bivariate Gaussian copula. The histogram of θ_c is shown in Fig. 6.7, which also confirms the significant positive relationship between u_{i1} and u_{i2}. Again, suppose we want to predict the sum of $y_{101,1}$ and $y_{101,2}$, both of which have the same covariate value of 5. We compare two approaches: the independent prediction and the dependent prediction using a copula. The independent prediction is the sum of posterior predictions $y^t_{101,1}$, $y^s_{101,2}$ without considering the permutation. For the dependent prediction using a copula, first a pair of $(u^t_{101,1}, u^t_{101,2})$ is generated from a bivariate Gaussian copula with parameter θ^t_c. Then we inverse two functions, $u^t_{101,1} = F_1(y^t_{101,1}; \beta^t_{01}, \beta^t_{11}, \alpha^t)$ and $u^t_{101,2} = F_2(y^t_{101,2}; \beta^t_{02}, \beta^t_{12}, \sigma^t)$, to get a pair of $(y^t_{101,1}, y^t_{101,2})$. Figure 6.8 shows a positive correlation between $y_{101,1}$, $y_{101,2}$ under the dependent prediction and this positive correlation affects the 97.5% percentile significantly compared with independent prediction. The posterior mean is 16.74 with the 95% CPDR of (7.80, 33.20) under the dependent prediction. The R code for estimating θ^t_c and predicting $y_{101,1}$, $y_{101,2}$ is as follows:

```
rho<-rep(NA,nrow(E1.sim$F1))
y1_5<-rep(NA,nrow(E1.sim$F1))
y2_5<-rep(NA,nrow(E1.sim$F1))
y1_5_ind<-rep(NA,nrow(E1.sim$F1))
y2_5_ind<-rep(NA,nrow(E1.sim$F1))
for (i in 1:nrow(E1.sim$F1))
{
  rho[i]<-cor(qnorm(E1.sim$F1[i,]),qnorm(E1.sim$F2[i,]))
  sigma<-matrix(c(1,rho[i],rho[i],1),ncol=2)
  F12<-rmvnorm(1,mean=rep(0,2),sigma)
  F1<-pnorm(F12[1]);F2<-pnorm(F12[2])
  y1_5[i]<-qgamma(F1,shape=E1.sim$alpha[i],rate=E1.sim$alpha[i]/(
E1.sim$b01[i]+5*E1.sim$b11[i]))
  y2_5[i]<-exp(qnorm(F2,E1.sim$b02[i]+5*E1.sim$b12[i],E1.sim$
sigma[i]))
  y1_5_ind[i]<-rgamma(1,shape=E1.sim$alpha[i],rate=E1.sim$alpha[i]
/(E1.sim$b01[i]+5*E1.sim$b11[i]))
  y2_5_ind[i]<-exp(rnorm(1,E1.sim$b02[i]+5*E1.sim$b12[i],E1.sim$
sigma[i]))
}
```

6.2 Copulas in Modelling Risk Dependence

We focus on the models for multiple run-off triangles. There are several papers on this topic. Shi and Frees (2011) and Shi (2014) use the elliptical copulas to address the dependencies introduced by various sources. They use the parametric bootstrap method to simulate the predictive distribution of outstanding liabilities. De Jong (2012) uses a Gaussian copula to model the dependence of payments from different triangles in the same calendar year.

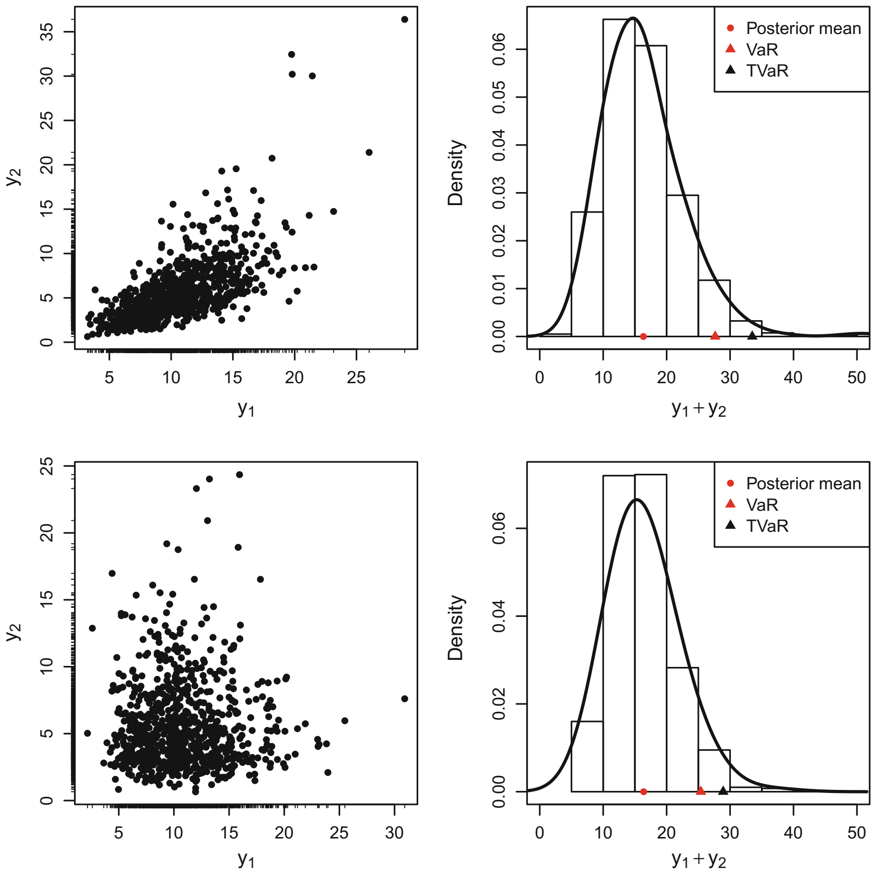


Fig. 6.8 $y_{101,1}$ versus $y_{101,2}$ and the predictive distribution of $y_{101,1} + y_{101,2}$ via the MCMC. The first row is from the desirable copula model. The second row is from the inappropriate independent model for the purpose of comparison. VaR and TVaR will be discussed in Sect. 6.2.2

One of the most important works is Zhang et al. (2012) which was awarded the ARIA prize by the Casualty Actuary Society. This annual prize, first awarded in 1997, is made to the author or authors of a paper published by the Journal of Risk and Insurance that provides the most valuable contribution to casualty actuarial science. This paper uses a Bayesian copula model to address the dependence between the different triangle payments in the same accident year and development year. This paper compares the goodness-of-fit of Clayton, Gumbel, Frank and Gaussian copulas and uses three different marginal regressions: a generalized linear regression, a non-linear growth curve model and a semi-parametric model.

6.2.1 Structural and Empirical Dependence Between Risks

We distinguish the two types of dependence since two different approaches are used to tackle them. In general, the risks an insurer faces often exhibit co-movement or dependencies. This means that knowledge about results for one risk can be used to better predict the results for another risk. Dependence between two risks may be due to known relationships (*structural dependence*), or simply due to the historically observed correlations (*empirical dependence*).

Structural dependence modelling

The structural co-movements can be accounted for in a regression modelling process. Structural dependencies include situations where loss variables are driven by common variables: for example, the cumulative claims of two benefits are both increasing with the development periods. This positive dependence can be modelled by using the covariate of development periods.

Empirical dependence modelling

The empirical co-movements are simply observed without any known (or capable of being modelled) relationships, i.e., the positive relationship of residuals from two models. For many types of risks, particularly in property and liability areas, co-movements are observed, but may not be easily explained. It is more likely necessary to construct dependency models that reflect observed and expected dependencies without formalising the structure of those dependencies with cause-effect models. The theory of copulas provides a comprehensive modelling tool that can reflect dependencies in a very flexible way.

6.2.2 The Effects of Empirical Dependence on Risk Measures

An insurer needs to hold much more than the expected value of unpaid claims liability to ensure the company's solvency with a quite large probability. In Australia, insurers typically add a *risk margin* to the mean of liability to get the estimation of reserve amount.

A risk margin is set consistently with risk measures. A risk measure is not calculated by summing up the contributions of different business lines, but more likely from the distribution of all risks combined. So it is necessary to consider the empirical dependence between different lines.

6.2.2.1 Risk Measures

Most risk measures can be classified as moment-based, tail-based, or probability transforms. The moment-based risk measures (including the standard deviation and semi-standard deviation) are not often used since they are not directly related to the solvency concept.

The most used risk measures are tail-based risk measures which emphasize large losses. The four tail-based risk measures, value at risk (VaR), tail value at risk (TVaR), excess tail value at risk (XTVaR), and expected policyholder deficit (EPD), are defined as follows:

- VaR is a percentile of a loss distribution.
- TVaR is the expected loss at a specified probability level and beyond.
- XTVaR is TVaR less the mean. When the mean is financed by other funding, capital is needed for losses above the mean, so subtracting the mean can capture this need.
- EPD is calculated by multiplying TVaR minus VaR by the probability level. If the probability level is chosen so that capital is VaR at that level, then TVaR minus VaR is the expected value of defaulted losses if there is default. Multiplying this quantity by the complement of the probability level yields the unconditional expected value of defaulted losses.

Probability transforms measure risk by shifting the probability towards the unfavourable outcomes and then computing a risk measure from the transformed probabilities. Most of the usual asset pricing formulas, like the capital asset pricing model and the Black-Scholes options pricing formula, can be expressed as transformed mean.

Table 6.3 The tail-based risk measures under different copula parameters in Example 6.2

Copula parameters	Loss	Mean	VaR	TVaR	XTVaR	EPD
	x_1	200.00	366.14	438.23	234.32	3.02
	x_2	147.31	295.88	369.13	223.07	3.80
$\theta_c = 1, \tau = 0$	$x_1 + x_2$	347.31	566.10	654.03	304.07	4.40
$\theta_c = 2, \tau = 0.5$	$x_1 + x_2$	347.31	667.59	826.50	473.59	7.95
$\theta_c = 4, \tau = 0.75$	$x_1 + x_2$	347.31	687.10	852.75	501.18	8.28

Example 6.2 (Empirical dependence) We illustrate the effects of empirical dependence on the risk measures by a hypothetical example. Consider two correlated loss variables x_1 and x_2 with the following distribution:

$$\begin{aligned} F(x_1, x_2) &= C\left(F_G(x_1; \alpha, \mu_1), F_{LN}\left(x_2; \mu_2, \sigma^2\right); \theta_c\right) \\ x_1 &\sim \text{Gamma}\left(\alpha, \frac{\alpha}{\mu_1}\right) \\ \log x_2 &\sim \text{N}\left(\mu_2, \sigma^2\right), \end{aligned}$$

where C is a Gumbel copula. The underlying parameters are specified as $\mu_1 = 200$, $\alpha = 5$, $\mu_2 = \log 130$ and $\sigma^2 = 0.25$. Consider three cases: $\theta_c = 1$ (i.e., the two risks are independent), $\theta_c = 2$ and $\theta_c = 4$. The two marginal distributions are positively skewed.

By doing simulation, we estimate the four tail-based risk measures for individual loss and the aggregated loss. Table 6.3 shows the results, implying the significant effects of empirical dependence on the tail-based risk measures. Figure 6.9 shows that when $\theta_c = 2$, larger x_1 and x_2 are more likely to be correlated with each other. This is because a Gumbel copula has a non-zero upper tail dependence as shown in Table 6.1.

6.3 Application to the Doctor and Hospital Benefits

Recall that Table 4.9 lists all the benefits in the WorkSafe Victoria. In "medical and like" benefit category, we have two sub-benefits: doctor and hospital. Intuitively, these two sub-benefits should be positively correlated. Here we focus on the models applied to the claims amounts rather than the PPCI method as in the previous two chapters.

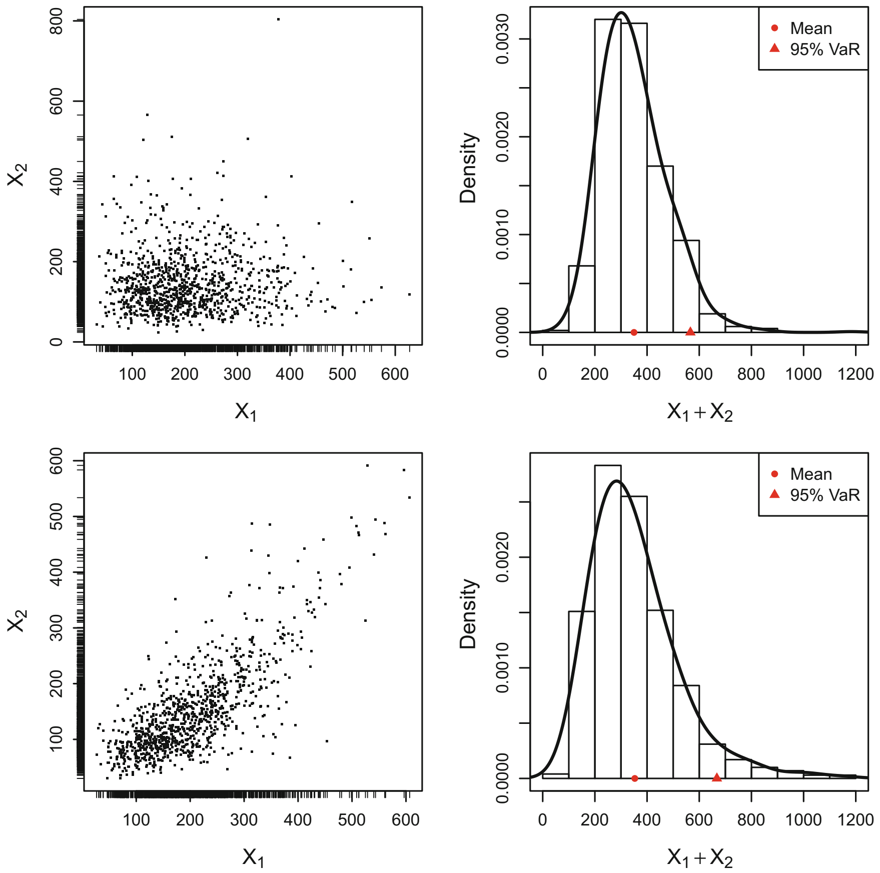


Fig. 6.9 x_1 versus x_2 and the distribution of $x_1 + x_2$. The first row is for $\theta_c = 1$. The second row is for $\theta_c = 2$

6.3.1 Preliminary GLM Analysis Using a Gaussian Copula

As a quick check of correlation between two triangles, we recommend starting from the least complicated models. We fit two chain ladder GLMs with a gamma error and a log link to the doctor benefit x and the hospital benefit y. The model is as follows:

$$
\begin{aligned}
F\left(x_{ij}, y_{ij}\right) &= C\left(F_1\left(x_{ij}; \alpha_1, \mu_{1i}, \gamma_{1j}\right), F_2\left(y_{ij}; \alpha_2, \mu_{2i}, \gamma_{2j}\right); \theta_c\right)\\
x_{ij} &\sim \text{Gamma}\left(\alpha_1, \frac{\alpha_1}{\mu_{1i}\gamma_{1j}}\right), i = 1, \ldots, 27, j = 1, \ldots, 27\\
y_{ij} &\sim \text{Gamma}\left(\alpha_2, \frac{\alpha_2}{\mu_{2i}\gamma_{2j}}\right), i = 1, \ldots, 27, j = 1, \ldots, 27,
\end{aligned}
$$

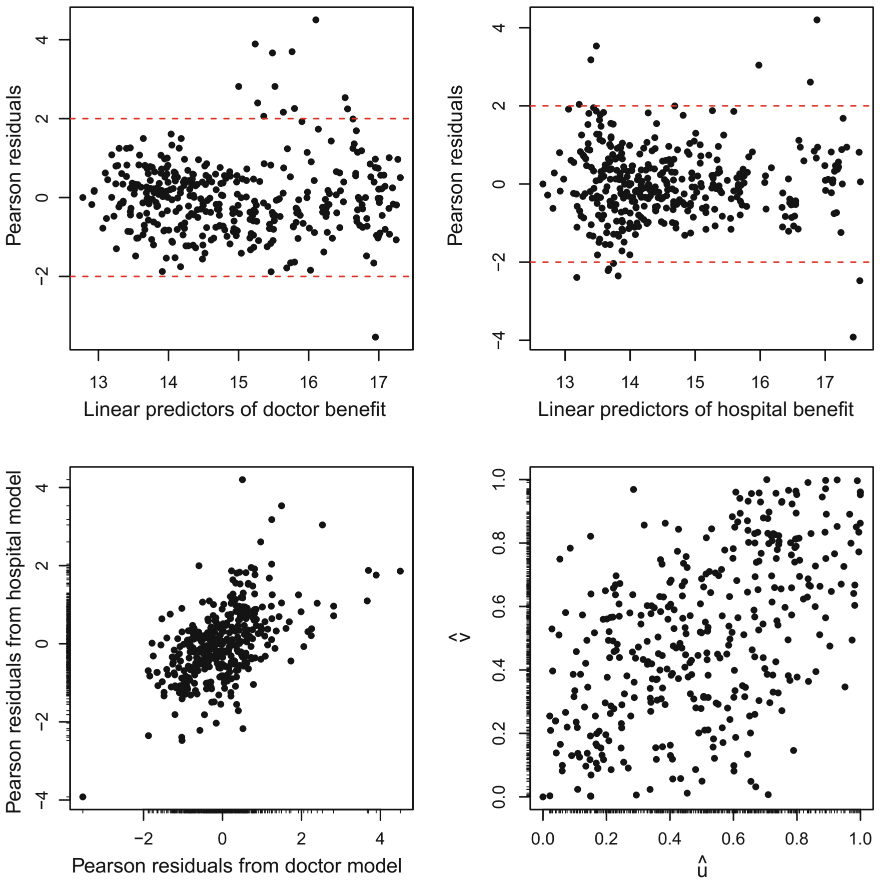


Fig. 6.10 The top two: the residual plots of two marginal regressions. The bottom two: the scatter plot of residuals and the scatter plot of $\hat{u}_{ij}$ versus $\hat{v}_{ij}$.

where F_1, F_2 are the cdfs of gamma distributions and C is a bivariate Gaussian copula.

For model inference, the IFME method is applied. We calculate the empirical cdfs, $\hat{u}_{ij} = F_1\left(x_{ij}; \hat{\alpha}_1, \hat{\mu}_{1i}, \hat{\gamma}_{1j}\right)$ and $\hat{v}_{ij} = F_2\left(y_{ij}; \hat{\alpha}_2, \hat{\mu}_{2i}, \hat{\gamma}_{2j}\right)$. Note that $\hat{\alpha}_1$,$\hat{\mu}_{1i}$, $\hat{\gamma}_{1j}$, $\hat{\alpha}_2$, $\hat{\mu}_{2i}$, $\hat{\gamma}_{2j}$ are the MLEs. We draw four Pearson residual plots: two scatter plots of residuals from marginal GLMs, the plot showing the relationship between two residuals, and the plot of $\hat{u}_{ij}$ versus $\hat{v}_{ij}$ in Fig. 6.10. It shows a significant positive empirical relationship.

6.3.1.1 The Predictive Distribution via a Parametric Bootstrap

The claims liability is simulated via the bootstrap method. The IFME of θ_c is $\hat{\theta}_c = \text{cor}\left[\Phi^{-1}\left(\hat{u}\right), \Phi^{-1}\left(\hat{v}\right)\right] = 0.5530$. We compare the bootstrap sample from the copula model (first row in Fig. 6.11) with the bootstrap sample from an independent model (second row in Fig. 6.11).

The 95% VaR from the copula model is larger than that from the independent model. We also list other tail-based risk measures in Table 6.4. Note that the estimated aggregated liability of both benefits is 707,407,135 dollars in the PwC report.

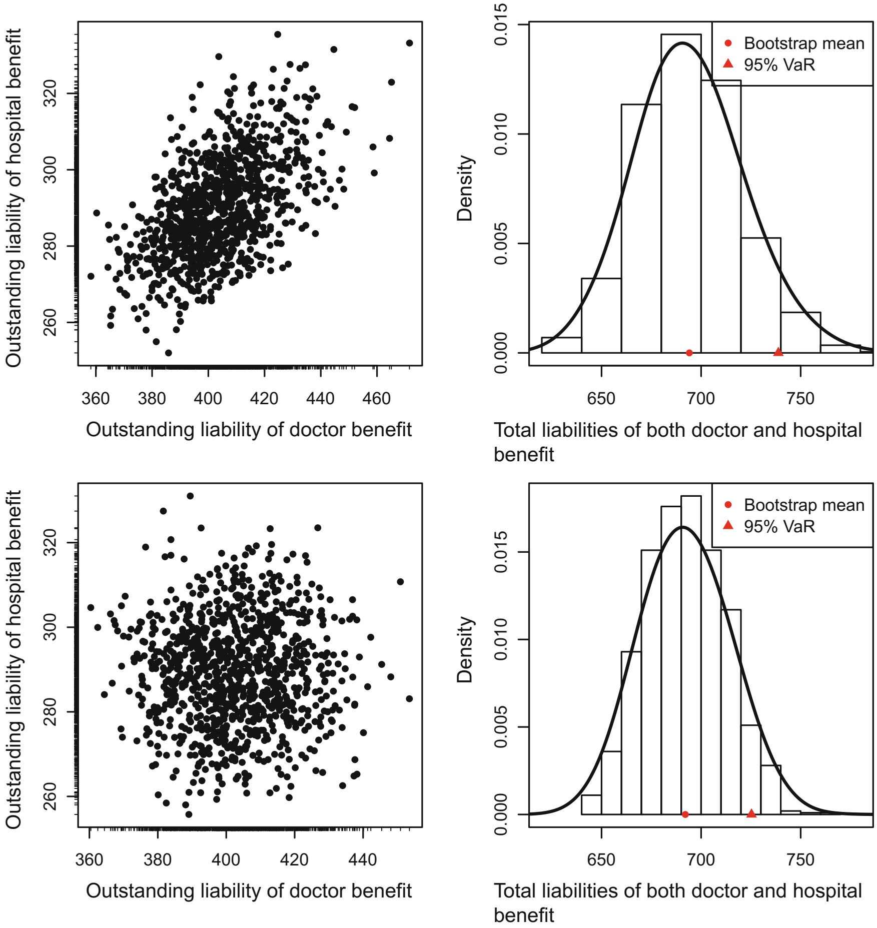


Fig. 6.11 The top two: the prediction of claims liability of two benefits made from the desirable copula model. The bottom two: the prediction of claims liability of two benefits made from the inappropriate independent model. The simulation is performed using bootstrap methods

Table 6.4 The tail-based risk measures of the aggregated liability via bootstrap methods

Model	Mean	VaR	TVaR	XTVaR	EPD
Copula model	692,205,659	737,515,967	747,729,301	55,523,642	510,667
Independent	693,343,113	727,254,943	737,508,270	44,165,157	512,666
Differences (%)	−0.2	1.4	1.4	25.7	−0.4

6.3.2 A Gaussian Copula with Marginal Bayesian Splines

We apply a Gaussian copula model with two marginal Bayesian natural cubic spline models to the two benefits, as follows:

$$\begin{aligned}
F\left(x_{ij}, y_{ij}\right) &= C\left[F_1\left(x_{ij}; \alpha_1, \theta_{ij}\right), F_2\left(y_{ij}; \alpha_2, \varphi_{ij}\right); \theta_c\right] \\
x_{ij} &\sim \text{Gamma}\left(\alpha_1, \frac{\alpha_1}{\theta_{ij}}\right) \\
y_{ij} &\sim \text{Gamma}\left(\alpha_2, \frac{\alpha_2}{\varphi_{ij}}\right) \\
\theta_{ij} &= A_i \times \exp\left(\sum_{h=1}^{27} \beta_h b_h\left(j\right)\right) \\
\varphi_{ij} &= B_i \times \exp\left(\sum_{h=1}^{27} \gamma_h b_h\left(j\right)\right) \\
\beta_h &\sim \text{DoubleExp}\left(0, \sigma_1^2\right), h = 1, \ldots, 27 \\
\gamma_h &\sim \text{DoubleExp}\left(0, \sigma_2^2\right), h = 1, \ldots, 27 \\
\theta_c &\sim \text{U}\left(0, 1\right),
\end{aligned}$$

where F_1, F_2 are the cdfs of gamma distributions and C is a bivariate Gaussian copula. All the claims are assumed to be settled by the development year 30. The IFME method is applied for the copula parameter estimation.

6.3.2.1 The Inferences for the Marginal Bayesian Splines

We draw the posterior mean and the 95% CPDR of the proportions of incremental payments to the ultimate claims payments for two benefits in Fig. 6.12. The increasing uncertainty in the tail developments is due to the lack of data. However, as stated in the discussion of previous chapter, we believe that the uncertainties should not increase dramatically. One approach to solving this problem is to assume strong priors for the tail developments.

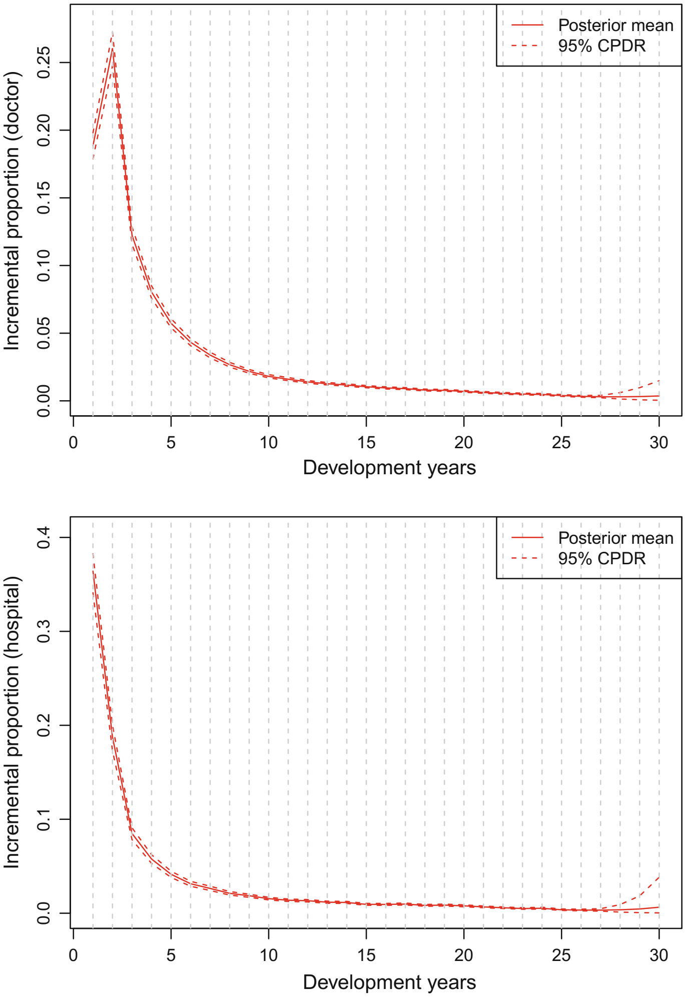


Fig. 6.12 Proportions of the incremental claims to the ultimate claims under non-informative priors

Under the non-informative priors, the posterior mean of $\exp\left[\sum_{h=1}^{27} \beta_h b_h\,(27)\right]$ was 0.003 with posterior standard deviation of 0.0004 and the posterior mean of $\exp\left[\sum_{h=1}^{27} \gamma_h b_h\,(27)\right]$ is 0.003 with posterior standard deviation of 0.0006. Accordingly, we assume the following strong priors for the tail developments in the development years 28, 29, 30:

$$\exp\left(\sum_{h=1}^{27} \beta_h b_h\,(j)\right) \sim \mathrm{N}\,(0.003, 0.0003)\,,\ j = 28, \ldots, 30$$

$$\exp\left(\sum_{h=1}^{27}\gamma_h b_h\left(j\right)\right) \sim \mathrm{N}\left(0.003, 0.0006\right), \, j = 28, \ldots, 30.$$

The resulting posterior distributions of proportions of incremental claims for both benefits are plotted in Fig. 6.13. Now the tail developments do not show as much volatility as in the model with non-informative priors. The Stan code is as follows:

```
amount.code<-"
data{
  int   N;                     // Number of observations
  int   n;                     // Number of future values
  int   K;                     // Number of accident year
  int   M;                     // Number of develop year
  int   H;                     // Number of basis functions
  vector[N]     amount_doc;    // Claims of doctor benefit
  vector[N]     amount_hos;    // Claims of hospital benefit
  matrix[M,H]   dev_basis;     // Basis functions
  int           acc[N];        // Accident years in upper triangle
  int           dev[N];        // Development years in upper triangle
  int           acc_p[n];      // Accident years in lower triangle
  int           dev_p[n];      // Development years in lower triangle
}
parameters{
  vector[H]                                  b1;
  vector[H]                                  b2;
  vector<lower=60*10^6,upper=150*10^6>[K]    ult1;
  vector<lower=50*10^6,upper=150*10^6>[K]    ult2;
  real<lower=0>                              alpha1;
  real<lower=0>                              alpha2;
  real<lower=0>                              sigma1;
  real<lower=0>                              sigma2;
}
transformed parameters{
  vector[N] means1;
  vector[N] means2;
  vector[M] dev_raw1;
  vector[M] dev_raw2;
  vector<lower=0>[M] dev_norm1;
  vector<lower=0>[M] dev_norm2;
  dev_raw1<-exp(dev_basis * b1);
  dev_raw2<-exp(dev_basis * b2);
  dev_norm1<-dev_raw1/sum(dev_raw1);
  dev_norm2<-dev_raw2/sum(dev_raw2);
  for (i in 1:N){
    means1[i]<-ult1[acc[i]]*dev_norm1[dev[i]];
    means2[i]<-ult2[acc[i]]*dev_norm2[dev[i]];
  }
}
model{
  b1 ~ cauchy(0,sigma1);  //sigma is a tuning parameters
  b2 ~ cauchy(0,sigma2);  //sigma is a tuning parameters
  for(i in 28:M){
   dev_norm1[i] ~ normal(0.003, 0.0003);
   dev_norm2[i] ~ normal(0.003, 0.0006);
  }
  for (i in 1:N){
    amount_doc[i] ~ gamma(alpha1, alpha1/means1[i]);
    amount_hos[i] ~ gamma(alpha2, alpha2/means2[i]);
  }
}
generated quantities{
  vector[n] means_p1;
  vector[n] means_p2;
  vector[N] u;
  vector[N] v;
  vector[N] residuals1;
  vector[N] residuals2;
  vector[N] log_lik1;
  vector[N] log_lik2;
  real      D1;
  real      D2;
  for (i in 1:n){
    means_p1[i]<-ult1[acc_p[i]]*dev_norm1[dev_p[i]];
    means_p2[i]<-ult2[acc_p[i]]*dev_norm2[dev_p[i]];
  }
  for (i in 1:N){
    u[i]<-gamma_cdf(amount_doc[i],alpha1,alpha1/means1[i]);
    v[i]<-gamma_cdf(amount_hos[i],alpha2,alpha2/means2[i]);
    residuals1[i]<-(amount_doc[i]-means1[i])/means1[i]*sqrt(alpha
        1);
    residuals2[i]<-(amount_hos[i]-means2[i])/means2[i]*sqrt(alpha
        2);
    log_lik1[i]<-gamma_log(amount_doc[i],alpha1,alpha1/means1[i]);
    log_lik2[i]<-gamma_log(amount_hos[i],alpha2,alpha2/means2[i]);
  }
  D1<-sum(-2*log_lik1);
  D2<-sum(-2*log_lik2);
}"
knots<-c(2:26)
dev_basis<-ns(c(1:30),knots=knots,Boundary.knots = c(1,27),
    intercept = T)
H<-ncol(dev_basis)
M<-nrow(dev_basis)
amount.stanfit<-stan(model_code = amount.code, data=c("amount_doc
","amount_hos","N","n","K","M","H","acc","dev","acc_p","dev_p
","dev_basis"),iter=800,chains=4, seed=10)
```

6.3.2.2 The Predictive Distribution via MCMC Methods

We aggregate the liabilities of two benefits via a bivariate Gaussian copula. Surprisingly, there is no significant difference between simulations of total liability from the copula model and from the independent model as shown in Fig. 6.14.

There are two reasons for this: one is that the marginal Bayesian model uncertainty overwhelms the dependence between them; the other is that the copula is used to model the dependence of incremental claims and the sum of incremental claims may display less dependence. We list the tail-based risk measures of the aggregated liability in Table 6.5.

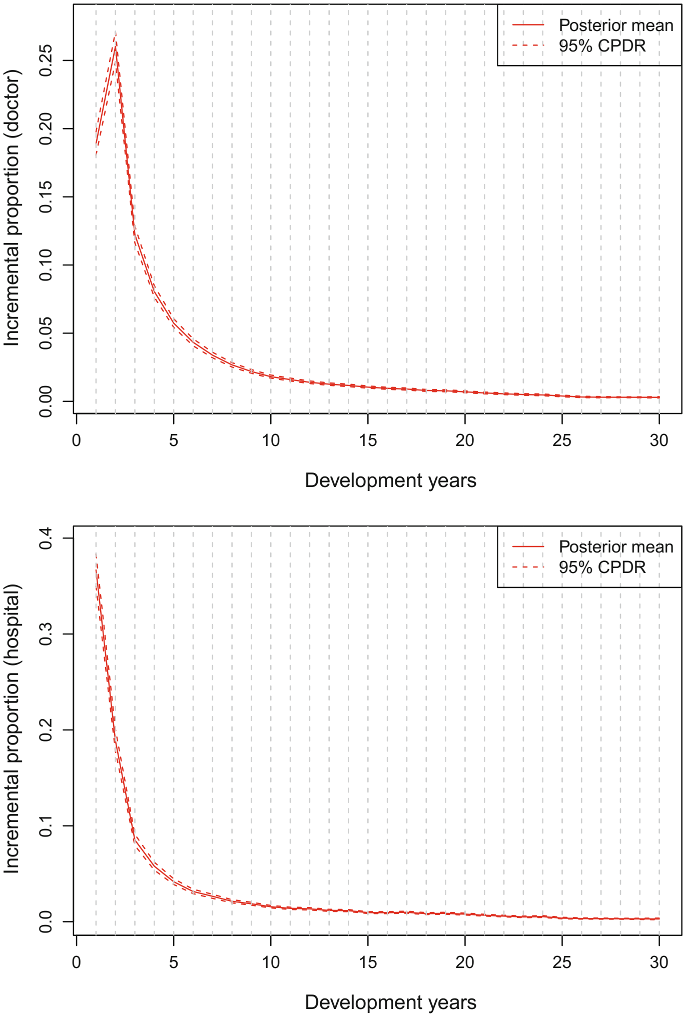


Fig. 6.13 Proportions of the incremental claims to the ultimate claims under strong priors

Table 6.5 The tail-based risk measures of the aggregated liability via MCMC methods. The PwC estimate is 707,407,135

Model	Mean	VaR	TVaR	XTVaR	EPD
Copula model	706,344,715	745,713,292	756,101,056	49,756,341	519,388
Independent	706,302,106	742,610,194	753,119,350	46,817,244	525,458
Differences (%)	0.01	0.42	0.40	6.28	1.16

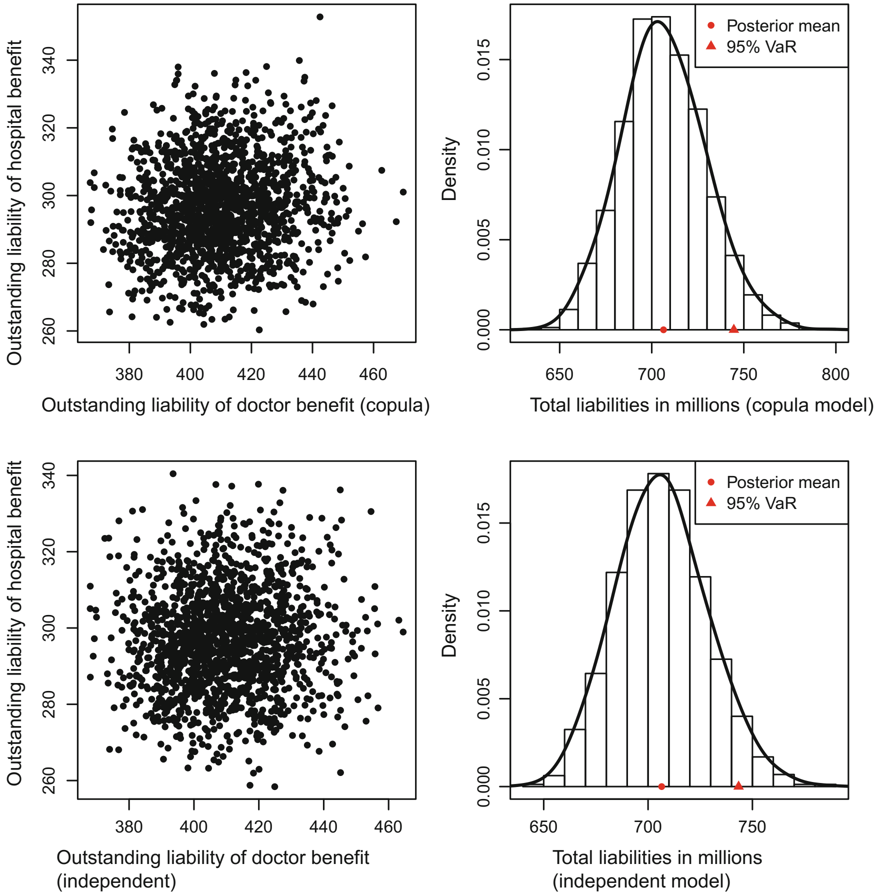


Fig. 6.14 The top two: the prediction of claims liability of two benefits made from the desirable copula model. The bottom two: the prediction of claims liability of two benefits made from the inappropriate independent model. The simulation is performed using MCMC methods

To end of this section, we point out that the copula model makes a difference if the claims payments in the next calendar year are predicted. As we did for the total claims liability, we simulate the claims payments in the next calendar year for both benefits from the copula model and from the independent model. The results are shown in Fig. 6.15 and Table 6.6. The empirical positive correlation is more obvious and it affects the XTVaR most significantly.

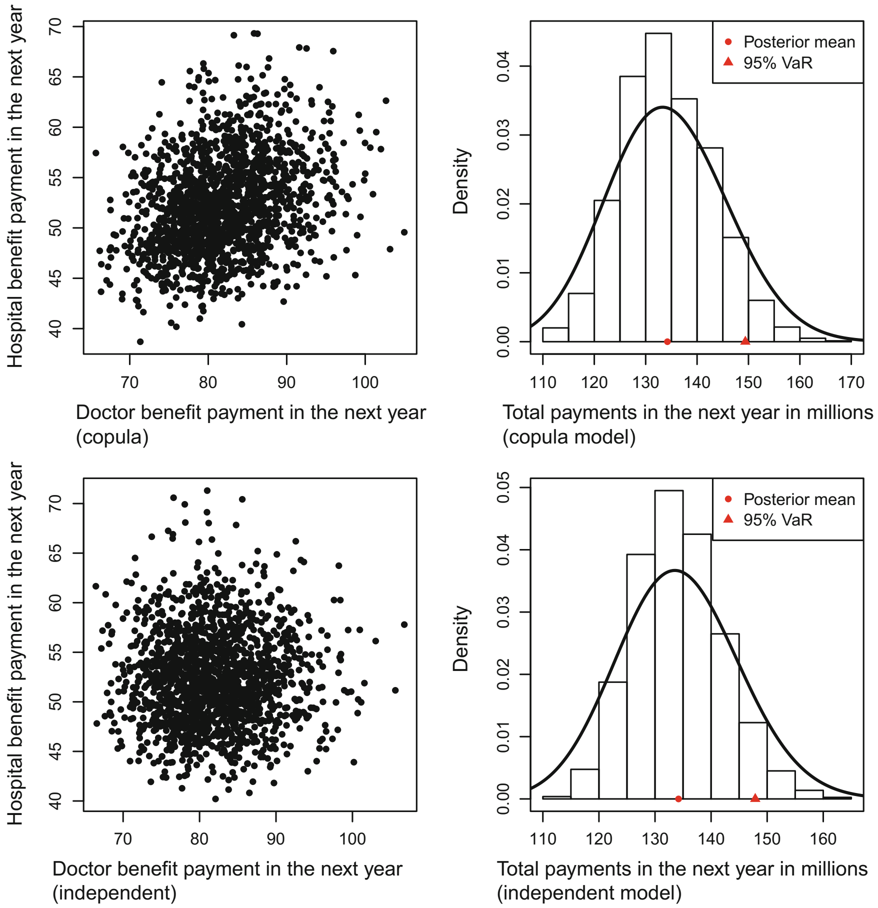


Fig. 6.15 The top two: the prediction of next year claims payment of two benefits made from the desirable copula model. The bottom two: the prediction of next year claims payment of two benefits made from the inappropriate independent model. The simulation is performed using MCMC methods

Table 6.6 The tail-based risk measures of the aggregated claims payments in the next calendar year via MCMC methods

Model	Mean	VaR	TVaR	XTVaR	EPD
Copula model	133,988,676	149,919,590	154,493,898	20,505,222	228,715
Independent	133,956,740	147,246,196	151,426,112	17,469,373	208,996
Differences (%)	0.02	1.82	2.03	17.38	9.44

6.4 Discussion

Copulas have a wide range of applications in finance, risk management, insurance etc. This chapter uses copulas to model the *contemporaneous* correlation, i.e., the dependence among different run-off triangles at the same development year and the same accident year. There are several actuarial papers considering using copulas to model other types of dependence, such as the common calendar years dependence due to claims inflation.

Another concern is the estimation method for the copula models. Here we apply the IFME method involving two consecutive steps: first make inference of the marginal regressions, then fix the parameters of the marginal regressions and infer the copula parameters. We have done some experiments to compare the Bayesian IFME method (applying MCMC to the marginal distribution and MLE to the copula consecutively) and the full Bayesian method (applying MCMC to the multivariate likelihood directly). They show that the Bayesian IFME method takes much less time with better convergence and similar inferences compared with the full Bayesian method. So we are confident with the Bayesian IFME method. Nevertheless, several papers develop a MCMC algorithm for the full Bayesian copula models (see the relevant literature in the next section).

In this chapter, we do not consider the selection of the optimal copula family, since a Gaussian copula fits well (at least visually) in all the problems considered. Genest and Rivest (1993) provide estimation and selection methods for Archimedean copulas. The tail dependence can be used to select a copula if the interest is in the tail behaviour.

6.5 Bibliographic Notes

A thorough discussion of copulas can be found in Joe (2014). An introduction to copulas is available in Nelsen (2013), which does not, however, contain the inference methods. Sklar (1959) introduces Sklar's theorem. Trivedi and Zimmer (2007) cover the main implementation and estimation of copulas. Genest and Rivest (1993) provide estimation and selection methods for Archimedean copulas. Embrechts and Hofert (2013) address the inference methods and goodness-of-fit tests for high-dimensional copulas. Kruskal (1958) discusses the measures of association in detail.

Pitt et al. (2006), Hoff (2007), Danaher and Smith (2011) and Smith (2011) discuss the Bayesian copula models and design efficient MCMC methods accordingly. All of them also consider the special case where there are discrete response variables.

Frees and Valdez (1998) introduced copulas to actuarial science. A general overview of copulas and their applications in actuarial science is provided by Embrechts et al. (2001), Venter (2002), Brehm et al. (2007) and Feldblum (2010).

Literature considering the dependence among run-off triangles includes Shi and Frees (2011) and Zhang et al. (2012), both of which use copulas to model the contemporaneous correlations among various lines of business: the former apply the bootstrap to estimate the predictive distribution of unpaid claims, while the latter apply the MCMC methods, which is closer to what we did in this chapter. De Jong (2012) uses copulas to accommodate the common calendar effect between triangles.

Shi et al. (2012) and Merz et al. (2013) model the contemporaneous dependence between run-off triangles and the common calendar effect within a run-off triangle via a Bayesian hierarchical log-normal model, which is equivalent to a Gaussian copula model with marginal log-normal regressions. Shi (2014) relaxes the marginal log-normal regression using elliptical copulas. Anas et al. (2015) use a hierarchical Archimedean copula to analyze the data from Shi and Frees (2011).

Czado et al. (2012) and Krämer et al. (2013) use copulas to model the dependence between claims occurrences and claims sizes. Meng and Gao (2018) discuss the claims reserving methods using both claims numbers and claims amounts but not in a copula framework.

References

Anas, A., Boucher, J. P., & Cossette, H. (2015). Modeling dependence between loss triangles with hierarchical Archimedean copulas. *ASTIN Bulletin*, *45*, 577–599.

Brehm, P. J., Perry, G., Venter, G. G., & Witcraft, S. (2007). *Enterprise risk analysis for property and liability insurance companies: A practical guide to standard models and emerging solutions*. New York: Guy Carpenter & Company.

Czado, C., Kastenmeier, R., Brechmann, E. C., & Min, A. (2012). A mixed copula model for insurance claims and claim sizes. *Scandinavian Actuarial Journal*, *2012*, 278–305.

Danaher, P. J., & Smith, M. S. (2011). Modeling multivariate distributions using copulas: Applications in marketing. *Marketing Science*, *30*, 4–21.

De Jong, P. (2012). Modeling dependence between loss triangles. *North American Actuarial Journal*, *16*, 74–86.

Embrechts, P., Lindskog, F., & McNeil, A. (2001). Modelling dependence with copulas and applications to risk management. https://people.math.ethz.ch/~embrecht/ftp/copchapter.pdf.

Embrechts, P., & Hofert, M. (2013). Statistical inference for copulas in high dimensions: A simulation study. *ASTIN Bulletin*, *43*, 81–95.

Feldblum, S. (2010). Dependency modeling. *Casualty Actuarial Society Study Notes*.

Fréchet, M. (1935). Generalisations du theoreme des probabilites totales. *Fundamenta Mathematicae*, *25*, 379–387.

Frees, E. W., & Valdez, E. A. (1998). Understanding relationships using copulas. *North American Actuarial Journal*, *2*, 1–25.

Genest, C., & Rivest, L.-P. (1993). Statistical inference procedures for bivariate Archimedean copulas. *Journal of the American statistical Association*, *88*, 1034–1043.

Hoff, P. D. (2007). Extending the rank likelihood for semiparametric copula estimation. *The Annals of Applied Statistics*, *1*, 265–283.

Joe, H. (2014). *Dependence modeling with copulas*. New York: Chapman & Hall.

Krämer, N., Brechmann, E. C., Silvestrini, D., and Czado, C. (2013). Total loss estimation using copula-based regression models. *Insurance: Mathematics and Economics*, *53*, 829–839.

Kruskal, W. H. (1958). Ordinal measures of association. *Journal of the American Statistical Association*, *53*, 814–861.

Meng, S. and Gao, G. (2018). Compound Poisson claims reserving models: Extensions and inference. *ASTIN Bulletin, 48*, 1137–1156.

Merz, M., Wüthrich, M. V., & Hashorva, E. (2013). Dependence modelling in multivariate claims run-off triangles. *Annals of Actuarial Science*, *7*, 3–25.

Nelsen, R. B. (2013). *An introduction to copulas*. New York: Springer.

Pitt, M., Chan, D., & Kohn, R. (2006). Efficient Bayesian inference for Gaussian copula regression models. *Biometrika*, *93*, 537–554.

Shi, P. (2014). A copula regression for modeling multivariate loss triangles and quantifying reserving variability. *ASTIN Bulletin*, *44*, 85–102.

Shi, P., & Frees, E. W. (2011). Dependent loss reserving using copulas. *ASTIN Bulletin*, *41*, 449–486.

Shi, P., Basu, S., & Meyers, G. G. (2012). A Bayesian log-normal model for multivariate loss reserving. *North American Actuarial Journal*, *16*, 29–51.

Sklar, M. (1959). Fonctions de répartition à n dimensions et leurs marges. *Publications de l'Institut de Statistique de l'Université Paris*, *8*, 229–231.

Smith, M. S. (2011). Bayesian approaches to copula modelling. *SSRN*, ID 1974297.

Trivedi, P. K., & Zimmer, D. M. (2007). *Copula Modeling: An Introduction for Practitioners*. Boston: Now Publishers.

Venter, G. G. (2002). Tails of copulas. *Proceedings of the Casualty Actuarial Society*, *89*, 68–113.

Zhang, Y., Dukic, V., & Guszcza, J. (2012). A Bayesian non-linear model for forecasting insurance loss payments. *Journal of the Royal Statistical Society A*, *175*, 637–656.

Chapter 7
Epilogue

Abstract In this final chapter, we summarize the three proposed Bayesian claims reserving models and suggest a Bayesian modelling procedure for use when facing a real problem. Finally, other considerations with respect to Bayesian methodology and actuarial applications are discussed.

7.1 The Three Claims Reserving Models

This monograph presents several Bayesian models to tackle the claims reserving problem in general insurance. These models are used to analyze the WorkSafe Victoria data set. Bayesian models provide a coherent way to incorporate the prior knowledge and combine it with the evidence from the data. This property is particularly useful when the actuarial judgements override the data. Another advantage of Bayesian models is that the Bayesian inferential engine can simulate the posterior distribution of parameters and the predictive distribution of the future value. This property is very important for application to the claims reserving problem, since claims reserving models are always complicated in terms of number of parameters and insurers are more interested in the distribution of unpaid claims than the point estimates.

We point out that the three proposed claims reserving models in this monograph are a compound model, a Bayesian natural cubic spline basis expansion model and a copula model with Bayesian margins. For the model inference, we rely on Stan, which implements the HMC method. Like MCMC, HMC simulates a Markov chain whose stationary distribution is the same as the target distribution. Compared with MCMC, HMC has a higher acceptance rate due to the "Hamiltonian dynamics" proposal.

7.1.1 A Compound Model

The PPCI method is used in the PwC report for the doctor benefit in WorkSafe Victoria, and we propose a compound model as a stochastic version of the PPCI method. The key point is to establish the relationship between the variance in a

© Springer Nature Singapore Pte Ltd. 2018

G. Gao, *Bayesian Claims Reserving Methods in Non-life Insurance with Stan*,
https://doi.org/10.1007/978-981-13-3609-6_7

single claim payment and the variance in PPCI. The distributional assumption of a single claim payment could be checked if we had the individual claims data.

The compound model discussed in Chaps. 5 and 6 is as follows:

$$\begin{aligned} y_{ij} &= \sum_{k=1}^{\mu_i} x_{ijk} \\ \mu_i &\sim \text{Distribution}_i \\ x_{ijk} &\sim \text{Gamma}\left(\alpha_{ij}, \beta_{ij}\right), \; k = 1, \ldots, \mu_i, \end{aligned}$$

where μ_i is the ultimate claims number of accident year i whose distribution is approximated by a Bayesian model, and x_{ijk} is the payment for the kth claim during the development year j whose distribution depends on both accident year and development year.

We define the payments per claim incurred during the development period j of accident year i as $\text{PPCI}_{ij} := y_{ij}/\text{E}\left(\mu_i\right)$. Note that $\text{E}(\text{PPCI}_{ij}) = \text{E}(x_{ijk})$. We use the posterior mean of μ_i as an estimate of $\text{E}\left(\mu_i\right)$. The relationship between the variance of PPCI_{ij} and the variance of x_{ijk} is

$$\text{Var}\left(x_{ijk}\right) = \frac{\left(\text{E}\left(\mu_i\right)\right)^2 \text{Var}\left(\text{PPCI}_{ij}\right) - \text{Var}\left(\mu_i\right)\left(\text{E}\left(\text{PPCI}_{ij}\right)\right)^2}{\text{E}\left(\mu_i\right)},$$

where all the quantities on the right hand side can be estimated by a MC sample. The distribution of y_{ij} conditional on μ_i is $\text{Gamma}\left(\mu_i \alpha_{ij}, \beta_{ij}\right)$, where $\alpha_{ij} = \text{E}\left(x_{ijk}\right)^2/\text{Var}\left(x_{ijk}\right)$ and $\beta_{ij} = \alpha_{ij}/\text{E}\left(x_{ijk}\right)$.

7.1.2 A Bayesian Natural Cubic Spline Basis Expansion Model

In the claims reserving models, the two challenging tasks are the derivation of the predictive distribution and the fit to the various shapes of development patterns. In the Bayesian framework, the first task is easily tackled by either the MCMC method or the HMC method. To deal with the second task, we rely on the chain ladder model or the basis expansion model.

The stochastic chain ladder model treats the development year as a factor variable, effectively introducing the same number of parameters as the number of development periods. So it can accommodate all the shapes of development patterns. However, the stochastic chain ladder model does not introduce the tail development.

The basis expansion model treats the development year as a continuous variable and expands the predictor space by including transformation of the predictor variable. In the Bayesian framework, we can shrink the non-significant parameters and interpolate the tail development.

Consider the Bayesian basis expansion model as discussed in Chap. 5:

$$
\begin{aligned}
y_{ij} &\sim \text{Gamma}\left(\alpha, \frac{\alpha}{\mu_{ij}}\right), i = 1, \ldots, I, j = 1, \ldots, J \\
\mu_{ij} &= P_i \times LR_i \times \exp\left(\sum_{h=1}^{H} \beta_h b_h\left(j\right)\right) \\
\beta_h &\sim \text{DoubleExp}\left(0, \sigma_h^2\right), h = 1, \ldots, H.
\end{aligned}
$$

The key part of this model is the natural cubic spline basis functions $\{b_h : h = 1, \ldots, H\}$ which expand the predictor space. We use the B-spline basis, an orthogonal set, generated by R function `ns( )`. We normally choose the knots at every unique value of **x**, which is analogous to the full rank smoothing splines.

Here we choose a gamma error distribution which could be replaced by other distributions such as a more general Tweedie distribution.

7.1.3 A Copula Model with Bayesian Margins

The copula model is used to aggregate the outstanding claims liabilities estimated from multiple triangles. We could assume any marginal regression for each triangle in the copula framework. In this monograph, we use the Gaussian copula which offers computational simplicity.

The copula model with Bayesian marginal regressions discussed in Chap. 6 is as follows:

$$
\begin{aligned}
F\left(x_{ij}, y_{ij}\right) = C\left[F_1\left(x_{ij}; \alpha_1, \theta_{ij}\right), F_2\left(y_{ij}; \alpha_2, \varphi_{ij}\right); \theta_c\right] \\
x_{ij} &\sim \text{Gamma}\left(\alpha_1, \frac{\alpha_1}{\theta_{ij}}\right) \\
y_{ij} &\sim \text{Gamma}\left(\alpha_2, \frac{\alpha_2}{\varphi_{ij}}\right) \\
\theta_{ij} &= A_i \times \exp\left(\sum_{h=1}^{H} \beta_h b_h\left(j\right)\right) \\
\varphi_{ij} &= B_i \times \exp\left(\sum_{h=1}^{H} \gamma_h b_h\left(j\right)\right) \\
\beta_h &\sim \text{DoubleExp}\left(0, \sigma_1^2\right) \\
\gamma_h &\sim \text{DoubleExp}\left(0, \sigma_2^2\right) \\
\theta_c &\sim \text{U}\left(0, 1\right),
\end{aligned}
$$

with non-informative priors for $\alpha_1, \alpha_2, A_i, B_i, \sigma_1^2, \sigma_2^2$. We fit the model using the IFME method (Joe 2014) which is not a full Bayesian analysis. It is possible to do a full Bayesian analysis by using a user-defined MCMC algorithm.

7.2 A Suggested Bayesian Modelling Procedure

A typical Bayesian modelling procedure includes: proposing a full probability model, calculating the posterior inference conditional on the data, modelling evaluation, and refinement. We suggest that a typical Bayesian modelling procedure should involve the following steps:

1. Define the problem. Different problems need different levels of effort. If we just need to get a point estimate of unpaid claims, the deterministic CL method or BF method may solve this problem well enough.
2. Visualize the data. We cannot change the data which is a reflection of real world, but we could change a model. Visualising the data helps us detect abnormal observations and choose a suitable model to analyze the data.
3. Fit a classical model, usually a GLM. This includes choosing the covariates, the mean function and the error distribution. It is good to try a simple model first, then go deeper into a more complicated model. In the GLM setting, lots of diagnostic tools are available and easily accessed. The mean function and the error distribution can be used in the next step.
4. Set up a Bayesian model and simulate from the posterior distribution. We turn to Bayesian modelling software such as BUGS or Stan to simulate from the posterior distributions. The detection of convergence was discussed in Sect. 3.2.1 and strategies for improving the convergence and efficiency were discussed in Sect. 3.2.2.
5. Make inferences from the MCMC or HMC sample. If the predictive distribution is required, we need to perform one further step to simulate the future values from the likelihood.
6. Model assessment and selection. We can compare different models using several information criteria. LOO cross-validation and WAIC can be easily derived using Stan, while DIC can be calculated automatically in BUGS.

We followed these six steps (though not strictly) in all the Bayesian modelling presented in this monograph. A variation to step 4 is to use a user-defined MCMC or HMC algorithm. We did this in the early stage of research for the examples discussed in Chaps. 2 and 3. We also used a user-defined RJMCMC algorithm in Sect. 4.3.1.

7.3 Other Considerations

We list some other considerations in Bayesian methodology and actuarial applications.

7.3.1 *Bayesian Methodology*

7.3.1.1 ODP Models and Tweedie Models in Stan

ODP models can be specified in BUGS via the zero trick. Indeed the zero trick can be used to define arbitrary likelihood function in BUGS (Lunn et al. 2000). However, Stan does not accept the zero trick and we have not yet worked out how to make a statement of the ODP model in Stan. In addition, Tweedie distributions are not built-in distributions in Stan.

7.3.1.2 Other Non-parametric Bayesian Models

We have seen the power of basis expansion models. Other Bayesian non-parametric models include Gaussian process models, Dirichlet process models etc. Further research could be done on these models and their applications.

7.3.1.3 Copulas Comparison and Selection

As we mentioned in Sect. 6.4, the comparison of different copulas is ignored in that chapter. The selection of a suitable copula could be based on the information criteria or the tail dependence. The goodness-of-fit for copulas is discussed in Genest et al. (2009).

7.3.1.4 Distributional Approximation

In Sect. 3.4, we have briefly reviewed variational Bayes methods, which are promising when dealing with large data sets. Other distributional approximation methods, such as pragmatic expectation (Minka 2001), are discussed in Bishop (2006). These methods deserve more attention in future research.

7.3.2 *Actuarial Applications*

7.3.2.1 Calendar Year Effect

The calendar year effect is not considered in this monograph. The obvious pattern in Fig. 4.17 indicates a significant calendar year effect. A possible approach is to incorporate a calendar year covariate (see Sect. 4.5).

7.3.2.2 Stochastic Reserving Methods for Other Benefits in WorkSafe Victoria

Three benefits in WorkSafe Victoria are investigated: the weekly benefit, the doctor benefit and the hospital benefit. These benefits are chosen since they are stable and less subject to changes in legislation than many others. It is desirable to propose stochastic versions of other reserving methods in the PwC report (Simpson and McCourt 2012) such as PPAC and PPCR.

7.3.2.3 One-Year Reserve Volatility

One key issue relating to the actual implementation of Solvency II is the estimation of the *one-year reserve volatility* (or *claims development results*). This issue is discussed in some recent literature, including Saluz et al. (2011), Christiansen and Niemeyer (2014) and Saluz (2015).

References

Bishop, C. M. (2006). *Pattern recognition and machine learning*. New York: Springer.

Christiansen, M., & Niemeyer, A. (2014). The fundamental definition of the solvency capital requirement in Solvency II. *ASTIN Bulletin*, *44*, 501–533.

Genest, C., Rémillard, B., & Beaudoin, D. (2009). Goodness-of-fit tests for copulas: A review and a power study. *Insurance: Mathematics and Economics*, *44*, 199–213.

Joe, H. (2014). *Dependence modeling with copulas*. New York: Chapman & Hall.

Lunn, D. J., Thomas, A., Best, N., & Spiegelhalter, D. (2000). WinBUGS—A Bayesian modelling framework: Concepts, structure, and extensibility. *Statistics and Computing*, *10*, 325–337.

Minka, T. P. (2001). Expectation propagation for approximate Bayesian inference. In *Proceedings of the Seventeenth Conference on Uncertainty in Artificial Intelligence* (pp. 362–369). Morgan Kaufmann Publishers Inc.

Saluz, A. (2015). Prediction uncertainties in the Cape Cod reserving method. *Annals of Actuarial Science*, *9*, 239–263.

Saluz, A., Gisler, A., & Wüthrich, M. V. (2011). Development pattern and prediction error for the stochastic Bornhuetter-Ferguson claims reserving method. *ASTIN Bulletin*, *41*, 279–313.

Simpson, L., & McCourt, P. (2012). *Worksafe Victoria actuarial valuation of outstanding claims liability for the scheme as at 30 June 2012*, Technical report. PricewaterhouseCoopers Actuarial Pty Ltd.

Appendix A
Derivations

A.1 Example 2.3

Since $\mathbb{E}\left(x_t\right) = \mathbb{E}\left(x_{t-1}\right)$ and $\operatorname{Var}\left(x_t\right) = \operatorname{Var}\left(x_{t-1}\right)$, we can easily get

$$\mathbb{E}\left(x_1\right) = 0 \text{ and } \operatorname{Var}\left(x_1\right) = \frac{1}{\lambda\left(1-\alpha^2\right)}.$$

Hence, this autoregressive process is uniquely determined by the following two distributions:

$$\begin{aligned} x_1 | \alpha, \lambda &\sim \mathrm{N}\left(0, \frac{1}{\lambda\left(1-\alpha^2\right)}\right) \\ x_t | x_{t-1}, \alpha, \lambda &\sim \mathrm{N}\left(\alpha x_{t-1}, \frac{1}{\lambda}\right), t = 2, 3, \ldots, n. \end{aligned}$$

A.1.1 The Joint Posterior Distribution

The joint posterior distribution of α and λ is

$$\begin{aligned} p\left(\alpha, \lambda | \boldsymbol{x}\right) &\propto p\left(\boldsymbol{x} | \alpha, \lambda\right) p\left(\alpha\right) p\left(\lambda\right) \\ &\propto p(\boldsymbol{x} | \alpha, \lambda) \frac{1}{\lambda} \\ &= p\left(x_n | x_{n-1}, x_{n-2}, \ldots, x_1, \alpha, \lambda\right) p(x_{n-1}, x_{n-2}, \ldots, x_1 | \alpha, \lambda) \frac{1}{\lambda} \\ &= p\left(x_n | x_{n-1}, \alpha, \lambda\right) p\left(x_{n-1} | x_{n-2}, \alpha, \lambda\right) \cdots p\left(x_1 | \alpha, \lambda\right) \frac{1}{\lambda} \\ &\propto \sqrt{\lambda} \exp\left[-\frac{\lambda}{2}\left(x_n - x_{n-1}\right)^2\right] \cdots \sqrt{\lambda\left(1-\alpha^2\right)} \exp\left[-\frac{\lambda\left(1-\alpha^2\right)}{2} {x_1}^2\right] \frac{1}{\lambda} \end{aligned}$$

© Springer Nature Singapore Pte Ltd. 2018

G. Gao, *Bayesian Claims Reserving Methods in Non-life Insurance with Stan*,
https://doi.org/10.1007/978-981-13-3609-6

$$= \lambda^{\frac{n}{2}-1}\left(1-\alpha^2\right)^{\frac{1}{2}}$$
$$\exp\left\{-\frac{\lambda}{2}\left[(x_n-\alpha x_{n-1})^2+\cdots+(x_2-\alpha x_1)^2+\left(1-\alpha^2\right){x_1}^2\right]\right\}.$$

Thus,

$$p\left(\alpha,\lambda|\boldsymbol{x}\right)=h_0\lambda^{\frac{n}{2}-1}\left(1-\alpha^2\right)^{\frac{1}{2}}\exp\left[-\frac{\lambda}{2}h\left(\boldsymbol{x},\alpha\right)\right],$$

where

$$h_0=\frac{1}{\int_0^\infty\int_{-1}^1\lambda^{\frac{n}{2}-1}\left(1-\alpha^2\right)^{\frac{1}{2}}\exp\left[-\frac{\lambda}{2}h\left(\boldsymbol{x},\alpha\right)\right]d\alpha d\lambda}$$

is called the normalizing constant and

$$h\left(\boldsymbol{x},\alpha\right)=(x_n-\alpha x_{n-1})^2+(x_{n-1}-\alpha x_{n-2})^2+\cdots+(x_2-\alpha x_1)^2+\left(1-\alpha^2\right){x_1}^2.$$

A.1.2 Two Marginal Posterior Distributions

The marginal posterior distribution of α is

$$\begin{aligned}p\left(\alpha|\boldsymbol{x}\right)&=\int\limits_0^\infty p\left(\alpha,\lambda|\boldsymbol{x}\right)d\lambda\\&\propto\int\limits_0^\infty\lambda^{\frac{n}{2}-1}\left(1-\alpha^2\right)^{\frac{1}{2}}\exp\left[-\frac{\lambda}{2}h\left(\boldsymbol{x},\alpha\right)\right]d\lambda\\&=\left(1-\alpha^2\right)^{\frac{1}{2}}\frac{\Gamma\left(\frac{n}{2}\right)}{\left(\frac{h(\boldsymbol{x},\alpha)}{2}\right)^{\frac{n}{2}}}\\&\propto\frac{\left(1-\alpha^2\right)^{\frac{1}{2}}}{h(\boldsymbol{x},\alpha)^{\frac{n}{2}}}.\end{aligned}$$

Thus,

$$p\left(\alpha|\boldsymbol{x}\right)=h_1\frac{\left(1-\alpha^2\right)^{\frac{1}{2}}}{h(\boldsymbol{x},\alpha)^{\frac{n}{2}}},$$

where

$$h_1=\int\limits_{-1}^1\frac{h(\boldsymbol{x},\alpha)^{\frac{n}{2}}}{\left(1-\alpha^2\right)^{\frac{1}{2}}}d\alpha.$$

The marginal posterior distribution of λ is

$$\begin{aligned} p\left(\lambda|\boldsymbol{x}\right) &= \int_{-1}^{1} p\left(\alpha, \lambda|\boldsymbol{x}\right) d\alpha \\ &\propto \int_{-1}^{1} \lambda^{\frac{n}{2}-1}\left(1-\alpha^2\right)^{\frac{1}{2}} \exp\left[-\frac{\lambda}{2} h\left(\boldsymbol{x}, \alpha\right)\right] d\alpha \\ &\propto \lambda^{\frac{n}{2}-1} \int_{-1}^{1} \exp\left[-\frac{\lambda}{2} h\left(\boldsymbol{x}, \alpha\right)\right] d\alpha \\ &\equiv \pi\left(\lambda\right). \end{aligned}$$

Thus $p\left(\lambda|\boldsymbol{x}\right) = \pi_0 \pi\left(\lambda\right)$, where $\pi_0 = 1/\int_0^\infty \pi\left(\lambda\right) d\lambda$.

A.1.3 Full Conditional Distribution of λ

It is easy to note that the full conditional distribution of λ is given by

$$\lambda|\boldsymbol{x}, \alpha \sim \text{Gamma}\left(\frac{n}{2}, \frac{h\left(\boldsymbol{x}, \alpha\right)}{2}\right).$$

So

$$\hat{\lambda} = \mathrm{E}\left(\lambda|\boldsymbol{x}\right) = \mathrm{E}\left(\mathrm{E}\left(\lambda|\alpha, \boldsymbol{x}\right)|\boldsymbol{x}\right) = \mathrm{E}\left(\left.\frac{n}{h\left(\boldsymbol{x}, \alpha\right)}\right|\boldsymbol{x}\right) = \int_{-1}^{1} \frac{n}{h\left(\boldsymbol{x}, \alpha\right)} p(\alpha|\boldsymbol{x}) d\alpha.$$

In Sect. 3.1.3 we show that the Rao-Blackwell estimate of $\hat{\lambda}$ is just based on the above argument.

A.2 Example 2.5

Consider a sample of size n from $\mathrm{N}\left(\mu, \sigma^2\right)$, denoted by $\boldsymbol{x}$. We want to test $H_0 : \mu = \mu_0$ versus $H_1 : \mu \neq \mu_0$ with σ^2 unspecified.

A.2.1 CLR and GLR

The conditional likelihood ratio (CLR) is

$$T^{\mathrm{C}}(\boldsymbol{x},\theta)=\frac{\sup_{\mu\neq\mu_0}p(\boldsymbol{x}|\mu,\sigma^2)}{\sup_{\mu=\mu_0}p(\boldsymbol{x}|\mu,\sigma^2)}=\frac{p(\boldsymbol{x}|\mu=\bar{\boldsymbol{x}},\sigma^2)}{p(\boldsymbol{x}|\mu=\mu_0,\sigma^2)}=\exp\left(\frac{n(\bar{\boldsymbol{x}}-\mu_0)^2}{-2\sigma^2}\right).$$

Since the posterior predictive p-value, p_B, is invariant under any strictly monotone data-free transformation of a discrepancy variable, we can use $n(\bar{\boldsymbol{x}}-\mu_0)^2/\sigma^2$ as the CLR. Similarly, the generalized likelihood ratio (GLR), $T^{\mathrm{G}}(\boldsymbol{x})$, can be calculated as $n(\bar{\boldsymbol{x}}-\mu_0)^2/s^2$, where $\bar{\boldsymbol{x}}$ and s^2 are the sample mean and the sample variance.

A.2.2 p_B Using CLR

The posterior predictive p-value, p_B, conditional on σ^2 is

$$p_B^{\mathrm{C}}\left(\sigma^2\right)=\Pr\left(\frac{n\left(\bar{X}-\mu_0\right)^2}{\sigma^2}\geq\frac{n(\bar{\boldsymbol{x}}-\mu_0)^2}{\sigma^2}\middle|H_0,\sigma^2\right),$$

which depends on the choice of the conditional prior $p(\sigma^2)$. Under the non-informative prior, $p(\sigma^2)\propto 1/\sigma^2$, the posterior distribution of σ^2 can be calculated as

$$\sigma^2|\boldsymbol{x}\sim\frac{ns_0^2}{\chi_n^2},$$

where $s_0^2=\sum_{i=1}^n(x_i-\mu_0)^2/n$ is the MLE of σ^2 under the null hypothesis H_0.

We have the following equation:

$$\begin{aligned}p_B^{\mathrm{C}}&=\mathrm{E}\left(p_B^{\mathrm{C}}\left(\sigma^2\right)|\boldsymbol{x},H_0\right)\\&=\mathrm{E}\left[\Pr\left(\frac{n\left(\bar{X}-\mu_0\right)^2}{\sigma^2}\geq\frac{n(\bar{\boldsymbol{x}}-\mu_0)^2}{\sigma^2}\middle|\mu_0,\sigma^2,\boldsymbol{x}\right)\middle|\mu_0,\boldsymbol{x}\right]\\&=\mathrm{E}\left[\Pr\left(\frac{\frac{n\left(\bar{X}-\mu_0\right)^2}{\sigma^2}}{\frac{ns_0^2}{\sigma^2}/n}\geq\frac{n(\bar{\boldsymbol{x}}-\mu_0)^2}{s_0^2}\middle|\mu_0,\sigma^2,\boldsymbol{x}\right)\middle|\mu_0,\boldsymbol{x}\right].\end{aligned}\tag{A.1}$$

Let:

$$u=\frac{n\left(\bar{X}-\mu_0\right)^2}{\sigma^2},\quad v=\frac{ns_0^2}{\sigma^2},\quad T_0(\boldsymbol{x})=\frac{n(\bar{\boldsymbol{x}}-\mu_0)^2}{s_0^2}.$$

Since $u|\mu_0,\sigma^2,\boldsymbol{x}\sim\chi_1^2$ does not depend on σ or $\boldsymbol{x}$, we have

$$u|\mu_0, \boldsymbol{x} \sim \chi_1^2 \text{ and } (u|\mu_0, \boldsymbol{x}) \perp \left(\sigma^2|\mu_0, \boldsymbol{x}\right).$$

Similarly we have

$$v|\mu_0, \boldsymbol{x} \sim \chi_n^2, \quad (v\,|\mu_0, \boldsymbol{x}) \perp (u|\,\mu_0, \boldsymbol{x}), \quad \left(\frac{u}{\frac{v}{n}}\middle|\mu_0, \boldsymbol{x}\right) \perp (\sigma^2|\mu_0, \boldsymbol{x}).$$

It follows, by continuation of (A.1), that

$$\begin{aligned}
p_B^{\mathrm{C}} &= \mathrm{E}\left[\Pr\left(\frac{u}{\frac{v}{n}} \geq T_0\left(\boldsymbol{x}\right)\middle|\mu_0, \sigma^2, \boldsymbol{x}\right)\middle|\, \mu_0, \boldsymbol{x}\right] \\
&= \mathrm{E}\left[\frac{\Pr\left(\frac{u}{\frac{v}{n}} \geq T_0\left(\boldsymbol{x}\right), \sigma^2|\mu_0, \boldsymbol{x}\right)}{p\left(\sigma^2|\mu_0, \boldsymbol{x}\right)}\middle|\, \mu_0, \boldsymbol{x}\right] \\
&= \int_{\sigma^2} \frac{\Pr\left(\frac{u}{\frac{v}{n}} \geq T_0\left(\boldsymbol{x}\right), \sigma^2|\mu_0, \boldsymbol{x}\right)}{p\left(\sigma^2|\mu_0, \boldsymbol{x}\right)} p\left(\sigma^2|\mu_0, \boldsymbol{x}\right) d\sigma^2 \\
&= \int_{\sigma^2} \Pr\left(\frac{u}{\frac{v}{n}} \geq T_0\left(\boldsymbol{x}\right), \sigma^2\middle|\mu_0, \boldsymbol{x}\right) d\sigma^2 \\
&= \Pr\left(\frac{u}{\frac{v}{n}} \geq T_0\left(\boldsymbol{x}\right)\middle|\mu_0, \boldsymbol{x}\right) \int_{\sigma^2} p\left(\sigma^2|\mu_0, \boldsymbol{x}\right) d\sigma^2 \\
&= \Pr\left(F_{1,n} \geq T_0\left(\boldsymbol{x}\right)|\boldsymbol{x}, \mu_0\right).
\end{aligned}$$

A.2.3 p_B Using GLR

The posterior predictive p-value using GLR is

$$\begin{aligned}
p_B^{\mathrm{G}} &= \Pr\left(\frac{n\left(\bar{X} - \mu_0\right)^2}{s^2} \geq \frac{n(\bar{\boldsymbol{x}} - \mu_0)^2}{s^2}\middle|\mu_0, \boldsymbol{x}\right) \\
&= \Pr\left(F_{1,n-1} \geq \frac{n(\bar{\boldsymbol{x}} - \mu_0)^2}{s^2}\middle|\mu_0, \boldsymbol{x}\right) \\
&= \Pr\left(F_{1,n-1} \geq T^{\mathrm{G}}\left(\boldsymbol{x}\right)\right) \\
&= \Pr\left(t_{n-1} > \sqrt{T^{\mathrm{G}}\left(\boldsymbol{x}\right)}\right).
\end{aligned}$$

Notice that T^{G} is a pivotal quantity, and p_B based on GLR is identical to the classical p-value based on the t-test.

A.3 Calculation of Equation (2.5)

To calculate p_B, we will first verify that 1's are uniformly placed given n_1. It can be shown that $\Pr\left(x_k = 1 \middle| \sum_{i=1}^{n} x_i = n_1\right) = n_1/n$, as follows:

$$\begin{aligned}
&\Pr\left(x_k = 1 \middle| \sum_{i=1}^{n} x_i = n_1\right)\\
&= \int_0^1 \Pr\left(x_k = 1 \middle| \sum_{i=1}^{n} x_i = n_1, \theta\right) p\left(\theta \middle| \sum_{i=1}^{n} x_i = n_1\right) d\theta\\
&= \int_0^1 \frac{\Pr\left(x_k = 1, \dot{x}_{-k} = n_1 - 1|\theta\right)}{\Pr\left(\sum_{i=1}^{n} x_i = n_1 \middle| \theta\right)} p\left(\theta \middle| \sum_{i=1}^{n} x_i = n_1\right) d\theta\\
&= \int_0^1 \frac{\theta\binom{n-1}{n_1-1}\theta^{n_1-1}(1-\theta)^{(n-1)-(n_1-1)}}{\binom{n}{n_1}\theta^{n_1}(1-\theta)^{n-n_1}} p\left(\theta \middle| \sum_{i=1}^{n} x_i = n_1\right) d\theta\\
&= \int_0^1 \frac{n_1}{n} p\left(\theta \middle| \sum_{i=1}^{n} x_i = n_1\right) d\theta = \frac{n_1}{n},
\end{aligned}$$

where $\dot{x}_{-k} = \sum_{i=1}^{n} x_i - x_k$.

Next,

$$p_{\mathrm{B}} = \sum_{i=0}^{10} \Pr\left(R\left(i, 10-i\right) \le 3\right) \Pr(n_1 = i|\boldsymbol{x}). \tag{A.2}$$

It follows that

$$\begin{aligned}
p_{\mathrm{B}} &= \Pr\left(r\left(x'\right) \le r\left(\boldsymbol{x}\right) \middle| \boldsymbol{x}\right)\\
&= \int_0^1 \left(\sum_{i=0}^{10}\sum_{j=1}^{3} \Pr\left(R\left(i, 10-i\right) = j\right) \Pr\left(n_1 = i|\theta\right)\right) p(\theta|\boldsymbol{x}) d\theta\\
&= \sum_{i=0}^{10} \int_0^1 \Pr\left(R\left(i, 10-i\right) \le 3\right) p\left(n_1 = i|\theta\right) p\left(\theta|\boldsymbol{x}\right) d\theta\\
&= \sum_{i=0}^{10} \int_0^1 \Pr\left(R\left(i, 10-i\right) \le 3\right) p\left(n_1 = i, \theta|\boldsymbol{x}\right) d\theta\\
&= \sum_{i=0}^{10} \Pr\left(R\left(i, 10-i\right) \le 3\right) \Pr(n_1 = i|\boldsymbol{x}).
\end{aligned}$$

We next calculate $p\left(n_1|\boldsymbol{x}\right)$ as follows:

$$\begin{aligned} p\left(n_1|\boldsymbol{x}\right) &= \int\limits_0^1 p(n_1, \theta|\boldsymbol{x})d\theta \\ &= \int\limits_0^1 p\left(n_1|\theta, \boldsymbol{x}\right) p(\theta|\boldsymbol{x})d\theta \\ &= \int\limits_0^1 p\left(n_1|\theta\right) p(\theta|\boldsymbol{x})d\theta \\ &\propto \int\limits_0^1 \binom{n}{n_1}\theta^{n_1+5}(1-\theta)^{n-n_1+5}d\theta \\ &\propto \binom{n}{n_1}\Gamma\left(n_1+6\right)\Gamma\left(n+6-n_1\right) \\ &\propto \frac{(n_1+5)!\,(n+5-n_1)!}{n_1!\,(n-n_1)!}, \end{aligned}$$

which implies that

$$p\left(n_1|\boldsymbol{x}\right) = \frac{\frac{(n_1+5)!(n+5-n_1)!}{n_1!(n-n_1)!}}{\sum_{i=0}^{n}\frac{(i+5)!(n+5-i)!}{i!(n-i)!}}.$$

Now the pmfs of $R\left(i, 10-i\right)$ and $n_1|\boldsymbol{x}$ are know. Finally, according to Eq. (A.2), p_B can be calculated as

$$p_{\mathrm{B}} = \sum_{i=0}^{10}\Pr\left(R\left(i, 10-i\right) \le 3\right)\Pr\left(n_1 = i|\boldsymbol{x}\right) = 0.1630.$$

Appendix B
Other Sampling Methods

B.1 A Simple Proof of the M-H Algorithm

The Metropolis-Hastings (M-H) algorithm is used to simulate a Markov chain whose stationary distribution is the target distribution. This Markov chain has a certain transaction matrix determined by the target distribution and by a proposal distribution.

Let $\mathscr{X}$ be a finite sample space and $\pi\left(x\right)$ a probability of interest on $\mathscr{X}$ (perhaps specified up to an unknown normalizing constant). The M-H algorithm at the tth iteration works as follows:

1. Propose a value from a proposal distribution $g(x^*|x^{t-1})$, where x^{t-1} is the state of x at the end of $t-1$ iteration or the initial value when $t=1$.
2. Calculate the acceptance ratio
$$A\left(x^*, x^{t-1}\right) = \frac{\pi\left(x^*\right) g(x^{t-1}|x^*)}{\pi\left(x^{t-1}\right)\ g(x^*|x^{t-1})}.$$
3. Accept x^* and set $x^t = x^*$ with probability $A\left(x^*, x^{t-1}\right)$ if $A\left(x^*, x^{t-1}\right) \leq 1$. Otherwise, reject x^* and set $x^t = x^{t-1}$.

The above M-H algorithm defines a Markov transaction matrix $\mathbf{K}$, whose entry, $\mathbf{K}\left(x^{t-1}, x^t\right)$, has the following expression:

$$\begin{cases} g\left(x^t|x^{t-1}\right), & \text{if } x^t \neq x^{t-1}, A\left(x^{t-1}, x^t\right) \geq 1 \\ g\left(x^t|x^{t-1}\right) A\left(x^{t-1}, x^t\right), & \text{if } x^t \neq x^{t-1}, A\left(x^{t-1}, x^t\right) < 1 \\ g\left(x^t|x^{t-1}\right) + \sum g\left(x^t|x^{t-1}\right)\left(1 - A\left(x^{t-1}, x^t\right)\right), & \text{if } x^t = x^{t-1}, \end{cases}$$

where $A\left(x^{t-1}, x^t\right)$ is the acceptance ratio. Note that the normalizing constant of π cancels out in all calculations. It is easy to show that the following equation holds:

$$\pi\left(x^{t-1}\right)\mathbf{K}\left(x^{t-1}, x^t\right) = \pi\left(x^t\right)\mathbf{K}\left(x^t, x^{t-1}\right).$$

© Springer Nature Singapore Pte Ltd. 2018

G. Gao, *Bayesian Claims Reserving Methods in Non-life Insurance with Stan*,
https://doi.org/10.1007/978-981-13-3609-6

Thus

$$\sum_{x^{t-1}} \pi\left(x^{t-1}\right) \mathbf{K}\left(x^{t-1}, x^{t}\right) = \sum_{x^{t-1}} \pi\left(x^{t}\right) \mathbf{K}\left(x^{t}, x^{t-1}\right) = \pi\left(x^{t}\right) \sum_{x^{t-1}} \mathbf{K}\left(x^{t}, x^{t-1}\right) = \pi\left(x^{t}\right).$$

The above equation says that no matter what the starting value is, after many iterations, the chance of being at x^t is approximately $\pi\left(x^t\right)$.

When $\mathscr{X}$ extends to the general space, many results are analogous to the results for discrete state-space space chains as we have shown here (see Robert and Casella 2013). Hence the M-H algorithm can be applied to most target distributions.

B.2 Adaptive Rejection Sampling

In adaptive rejection sampling, we assume $\pi\ (x)$ is log-concavity and denote $h\ (x) = \log\left(\pi\ (x)\right)$. Suppose that $h\ (x)$ and $h'\ (x)$ have been evaluated at k abscissae in $\mathscr{X} : x_1 \leq x_2 \leq \ldots \leq x_k$. Let $T_k = \{x_i : i = 1, 2, \ldots, k\}$.

Define the envelope function on T_k as $\exp u_k\ (x)$ where $u_k\ (x)$ is a piecewise linear upper hull formed from the tangents to $h\ (x)$ at the abscissae in T_k. The tangents at x_i and x_{i+1} intersect at

$$z_i = \frac{h\ (x_{i+1}) - h\ (x_i) - x_{i+1}h'\ (x_{i+1}) + x_i h'\ (x_i)}{h'\ (x_i) - h'\ (x_{i+1})}, \text{ for } i = 1, \ldots, k-1.$$

We add z_0 as the lower bound of $\mathscr{X}$ and z_k as the upper bound of $\mathscr{X}$. Then $u_k\ (x) = h\ (x_i) + (x - x_i)\, h'\ (x_i)$ for $x \in \left[z_{i-1}, z_i\right], i = 1, \ldots, k$.

Define the squeezing function on T_k as $\exp l_k\ (x)$, where $l_k\ (x)$ is a piecewise linear lower hull formed from the chords between adjacent abscissae in T_k. For $x \in \left[x_j, x_{j+1}\right], j = 1, 2, \ldots, k-1$,

$$l_k\ (x) = \frac{\left(x_{j+1} - x\right) h\left(x_j\right) + \left(x - x_j\right) h\left(x_{j+1}\right)}{x_{j+1} - x_j}.$$

For $x < x_1$ and $x > x_k$, we define $l_k\ (x) = -\infty$.

Note that the envelope and squeezing functions are piecewise exponential functions. The concavity of $h\ (x)$ ensures that $l_k\ (x) \leq h\ (x) \leq u_k\ (x)$ for all x in $\mathscr{X}$. To independently simulate n values from $\pi\ (x)$ by the adaptive rejection sampling, we perform the following steps until n values are accepted:

1. Initialisation step. Initialize the abscissae in T_k. If $\mathscr{X}$ is unbounded, make sure $h'\ (x_1) > 0$ and $h'\ (x_k) < 0$. Calculate the functions $u_k\ (x)$ and $l_k\ (x)$. Also calculate
$$s_k\ (x) = \frac{\exp\left(u_k\ (x)\right)}{\int_{\mathscr{X}} \exp\left(u_k\ (x)\right) dx}.$$

2. Sampling step. Sample a value x^* from $s_k(x)$ (a piecewise exponential distribution) and a value u from U (0, 1). Accept it if $u \le \exp\left[l_k(x^*) - u_k(x^*)\right]$. Otherwise, calculate $h(x^*)$ and $h'(x^*)$, accept it if $u \le \exp\left[h(x^*) - u_k(x^*)\right]$.
3. Updating step. If $h(x^*)$ and $h'(x^*)$ were evaluated at the sampling step, include x^* in T_k to form T_{k+1}, relabel the elements of T_{k+1} in ascending order, construct functions $u_{k+1}(x)$, $l_{k+1}(x)$ and $s_{k+1}(x)$ on the basis of T_{k+1}. Return to the sampling step if n values have not yet been accepted.

In a Gibbs sampler, the full conditional distribution of a particular parameter θ can be written as

$$h(\theta|\cdot) \propto \prod_j g_j\left(\theta_j|\Omega_j\right),$$

where $g_j(\theta_j|\Omega_j)$ is a function containing θ_j, and Ω_j is a set of other parameters and data. When $h(\theta|\cdot)$ is not a standard distribution but every $g_j(\theta_j|\Omega_j)$ is log-concave, we can apply the adaptive rejection sampling to $h(\theta|\cdot)$.

B.3 Slice Sampling

Slice sampling is another MCMC method. This was introduced by Neal (2003) and it is one of the building blocks of BUGS. Slice sampling simulates a value uniformly from underneath the pdf curve $\pi(x)$ without need to reject any points. Here we give a brief summary of how slice sampling works. The tth iteration of a slice sampling consists of the following three steps:

1. Draw a value y from U $\left(0, g\left(x^{t-1}\right)\right)$ (i.e., a vertical line under $g\left(x^{t-1}\right)$), where x^{t-1} is the ending value of $t-1$th iteration, and g is a function proportional to the target distribution $\pi(x)$. Define a horizontal slice $S = \{x : g(x) > y\}$.
2. Find a suitable interval I containing much of the slice S. Ideally, we can solve $g(x) > y$ and find the exact slice. But this is not always feasible. Generally, we use a "stepping out" procedure to find an interval containing much of the slice. We assume w as a typical length of a unit interval, m as an integer limiting the length of interval to mw.

   a. Randomly place a unit interval of length w around x^{t-1}. First choose a value u from U (0, 1), then set $L = x^{t-1} - wu$ and $R = L + w$. The interval (L, R) covers x^{t-1}.
   b. Expand the unit interval. Choose a value v from U (0, 1), then set the maximum number of unit intervals on the right side as the largest integral smaller than mv, denoted by J, and the maximum number of unit intervals in the left side as $K = m - 1 - J$. Calculate the ending points of the expanded interval as follows:

$$L = x^{t-1} - wu - wJ, \; R = L + w.$$

c. Adjust the interval. If $J > 0$ and $y < g(L)$, repeat set the new L as $L - w$ and the new J as $J - 1$ until $J = 0$ or $y > g(L)$; if $K > 0$ and $y < g(R)$, repeat set the new R as $R + w$ and the new K as $K - 1$ until $K = 0$ or $y > g(R)$. Return the final interval $I = (L, R)$.

3. Draw a new value x^t uniformly from S. Repeatedly draw a value uniformly from an interval which is initially equal to I but shrinks each time when a draw is not in the slice S, until a value is found within $S \cap I$. Note that the interval I found from "stepping out" procedure may overlap S.

Neal (2003) gave detailed proof of slice sampling which is not discussed here.

References

Neal, R. M. (2003). Slice sampling. *Annals of Statistics*, *31*, 705–741.

Robert, C., & Casella, G. (2013). *Monte Carlo statistical methods*. New York: Springer Science & Business Media.

Index

© Springer Nature Singapore Pte Ltd. 2018

G. Gao, *Bayesian Claims Reserving Methods in Non-life Insurance with Stan*,
https://doi.org/10.1007/978-981-13-3609-6