
Computat
Biology

Applied Bioinformatics and Biostatistics in Cancer Research

Series Editors

Jeanne Kowalski
John Hopkins University, Baltimore, MD, USA

Steven Piantadosi
Cedars Sinai Medical Center, Los Angeles, CA, USA

For other titles published in this series, go to
http://www.springer.com/series/7616

Tuan Pham
Editor

Computational Biology

Issues and Applications in Oncology

Springer

Editor
Tuan Pham
Associate Professor
School of Engineering and Information Technology
The University of New South Wales
Canberra, ACT 2600, Australia
t.pham@adfa.edu.au

ISBN 978-1-4419-0810-0 e-ISBN 978-1-4419-0811-7
DOI 10.1007/978-1-4419-0811-7
Springer New York Dordrecht Heidelberg London

Library of Congress Control Number: 2009935696

Printed on acid-free paper

Springer is part of Springer Science+Business Media (www.springer.com)

Preface

Computational biology is an interdisciplinary research that applies approaches and methodologies of information sciences and engineering to address complex problems in biology. With rapid developments in the *omics* and computer technologies over the past decade, computational biology has been evolving to cover a much wider research domain and applications in order to adequately address challenging problems in systems biology and medicine. This edited book focuses on recent issues and applications of computational biology in oncology. This book contains 11 chapters that cover diverse advanced computational methods applied to oncology in an attempt to find more effective ways for the diagnosis and cure of cancer.

Chapter 1 by Chen and Nguyen addresses an analysis of cancer genomics data using partial least squares weights for identifying relevant genes, which are useful for follow-up validations. In Chap. 2, Zhao and Yan report an interesting biclustering method for microarray data analysis, which can handle the case when only a subset of genes coregulates under a subset of conditions and appears to be a novel technique for classifying cancer tissues. As another computational method for microarray data analysis, the work by Lê Cao and McLachlan in Chap. 3 discusses the difficulties encountered when dealing with microarray data subjected to selection bias, multiclass, and unbalanced problems, which can be overcome by careful selection of gene expression profiles. Novel methods presented in these chapters can be applied for developing diagnostic tests and therapeutic treatments for cancer patients.

Ductal carcinoma in situ (DCIS) is known as the earliest possible clinical diagnosis of breast cancer and performed with screening mammography that has detected small areas of calcification in the breast. Chapter 4 by Macklin et al. presents a biophysics- and agent-based cellular model of DCIS, which is modular in nature and can be extended to incorporate more advanced biological hypotheses. Chapter 5 by Verma gives an overview of a state of the art for the classification of suspicious areas in digital mammograms and presents a multicluster class-based approach for classifying such areas in benign and malignant cases.

The work in Chap. 6 by To and Pham presents several methods using evolutionary computation algorithms for classification of oncology data. Evolutionary computation is effective in this study because it can offer efficiency in searching in high-dimension space, particularly in nonlinear optimization and hard optimization

problems. In Chap. 7, Solé et al. provide a thorough review of genetic association studies on SNP-array analysis techniques as well as many existing bioinformatics tools for carrying out such analyses.

Image analysis of cancer cells and tumors is an important research area in computational biology and bioinformatics. Image-based study enables an efficient way for gaining better understanding of the angiogenesis and genetic basis of cancer. Vallotton and Soon report in Chap. 8 several image analysis tools and techniques for the automatic identification of objects in image sequences and quantification of their dynamic behaviors. Chapter 9 by Tran and Pham reports several recent developments in cell classification for high-content screening, which can be useful computerized tools for automated analysis of large image data and for aiding the process of drug discovery.

Chapter 10 by Daskalaki et al. reviews important cellular processes for cancer onset, progression, and response to anticancer drugs and provides a summary of existing databases and tools for computational models in oncology. This chapter also presents several frameworks for modeling cancer-related signaling pathways. Finally, Le et al. discuss in Chap. 11 laser speckle imaging for real-time monitoring of blood flows and vascular perfusion; with proper experimental setups and quantitative analyses, this technology can offer its potential applications for research and development in diagnostic radiology and oncology.

Besides the availability of genomic data, life-science researchers study proteomics to gain insight into the functions of cells by learning how proteins are expressed, processed, recycled, and their localization in cells. Functional proteomics involves the use of mass spectrometry data to study the regulation, timing, and location of protein expression. Interaction studies seek to understand how proteins pair between themselves and other cellular components interact to constitute more complex models of the molecular machines. Chapter 12 by Jin et al. gives an overview of bioinformatics algorithms for biomarker discovery, validation, clinical application of proteomic biomarkers, and related biological backgrounds.

Many thanks to all the authors for their timely effort in contributing chapters to this edited book. Gratitude is expressed to Rachel R. Warren, the Editor of Cancer Research, Springer Science, along with Jeanne Kowalski at Sidney Kimmel Cancer Center at Johns Hopkins and Steven Piantadosi at the Cedars-Sinai Medical Center, in acknowledgment of their invitation, encouragement, and support in making this work a valuable contribution to the endeavor of exploring advanced mathematics, statistics, computer science, information technology, and engineering computation for solving challenging problems in oncology.

Canberra, Australia *Tuan Pham*

Contents

Chapter 1
Identification of Relevant Genes from Microarray Experiments based on Partial Least Squares Weights: Application to Cancer Genomics

Ying Chen and Danh V. Nguyen

Abstract In microarray genomics expression data obtained from hybridization of different cancer tissue samples or samples from cancer and normal cellular conditions, it is of interest to identify differentially expressed genes for follow-up validation studies. One approach to the analysis of genomics expression data is to first reduce the dimensionality using partial least squares (PLS), which has been useful in various cancer microarray data applications (Nguyen and Rocke, Bioinformatics 18:39–50, 2002a). PLS involves reducing the dimensionality of the gene expression data matrix by taking linear combinations of the genes/predictors (referred to as PLS components). However, the weights assigned to each gene and at each dimension are nonlinear functions of both the genes and outcome/response variable, making analytical studies difficult. In this paper, we propose a new measure for identifying relevant genes based on PLS weights called random augmented variance influence on projection (RA-VIP). We compare the relative performance of RA-VIP in terms of its sensitivity and specificity for identifying truly informative (differentially expressed) genes to two previously suggested heuristic measures, both based on PLS weights, namely the variable influence on the projection (VIP) and the PLS regression B-coefficient (denoted B-PLS). The methods are compared using simulation studies. We further illustrate the proposed RA-VIP measure on two microarray cancer genomics data sets involving acute leukemia samples and colon cancer and normal tissues.

1.1 Introduction

Partial least squares (PLS), introduced as a latent variable modeling technique in econometrics by Wold (1966), has been found to be a versatile technique in numerous application areas, particularly with high-dimensional genomics expression data

D.V. Nguyen (✉)
Division of Biostatistics, University of California School of Medicine, Davis, CA, USA
e-mail: ucdnguyen@ucdavis.edu

T. Pham (ed.), *Computational Biology: Issues and Applications in Oncology*,
Applied Bioinformatics and Biostatistics in Cancer Research,
DOI 10.1007/978-1-4419-0811-7_1,

(Boulesteix 2004; Nguyen and Rocke 2002a). PLS applications typically involve reducing the dimensionality of the predictor data matrix by taking linear combinations of the predictor variables/genes. (The linear combinations are referred to as PLS components.) However, unlike linear dimension reduction, such as principal components analysis, the weights or coefficients in the PLS components involve nonlinear functions of the response variable(s). (See Nguyen and Rocke (2002a,b, 2004) for applications in cancer classification based on gene expression profiles and characterizations of the nonlinear structure of PLS weights.) This makes analytical characterization intractable. Numerical studies may shed light on the utility of PLS weights in the dimension reduction process for specific applications. Motivated by applications in microarray gene expression data, one of our primary aims in this work is to systematically examine the utility of the PLS weights for identification/selection of relevant genes in PLS dimension reduction.

Two measures indicating the relative contribution of each variable to the PLS dimension reduction, both based on PLS weights, are (1) the variable influence on the projection (VIP) and (2) the PLS (regression) B-coefficient (denoted B-PLS); see Wold et al. (1993) and also SAS PROC PLS (SAS Institute, Inc., Cary, NC). We study the ability of selection rules based on VIP and B-partial least squares (B-PLS) values assigned to each gene as a basis for selecting relevant genes in microarray data. Additionally, we propose and study an alternative measure called random augmentation VIP (RA-VIP). The RA-VIP measure is based on estimating the "null distribution" of VIP values and assigning a p value to each gene for selection. In the current work, we focus on the binary classification data or two-group comparison case, where the outcome is a vector of indicators for two groups, namely $\mathbf{y} = (y_1, \ldots, y_n)'$ with $y_i = 0$ or 1, and $\mathbf{X}$ is an $n \times p$ gene expression matrix for p genes with $n \ll p$.

In applications heuristic rules are suggested for ranking each variable based on VIP and B-PLS coefficients and variable selection are based on the resultant ranking. For example, Wold et al. (1993) suggest that a VIP value greater than 1 is an indication of a relevant or important variable, while a VIP value less than 0.8 is an indication of an irrelevant or unimportant variable. With respect to B-PLS coefficients, variables with small absolute B-PLS coefficients are considered uninformative and variables with coefficients larger than about half the maximum of the B-PLS values are considered to be of interest. Although these heuristic rules are based on experience with high-dimensional data mostly in chemometrics, drug discovery, and spectrometric calibration applications, the suggested VIP and B-PLS thresholds are somewhat arbitrary and their usefulness specifically to genomics data requires evaluation. Our proposed selection of variables based on RA-VIP measure provides a more formal selection of relevant genes/variables based on p values relative to the null distribution. Furthermore, in this work we evaluate the performance of the proposed RA-VIP measure and compare its performance relative to the above informal rules based on VIP and B-PLS coefficients using simulation studies.

Our work here is organized as follow. In Sect. 1.2, we describe the measures VIP and B-PLS and provide details on the variable selection rules based on them. The proposed method RA-VIP is also presented there. In Sect. 1.3 we describe

two simulation models to evaluate the relative performance of the three measures, namely VIP, B-PLS, and RA-VIP, to select truly informative genes. The first set of simulation studies is based on a normal model for gene expression data with cluster-specific correlation. The second set of simulation studies is based on resampling a colon cancer and normal tissue microarray data, using the observed (realistic) correlation structure of the data. The simulation results, summarizing the sensitivity and specificity of the three variable selection measures, are given in Sect. 1.4. Applications of the proposed RA-VIP measure to an acute leukemia data and a colon tissue microarray data are summarized in Sect. 1.5. We conclude with a brief discussion in Sect. 1.6.

1.2 Methods

1.2.1 Partial Least Squares Dimension Reduction

Let $\mathbf{y} = (y_1, \ldots, y_n)'$ be the vector of binary response variable representing classes of biological samples or clinical outcomes, $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_p]$ be a $n \times p$ matrix of gene expression values, and $\mathbf{x}_j = (x_{1j}, \ldots, x_{nj})'$ is the vector of expression values for gene j across the n samples. PLS achieves dimension reduction by constructing orthogonal linear combination of the original variables $(X_1, \ldots, X_p)$ that maximize the covariance between the response variable $(\mathbf{y})$ and linear combinations of the predictor variables $(\mathbf{X})$. More precisely, denote the kth PLS component as the linear combination $\mathbf{t}_k = \mathbf{X}\mathbf{w}_k$, where $\mathbf{w}_k = (w_{1k}, \ldots, w_{pk})'$ is the kth PLS weight vector. The vector $\mathbf{w}_k$ contains individual weights assigned to each gene j, $1 \le j \le p$, at the kth dimension. More precisely, the vector of PLS weights for dimension k is obtained as

$$\mathbf{w}_k = \arg\max_{\mathbf{w}} \ \mathrm{Cov}^2(\mathbf{y}, \mathbf{X}\mathbf{w}) = \arg\max_{\mathbf{w}} \ \mathbf{w}'\mathbf{X}'\mathbf{y}\mathbf{y}'\mathbf{X}\mathbf{w},$$

where $\mathbf{w}_k{}'\mathbf{w}_k = 1$ and $\mathbf{t}_k{}'\mathbf{t}_j = 0$, for $j < k$, $k = 1, \ldots, A$ and $A < \mathrm{rank}(\mathbf{X})$. Typically the number of dimensions retained, A, is small (e.g., 1–5) in practice with gene expression data. The matrix of PLS components can be more succinctly expressed as $\mathbf{T} = [\mathbf{t}_1, \mathbf{t}_2, \ldots, \mathbf{t}_A] = \mathbf{X}\mathbf{W}$, where $\mathbf{W} = [\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_A]$. See Nguyen and Rocke (2002a) for details and also references therein. Also, see Boulesteix and Strimmer (2006) for a review of PLS applications in genomics and bioinformatics.

1.2.2 Variable Selection Measures as Functions of PLS Weights

The kth PLS component $\mathbf{t}_k$, $k = 1, \ldots, A$, is a linear combination of $\mathbf{X}$: $\mathbf{t}_k = \mathbf{X}\mathbf{w}_k = w_{1k}\mathbf{x}_1 + w_{2k}\mathbf{x}_2 + \cdots + w_{pk}\mathbf{x}_p$. Thus, the coefficients $\{w_{jk}\}_{j=1}^{p}$ can be

interpreted as the relative contribution of each gene/variable in dimension k (i.e., to PLS component k). One sensible measure of the overall contribution of gene j to the PLS dimension reduction is to combine the weights for gene j across the A-retained PLS dimensions. We briefly describe two previously proposed measures, called VIP and B-PLS, in Sects. 1.2.3 and 1.2.4, respectively. We then propose a new alternative measure, based on VIP, called random augmented VIP (RA-VIP) in Sect. 1.2.5. These variable selection measures are all functions of the PLS weights across the A PLS dimensions.

1.2.3 Variable Influence Projection in Partial Least Squares

The VIP measure (Wold et al. 1993) is a weighted sum of the normalized PLS weights across the A PLS dimensions. More precisely, the VIP for gene j is

$$\mathrm{VIP}_j = \left\{ p \sum_{k=1}^{A} v_k^* (w_{jk}/\|\mathbf{w}_k\|)^2 \right\}^{1/2}, \qquad j = 1, \ldots, p, \tag{1.1}$$

where $v_k^* = \mathrm{SSY}_k / \sum_{k=1}^{A} \mathrm{SSY}_k$, SSY_k is the amount of variance of Y explained by the PLS component $\mathbf{t}_k$. The denominator in v_k^* is the total response variation explained by all A PLS components. Note that $p^{-1} \sum_{i=1}^{p} \mathrm{VIP}_j{}^2 = 1$, so that the sum of squared VIP values for all genes sum to p. We note that in the VIP_j measure, the contribution of the gene from dimension k is weighted by the relative amount of response variation explained by PLS component $\mathbf{t}_k$, namely v_k^*. A heuristic rule was suggested by Wold et al. (1993) that a VIP value greater than 1 is an indication of a relevant or important variable, while a VIP value less than 0.8 is an indication of an irrelevant or unimportant variable. We evaluate the relative performance of this selection rule in the context of high-dimensional microarray data in the simulation studies of Sect. 1.3.

1.2.4 B-Partial Least Squares (B-PLS) Regression Coefficient

Another relative measure based on PLS weights, denoted as B-PLS, was also proposed in Wold et al. (1993) to assess the relevance of the predictor variables. B-PLS is simply the PLS regression coefficient given by

$$\mathbf{B} = \mathbf{W}(\mathbf{P}'\mathbf{W})^{-1}(\mathbf{T}'\mathbf{T})^{-1}\mathbf{T}'\mathbf{y}, \tag{1.2}$$

where $\mathbf{P}$ is the loading matrix (in the decomposition $\mathbf{X} = \mathbf{T}\mathbf{P}' + \mathbf{E}$ with $\mathbf{E}$ denoting the X-residual matrix; see e.g., Helland (1988)). If the number of PLS components

are chosen to be p, which is the same as the dimension of original X matrix, the coefficient $\mathbf{B} = (B_1, \ldots, B_p)'$ is just the vector of coefficients from the ordinary least squares regression of Y on $\{X_j\}_{j=1}^{p}$. Predictor variables with small absolute B-PLS coefficients (e.g., $|B_j| < 0.5\max_{1\le j\le p}\{|B_j|\}$) are considered uninformative and coefficients larger than about half the maximum of the B_j values are considered to be relevant variables (Wold et al. 1993).

1.2.5 Random Augmentation VIP

The heuristic rules for identifying relevant variables in PLS dimension reduction, based on B-PLS regression coefficients and VIP summarized above are easy to compute as they are simply based on quantities that are direct by-products of the PLS algorithm. Although based on empirical observations in practice, the heuristic rules for variable selection based on B-PLS and VIP may still appear somewhat arbitrary. We propose an alternative approach to identify relevant genes by more formally comparing the observed average VIP for each gene to the null distribution of VIP when the expression data matrix ($\mathbf{X}$) is uncorrelated with the outcome ($\mathbf{y}$). To estimate the null VIP distribution, the predictor matrix $\mathbf{X}$ is augmented with a set of random "noise" variables. The set of random noise variables are created by randomly permuting the values in each gene variable, respectively. Thus the new augmented gene expression matrix (predictor matrix) consists of original gene variables along with their randomly permuted versions. PLS dimension reduction is then applied to the augmented predictor matrix defined as $\mathbf{Z} = [\mathbf{X}, \mathbf{X}^*]$, where $\mathbf{X}^*$ is the randomly permuted (noise) version of $\mathbf{X}$. The null VIP distribution is then used to obtain a p value for each gene which can then be used for identifying potential relevant genes. The proposed procedure, which we refer to as random augmentation VIP (RA-VIP), is given in more details below:

1. Randomly permute each variable in $\mathbf{X}$ to get corresponding random noise predictor matrix $\mathbf{X}^*$.
2. Create augmented predictor data matrix $\mathbf{Z} = [\mathbf{X}, \mathbf{X}^*]$ of size $n \times 2p$.
3. Apply PLS dimension reduction using $\mathbf{Z}$ and response vector $\mathbf{y}$ and compute $(\mathrm{VIP}_1, \ldots, \mathrm{VIP}_p)$ and $(\mathrm{VIP}_1^*, \ldots, \mathrm{VIP}_p^*)$ corresponding to genes $(X_1, \ldots, X_p)$ and noise genes $(X_1^*, \ldots, X_p^*)$, respectively.
4. Repeat steps 1–3 L times producing $\{\mathrm{VIP}_{jl}, j = 1, \ldots, p\}_{l=1}^{L}$ and $\{\mathrm{VIP}_{jl}^*, j = 1, \ldots, p\}_{l=1}^{L}$.
5. Define the observed (average) VIP value for gene j by $\mathrm{VIP}_{j,\mathrm{Obs}} = L^{-1} \sum_{l=1}^{L} \mathrm{VIP}_{jl}$, for $j = 1, \ldots, p$.
6. Compute the p value for gene j as: $p_j = \sum_{l=1}^{L} \#\{\nu : \mathrm{VIP}_\nu^* > \mathrm{VIP}_{j,\mathrm{Obs}}, \nu = 1, \ldots, p\}/(pL)$, for $j = 1, \ldots, p$.

1.3 Design of Simulation Studies

We consider a simulation design to assess the performance of the proposed random augmentation PLS for variable selection and compare to the methods described in Sect. 1.2. We focus here on a binary outcome. Define the vector of binary outcome $\mathbf{y} = (0, \ldots, 0, 1, \ldots, 1)'$ and the matrix of gene expression data $\mathbf{X} = (\mathbf{X}_1|\mathbf{X}_2)_{n\times p}$, where n is the number of samples, p is the number of genes, $\mathbf{X}_1$ is the $n \times d$ expression matrix corresponding to d truly *informative* genes (i.e., genes differentially expressed between the two outcome classes/groups), and $\mathbf{X}_2$ is the expression matrix for the remaining $p-d$ *uninformative* genes. Because naturally there are groups of up- and down-regulated genes when monitoring gene expression globally, we further divide the d informative genes into two gene clusters. Genes in the first cluster are up-regulated in samples for group $Y = 0$ while the genes in the second cluster are up-regulated in samples for group $Y = 1$. The remaining uninformative genes do not differ across groups on average, although data with substantial noise variability can render identification of the d informative genes difficult.

We considered several design parameters in the simulation studies, including (a) the proportion of informative genes, (b) the level of difficulty in identification task [signal-to-noise ratio (SNR)], and (c) the overall strength of the correlation among genes. The number of truly informative (relevant) genes were set to 10, 50, 100, and 150 corresponding to 1%, 5%, 10%, and 15% of a total of $p = 1{,}000$ genes, respectively. Variability in performance of the methods are examined with respect to these main factors.

A challenge with designing a simulation model generally is the incorporation of a sensible correlation structure among genes. A simple correlation structure within a specific probability model will allow for more precise assessment of the effects of specific factors, such as the proportion of truly informative genes or the overall strength of the correlation. However, these advantages in assessment and control of experimental factors on the performance of the methods must be balanced with a correlation structure/model that reflects the observed/empirical correlation structures with real data. Therefore, we considered two simulation models. In the first model and associated set of simulations, we designed a simple correlation structure in a multivariate normal probability model where informative genes have the same correlation within cluster and uninformative genes are uncorrelated. A second set of simulations, based on resampling the real data, preserves the observed, more complex, correlation structure. Details specific to these two simulation models are further described in Sects. 1.3.1 and 1.3.2.

To evaluate the performance of various methods we used both sensitivity (Sen) and specificity (Spec). Sensitivity is the proportion of true informative genes correctly identified and 1 − Spec is the proportion of uninformative genes incorrectly identified as informative. Methods with larger Sen and smaller 1 − Spec are considered to have better overall performance. See Table 1.1 for a summary of these measures.

Table 1.1 Sensitivity is Sen $= a/(a+c)$, specificity is Spec $= d/(b+d)$, and $1 - \text{Spec} = b/(b+d)$.

	Informative genes	Uninformative genes
Genes identified as informative	a	b
Genes identified as uninformative	c	d

1.3.1 Simulation Based on Normal Model with Cluster-Specific Correlation

We assume a normal model for suitably transformed (and normalized) gene expression data. Denote the true group-specific mean expression levels and standard deviations for gene j by $\mu_{j1} = E(X_j|Y = 0)$, $\mu_{j2} = E(X_j|Y = 1)$, $\sigma_{j1} = \{\text{Var}(X_{j1}|Y = 0)\}^{1/2}$, and $\sigma_{j2} = \{\text{Var}(X_{j2}|Y = 1)\}^{1/2}$. The SNR of gene j is defined as:

$$\text{SNR}_j = \frac{\mu_{j1} - \mu_{j2}}{\sigma_{j1} + \sigma_{j2}}, \qquad j = 1, 2, \ldots, p. \tag{1.3}$$

We used a real colon tissue microarray data set, described in more details in Sect. 1.3.2, to set the model parameters μ_{jk} and σ_{jk} (for $k = 1, 2$, $j = 1, \ldots, p$). More precisely, the mean for gene j in group 1, μ_{j1}, is assigned the corresponding sample mean and similarly for the standard deviation parameters. Given a SNR value in (1.3), the true mean parameter μ_{j2} is then computed from (1.3). To study dependence of the methods on the SNR, we considered three distributions for SNR corresponding to low, moderate and high mean SNR levels. The SNR levels are obtained from a normal distribution with parameters $(\mu_{SNR}, \sigma_{SNR}) = (0.5, 0.5/4)$, $(1.0, 1.0/4)$, or $(1.5, 1.5/4)$ corresponding to low, moderate, and high average levels of SNR, respectively. Here μ_{SNR} and σ_{SNR} denote the mean and standard deviation parameters of normal distribution from which SNR_j were generated. (These parameters were chosen after a careful study of the distributions of the observed/estimated signal to noise ratio corresponding to the leukemia and colon data sets described later in Sect. 1.5.)

Next, the sets of informative genes, consisting of both up- and down-regulated genes, are selected (randomly once). Denote the sets of informative up- and down-regulated genes by G_1 and G_2, respectively. For a simple model, we take the mean absolute SNR of down-regulated genes to be the same as up-regulated genes. Furthermore, we assume that genes in different clusters are uncorrelated and genes in the same cluster have the same correlation. More specifically, the cluster-specific correlation structure (within gene cluster G_k) is

$$\mathbf{R}_k = \begin{bmatrix} 1 & \rho_k & \cdots & \rho_k \\ \rho_k & 1 & \cdots & \rho_k \\ \vdots & \vdots & \ddots & \vdots \\ \rho_k & \rho_k & \cdots & 1 \end{bmatrix}_{d_k \times d_k}, \qquad k = 1, 2$$

where ρ_k is the common correlation coefficient in cluster G_k, $d_k = |G_k|$ with $d = d_1 + d_2$. The remaining uninformative genes are assumed to be uncorrelated. Thus, the correlation structure of all p genes is

$$\mathbf{R} = \begin{bmatrix} \mathbf{R}_1 & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{R}_2 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{I}_{p-d} \end{bmatrix},$$

where $\mathbf{I}_a$ denotes the identity matrix of size a. The data is generated from a multivariate normal model with mean $\{\mu_{jk}\}$ and with correlation matrix $\mathbf{R}$. To reflect the small sample size, relative to the number of genes, in microarray data, the simulated data sets are obtained with size $n = 60$ and $p = 1{,}000$. For each simulation configuration (proportion of truly informative genes, SNR, and correlation level), $N = 300$ Monte Carlo data sets were generated.

1.3.2 Resampling-Based Simulation from Real Data

Our second simulation model is based on resampling the original (real) gene expression matrix and preserves the observed correlation matrix among the p genes. As an example, we use a colon tissue microarray data set from Alon et al. (1999). This data set contains the expressions of $p = 2{,}000$ genes on 62 tissues samples hybridized to Affymetrix Hum6000 array. Among the $n = 62$ samples, $n_1 = 40$ are from colon cancer and $n_2 = 22$ are from normal tissues. The GLog Average (GLA) expression index based on the generalized logarithm of the perfect match probes (Zhou and Rocke 2005) was used.

This resampling procedure involves three main steps (1) removing the potential observed differences between groups in the original data, (2) resampling from the data after the first step, and (3) randomly selecting informative genes by adding a mean difference between the two groups that is proportional to the observed standard deviation for that specific gene j, $\delta\widehat{\sigma}_j$. Note that although mean differences between the two groups are removed, the correlation structure is preserved due to invariance to linear transformation. These steps, in more details, are:

- Remove systematic differences between groups for each gene $j = 1, \ldots, p$. This is obtained by subtracting the group-specific mean for each gene:

$$x_{ijk}^{\mathrm{C}} = x_{ijk} - \widehat{\mu}_{jk} + \widehat{\mu}_j, \qquad k = 1, 2,$$

where x_{ijk} is the original expression value for gene j from sample i in group k, $\widehat{\mu}_{jk}$ is the sample mean of gene j in group k, and $\widehat{\mu}_j$ is the sample mean for gene j over the combined groups. Thus, there is no difference in mean expression between groups for all genes and the original (observed) correlation structure is preserved based on data matrix $\mathbf{X}^{\mathrm{C}}$ (with (i, j)th entry given by x^{C}_{ijk}).

- Observations in $\mathbf{X}^{\mathrm{C}}$ are resampled with replacement N times, resulting in N resampled/simulated gene expression data matrices $\mathbf{X}^{(1)}, \ldots, \mathbf{X}^{(N)}$, each of size $n \times p$.
- A fixed number d of informative genes (selected randomly once) are chosen by subtracting $\delta(\widehat{\sigma}_{j1} + \widehat{\sigma}_{j2})$ from informative gene $j = 1, \ldots, d$ in group 2 (corresponding to $Y = 1$). Thus, the standardized effect size for informative gene j is δ, which ranges from 0.5 to 1.5 as for the SNR described above for the normal distribution model. As before, the proportion of informative genes ranges from 1 to 15%.

1.4 Simulation Results

1.4.1 Result Based on Normal Model with Cluster-Specific Correlation

Based on the normal model described in Sect. 1.3.1, 300 simulated data sets were generated. From preliminary simulation studies (not shown) the effect of the overall correlation strength among genes may be a factor of interest, although the results are more dominated by two other factors (1) the proportion of true informative genes and (2) the signal-to-noise ratio (SNR) of relevant genes in the normal data model. Thus, we focus here on the results for studies that fixed the maximum correlation among relevant genes (which was set to be a maximum of 0.8).

Figure 1.1 displays the relative performance based on VIP, B-PLS, and the proposed RA-VIP in terms of sensitivity and specificity. Note that high sensitivity and low 1-specificity is desirable. For the popular measure VIP, as the proportion of truly informative genes increases (from 1 to 15%), the specificity improves (i.e., 1-specificity decreases). This is apparent across all levels of SNR, although more so in the weak SNR case. However, sensitivity appears uniformly high for gene identification based on VIP. Thus, overall, this suggests that VIP has better performance when there are more truly informative genes in the data set relative to the total p genes. One such example in practice is the case where cancer tissues are compared to normal tissues (or radiation treatment) in microarray studies, where one would expect more broader/global changes in the global expression of the repertoire of genes. In addition, when the SNR becomes weaker, i.e., the gene identification task becomes more difficult, the specificity of VIP is poor, as expected.

Interestingly, selection based on B-PLS coefficients has high specificity in all three cases of weak, moderate, and strong SNR. However, the sensitivity of B-PLS

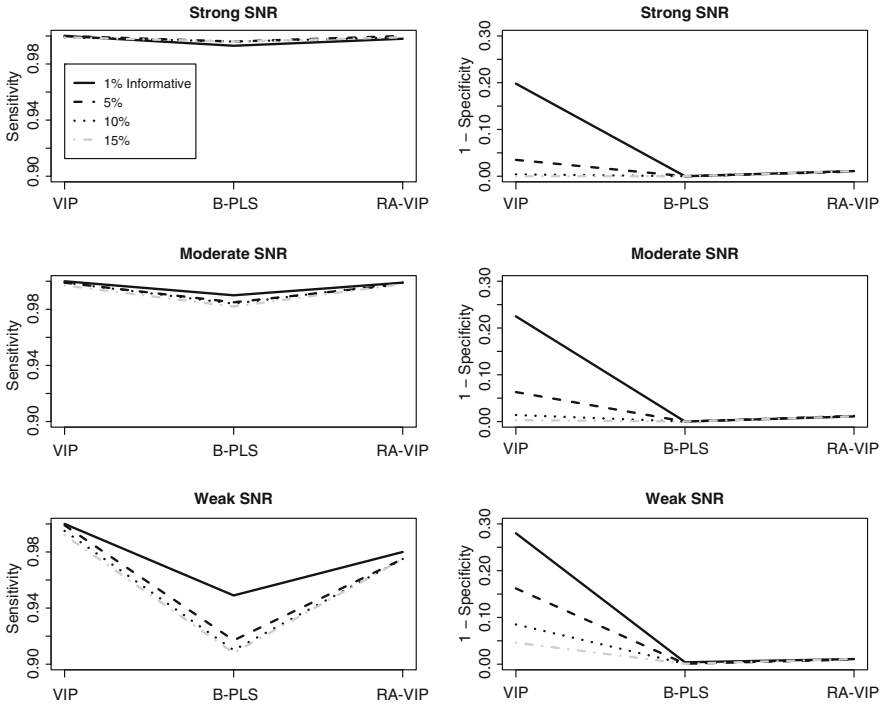

Fig. 1.1 The sensitivity and 1-specificity of VIP, B-PLS, and RA-PLS gene selection methods based on a normal data model with cluster-specific correlation. Plotted values are averaged over 300 Monte Carlo data sets of size 60 × 1,000 and for each proportion of truly informative genes between 1 and 15%

deteriorates noticeably relative to VIP (and RA-VIP) when the SNR becomes weak. See the first column of Fig. 1.1. Thus, these results suggest that B-PLS coefficients perform favorably with respect to specificity while VIP performs favorably in terms of sensitivity.

From Fig. 1.1, it can be seen that the proposed method, based on RA-VIP, performs well overall with respect to both sensitivity and specificity. Thus, RA-VIP appears to be able to balance both criterions, which are important in gene selection/identification. RA-VIP has relatively high specificity that is competitive with B-PLS and high sensitivity that is competitive with VIP. This observed overall superior performance of RA-VIP appears to hold across all levels of SNR (from weak to strong) and the proportion of truly informative genes. This later point has important implications in practice, because one does not know a priori how many genes are truly informative and the strength/level of gene expression relative to background noise.

1.4.2 Resampling-Based Simulation Result

Although the above simulation study, utilizing a normal model with cluster-specific correlation, provided some insights on the performance of the three approaches to selection of informative genes based on PLS weights (namely VIP, B-PLS, and RA-VIP), the correlation structure among the genes is not fully modeled. Postulating a sensible correlation structure among the thousands of genes is a difficult task. Thus, we consider an alternative resampling-based simulation approach to study the relative performance of the three measures that preserves the observed correlation structure. This approach allows for a more realistic model of the correlation structure. For illustration, we use the observed correlation structure of the colon (cancer/normal) tissue data set as a model (see Sect. 1.3.2). The parameter settings of this resampling-based simulation are similar to the normal model and we also used 300 resampled data sets. The relative performances of the methods in terms of sensitivity and specificity are summarized in Fig. 1.2. The results suggest that under a real correlation structure among genes, the above conclusions regarding the

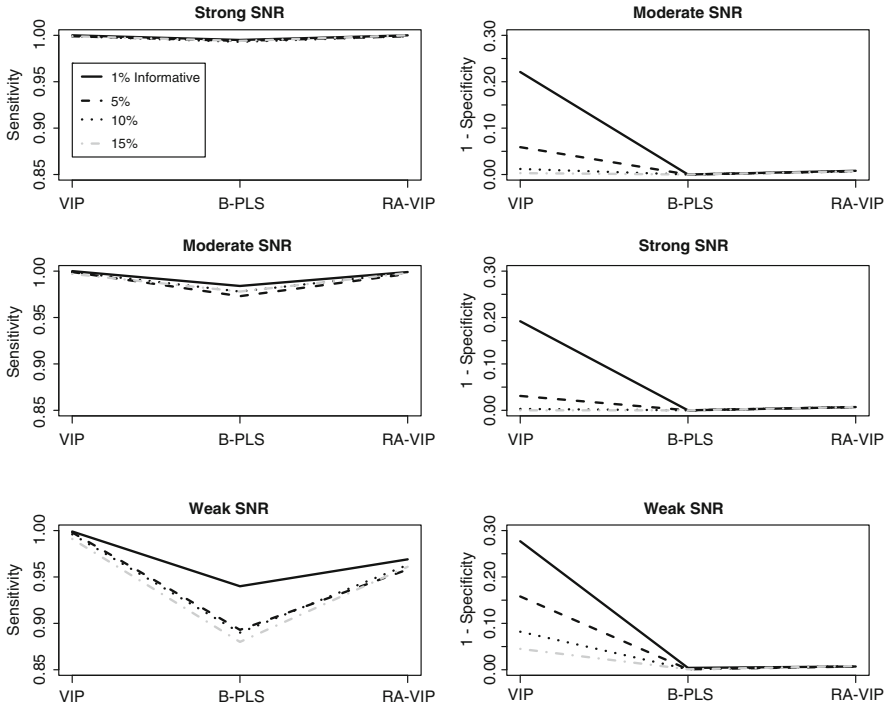

Fig. 1.2 The sensitivity and 1-specificity of VIP, B-PLS, and RA-PLS gene selection methods based on real correlation structure among genes in the colon/normal tissues data set. Plotted values are averaged over 300 resampling data sets preserving the original observed correlation matrix among genes in the colon/normal data set (of size $62 \times 2{,}000$. See also Fig. 1.1 caption for details

relative performances of VIP-PLS, B-PLS, and RA-VIP (for the more simplified correlation structure in a normal model) also hold. These results, taken together with the simulation results above in Sect. 1.4.1, indicate that RA-VIP is a reliable measure to identify relevant genes. In Sect. 1.5, we apply the RA-VIP measure to select relevant genes in two cancer microarray gene expression data sets.

1.5 Applications to Microarray Gene Expression Data

We apply the proposed RA-VIP measure to identify relevant genes in two publicly available microarray gene expression data sets. The first data set is the well-known acute leukemia data set (Golub et al. 1999) using Affymetrix high density oligonucleotide array Hu6800 (HuGeneFL). The original data set contains the expressions of 7,129 genes on 72 cases, consisting of 25 acute myeloid leukemia (AML) and 47 acute lymphoblastic leukemia (ALL) cases. For the second illustrative example, we also apply the proposed measure based on PLS weights to a colon data set (Alon et al., 1999) consisting of expression profiles on 2,000 genes from 40 colon cancers and 22 normal tissue samples. The experiment used Affymetrix Hum6000 array.

Although there are numerous proposed indices of gene expression for Affymetrix data, it is not the focus of the current study (see, e.g., Li and Wong 2001; Irizarry et al. 2003). In the current work, we choose the GLog Average (Zhou and Rocke 2005) algorithm to preprocess the raw intensity data to get an expression index for each gene, although alternative expression indices can be used. Table 1.2 summarizes the number of genes selected based on the RA-VIP measure for p value thresholds ranging from $p = 0.001$ to $p = 0.05$. We present the results for the number of PLS components ranging from 1 to 4 components and note

Table 1.2 Gene selection based on RA-VIP for the leukemia and colon microarray data sets for various p value threshold ranging from 0.001 to 0.05

Data set (n_1, n_2)	No. of PLS components (A)	p Value threshold			
		0.001	0.005	0.01	0.05
Leukemia (25 AML, 47 ALL)	1	427	667	812	1,362
	2	418	663	796	1,443
	3	439	644	804	1,434
	4	412	665	821	1,441
Colon (22 cancer, 40 normal)	1	72	125	169	346
	2	29	57	100	277
	3	26	57	101	277
	4	31	62	107	283

Given are the number of genes selected by RA-VIP measure for each p value threshold and PLS dimension

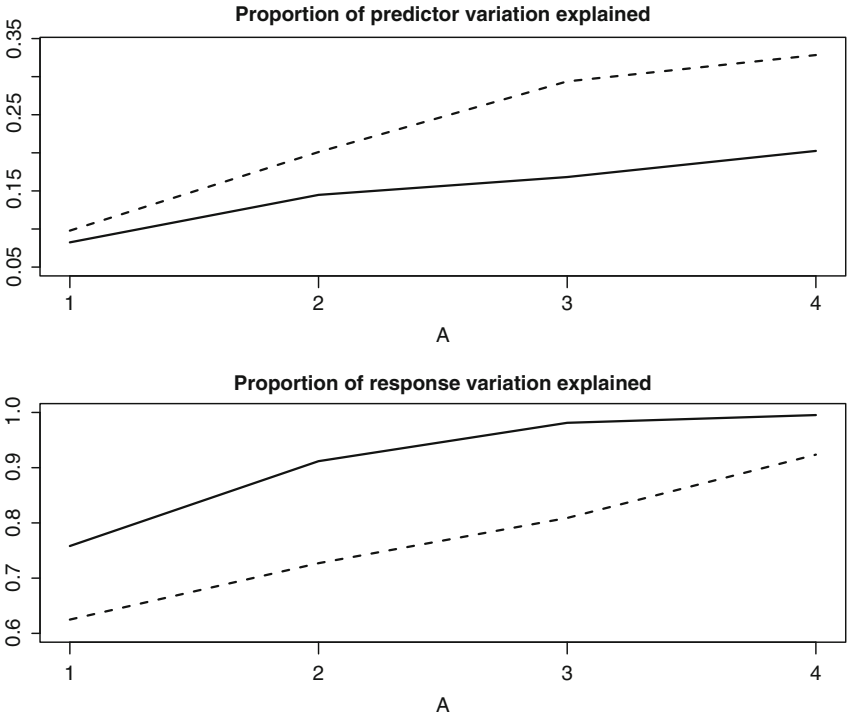

Fig. 1.3 Proportion of predictor and response variation explained for the leukemia (*solid line*) and colon (*dashed line*) data sets

that the set of genes identified/selected depends on the number of PLS dimensions/components (A). Cross validation can used to choose the number of PLS components, A, although the use of cross validation in this context will require further evaluation. Descriptively, one can also choose A based on the proportion of predictor and response variation explained. For example, Fig. 1.3 provides the proportion of variation explained for $A = 1, 2, 3, 4$ for both the leukemia and colon data sets. At $A = 3$ the response variation explained is high (0.81/0.98 for colon/leukemia data). The predictor variation explained is relatively low for both data sets, although the trend in the proportion of predictor variation explained appears to be flattening out. For illustration we provide the top 40 genes selected based on RA-VIP p values for $A = 3$ in Tables 1.3 and 1.4 for the leukemia and colon data sets, respectively.

1.6 Discussion

Partial least square, introduced by Herman Wold in 1966 as a latent variable modeling approach, has been useful as a regression modeling technique in chemometrics over the last several decades. Nguyen and Rocke (2002a) viewed PLS as a

Table 1.3 Top 40 genes selected based on RA-VIP for the acute leukemia (AML/ALL) microarray data set ranked by p value

Probe/gene	Name/description
AB002559_at STXBP2	Syntaxin binding protein 2
AF009426_at C18orf1	Chromosome 18 open reading frame 1
D10495_at PRKCD	Protein kinase C, delta
D14658_at SPCS2	Signal peptidase complex subunit 2 homolog (*S. cerevisiae*)
D14664_at CD302	CD302 molecule
D21262_at NOLC1	Nucleolar and coiled-body phosphoprotein 1
D26579_at ADAM8	ADAM metallopeptidase domain 8
D29963_at CD151	CD151 molecule (Raph blood group)
D42043_at RFTN1	Raftlin, lipid raft linker 1
D49950_at IL18	Interleukin 18 (interferon-gamma-inducing factor)
D50918_at 39697	Septin 6
D63874_at HMGB1	High-mobility group box 1
D86479_at AEBP1	AE binding protein 1
D86967_at EDEM1	ER degradation enhancer, mannosidase alpha-like 1
D86970_at TIAF1/ MYO18A	TGFB1-induced anti-apoptotic factor 1 / myosin XVIIIA
D87076_at PHF15	PHD finger protein 15
D88270_at VPREB1	Pre-B lymphocyte gene 1
D88422_at CSTA	Cystatin A (stefin A)
D89667_at PFDN5	Prefoldin subunit 5
HG1612-HT1612_at	–
HG2788-HT2896_at	–
HG2855-HT2995_at	–
HG3254-HT3431_at	–
HG3494-HT3688_at	–
J03473_at PARP1	Poly (ADP-ribose) polymerase family, member 1
J03589_at UBL4A	Ubiquitin-like 4A
J03798_at SNRPD1	Small nuclear ribonucleoprotein D1 polypeptide 16 kDa
J04615_at SNRPN/ SNURF	Small nuclear ribonucleoprotein polypeptide N / SNRPN upstream reading frame
J04990_at CTSG	Cathepsin G
J05243_at SPTAN1	Spectrin, alpha, nonerythrocytic 1 (alpha-fodrin)
K01396_at SERPINA1	Serpin peptidase inhibitor, clade A (alpha-1 antiproteinase, antitrypsin), member 1
K01911_at NPY	Neuropeptide Y
L05148_at ZAP70	Zeta-chain (TCR) associated protein kinase 70 kDa
L07633_at PSME1	Proteasome (prosome, macropain) activator subunit 1 (PA28 alpha)
L08246_at MCL1	Myeloid cell leukemia sequence 1 (BCL2-related)
L09717_at LAMP2	Lysosomal-associated membrane protein 2
L13278_at CRYZ	Crystallin, zeta (quinone reductase)
L19437_at TALDO1	Transaldolase 1
L20010_at HCFC1	Host cell factor C1 (VP16-accessory protein)
L21954_at TSPO	Translocator protein (18 kDa)

All p values < 0.00005

Table 1.4 Top 40 genes selected based on RA-VIP for colon cancer/normal tissue microarray data set ranked by p value

Probe/gene name/description	p Value
H20709 3′ UTR 1 173155 MYOSIN LIGHT CHAIN ALKALI, SMOOTH-MUSCLE ISOFORM	0.000588235
J03544 gene 1 Human brain glycogen phosphorylase mRNA, complete cds	0.000588235
M76378 gene 1 Human cysteine-rich protein (CRP) gene, exons 5 and 6	0.000588235
M63391 gene 1 Human desmin gene, complete cds	0.000588235
Z50753 gene 1 *Homo sapiens* mRNA for GCAP-II/uroguanylin precursor	0.000588235
R87126 3′ UTR 2a 197371 MYOSIN HEAVY CHAIN, NONMUSCLE (*Gallus gallus*)	0.000588235
X12671 gene 1 Human gene for heterogeneous nuclear ribonucleoprotein (hnRNP) core protein A1.	0.000588235
H43887 3′ UTR 2a 183264 COMPLEMENT FACTOR D PRECURSOR (*H. sapiens*)	0.000588235
T86473 3′ UTR 1 114645 NUCLEOSIDE DIPHOSPHATE KINASE A (HUMAN)	0.000588235
R44887 3′ UTR 2a 33869 NEDD5 PROTEIN (*Mus musculus*)	0.000588235
X86693 gene 1 *H. sapiens* mRNA for hevin-like protein	0.000588235
M36634 gene 1 Human vasoactive intestinal peptide (VIP) mRNA, complete cds	0.000588235
J05032 gene 1 Human aspartyl-tRNA synthetase alpha-2 subunit mRNA, complete cds	0.000588235
H08393 3′ UTR 2a 45395 COLLAGEN ALPHA 2(XI) CHAIN (*H. sapiens*)	0.000588235
H06524 3′ UTR 1 44386 GELSOLIN PRECURSOR, PLASMA (HUMAN)	0.000588235
T95018 3′ UTR 2a 120032 40S RIBOSOMAL PROTEIN S18 (*H. sapiens*)	0.000714286
D25217 gene 1 Human mRNA (KIAA0027) for ORF, partial cds	0.000714286
M26697 gene 1 Human nucleolar protein (B23) mRNA, complete cds	0.000714286
M22382 gene 1 MITOCHONDRIAL MATRIX PROTEIN P1 PRECURSOR (HUMAN)	0.000714286
Z49269 gene 1 *H. sapiens* gene for chemokine HCC-	0.000714286
T92451 3′ UTR 1 118219 TROPOMYOSIN, FIBROBLAST AND EPITHELIAL MUSCLE-TYPE	0.000714286
H49870 3′ UTR 2a 178915 MAD PROTEIN (*H. sapiens*)	0.000714286
U26312 gene 1 Human heterochromatin protein HP1Hs-gamma mRNA, partial cds	0.000714286
X04106 gene 1 Human mRNA for calcium-dependent protease (small subunit)	0.000714286
X54942 gene 1 *H. sapiens* ckshs2 mRNA for Cks1 protein homolog	0.000714286
R44301 3′ UTR 2a 34262 MINERALOCORTICOID RECEPTOR (*H. sapiens*)	0.000714286
T51534 3′ UTR 1 72396 CYSTATIN C PRECURSOR (HUMAN)	0.001290323
R36977 3′ UTR 1 26045 P03001 TRANSCRIPTION FACTOR IIIA	0.001290323
X12369 gene 1 TROPOMYOSIN ALPHA CHAIN, SMOOTH MUSCLE (HUMAN)	0.001714286
X70326 gene 1 *H. sapiens* MacMarcks mRNA	0.001714286
U09564 gene 1 Human serine kinase mRNA, complete cds	0.001714286
X17651 gene 1 Human Myf-4 mRNA for myogenic determination factor	0.001714286

(continued)

Table 1.4 (continued)

Probe/gene name/description	*p* Value
X87159 gene 1 *H. sapiens* mRNA for beta subunit of epithelial amiloride-sensitive sodium channel	0.001794872
H40095 3′ UTR 1 175181 MACROPHAGE MIGRATION INHIBITORY FACTOR (HUMAN)	0.001794872
U22055 gene 1 Human 100-kDa coactivator mRNA, complete cds	0.001794872
X56597 gene 1 Human humFib mRNA for fibrillarin	0.001794872
X14958 gene 1 Human hmgI mRNA for high mobility group protein Y	0.001904762
X12466 gene 1 Human mRNA for snRNP E protein	0.001904762
U32519 gene 1 Human GAP SH3 binding protein mRNA, complete cds	0.001904762
D31885 gene 1 Human mRNA (KIAA0069) for ORF (novel protein), partial cds	0.003333333

dimension reduction method and proposed the use of PLS components in molecular cancer classification based on microarray gene expression profiles. Because of the complexity and nonlinear structure of the PLS weights, works in selecting/identifying the variables that contribute to the PLS dimension reduction above background noise level have been limited. In this work, we evaluated the relative usefulness of two heuristic rules for determining important variables (genes) based on VIP and B-PLS coefficient measures, both functions of the PLS weights. We also proposed an additional measure called RA-VIP and assigned a p value to each gene, determined from the null distribution where there is no association between gene expression and the binary outcome variable. We assessed the sensitivity and specificity of these approaches using simulation studies where the set of truly informative genes are known by design. Based on the simulation studies, selection of genes based on RA-VIP have overall advantages relative to gene selection based on the heuristic rules using VIP and B-PLS coefficients. A subset of genes can be identified based on RA-VIP p values, allowing for follow-up expression analysis such as RT-PCR.

Although we have focused the current work on the case of binary outcome/response, such as experiments comparing two cancer types or subtypes or between cancer and normal cellular states, the case of three or more groups (or continuous) outcome can be accommodated directly. This is feasible since PLS can handle multiple outcome variables (as well as continuous outcomes). For multiple categories with more than two groups, multiple indicators variables can be created for each group (e.g., for each cancer type) and PLS can be applied to the response matrix of response indicators.

Acknowledgments Support for this work includes the National Institute of Health (NIH) grants UL1DEO19583, RL1AG032119, RL1AG032115, HD036071, and UL1RR024146. This publication was also made possible by Grant Number UL1 RR024146 from the National Center for Research for Medical Research. Its contents are solely the responsibility of the authors and do not necessarily represent the official view of NCRR or NIH.

References

Alon U, Barkai B, Notterman DA et al (1999) Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays. Proc Natl Acad Sci 96:6745–6750

Boulesteix A (2004) PLS dimension reduction for classification with microarray data. Stat Appl Genet Mol Biol 3:Article 33

Boulesteix A, Strimmer K (2006) Partial least squares: A versatile tool for the analysis of high-dimensional genomic data. Brief Bioinform 8:33–44

Golub TR, Slonin DK, Tamayo P et al (1999) Molecular classification of cancer: Class discovery and class prediction by gene expression monitoring. Science 286:531–537

Helland IS (1988) On the structure of partial least squares. Commun Stat Simul Comput 17: 581–607

Irizarry RA, Hobbs B, Collin F, Beazer-Barclay YD, Antonellis KJ, Scherf U, Speed TP (2003) Exploration, normalization, and summaries of high density oligonucleotide array probe level data. Biostatistics 4:249–264

Li C, Wong WH (2001) Model-based analysis of oligonucleotide arrays: Expression index computation and outlier detection. Proc Natl Acad Sci USA 98:31–36

Nguyen DV, Rocke DM (2002a) Tumor classification by partial least squares using microarray gene expression data. Bioinformatics 18:39–50

Nguyen DV, Rocke DM (2002b) Multiclass cancer classification via partial least squares with gene expression profiles. Bioinformatics 18:1216–1226

Nguyen DV, Rocke DM (2004) On partial least squares dimension reduction for microarray-based classification: A simulation study. Comput Stat Data Anal 46:407–425

SAS Institute, Inc. SAS/STAT User's Guide (1999) Cary, NC

Wold H (1966) Estimation of principal components and related models by iterative least squares. In: Krishnaiah PR (ed) Multivariate analysis. Academic, New York

Wold S, Johansson E, Cocchi M (1993) PLS-partial least-squares projections to latent structures. In: Kubinyi H. (ed) 3D QSAR in drug design: Theory, methods and applications. ESCOM, Leiden, Holland

Zhou L, Rocke DM (2005) An expression index for Affymetrix GeneChips based on generalized logarithm. Bioinformatics 21:3983–3989

Chapter 2
Geometric Biclustering and Its Applications to Cancer Tissue Classification Based on DNA Microarray Gene Expression Data

Hongya Zhao and Hong Yan

Abstract Biclustering is an important tool in microarray data analysis when only a subset of genes coregulates under a subset of conditions. It is a useful technique for cancer tissue classification based on gene expression data. Unlike standard clustering analysis, biclustering methodology can perform simultaneous classification on the two dimensions of genes and conditions in a data matrix. However, the biclustering problem is inherently intractable and computationally complex. In this chapter, we present a novel geometric perspective of a biclustering problem and the related geometric algorithms. In the view of geometrical interpretation, different types of biclusters can be mapped to the linear geometric structures, such as points, lines, or hyperplanes in a high-dimensional data space. Such a perspective makes it possible to unify the formulation of biclusters and thus the biclustering process can be interpreted as a search for linear geometries in spatial space. Based on the linear geometry formulation, we develop Hough transform-based biclustering algorithms. Considering the computational complexity in searching the existence of noise in microarray data, and the biological meanings of biclusters, we propose several methods to improve the geometric biclustering algorithms. Simulation studies show that the algorithms can discover significant biclusters despite the increased noise level and regulatory complexity. Furthermore, the algorithms are also effective in extracting biologically meaningful biclusters from real microarray gene expression data.

2.1 Introduction

DNA microarray technology is a high-throughput and parallel platform that can provide expression profiling of thousands of genes in different biological conditions, thereby enabling rapid and quantitative analysis of gene expression patterns on a global scale. It aids the examination of gene functions at the cellular level, revealing

H. Yan (✉)
Department of Electronic Engineering, City University of Hong Kong, Kowloon, Hong Kong and School of Electrical and Information Engineering, University of Sydney, NSW 2006, Australia
e-mail: h.yan@cityu.edu.hk

T. Pham (ed.), *Computational Biology: Issues and Applications in Oncology*,
Applied Bioinformatics and Biostatistics in Cancer Research,
DOI 10.1007/978-1-4419-0811-7_2,

how multiple gene products work together to produce physical and chemical responses to both static and changing cellular needs. This mode of analysis has been widely used to observe gene expression variation in oncology (Alizadeh et al. 2000; Ochs and Godwin 2003; Stoughton 2005; Cowell and Hawthorn 2007).

As an increasing number of large-scale microarray experiments are carried out, analysis of the expression data produced by these experiments remains a major challenge. A key step of the analysis is the identification of groups of genes that exhibit similar expression patterns. Therefore clustering analysis has emerged as one of the most valuable tools to elicit complex structures and gather information about how genes work in combination with microarray data. A number of clustering methods have been proposed for the analysis of gene functions (Golub et al. 1999; Dudoit et al. 2002; Desper et al. 2004; Wu et al. 2004; Wang et al. 2008).

Gene expression data are usually arranged in a matrix, where each gene corresponds to a row and a condition to one column. Each element of the matrix represents the expression level of a gene under an experimental condition. The values of each element are usually the logarithms of the relative abundance of the mRNA measured in microarray experiments. Thus, clustering methods can be applied to group genes by comparing rows or conditions by comparing columns. However, conventional clustering methods have their limitations: they require that the related genes (conditions) behave similarly across all measured conditions (genes) in one cluster. In fact, an interesting cellular process may be involved in a subset of genes coregulated or coexpressed only under a subset of conditions, but to behave almost independently under other conditions. As such, some genes may participate in more than one function, resulting in one regulation pattern in one context and a different pattern in another. Discovering such local expression patterns may be the key to uncovering many genetic pathways that are not apparent otherwise. Thus, it is highly desirable to move beyond the clustering paradigm and to develop approaches capable of discovering local patterns in microarray data (Cheng and Church 2000; Madeira and Oliveira 2004; Prelic et al. 2006).

Beyond the traditional clustering methods, biclustering can identify a subset of genes that are coregulated across a subset of conditions in microarray analysis. Biclustering was first investigated by Hartigan (1972). It was first applied to expression matrices for simultaneous clustering of both genes and conditions by Cheng and Church (2000). Since then, the research literature on biclusters has been booming. Comprehensive surveys about the biclustering algorithms are published by Madeira and Oliveira (2004) and Tanay et al. (2006). Cheng and Church (2000) define the concept of δ-biclusters and a greedy algorithm for finding δ-biclusters. Yang et al. (2002) improved their method by allowing missing gene expression values in gene expression matrices. The algorithms proposed by Tanay et al. (2002) and Prelic et al. (2006) focused on the biclustering types of coherent evolution. In their method, the original expression matrices are discretized and the algorithms are applied to binary matrices. The order-preserving sub-matrix (OPSM) was proposed by Ben-Dor et al. (2002), in which all genes contain the same linear ordering and employ a heuristic algorithm for the OPSM problem. Ihmels et al. (2002) used gene signature and condition signature to find biclusters with both up- and down-regulated

expression values. When no a priori information of the matrix is available, they proposed a random iterative signature algorithm (ISA) (Ihmels et al. 2002; 2004). A random algorithm, xMOTIF, is presented by Murli and Kasif (2003). A systematic comparison of the different biclustering methods was made by Prelic et al. (2006).

In general, existing algorithms perform biclustering by adding or deleting rows and/or columns in the data matrix in optimal ways such that a merit function is improved by the action. A different viewpoint of biclustering can be formulated in terms of the spatial geometrical distribution of points in data space. The biclustering problem is tackled as the identification and division of coherent submatrices of data matrices into geometric structures (lines or planes) in a multidimensional data space (Gan et al. 2008). Such perspective makes it possible to unify the formulations of different types of biclusters and extract them using an algorithm for detecting geometric patterns, such as lines and planes. And it also opens a door to biclustering based on the detecting geometric pattern method, such as of lines, planes, and shapes.

Therefore, pattern recognition-based methods have been developed for data biclustering (Liew et al. 2005; Zhao and Yan 2007; Gan et al. 2008; Zhao et al. 2008). In these algorithms, the well-known Hough transform (HT) is employed to detect the geometric patterns. Statistical properties of the HT, such as robustness, consistency, and convergence, make it more suitable for biclustering analysis of microarray data than the traditional methods (Goldenshluger and Zeevi 2004). Especially it is noted for its ability to identify geometric patterns in noisy data because noise is one of the major issues in microarray data analysis.

However, the direct HT-based biclustering algorithm becomes ineffective in terms of both computing time and storage space. To overcome these difficulties, a subdimension-based method has been introduced into the biclustering algorithm. For example, the geometrical biclustering (GBC) algorithm only performs the HT in 2D column-pair spaces (Zhao et al. 2008).

After obtaining sub-biclusters with HT, we need to merge the small sub-biclusters into the larger ones. Different criteria are considered for the different geometric biclustering algorithms (GBCs). In GBC, the common genes and conditions of sub-biclusters are used based on the properties of a system of equations representing genes and conditions (Zhao et al. 2008). That is, two sub-biclusters are combined if they satisfy the numerical similarity measure. The algorithm is presented in Sect. 2.3.2. Obviously, the combination of common genes and conditions can be too strict to form meaningful biclusters of larger sizes. Because of noise in microarray data, many genes are filtered out after the merging step. It is found in the application that the number of genes in the identified biclusters is often small and the outcome is sensitive to noise. As such, the geometric properties of biclusters are ignored in the combination steps.

To overcome the shortcomings of GBC, an improved biclustering algorithm (RGBC) is proposed within the framework of probabilistic relaxation labeling in Sect. 2.3.3. Relaxation labeling processes are widely used in many different domains including image processing, pattern recognition, and article intelligence (Rosenfeld et al. 1976; Kittler and Illingworh 1985; Kittler 2000). They are iterative procedures

that aim to reduce the ambiguity and noise effect to select the best labels for all objects. In the relaxation-based geometric biclustering (RGBC) algorithm, the genes are labeled during the merging step based on the distance of the data points to the identified hyperplanes. According to this criterion, outlier genes with larger distance are deleted from the new sub-biclusters and genes whose expressions are close to the hyperplanes are merged. Thus, consistent large-size biclusters can be discovered.

In addition to the computational focus, it should be emphasized that the genes in one bicluster are involved in a similar biological function, process, or location. As well, accepted standards of function categories, gene ontology (GO), are employed to supervise the combining procedures in the GBC, named as GBFM in Sect 2.3.4. In most of the biclustering algorithms, GO is only used to infer the biological relevance of the obtained biclusters if the enrichment of the function categories within the biclusters is statistically significant (Prelic et al. 2006). The GBFM algorithm makes the forming of significant biclusters consistent with the gene function categories in GO. That is, the sub-biclusters are automatically merged into large biclusters if they are in the similar function categories in the framework of GO. The procedures directly incorporate the information of gene function modules into the biclustering process. Thus not only are the numerical characteristics in biclustering patterns identified, but the biological functions of biclusters are also considered in GBFM.

Unlike most biclustering algorithms, the novel geometric perspective of biclusters inspires the search of structures of known geometries in spatial space. So a series of GBCs are proposed based on the hyperplane detection methodology. The experiment results on both synthetic and real-gene expression datasets demonstrate that the algorithms are powerful, flexible, and effective. It is believed that such a novel approach to the problem of biclustering should be valuable to the scientific communities that frequently deal with detecting interesting patterns hidden in vast amounts of seemingly random experimental data (Gan et al. 2008; Zhao et al. 2008). In this chapter, we show that the biclustering methods are effective for cancer data analysis.

2.2 Geometric Biclustering Patterns

2.2.1 Bicluster Types

Gene expression data are usually arranged in a data matrix $D_{N\times n}$ with N genes and n experimental conditions. Element d_{ij} of $D_{N\times n}$ represents the expression level of the ith gene under the jth experimental condition. The value of d_{ij} is usually the logarithm of the relative abundance of the mRNA measured in microarray experiments.

Traditional clustering attempts to group all objects (genes or conditions) into different categories to uncover any hidden patterns. However, if we try to cluster

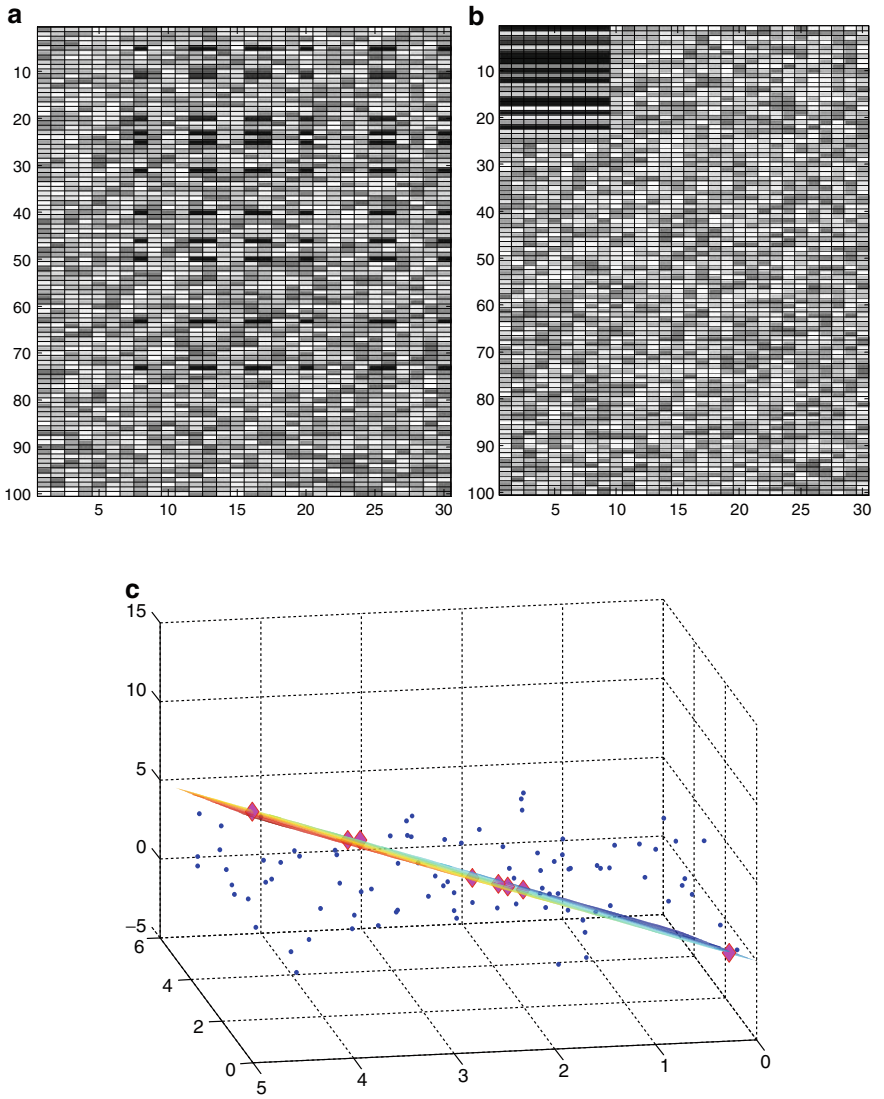

Fig. 2.1 Visualization of geometric patterns of biclustering: (**a**) The intensity representation of a microarray data matrix. (**b**) A hidden bicluster pattern embedded in the matrix by permuting the rows and columns appropriately. From a geometric point of view, different types of biclusters are considered as the points, lines, and hyperplanes in high-dimensional space. (**c**) One identified plane passing by the points in one additive bicluster in the three-dimensional space for demonstration

$D_{N\times n}$ as demonstrated in Fig. 2.1a using all measurements, we would not uncover any useful patterns although they actually exist in $D_{N\times n}$. By relaxing the clustering constraint that related objects must behave similarly across all measurements, some "localized" patterns can be uncovered readily as demonstrated in Fig. 2.1b.

Table 2.1 Illustration of a gene expression data matrix and five types of biclusters: (a) the left matrix is a simplified gene expression data one and (b) the right matrix includes the expression sets of five types of biclusters and the linear mathematical expression

	c_1	c_2	c_3	c_4	c_5	c_6
g_1	10	10	10	14	7	10
g_2	10	14	10	24	35	10
g_3	15	15	15	12	20	35
g_4	20	20	20	17	25	40
g_5	25	25	25	22	30	45
g_6	10	14	30	27	35	50

Type of biclusters	Bicluster set	Linear geometric expression of bicluster
Constant	$(\{g_1, g_2\}, \{c_1, c_3, c_6\})$	$x_1 = x_2 = [10\ 10\ 10]$
		$y_1 = y_3 = y_6 = [10\ 10]^{\mathrm{T}}$
Constant row	$(\{g_1, g_3, g_4, g_5\}, \{c_1, c_2, c_3\})$	$x_1 = x_3 - 5 = x_4 - 10 = x_5 - 15$
		$y_1 = y_2 = y_3 = [10\ 15\ 20\ 25]^{\mathrm{T}}$
Constant column	$(\{g_2, g_6\}, \{c_1, c_2, c_5\})$	$x_2 = x_6 = [10\ 14\ 35]^{\mathrm{T}}$
		$y_1 = y_2 - 4 = y_5 - 15$
Additive	$(\{g_3, g_4, g_5, g_6\}, \{c_3, c_4, c_5, c_6\})$	$x_3 = x_4 - 5 = x_5 - 10 = x_6 - 15$
		$y_3 = y_4 + 3 = y_5 - 5 = y_6 - 20$
Multiplicative	$(\{g_1, g_6\}, \{c_5, c_6\})$	$x_1 = 0.2 \times x_6$
		$y_5 = 0.7 \times y_6$

Biclustering algorithms perform clustering in the gene and condition dimensions simultaneously. In the case of a genome, a bicluster is regarded as being a set of genes that exhibit similar biological functions under a subset of experiment conditions (Barkow et al. 2006). Denoting the index of $D_{N\times n}$ as $G = \{g_1, \ldots, g_N\}$ and $C = \{c_1, \ldots, c_n\}$, we have $D = (G, C) \in \Re^{|G|\times|C|}$. Thus, during data analysis a bicluster $B = (X, Y)$ appears as a submatrix of D with some specific patterns, where $X = \{N_1, \ldots, N_x\} \subseteq G$ and $Y = \{n_1, \ldots, n_y\} \subseteq C$ are a separate subset of G and C.

In biclustering literature, several types of coherent patterns are defined as able to capture important biological phenomena. Table 2.1a is a simplified example of a gene expression matrix including five types of biclusters. As listed in the second column of Table 2.1b, these biclusters are (a) a constant bicluster: e.g., $(\{g_1, g_2\}, \{c_1, c_3, c_6\})$, (b) constant rows $(\{g_1, g_3, g_4, g_5\}, \{c_1, c_2, c_3\})$, (c) constant columns, $(\{g_2, g_6\}, \{c_1, c_2, c_5\})$, (d) additive coherent values, where each row or column can be obtained by adding the constant to another row or column, e.g., $(\{g_3, g_4, g_5, g_6\}, \{c_3, c_4, c_5, c_6\})$, and (e) multiplicative coherent values, where each row or column can be obtained by multiplying another row or column by a constant value, e.g., $(\{g_1, g_6\}, \{c_5, c_6\})$.

The biological meanings of these biclusters can be inferred from the relations between their genes and conditions. For example, in a constant bicluster, a subset of genes always displays the same expression level under a subset of conditions. In an additive bicluster, expression levels of a subset of genes under one condition are always higher (or lower) by a constant than under another condition. The study of

common fluctuations of the expression levels in these biclusters is useful in practical applications, such as cancer tissue classification (Alizadeh et al. 2000).

2.2.2 Geometric Expressions of Biclusters

In this section, the corresponding relations between the biclusters and geometric structures are demonstrated. Some properties of the linear expressions are also discussed in the complete and subdimensional data spaces and extended to the GBCs.

In recent years, various algorithms have been proposed to detect the different types of biclusters. Most of these algorithms employ data mining techniques to search for the best possible submatrices. The general strategy in these algorithms can be described as permuting rows and/or columns of the data matrix in a number of ways such that an appropriate merit function is improved by the action. Obviously, the form of the merit function depends greatly on the types of biclustering patterns to be uncovered (Madeira and Oliveira 2004).

In contrast to the existing permutation-based approach, a novel geometric perspective for the biclustering problem is inspired by Gan et al. (2008). According to their viewpoint, submatrices are mapped to be the points, lines, or planes with some special patterns in the high-dimensional data space. Thus instead of searching for coherent Bs in D by the permutation processes, the biclustering problem is transformed into the detection of specific geometric structures formed by the spatial arrangement of these data points. This perspective first provides a unified formulation for extracting different types of biclusters simultaneously. Furthermore, the geometric view makes it possible to perform biclustering with the generic line- or plane-finding algorithms.

For example, the condition set Y in $B = (X, Y)$ spans a $||Y||$-dimensional space, and the expression of every gene in X corresponds to a point in the spatial space. The five different types of biclusters can be uniquely mapped to the linear geometric structures with the equation $\sum_i a_i x_i = 0$ in the space. Figure 2.1c demonstrates the formed plane of one additive bicluster in a 3D data space.

In general, when one bicluster is embedded in a larger data matrix, the points or lines defined by the bicluster sweep out of a hyperplane in the spatial space. It is theoretically feasible to apply any plane-finding algorithm, such as the well-known HT, widely used in image analysis, to identify the biclusters in a microarray data matrix.

However, computational complexity makes the direct application difficult with a large number of genes and conditions. Thus the GBC algorithms are improved step by step. The following splitting technique plays an important role in coping with the NP-hard problem in biclustering (Zhao et al. 2008). Considering the linear relations within one bicluster, it is unnecessary to express its coherent pattern with all variables (conditions) in one equation. Based on the theoretical property of equivalent expression, one system of equations with two or three variables can be used to describe the biclustering pattern instead of one multivariable linear

Table 2.2 The linear expressions of the five biclustering types and the geometrical structures in the 2D data space, a_1–a_2 and ρ–θ Hough space, respectively

Type	Equation	Pattern in data space	Pattern in parameter space $[a_1, a_2]$	Pattern in polar space $[\theta, \rho]$
Constant (**C**)	$x_i = x_j = a$	A point on diagonal line	A line passing [1,0]	A sinusoidal curve passing $[-\pi/4, 0]$
Constant rows (R)	$x_i = x_j$	Points on diagonal line	Lines passing [1,0]	Sinusoidal curves passing $[-\pi/4, 0]$
Constant columns (O)	$x_i = a_i$, $x_j = a_j$ $(a_i \neq a_j)$	A point off diagonal line	A line passing $[a_1, 0]$ $(k \neq 1)$	A sinusoidal curve passing $[\theta, 0]$ $(\theta \neq -\pi/4)$
Additive (**A**)	$x_j = x_i + b_{ij}$ $(b_{ij} \neq 0)$	Points off diagonal line	Lines passing $[1,a_2]$ $(b \neq 0)$	A sinusoidal curve passing $[-\pi/4, \rho]$ $(\rho \neq 0)$
Multiplicative (M)	$x_j = a_{ij} x_i$ $(a_{ij} \neq 1)$	Points on the straight line passing origin	Lines passing $[a_1,0]$ $(k \neq 1)$	Sinusoidal curves passing $[\theta, 0]$ $(\theta \neq -\pi/4)$

equation. For example, instead of finding a constant row pattern in Table 2.1 satisfying $x_1 = x_3 - 5 = x_4 - 10 = x_5 - 15$ in a 4D space, we can detect the same pattern with $x_1 = x_3 - 5$, $x_3 = x_4 - 5$, and $x_5 = x_3 - 10$ in three 2D spaces.

As listed in Table 2.2, all five types of biclusters in Table 2.1 can be generalized into the linear relation of $x_j = a_{ij} x_j + b_{ij}$ in 2D space although they appear to be substantially different to each other. Considering the clear biological relevance in biclustering, the additive and multiplicative patterns are emphasized, which can be described by $x_j = \pm x_i + b_{ij}$ and $x_j = a_{ij} x_i$, respectively. Obviously, the first three types of biclusters are special cases of the two models when $b_{ij} = 0$ or $a_{ij} = 1$ in 2D space.

2.3 Geometric Biclustering Algorithms

Based on the geometric interpretation of biclusters discussed above, a series of GBCs are proposed to search for linear patterns. Considering the computational complexity of searching and the biological functions of biclusters, these algorithms commonly employ the HT-based technology to detect the genes of interest with linear structures in the subspaces. Then based on the resulting sub-biclusters, different strategies are considered to combine the small ones into the large biclusters.

For example, the common genes and conditions of sub-biclusters are considered in GBC based on the properties of a system of linear equations (Zhao et al. 2008).

That is, two sub-biclusters are combined if they satisfy the numerical similarity measure. The algorithm is presented in Sect 2.3.2. However, the criterion is too strict to discover large-size biclusters in many cases. Because of noise in microarray data, many genes are filtered out after the merging step. As such, the geometric property of biclusters is also ignored during the combination. Thus in Sect 2.3.3, the RGBC algorithm is proposed within the relaxation labeling framework for the combination of sub-biclusters based on their geometric properties. Considering the biological functions of biclusters in microarray experiments, the information of gene ontology (GO) annotations are incorporated into the merging step in GBFM (as discussed in Sect 2.3.4). That is, the sub-biclusters are automatically merged into large biclusters if they are in the same function categories. The algorithm balances the numerical characteristics in a gene expression matrix and the gene functions in the biological activities.

2.3.1 Hough Transformation for Line Detection

In the framework of GBCs, line detection plays an important role. The well-known HT is employed to detect the hyperplanes in subdimensional data spaces. The HT is a widely used classical method for extracting lines and curves in images through a voting process in the parameter space (Ballard and Brown 1982; Illingworth and Kittler 1988). Its statistical properties, such as robustness, consistency, and convergence, make it more suitable for microarray data analysis than some traditional regression methods (Goldenshluger and Zeevi 2004). Especially it is noted for its ability to identify geometric patterns in noisy data because noise is one of the major issues in microarray data. In this subsection, the HT is briefly reviewed first and its extension is then described.

2.3.1.1 The Classical Hough Transformation

Given a set of points $\{\boldsymbol{x_i} = (x_{1i}, x_{2i}) \in R^2 : i = 1, \ldots, n\}$, the objective is to infer the parameters (a_1, a_2) of the line $x_{2i} = a_1 x_{1i} + a_2$ which fit the data $\{\boldsymbol{x_i}\}$ optimally. The key to the HT is to view each point as generating a line comprising all pairs (a_1 and a_2) that are consistent with this point. For example, for the ith point, this line is given by $L_i = \{(a_1, a_2) : a_2 = a_1 x_{1i} + x_{2i}\}$ in the Hough domain. The collinearity in the original set of points will manifest itself in a common intersection of lines in the Hough domain.

The HT algorithm is implemented by identifying the cells with the desired quantization in the Hough domain that receive the largest number of counts. We denote the cell centered at $\xi = (a_1, a_2)$ as $D(\xi)$. The HT algorithm looks for a point $\hat{\xi} = (\hat{a}_1, \hat{a}_2)$ in the Hough domain such that the maximal number of lines L_i cross over the cell $D\left(\hat{\xi}\right)$. The HT estimators $\hat{\xi}$ maximize the object function with respect to $\xi = (a_1, a_2)$

$$M(\xi) := \frac{1}{n} \sum_{i=1}^{n} \mathbf{1}\{D(\xi) \cap L_i \neq \emptyset\},$$

where $\mathbf{1}\{D(\xi) \cap L_i \neq \emptyset\}$ is the number of lines that satisfy the given nonempty condition.

In practice, polar coordinates are used to describe the line in Hessian normal form instead of the direct parameter space. This allows for the detection of vertical lines ($\theta = \pi/2$) in the data set, and moreover guarantees an isotropic error in contrast to the parameterization. This leads the following parameter form of a line:

$$\rho = x_1 \cos\theta + x_2 \sin\theta,$$

where ρ is the distance of a line to the original point and θ is the angle of the normal to the line with the horizontal axis. Since ρ is limited from $-\left(x_1^2 + x_2^2\right)^{1/2}$ to $\left(x_1^2 + x_2^2\right)^{1/2}$ and θ is limited from $-\pi/2$ to $\pi/2$, the original dynamic ranges of the parameters in the Hough domain are compressed and a small accumulator array is sufficient to find all lines (Ballard and Brown 1982). Note that if the polar equation of a line is used, the points in data space are mapped to sinusoidal curves as demonstrated in Fig. 2.2. The last three columns of Table 2.2 also list the corresponding geometric patterns of biclusters in 2D data space, a_1–a_2 and ρ–θ Hough space, respectively.

2.3.1.2 Generalization of the Hough Transformation

The classical HT only detects lines in a given 2D space as discussed. The HT can be generally developed to n-dimension space. The point $\mathbf{x} \in R^m$ lying on such a hyperplane H can be described by the linear equation $\mathbf{n}^{\mathrm{T}}\mathbf{x} = 0$ where $\mathbf{n}^{\mathrm{T}}$ is a nonzero vector orthogonal to H. After normalization, $||\mathbf{n}|| = 1$ and the normal vector $\mathbf{n}$ is uniquely determined by H if we additionally require $\mathbf{n}$ to lie on one hemisphere of the unit sphere. In terms of spherical coordinates, $\mathbf{n}$ can be rewritten as

$$\mathbf{n} = \begin{bmatrix} \cos\varphi & \sin\theta_1 & \sin\theta_2 & \cdots & \sin\theta_{m-2} \\ \sin\varphi & \sin\theta_1 & \sin\theta_2 & \cdots & \sin\theta_{m-2} \\ & \cos\theta_1 & \sin\theta_2 & \cdots & \sin\theta_{m-2} \\ & & \ddots & & \vdots \\ & & \cos\theta_1 & \cos\theta_2 & \cdots \cos\theta_{m-2} \end{bmatrix} \tag{2.1}$$

with $(\varphi, \theta_1, \ldots, \theta_{m-2}) \in [0, \pi)^{m-1}$. By substituting $\mathbf{n}$ in (2.1) into the original linear equation, the generalized HT of $\mathbf{x} \in R^m$ is

$$\begin{aligned} \mathbf{x} \rightarrow \Big\{ & (\varphi, \theta_1, \ldots, \theta_{m-2}) \in [0, \pi)^{m-1} \,|\theta_{m-2} \\ & = \arctan\left(\sum_{i=1}^{m-1} v_i \frac{x_i}{x_m} \right) + \frac{\pi}{2} \Big\} \quad \text{for } x_m \neq 0, \end{aligned}$$

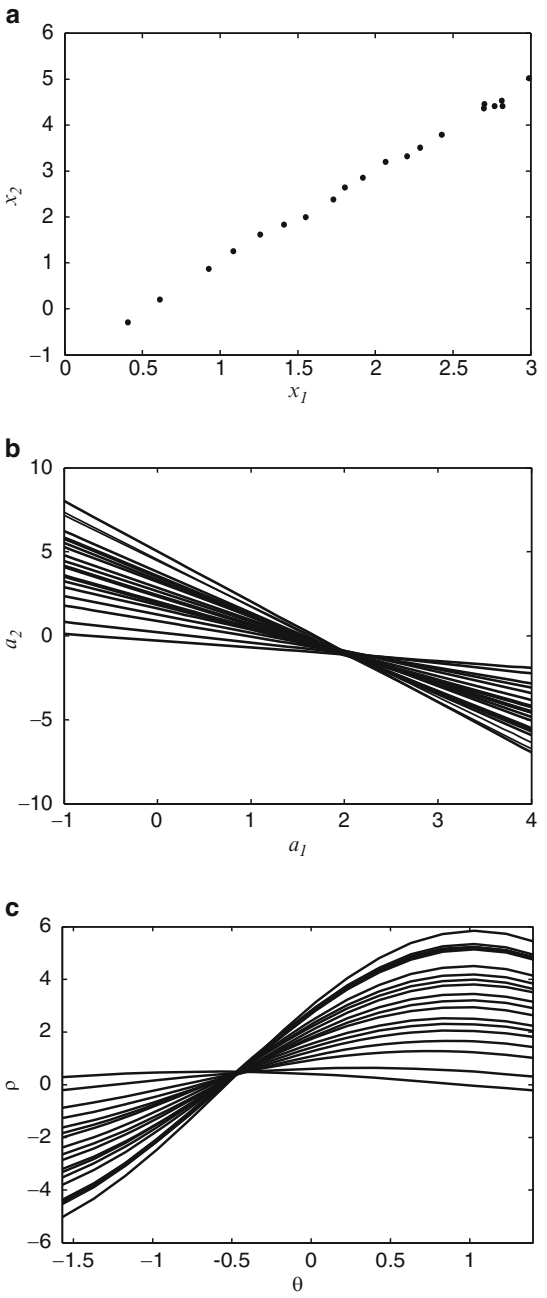

Fig. 2.2 Illustration of the Hough transform: (**a**) Points in the data space. According to the Hough transform, a point (or line) in data space is mapped to a line (or point) in the parameter space. (**b**) The corresponding lines in the linear parameter space. (**c**) The corresponding sinusoids in the polar parameter space. The collinearity in the original set of data points manifests itself in a common intersection of lines in the Hough space

where

$$v_i := \begin{cases} \cos\varphi \prod_{j=1}^{m-3} \sin\theta_j, & i = 1, \\ sin\varphi \prod_{j=1}^{m-3} \sin\theta_j, & i = 2, \\ \prod_{j=1}^{i-2} \cos\theta_j \prod_{j=i-1}^{m-3} \sin\theta_j, & i > 2 \end{cases}$$

and we set $\theta_{m-2} = 0$ for $x_m = 0$ for continuity (Ballard 1981; Theis et al. 2007).

Theoretically, it is feasible to directly employ HT in the complete data space to detect the geometric structures of biclusters (Gan et al. 2008). However, the original HT-based algorithm becomes ineffective, in terms of both computing time and storage space, as the number of conditions increases. Thus the HT is only used in some subspaces of microarray data to improve the efficiency in the following biclustering algorithms.

2.3.2 Geometric Biclustering Algorithm

In the proposed GBC (Zhao et al. 2008), the HT is first used to detect the lines in all 2D subspaces, named the column-pair space, and the sub-biclusters are then recorded. Considering the close relationships between the biological meanings and types of biclusters, we need to classify them in the next step. As discussed in 2.2.2, it is enough to consider the additive (A) and multiplicative (M) models. In the GBC algorithm, a visualization tool, additive and multiplicative pattern plot (AMPP) is developed for this task.

2.3.2.1 Additive and Multiplicative Pattern Plot

The AMPP is implemented as follows. Given $\{(x_{1i}, x_{2i}) : i = 1, \ldots, k\}$, it is assumed that there are k points on a line detected using the HT in a column-pair space. Now we try to separate the k points into two types of biclusters. The difference of $d_i = x_{1i} - x_{2i}$ and $r_i = \arctan(x_{1i}/x_{2i})$ are employed in AMPP to show the difference between the additive and multiplicative patterns as demonstrated in Fig 2.3. The horizontal axis of AMPP represents the change of additive patterns d_i and the vertical axis the multiplicative patterns r_i. In AMPP, $r_i = \arctan(x_{1i}/x_{2i})$ is used instead of the direct ratio x_{1i}/x_{2i} to reduce the dynamics range of the ratio.

Based on AMPP, the boxplot is used to classify the points into different patterns. The boxplot was first proposed by J. Tukey, as simple graphical summaries of the distribution of variables (Celveland 1993). In a boxplot, the middle line represents the median and the lower and upper sides of the rectangle show the medians of the lower and upper halves of the data. Along the horizontal boxplot, the points in the box are considered to be shifted with their median in an A pattern and the points in the box of the vertical boxplot are considered to be multiplied by their median in an M pattern. The points in their intersection set are considered as the overlapped

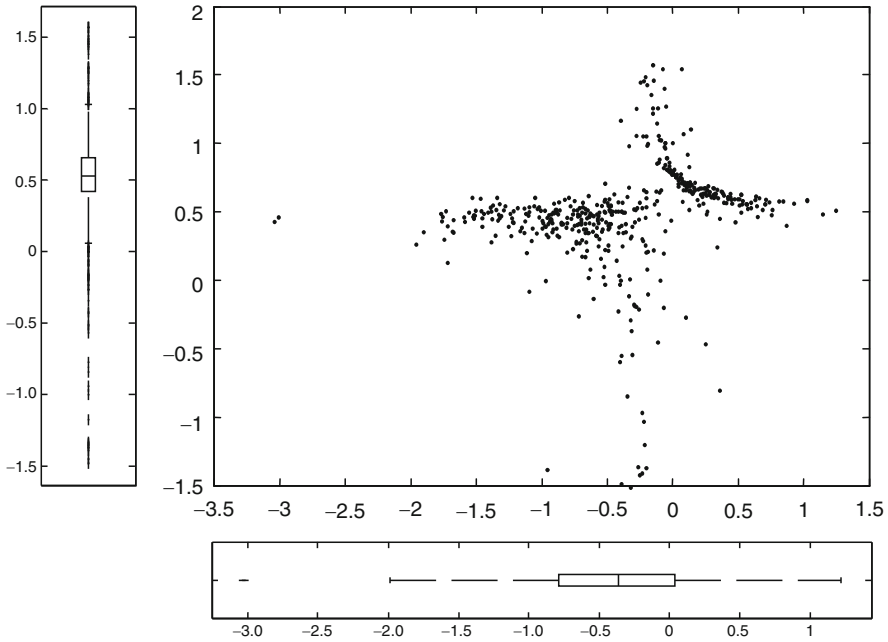

Fig. 2.3 Additive and multiplicative pattern plot (AMPP) of microarray data. $d_i = x_i - y_i$ and $r_i = \arctan(x_i / y_i)$ are scaled as horizontal and vertical axes. The corresponding points in the boxes of boxplots are separately divided into two models

points in the two patterns, that is, a constant pattern. The horizontal and vertical boxplots are also shown in Fig. 2.3. The visualization plot can classify the points detected using the HT in a column-pair space.

2.3.2.2 GBC Algorithm

Based on the HT and AMPP, the GBC algorithm can identify a set of maximal biclusters, where a sub-bicluster $B = (X, Y)$ is defined as a maximal one if and only if no T' exists such that $T \subset T'$ (that is $I \subset I'$ and $J \subset J'$) (Madeira and Oliveira 2004). A simplified flowchart of the GBC is shown in Fig. 2.4. In the GBC, the HT is used to detect the lines in all column-pair spaces and record the sub-biclusters as $B_{ij} = \left(G_{ij}, \{i, j\}\right)$. Obviously, these sub-biclusters are the maximal ones in their column-pair space related to two conditions. As presented in Sect. 2.3.2.1, AMPP is used to classify B_{ij}s into different types.

The following problem illustrates how to combine the smaller sub-biclusters into the maximal ones. The following property of sub-biclusters provides one solution to the merging step: Let $B_{\max} = (G_{\max}, C_{\max})$ be one maximal bicluster and $\{T_i = (I_i, J_i)\}$ be a set of maximal sub-biclusters in column-pair space. If $J_i \subseteq C_{\max}$,

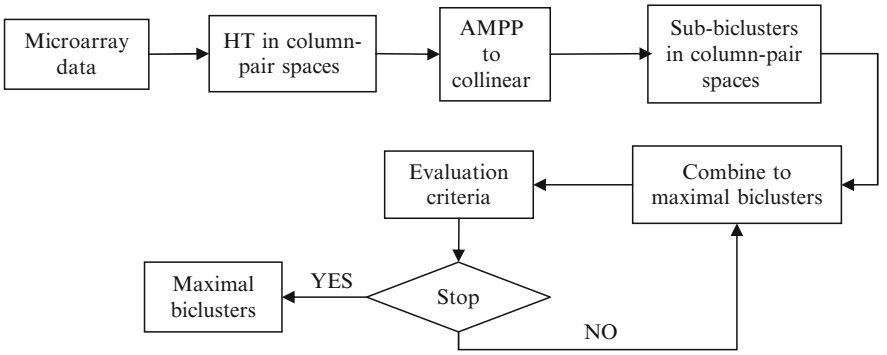

Fig. 2.4 The overall flow of GBC algorithm

then $G_{\max} \subseteq I_i$ (Yoon et al. 2005). So, the given two sub-biclusters $B_s = (I_s,\ J_s)$ and $B_t = (I_t,\ J_t)$ are combined to form a larger one using the operation

$$B_r = B_s \oplus B_t = (I_s \cup I_t, J_s \cap J_s)\,. \tag{2.2}$$

Thus the number of genes in the merged biclusters becomes smaller and smaller with the combination. The combination is stopped if the number of genes in the merged biclusters is fewer than the given parameter δ. The biclusters are also filtered out if the elements overlap by more than 25% or the number of conditions is fewer than the given parameter ζ. The overall GBC algorithm is summarized as follows.

Geometric Biclustering Algorithm (GBC)

Input: Microarray data matrix $D\ (G, C)$; quantization step size in the HT parameter space q; minimum number of genes and condition to form a bicluster δ and ζ.
Output: a series of biclusters
HT_ALG: perform Hough transformation in a column-pair space
AMPP_ALG: classify the collinear points

(1) Perform HT in all column-pair spaces

$$[G_{ij}, C_{ij}] = \text{HT\_ALG}(D(G,\ i),\ D(G,\ j),\ q)\ \forall\ i, j\ \in\ C;$$

(2) AMPP to classify the collinear points

$$[B_{ij}\ \text{Cons},\ B_{ij}\text{\_Add},\ B_{ij}\text{\_Mul}] = \text{AMPP\_ALG}(G_{ij}, C_{ij});$$

(3) Combine sub-biclusters

$$\text{for}\ \forall\ B^i\text{\_Cons} = (G^i\text{\_Cons},\ C^i\text{\_Cons}),\ B^j\text{\_Cons} = (G^j\text{\_Cons},\ C^j\text{\_Cons}) \in \{B_{ij}\text{\_Cons}\}$$

$$\text{if}\ C^i\text{\_Cons} \cap C^j\text{\_Cons} \neq \emptyset$$

then $C^{ij}\_\text{Cons} = C^{i}\_\text{Cons} \cup C^{j}\_\text{Cons}$

$$G^{ij}\_\text{Cons} = G^{i}\_\text{Cons} \cap G^{j}\_\text{Cons};$$

(4) Filter biclusters

if $||G^{ij}\_\text{Cons}|| > \delta$ and $||C^{ij}\_\text{Cons}|| > \zeta$

then $B^{ij}\_\text{Cons} = (G^{i}\_\text{Cons}, C^{i}\_\text{Cons})$;

if B^{ij}_Cons is overlapped by any element in any bicluster recorded

then repeat (3);

else

repeat (3) with output B^{ij}_Cons as one merged bilcluster;

2.3.2.3 Applications

The data from the synthetic model enable us to evaluate the performance of GBC to identify the known grouping. Zhao et al. (2008) investigate two important questions in a simulation study: whether the algorithm is robust against noise and whether it has the ability to identify multiple overlapping biclusters. In order to compare the performance of different biclustering methods, the following gene matching score, similar to the one used in Prelic et al. (2006) and Liu and Wang (2007). Let $B_1 = (G_1, C_1)$ and $B_2 = (G_2, C_2)$ be two sets of biclusters. The following score was first proposed to evaluate the recovery and representation of true biclusters in Zhao et al. (2008),

$$S(B_1, B_2) = \max_{B_1} \max_{B_2} \frac{|G_1 \cap G_2| + |C_1 \cap C_2|}{|G_1 \cup G_2| + |C_1 \cup C_2|}, \tag{2.3}$$

which is symmetric about B_1 and B_2. Compared with the original score, this score is more consistent with the cases in the experimental data analysis because the real underlying patterns of gene expression are completely unknown in microarray experiments. Therefore, it is enough in real data analysis to select some best and meaningful biclusters from the large number of detected ones as candidates for further biological verification. In the following section, we denote B_{opt} as the set of implanted biclusters and B as the set of resulting biclusters produced by a biclustering method. $S\left(B_{\text{opt}}, B\right)$ represents the best results of biclusters identified by the algorithm. The larger the scores are, the better the identified patterns are.

The performance of the GBC algorithm is investigated with noisy and overlapping biclusters in microarray data in comparison with other biclustering algorithms.

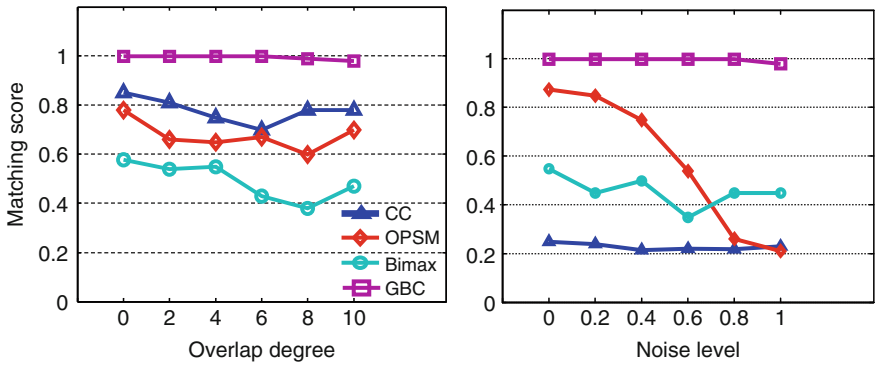

Fig. 2.5 Results of the simulation study for overlapping additive biclusters with different overlap degrees (*left*) and nonoverlapping additive ones with different noise levels (*right*)

In Zhao et al. (2008), the following algorithms are performed, such as CC (Cheng and Church 2000), OPSM (Ben-Dor et al. 2002), and Bimax (Prelic et al. 2006), which can be downloaded from the software toolbox BicAT (Barkow et al. 2006). Considering the M pattern can be mapped to A pattern by taking a log transformation, the synthetic expression data are generated with the approach described in Prelic et al. (2006).

Figure 2.5 summarizes the performance of different biclustering methods. In the case of the overlapped biclusters of different degrees, GBC is hardly affected by the overlap degree of the implanted biclusters. In the combination steps, all specific submatrices satisfying the conditions are identified so that the details of sub-biclusters can be tracked and used to detect all overlapping biclusters. In CC, however, the random values are used to replace the discovered biclusters in a given matrix to find more biclusters. The gene matching scores of the CC algorithm are higher than those of the other two existing methods, but are still lower than the ones from the GBC algorithm. The Bimax algorithm appears to be little sensitive to the increased overlap degrees. The first normalization step in Bimax may cause this problem. Because the range of expression values after normalization becomes narrower with increased overlap, the differences between normal and significant expression values blur and are more difficult to separate. As to CC and OPSM, the performance is not significantly affected by the overlap degree.

With the increasing of noise levels, GBC algorithm also shows better performance than the other algorithms. The HT is well-known to be robust against noise and this is why GBC, which is developed based on the HT, has a superior performance. In contrast, the significant decrease of the performance may be caused by the greedy algorithm in OPSM: only a single bicluster is considered for the linear ordering number of every column. And the CC algorithm computes the similarity of the selected gene expression data only and can easily be trapped at local optimal points. To implement Bimax, the synthetic data should be discretized to binary values with a threshold. Since noise blurs the difference between background and biclusters, the binarization process can degrade the biclustering performance.

Besides the computational performance of biclustering algorithms, the researchers are more interested in the biological relevance of the detected biclusters. So the GBC is applied to the real microarray data of multiple human organs (Son et al. 2005). The dataset captured 18,927 unique genes for 19 different organs from 158 normal human tissues of 30 donors. The data can be downloaded at the Web site http://home.ccr.cancer.gov/ontology/oncogenomics/. Two procedures are performed to explore the expression patterns of the human organ in Zhao et al. (2008).

In the first case, the gene expressions of the different organs are calculated for the analysis with the mean values and one $5,298 \times 19$ mean expression matrix is obtained after filtering. In the framework of GBC, $19 \times 18/2 = 171$ sub-biclusters are first obtained in column-pair spaces. In all these sub-biclusters, the number of columns is always two and that of rows is the peak count of accumulator arrays after the HT in the corresponding parameter space. We show the heat map of all sub-biclusters in Fig. 2.6. The indices of the row and column in Fig. 2.6 refer to 19 different organs, and the values of the cross points are the number of genes in the corresponding sub-bicluster in their column-pair space. The diagonal values are set to zero. Obviously, the square matrix is symmetric. We use different gray scales to represent different count values. The darker the intensity is, the larger the value is. For example, the largest value of the square matrix is 468 in the sub-bicluster

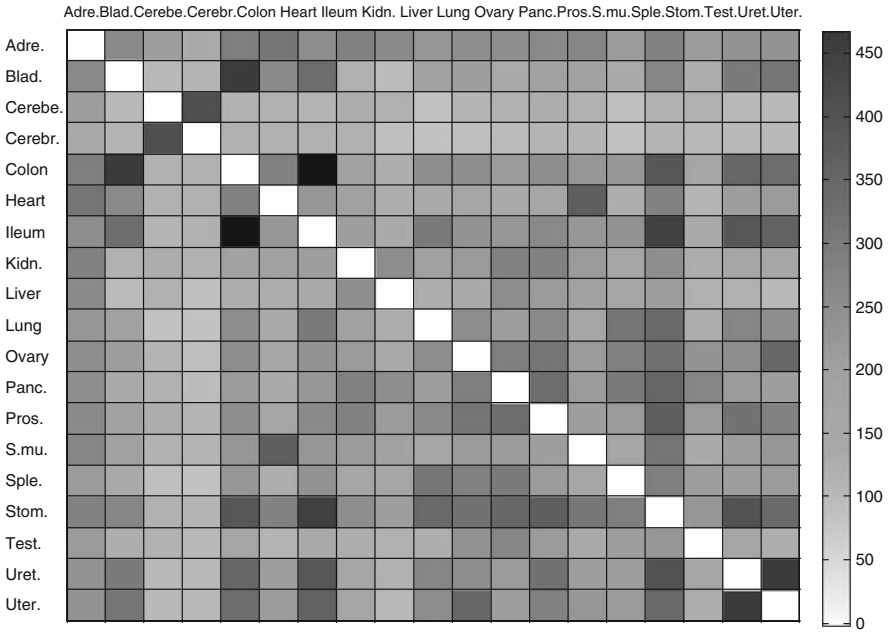

Fig. 2.6 Heat map of the symmetric square matrix of highest count in the column-pair space. The rows and columns represent 19 organs. The cross values are the highest count number of the accumulator array after the HT. The diagonal values are set to zero

composed of colon and ileum, that is, their gene expression patterns are very similar, which is in logical agreement with the known functions of the organs. In the following steps, the sub-biclusters are merged into the maximal biclusters. It is also discovered that the procedures of combination nearly coincide with their corresponding organ functions. For example, the similar organs such as colon, ileum, bladder, and stomach are merged into one significant block with the largest number of common genes after the first iteration of the algorithm.

Besides the computation with the mean expression matrix, the biclustering algorithm can be directly performed with the whole expression matrix. In Son et al. (2005), t testing of the mean expression matrix is the main analysis tool. Obviously, the information among the samples of the same organ is ignored. The GBC is applied to the whole 5,298 × 158 expression matrix. The best biclusters with the significant functions of every organ are listed in Table 2.3 and compared to the results of Son et al. (2005). In Table 2.3, the third column is the number of genes given in Son et al. (2005) with respect to the organ-specific GO, and the second and fourth column show the number of columns and rows in our biclusters, respectively. The most significant GO term and corresponding p values are provided in the fifth and sixth column, respectively. Although all samples of every organ are considered instead of the mean values, the number of genes in each bicluster is more than that in Son et al. (2005) except for the bladder. In addition, the significant biclusters in the colon, ileum, ovary, stomach, and uterus are detected with GBC, which were not detected in Son et al. (2005).

Table 2.3 The significant gene ontology of the 19 organs

Organ	# Samples	# Genes in Wang et al. (2008)	# Genes in GBC	GO term	p Values
Adrenal	9	2	6	GO: 0015247	4.7×10^{-7}
Bladder	9	104	62	GO: 0005604	3.2×10^{-6}
Cerebellum	6	4	17	GO: 0007420	5.4×10^{-5}
Cerebrum	7	5	24	GO: 0030594	7.1×10^{-9}
Colon	8	–	11	GO: 0045078	2.6×10^{-7}
Heart	7	5	35	GO: 0008016	2.9×10^{-6}
Ileum	10	–	8	GO: 0006629	8.3×10^{-7}
Kidney	10	11	23	GO: 0006811	7.9×10^{-8}
Liver	10	54	76	GO: 0016491	3.4×10^{-6}
Lung	9	17	21	GO: 0006955	1.2×10^{-5}
Ovary	5	–	25	GO: 0007338	5.7×10^{-7}
Pancreas	6	6	13	GO: 0007586	3.2×10^{-6}
Prostate	8	3	22	GO: 0006334	3.2×10^{-6}
S. muscle	9	10	38	GO: 0008307	2.4×10^{-9}
Spleen	10	7	16	GO: 0001584	5.6×10^{-6}
Stomach	10	–	9	GO: 0042894	5.3×10^{-7}
Testicle	7	25	27	GO: 0019953	7.1×10^{-10}
Ureter	8	4	18	GO: 0006366	6.8×10^{-6}
Uterus	10	–	7	GO: 0007275	9.4×10^{-7}

2.3.3 Relaxation-Based Geometric Biclustering Algorithm

Based on the sub-biclusters detected by HT in column-pair spaces, GBC makes use of the combining criterion that any two sub-biclusters with at least one common condition can be combined into a new one where the common genes are kept. Obviously, the criterion is too strict to discover large biclusters in microarray data because of noise. As such, the geometric property of biclusters is ignored during the combination.

Considering the recognition of geometric structures in noise data, an improved algorithm RGBC is proposed within the framework of probabilistic relaxation labeling. Relaxation labeling processes are widely used in many different domains including image processing, pattern recognition, and article intelligence. They are iterative procedures that aim to reduce the ambiguity and noise effect to select the best labels for all objects.

In the merging step of RGBC, the genes are labeled based on the distance of data points to the identified hyperplanes. Thus the outlier genes with large distance are deleted from the new sub-biclusters and others close to the hyperplanes are merged. In the following sections, a brief introduction to the relaxation labeling scheme, the RGBC algorithm and its applications are presented, respectively.

2.3.3.1 Nonlinear Probabilistic Relaxation Labeling

The relaxation labeling technique was first proposed by Rosenfeld et al. (1976). In a relaxation procedure, the contextual information is employed to classify a set of interdependent objects by allowing interactions among the possible classifications of related objects. Probabilistic relaxation has been successfully applied to many image processing tasks, such as scene labeling, pixel labeling, shape matching, line and curve enhancement, handwriting character recognition, and breaking substituting ciphers (Kittler and Illingworh 1985; Fu and Yan 1997).

A nonlinear probabilistic relaxation model can be described as follows. In general, there are five parts in a labeling problem:

1. A set of n objects: $\mathrm{A} = \{a_1, \ldots, a_n\}$ to be labeled.
2. A set of m labels: $\Lambda = \{\lambda_1, \ldots, \lambda_m\}$ for each object.
3. The weight of the influence on one object from others. We use d_{ij} to denote the influence coefficients of a_i from a_j, which satisfy $\sum_{j=1}^{n} d_{ij} = 1$.
4. For each pair of objects a_i and a_j, a compatibility coefficient matrix R_{ij} with size $m \times m$ is defined. The element $r_{ij}(\lambda, \lambda')$ $(\lambda, \lambda' \in \Lambda)$ of R_{ij} represents the compatibility of labeling λ on object a_i with λ' on object a_j, which satisfies the condition

$$\begin{cases} 0 < r_{ij}(\lambda, \lambda') \leq 1, & \lambda \text{ and } \lambda' \text{ compatible;} \\ r_{ij}(\lambda, \lambda') = 0, & \lambda \text{ and } \lambda' \text{ independent;} \\ -1 \leq r_{ij}(\lambda, \lambda') < 0, & \lambda \text{ and } \lambda' \text{ incompatible.} \end{cases}$$

5. For each object a_i, there is a set of initial probability $p_i^{(0)}(\lambda)$ $(\lambda \in \Lambda)$ satisfying $\sum_{\lambda\in\Lambda} p_i^{(0)}(\lambda) = 1$ where $0 \le p_i^{(0)}(\lambda) \le 1$.

A relaxation scheme actually corresponds to a recurrent dynamic system which depends on the updating rule of the system (Fu and Yan 1997). In general, the updated probability for λ of a_i at the $(k+1)$th iteration is

$$p_i^{(k+1)}(\lambda) = \frac{p_i^{(k)}(\lambda)\left[1 + q_i^{(k)}(\lambda)\right]}{\sum_{\lambda\in\Lambda} p_i^{(k)}(\lambda)\left[1 + q_i^{(k)}(\lambda)\right]}, \tag{2.4}$$

where the updating correction is

$$q_i^{(k)}(\lambda) = \sum_{j=1}^{n} d_{ij}\left(\sum_{\lambda'\in\Lambda} r_{ij}(\lambda,\lambda')\, p_j^{(k)}(\lambda')\right). \tag{2.5}$$

The key factors in the relaxation scheme are the initial probability estimate $p_i^{(0)}(\lambda)$ for label assignment and the calculation of the compatibility coefficients $r_{ij}(\lambda, \lambda')$. However, there are no general methods for initial probability assignment. In RGBC, the procedure of merging sub-biclusters is mapped into a relaxation framework that can deal with outliers and high noise effectively by using robust initial probability estimation and compatibility-coefficient assignment techniques as discussed below.

Given a series of sub-biclusters detected by HT in subspaces, for example, two of them are denoted as $B_1 = (G_1, C_1)$ and $B_2 = (G_2, C_2)$ and their hyperplanes as l_1 and l_2, respectively. To determine whether the two sub-biclusters can be combined, the expression value b_{ij} in $B_{12}^0 = (G_{12}, C_{12}) = (G_1 \cap G_2, C_1 \cap C_2)$ is considered as an object. Thus one of the two labels λ_0 and λ_1 is assigned to each object and the objects labeled λ_0 are kept in the combined sub-bicluster and those labeled λ_1 are deleted. In geometric biclustering, the points belonging to one bicluster should be on, or close to, the detected hyperplane. Therefore, it is reasonable to employ the distances of points to the hyperplanes in the labeling algorithm. To test the significance of b_{ij} on the distance, we delete the ith row and jth column from B_{12} to calculate the sum of the distances sd_{ij} to the merged hyperplanes. Then, it can be concluded that the object b_{ij} is more significant than the other objects in terms of its contribution to the distance measured when sd_{ij} is smaller, that is, b_{ij} may be an outlier; and vice versa.

In the relaxation scheme, the initial probabilities of every object should be estimated first. However, there is no general method to calculate the probabilities. Inspired by the geometric view of biclusters, the empirical estimation of the distances sd_{ij} is employed as initial probabilities of the object b_{ij} in RGBC

$$\begin{cases} p_{ij}^{(0)}(\lambda_0) = \frac{1}{T}\sum\limits_{t=1}^{T} H\left(sd_{gh}, sd_{ij}\right) \\ p_{ij}^{(0)}(\lambda_1) = 1 - p_{ij}^{(0)}(\lambda_0) \end{cases}, \tag{2.6}$$

where

$$H\left(sd_{gh}, sd_{ij}\right) = \begin{cases} 1 & sd_{gh} \geq sd_{ij} \\ 0 & sd_{gh} < sd_{ij} \end{cases}$$

and T is the number of the objects in B_{12}^{0}. Obviously according to (2.6), the smaller the distance to the hyperplane, the higher the initial probability of the gene being labeled λ_0 will be.

When the initial probabilities of each object are determined, the following compatibility coefficients based on statistical correlation are calculated

$$r_{ij,gh}\left(\lambda, \lambda'\right) = \frac{\sum_{ij,gh}\left(p_{ij}\left(\lambda\right) - \bar{p}_{ij}\left(\lambda\right)\right)\left(p_{gh}\left(\lambda'\right) - \bar{p}_{gh}\left(\lambda'\right)\right)}{\sigma\left(\lambda\right)\sigma\left(\lambda'\right)}, \tag{2.7}$$

where $p_{ij}\left(\lambda\right)$ is the initial probability of the object on the ith row and jth column with label λ, $\bar{p}_{ij}\left(\lambda\right)$ is the mean of $p_{ij}\left(\lambda\right)$ for all objects along the ith row and jth column, and $\sigma\left(\lambda\right)$ is the standard deviation of $p_{ij}\left(\lambda\right)$ (Rosenfeld et al. 1976).

2.3.3.2 Algorithms

The RGBC algorithm is based on the detection of hyperplanes in data subspaces with the HT and the combination of sub-biclusters with the relaxation labeling method, which is illustrated in Fig 2.7.

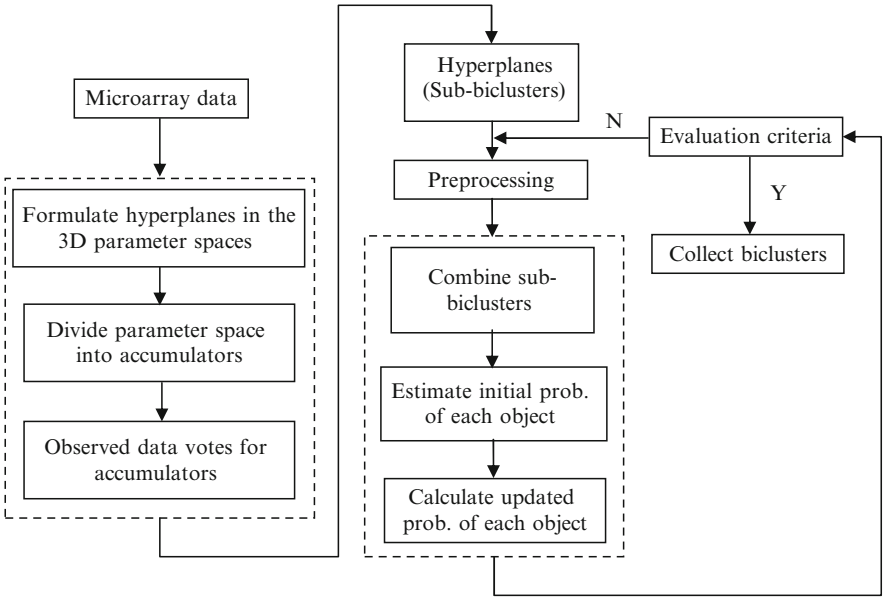

Fig. 2.7 The overall flow diagram of the RGBC algorithm

As described in Sect 2.3.1, the generalized HT is used to identify geometric patterns in some but not all subdimensional spaces. The strategy of data space partition has been employed in subspace clustering methods (Lam and Yan 2006). Instead of 2D pair-column space, 3D subspace is employed to reduce the computation complexity. The detected sub-biclusters contain the maximal number of genes in their 3D subspaces related to three conditions.

The next step is to combine the sub-biclusters into larger ones. The problem is transformed into the labeling procedure to delete outlier genes. With the analysis of geometric distances in biclusters, the probabilistic relaxation labeling framework is applied to the merging step in the algorithm. Considering the number of genes is much larger than that of conditions in microarray data, only the rows of the outliers are deleted and the columns are retained in RGBC

The combination is stopped after all sub-biclusters are considered, or if the number of genes in the merged biclusters is fewer than the given parameter δ. The biclusters are filtered if the number of conditions is fewer than a given parameter ζ. The overall RGBC algorithm is summarized as follows.

Relaxation-Label Geometric Biclustering (RGBC) Algorithm

Input: A microarray data matrix $D\,(G, C)$.
Output: A set of geometric biclusters.
GHT: The generalized HT and the outputs are the identified sub-biclusters and the linear equation of their geometric patterns.
PRL: The probabilistic relaxation labeling algorithm and the outputs are the uniform sub-biclusters combined.

(1) Perform the GHT in 3D sub-spaces to form the geometric sub-biclusters.

For $i = 1{:}||C||$

do $C_i = [i\, i + 1 i + 2]$

$[sB_i,\ l_i] = \text{GHT}\ (D(G, C_i), q)$ where $sB_i = (sB_i\_G,\ sB_i\_C)$;

(2) Perform PRL to combine the sub-biclusters into uniform large-sized ones.

For $\forall$ i, j $\in$ sBs

do $[sB_{ij}\_G,\ sB_{ij}\_C] = \text{PRL}\ (sB_i,\ sB_j)$;

(3) Filter sub-biclusters until no sub-biclusters can be combined.

If $||sB_{ij}\_G|| > \delta$

If $||sB_{ij}\_C|| > \zeta$

If $||sB_{ij}|| \cap ||sB_{st}|| < \beta||sB_{ij}||$ where $\forall\ sB_{st} \in \{sB_{ij}\}$

then output $B_{ij} = sB_{ij}$

2.3.3.3 Applications

Similar to the simulation study in GBC, the synthetic model in Prelic et al. (2006) is employed to investigate the capability of the method to identify the known grouping with noise and outliers. The proposed method is compared with several biclustering algorithms in our simulation study, such as GBC (Zhao et al. 2008), CC (Cheng and Church 2000), OPSM (Ben-Dor et al. 2002), ISA (Ihmels et al. 2002; 2004), xMotif (Murli and Kasif 2003), and Bimax (Prelic et al. 2006) algorithms, some of which can be downloaded from the software BicAT http://www.tik.ee.ethz.ch/sop/bimax (Barkow et al. 2006).

In the first scenario, a numerical comparison is made between the number of outliers and the algorithms' performance without noise as shown in Fig. 2.8a. The x-axis shows the different percentages of outlier genes in the expression matrices and y-axis shows the matching scores calculated by (2.3).

Obviously, the patterns in Fig. 2.8a demonstrate significant difference among the scores of different biclustering algorithms. In comparison to the higher scores of GBC, OPSM, Bimax, and RGBC, the scores of CC, ISA, xMotif are very low in all synthetic cases and unsuitable for synthetic data. The scores of CC are almost equal to 0.31 since the discovered biclusters by CC have to be marked with random values and thus the expression values of outlier genes are degraded. The xMotif method is mainly designed to find biclusters with coherent row values, and thus the types of underlying biclusters in our simulation are not well suited for this algorithm. There is not even any output in the synthetic data with noise. A similar argument can also be applied to the case of ISA. Indeed, only up- and downregulated expressions are used in ISA so that the outliers are emphasized and some rows and columns in real biclusters containing elements of normal expression levels are missed, especially in additive patterns.

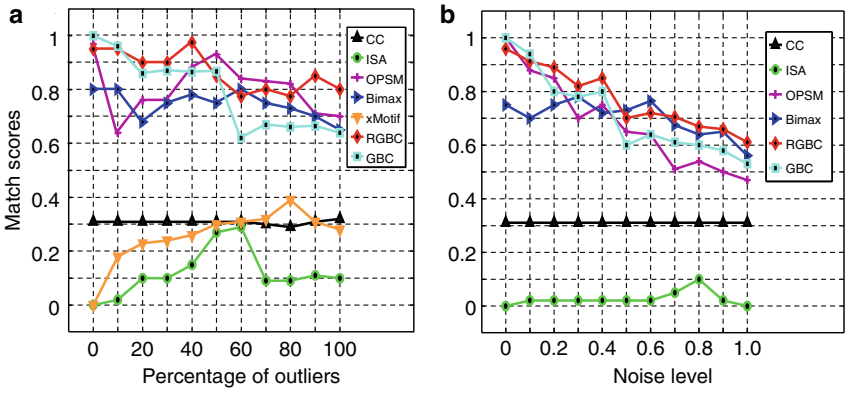

Fig. 2.8 Results of the simulation study for the RGBC algorithm. (**a**) The comparison of the number of outliers vs. the algorithms' performance without noise. The x-axis shows the different percentages of outlier genes in the expression matrices and y-axis shows the matching scores. (**b**) The effect of noise on the biclustering algorithms. The x-axis shows the different levels of noise and y-axis the matching scores

Comparatively, GBC shows the best performance and the score is equal to the one without outliers. With the increasing percentage of outliers, however, the scores rapidly decrease especially near the percentage range of 50–60. In GBC, the strict criterion of combination makes the size of biclusters small. OPSM is also sensitive to the number of outlier genes. In the absence of outlier genes, the better biclusters can also be detected with OPSM. OPSM is also suitable to identify the additive biclusters because the changes along the condition dimension represent optimal order preserving submatrices. On the other hand, the decrease in performance may be caused by the greedy algorithm in OPSM: only a single bicluster is considered for the linear ordering number of every column. Bimax shows the better and stable performance. The proposed RGBC algorithm performs well and is not significantly affected by outlier genes. The results imply that the criterion of geometrical distance within the relaxation label framework is effective for the outliers in microarray data. If an outlier is in one of the sub-biclusters in the combination step, its distance to the combined hyperplanes will be larger than those of other genes. Thus, the outliers can be easily identified and deleted from the new combined sub-biclusters.

The second artificial scenario, where the data are uniformly distributed without outlier genes, serves as a basis to assess the sensitivity of the methods to noise in the data (Fig. 2.8b). Similar to Fig. 2.8a, the lower scores are recorded in CC and ISA. For varying noise levels, the influence of noise becomes significant and the scores of all biclustering algorithms are decreased. Overall, GBC, OPSM, Bimax, and RGBC show better performance in this case.

In the absence of noise, GBC and OPSM show the best performance and the scores are equal to one. With the increase in noise, however, the scores rapidly decrease. Comparatively, the performance of Bimax is stable to the noise. To implement Bimax, the synthetic data should be discretized to binary values with the predefined threshold or percentage. In this experiment, we set the top 10% altered genes to one and the rest to zero in the simulation study. Since noise blurs the difference between background and biclusters, the binarization process may degrade the performance of Bimax especially at high level noise.

Comparatively, the trends of scores in GBC and RGBC are very close. Some scores of GBC are a little higher than those of RGBC in the case of low noise levels. One potential reason of the phenomenon is the way the HT work: both of them begin with the resulting sub-biclusters of the HT in subdimensional spaces. Indeed, the HT used in the algorithms plays an important role in the robustness to noise. Overall, the performance of GBC is better at detecting the geometrical biclusters in the ideal situation without noise and outliers and the improved RGBC is more effective in real microarray data analysis.

Furthermore, the RGBC algorithm is applied to two microarray datasets obtained for the yeast cell cycle (Cho et al. 1998) and colon cancer (Alon et al. 1999).

Yeast cell cycle data. The yeast cell data show the fluctuation of expression levels of 6,220 genes at 17 time points. According to the five-phase criterion, a subset of 384 genes is adopted whose expression levels peak at different time points corresponding to the five phases of cell cycles (Cho et al. 1998). The labeled set is analyzed by many clustering and classification algorithms in microarray data

Table 2.4 The matching scores of different biclustering algorithms in the labeled yeast cell cycle microarray data

	Bicluster				
Algorithm	CC	OPSM	Bimax	GBC	RGBC
Early G_1	0.26	0.17	0.23	0.29	0.38
Late G_1	0.60	0.54	0.35	0.54	0.41
S phase	0.19	0.46	0.57	0.32	0.43
G_2	0.24	0.25	0.21	0.37	0.36
M phase	0.44	0.42	0.37	0.31	0.65
Mean	0.35	0.37	0.38	0.37	0.45
Std	0.11	0.09	0.06	0.10	0.05

analysis (Lam and Yan 2006). The 17 time points are divided into five cycle phases: early G_1 phase, late G_1 phase, S phase, G_2 phase, and M phase. Every gene in the set is also labeled with one of the phases according to their biological functions. Thus, given the labeled functional subset, we can calculate the match scores to compare the performance of different biclustering algorithms. The data set can be downloaded from http://faculty.washington.edu/kayee/model/.

CC, OPSM, Bimax, GBC, and RGBC are applied to the data set and the results are shown in Table 2.4. The match scores of biclusters are calculated for the subset of five phases and the mean and standard deviation of every algorithm are also listed at the bottom of Table 2.4. For example, the mean of match scores in CC is nearly 0.35 with std 0.12. The high means imply that the resulting biclusters are consistent with the labeled functional subsets. If the std is high, however, the performance of the algorithm significantly fluctuates among the different phases. Obviously, there is little difference among the performance of CC, OPSM, Bimax, and GBC. The proposed RGBC shows the better results with the higher mean of 0.45 and lower std of 0.05.

Colon cancer dataset. Unlike for the labeled data set in the first case, we select the unlabeled microarray data in this section. In most microarray research, the biological function of spotted genes on microarrays is unknown and the experiments are designed to explore the functionally characterized genes under some conditions. Therefore, the genes in one bicluster are considered to be regulated in a synchronized fashion under the conditions. The microarray experiment of colon cancer originated in Alon et al. (1999), where it is of interest to separate the coregulated families of genes in cancerous from noncancerous tissues. The matrix contains 40 tumor and 22 normal colon tissues over 2,000 genes which are chosen with the highest minimal intensity across all the samples (Alon et al. 1999). The dataset can be downloaded from http://microarray.princeton.edu/oncology/.

Biclustering algorithms such as CC, OPSM, Bimax, GBC, and RGBC are applied to the colon cancer data set. To demonstrate the capability of detecting large biclusters, a bicluster obtained from a method should include many genes of either cancerous or noncancerous tissues only. The best biclusters, using different methods, are listed in Table 2.5. For every algorithm, there are two rows: one for tumor samples and the other for normal ones. In CC, for example, one best bicluster for

Table 2.5 The results of the biclusters detected in the non-labeled colon cancer dataset

	Bicluster			
Algorithm	# Genes	# Samples	# Tumor	# Normal
CC	50	19	26	7
	33	23	14	9
OPSM	31	13	8	5
	85	9	3	6
xMotif	14	23	18	5
	16	8	1	7
ISA	37	3	3	0
	24	3	0	3
GBC	11	15	12	3
	6	15	1	4
RGBC	25	18	16	2
	17	13	0	13

a tumor contains 50 genes and 33 tissues of which 26 are tumor samples and 7 are normal, and the best normal sample contains 19 genes and 23 tissues of which 9 are normal and 14 are tumorous. In Table 2.2, we discover that the number of samples in the biclusters of ISA is small, although the complete separation is identified. Similarly, the number of genes is large in the biclusters of OPSM. In the merging step of GBC, the number of genes is decreased and that of samples is increased step by step to find the common genes and samples. Comparatively, the results show that the RGBC method can find high-quality biclusters.

2.3.4 Geometric Biclustering Using Functional Modules (GBFM)

Based on the sub-biclusters detected by HT in subspaces, different strategies can be employed for the combination. In genome research, these genes in new combined biclusters are involved in a similar function, process, or location. However, the previous algorithms only focus on the computational solutions, such as the numerical similarity measure or noise in the data. And the biological relevance of the obtained biclusters are tested in the last step with the gene ontology (GO). In geometric biclustering using functional modules (GBFM), however, we directly incorporate the biological information into the merging process based on gene ontology (GO) annotations. The algorithm balances the numerical characteristics in a gene expression matrix and the gene functions in the biological activities.

2.3.4.1 Gene Annotation and Functional Modules

The soundness of clustering in the analysis of gene expression profiles and gene function prediction is based on the hypothesis that genes with similar expression

profiles may imply strong correlations with their functions in the biological activities. On the one hand we can therefore discover the coexpressed genes in column-pair space by HT. On the other hand, the challenge faced in our biclustering is how to combine the underlying biological phenomena into the identification of biclustering. Currently, GO provides us with a systematic tool to mine the functional and biological significance of genes and gene products in the resulting clusters. We analyze the correlation between the expression similarity of genes and their corresponding functional similarity according to GO, and then propose a novel algorithm to identify the functional biclusters. In this section, we present the statistical methodology of annotating clusters with GO.

The GO project provides a controlled vocabulary for various genomic databases of diverse species in such a way that it can show the essential features shared by all the organisms (The Gene Ontology Consortium 2000). Figure 2.9 shows the top levels of the gene ontology.

At the first level, known genes are classified into three categories, i.e., Molecular Function (MF), Cellular Component (CC), and Biological Process (BP). Different

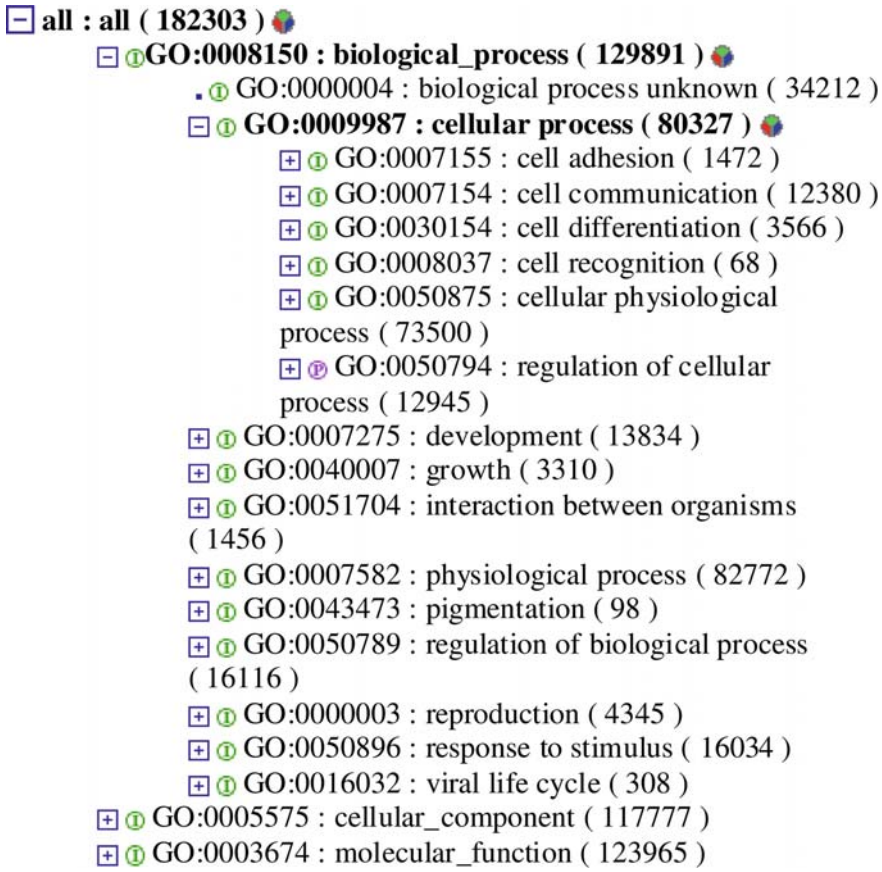

Fig. 2.9 A screen shot of a tree view of GO terms

gene function categories in GO have very sophisticated relationships, such as "part of" or "is". They can be structured as directed acyclic graphs (DAGs) that represent a network in which each term may be a child of one or more parents. GO has become a well-accepted standard in organizing gene function categories. Furthermore, many studies are implicitly based on the assumption that gene products that are biologically and functionally related maintain this similarity both in their expression profiles as well as in their GO annotation.

In a typical microarray experiment, the express matrix is a long list of genes with corresponding expression measurements under different conditions. This list is only the starting point for a meaningful biological explanation. Their relevant biological processes or functions can be identified from gene expression data by scoring the statistical significance of predefined functional gene groups, e.g., based on Gene Ontology (GO). In our algorithm, we will use the p value to assess the significance of a particular function group of genes within a cluster.

Given a set of genes and one of the three ontology terms, we first find the set of all unique GO terms within the ontology that are associated with one or more genes of interest. Next, for each term we determine how many of these genes are annotated at the node. We can ask if there are more genes of interest at the node than one might expect by chance. If that is true, then that term can be thought of as being overrepresented in the data. We calculate the p value to assess the significance of a particular function group within a cluster. The hypergeometric distribution is used to model the probability of observing at least k genes from a cluster of n genes by chance in a category containing f genes from a total genome size of g genes. The p value is given by

$$P(i \geq k) = \sum_{i=k}^{f} \frac{\binom{f}{i}\binom{g-f}{n-i}}{\binom{g}{n}}, \tag{2.8}$$

which is the probability of observing something as extreme or more extreme than was observed. Thus, the test measures whether a cluster is enriched with genes from a particular category to a greater extent than would be expected by chance (Berrize et al. 2003).

For example, if the majority of genes in a cluster have the same biological function, then it is unlikely that this happens by chance and the category's p value would be close to 0. In our biclustering algorithm, the p value of each category present in the sub-bicluster is first calculated. Given a cutoff threshold θ_p, the categories whose p value is larger than θ_p are eliminated without further consideration. The result is a set of significant GO function categories. In essence, the procedure is involved in multiple hypothesis testing. We use the methodology of controlling FDR instead of the Bonferroni correction for multiple hypotheses testing.

There are two direct ways to annotate the biclusters with the set of significant function categories. One method is to keep all significant function categories as annotation candidates. However, the annotation might become ambiguous if genes in

one cluster are assigned with too many function categories. The other way is to annotate a cluster with the category that has the smallest p value. Choosing the most significant category to represent the cluster is reasonable. However, we note the fact that few genes annotated at the categories will typically have the smaller p values, and often they are not very interesting. We choose typical GO terms with a reasonable number of genes and small p values in our biclustering algorithm.

2.3.4.2 Algorithms

Based on the detection of sub-biclusters with the HT in column-pair spaces and the incorporation of GO function categories, the algorithm GBFM is introduced to identify the gene functional biclusters.

In the proposed algorithm, we take the GO into account when we merge the sub-biclusters. The GO terms of each sub-bicluster are first determined with the corresponding p values. Then they are combined if their GO terms are similar. We stop the combination if the sub-biclusters are not annotated by GO or the number of genes in the merged biclusters is fewer than the given parameter δ. We also filter out biclusters whose number of conditions is fewer than a given parameter ζ. The GBFM is summarized into several major steps as shown below.

Geometric Biclustering using Functional Modules (GBFM) Algorithm

Input: Microarray data matrix $D(G, C)$; the quantization step size in ρ-θ space q; the minimum number of genes in one bicluster δ; the minimum number of conditions in one bicluster ζ; the significant level in the hypothesis testing of GO annotation α; the percent of element in one set β.

Output: The maximal biclusters significantly annotated by GO.

HT: Perform the HT in a column-pair space and the outputs are the corresponding indexes of genes and conditions in the identified sub-biclusters.

MHT_GO: Perform the hypothesis testing of sub-biclusters related to GO and the outputs are the corresponding GO term (sT–GO) and the significant level (sS–GO).

Step 1: Perform the HT to detect the specific lines in all column-pair spaces to form the sub-biclusters.

$$\text{for } \forall\ i, j \in C, \text{ then } [sB_{ij}\_\text{Add}, sB_{ij}\_\text{Mul}] = \text{HT}\ (D(G, i), D(G, j), q)$$

where

$$sB_{ij}\_\text{Add} = [sG_{ij}\_\text{Add}, sC_{ij}\_\text{Add}] \text{ and } sB_{ij}\_\text{Mul} = [sG_{ij}\_\text{Mul}, sC_{ij}\_\text{Mul}]$$

Step 2: Multiple hypothesis testing of sub-biclusters with α

$$[sT_{ij}\_\text{GO}, sS_{ij}\_\text{GO}] = \text{MHT\_GO}(sB_{ij}, \alpha)$$

Step 3: Filter sub- biclusters until no sub-biclusters can be combined

If $sS_{ij}\_\text{GO} < \alpha$

If $||sG_{ij}|| > \delta$

If $||sC_{ij}|| > \zeta$

If $||sB_{ij}|| \cap ||sB_{st}|| < \beta||sB_{ij}||$ where $\forall\ sB_{st} \in \{sB_{ij}\}$

Then output $B_{ij} = sB_{ij}$

Step 4: Combine the sub-biclusters with the same functional modules and common conditions

$\forall\ sB_i = [sG_i,\ sC_i], sB_j = [sG_j,\ sC_j] \in \{sB_{ij}\_\ \text{Add}\}$ (or $\{sB_{ij}\_\text{Mul}\}$)

If $sT_i\_\text{GO} \cap sT_j\_\text{GO} \neq \emptyset$

If $sC_i \cap sC_j \neq \emptyset$

then $sB_{ij} = [sG_{ij},\ sC_{ij}]$ where $sG_{ij} = sG_i \cap sG_j$ and $sC_{ij} = sC_i \cup sC_j$

2.3.4.3 Applications

The technology of cDNA microarrays is used to explore the variation in the expression of approximately 8,000 unique genes in 60 cell lines used in the National Cancer Institute's screen for anticancer drugs. Such cell lines differ from both normal and cancerous tissue, the inaccessibility of human tumors and normal tissue makes it likely that such cell lines will continue to be used as experimental models for the foreseeable future. In fact, the classification of the cell lines based solely on the observed patterns of gene expression can reveal a correspondence to the ostensible origins of the tumors from which the cell lines are derived. The assessment of gene expression patterns in a multitude of cell and tissue types should also lead to increasingly detailed maps of the human gene expression program and provide clues as to the physiological roles of uncharacterized genes (Ross et al. 2000).

The original database of the cDNA microarray experiment is available at http://genome-www.stanford.edu/nci60 and http://discover.nci.nih.gov. We use the processed $1{,}328 \times 60$ expression data matrix and perform the GBFM biclustering analysis and discover relationships between phenotypic properties of the 60 cell lines and 1,328 significant genes.

In the first step, we perform the HT in column-pair space to detect the different types of sub-biclusters. Figure 2.10 demonstrates three cases of sub-biclusters in ρ–θ parameter spaces. The gene expression data in the sub-biclusters are mapped to the sinusoidal curves. As demonstrated in Fig. 2.10a, the corresponding expressions

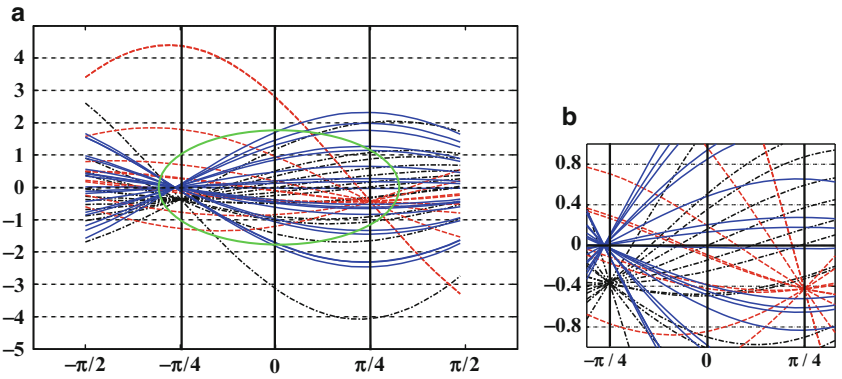

Fig. 2.10 (**a**) Curve intersecting points in the ρ–θ space corresponding to detected lines in column-pair space using the HT. The intersecting points of interest are on $\theta = \pm\pi/4$ or $\rho = 0$ in the ρ–θ space. The curve intersection regions are zoomed in and shown in (**b**)

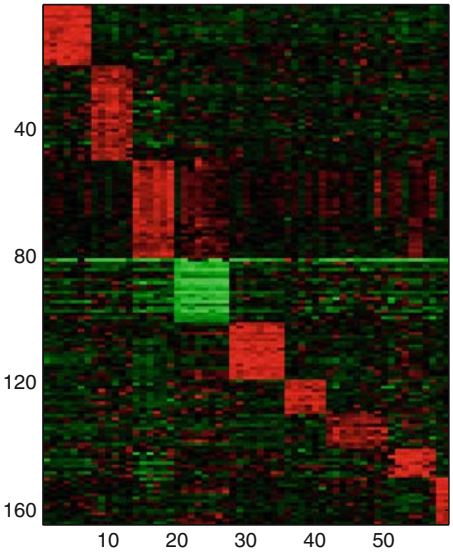

Fig. 2.11 Some biclustering results of the GBFM for the human cancer cell lines. The row contains 164 selected genes, the column contains 59 cell lines, and the corresponding expression values are demonstrated with one color map

are classified into the different types according to the intersecting positions, such as the multiplicative type if the blue curves intersect on the horizontal axis. And the relations can be clearly demonstrated with the zoom-in plot in Fig. 2.10b.

The sub-biclusters are then merged step by step into the maximal biclusters depending on the gene functional categories. According to the experimental design, the 60 cell lines are derived from 10 tumor tissues including colon (7), central nervous system (6), leukaemia (6), melanoma (8), renal (8), ovarian (6), lung (9), breast (7), prostate (2), and unknown tissues (1), where the numbers in the brackets are the number of samples from the same tissue. We present the nine corresponding biclusters of the first tissues in Figs. 2.11 and 2.12 and the corresponding GO terms with p values are listed in Table 2.6.

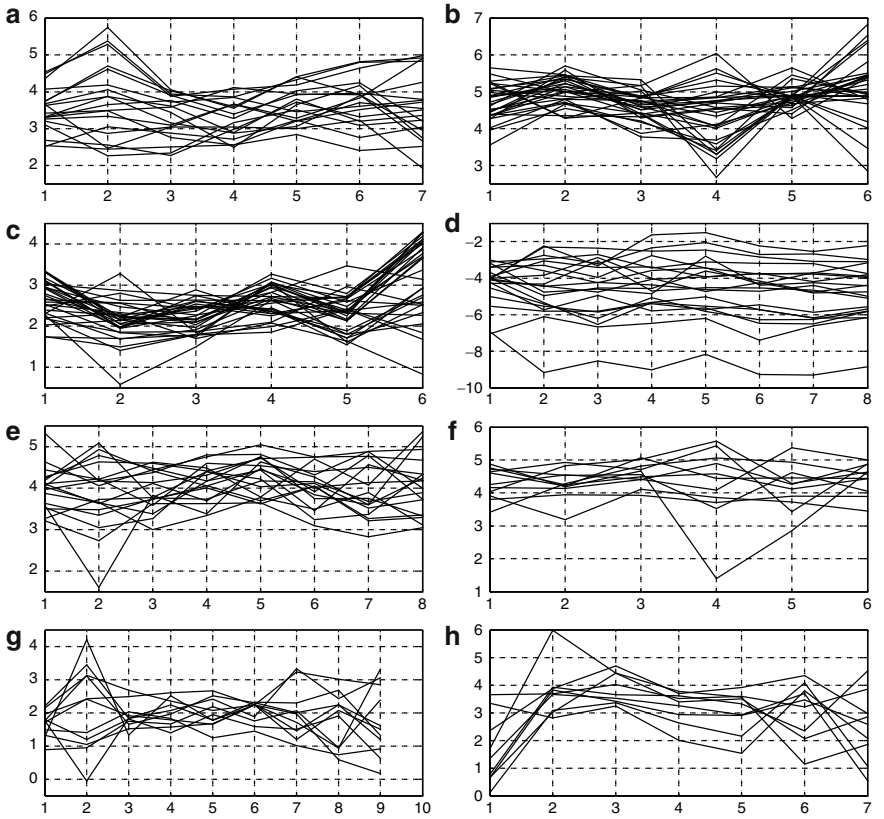

Fig. 2.12 Gene expression date of all biclusters in Fig. 2.8 except those for prostate. The x-axis shows the index of the cell lines from the tissues and the y-axis shows the expression values. (**a**) Colon, (**b**) central nervous system, (**c**) leukaemia, (**d**) melanoma, (**e**) renal, (**f**) ovarian, (**g**) lung, (**h**) breast cancer cell lines

Table 2.6 Significant gene ontology terms for nine human cancer cell lines

Cell line	Bicluster size	GO term	p Value
Colon cancer	19×7	GO: 0045078: positive regulation of interferon-gamma biosynthetic process	4.6×10^{-6}
CNS	30×6	GO:0003013: circulatory system process	7.3×10^{-5}
Leukaemia	31×6	GO: 0006352: transcription initiation	8.1×10^{-6}
Melanoma	20×8	GO:0016020: membrane	2.9×10^{-5}
Renal cancer	18×8	GO:0031012:extracellular matrix	3.4×10^{-6}
Ovarian cancer	11×6	GO:0007155: cell adhesion	1.2×10^{-5}
Lung cancer	11×9	GO:0051726:regulation of cell cycle	5.7×10^{-7}
Breast cancer	9×7	GO:0007010: cytoskeleton organization and biogenesis	4.7×10^{-6}
Prostate cancer	15×2	GO: 0006334nucleosome assembly	3.2×10^{-5}

In Fig. 2.8, the row contains 164 selected genes and the column contains the 59 conditions, and the corresponding expression values are demonstrated with one color map. Obviously the patterns of nine tissues are significantly different. The detailed fluctuations of every bicluster except prostate are plotted in Fig. 2.12, where the x-axis shows the index of condition and the y-axis shows the expression values.

Our analysis also produces results similar to those in Ross et al. (2000). We discover that 20 genes are highly expressed in the melanoma-derived lines. This set is enriched for genes with known roles in melanocyte membrane such as TYR, DCT, MLANA, and SPON1. However, the flat fluctuations of the gene expressions in Fig. 2.12 shows that there is less difference in gene expression in melanoma-derived cell lines than the others. It is also discovered in our studies that the genes in the bicluster of rental-derived cells are characterized by their synthesis or modification of an extracellular matrix (THBS1, CATSL1, CATSA). More biological meanings of the biclusters are summarized in Table 2.6.

2.4 Conclusions

In this chapter, a novel interpretation of the biclustering problem is presented in terms of geometric distributions of data points in a high-dimensional data space. From this perspective, the biclustering problem becomes that of detecting structures of known linear geometries in the high-dimensional data space. Thus, the typical types of biclusters are just different spatial arrangements of the hyperplanes in the high-dimensional data space. This novel perspective allows us to perform biclustering geometrically using a hyperplane detection algorithm, such as the HT. In comparison with the original search in complete space, the HT can be employed in subspaces to deduce the computational complexity. With the sub-biclusters, different strategies are used for the following combination step to form large biclusters, such as the common variable in GBC, relaxation labeling in RGBC, and GO annotation in GFBC. The experiment results on both synthetic and real gene expression datasets have demonstrated that the geometric algorithm is very effective for DNA microarray data analysis and cancer tissue classification. Since biclustering can partition the input data in both row and column directions simultaneously and can produce overlapping biclusters, we expect that the technique will find more and more applications in many problems of computational ontology, in which there is a need to detect coherent patterns in the data.

Acknowledgement This work is supported by a grant from the Hong Kong Research Grant Council (Projects CityU 122506 and 122607).

References

Alizadeh AA, Eisen MB, Davis RE et al (2000) Distinct types of diffuse large B-cell lymphoma identified by gene expression profiling. Nature 403:503–511

Alon U, Barkai N, Notterman DA et al (1999) Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays. Proc Natl Acad Sci USA 96:6745–6750

Ballard DH (1981) Generalizing the Hough transform to detect arbitrary shapes. Pattern Recogn 13:111–122

Ballard DH, Brown CM (1982) Computer vision. Prentice-Hall, Englewood Cliffs, NJ

Barkow S, Bleuler S, Prelic A et al (2006) BicAT: a biclustering analysis toolbox. Bioinformatics 22:1282–1283

Ben-Dor A, Chor B, Karp R et al (2002) Discovering local structure in gene expression data: the order-preserving sub-matrix problem. In: Myers G et al (eds) Annual conference on research in computational molecular biology. Proceedings of the 6th annual international conference on computational biology. ACM, New York, pp 49–57

Berrize GF, King OD, Bryant B et al (2003) Characterizing gene sets with FuncAssociate. Bioinformatics 19:2502–2504

Celveland WS (1993) Visualizing data. At & T Bell Labloratories, Murray Hill, NJ

Cheng Y, Church GM (2000) Biclustering of expression data. In: Proceedings of 8th international conference on intelligent systems for molecular biology (ISMB'00), pp 93–103

Cho RJ, Campbell MJ, Winzeler EA et al (1998) A genome-wide transcriptional analysis of the mitotic cell cycle. Mol Cell 2:65–73

Cowell JK, Hawthorn L (2007) The application of microarray technology to the analysis of the cancer genome. Curr Mol Med 7:103–120

Desper R, Khan J, Schaffer A (2004) Tumor classification using phylogenetic methods on expression data. J Theor Biol 228:477–496

Dudoit S, Fridlyand J, Speed TP (2002) Comparison of discrimination methods for the classification of tumors using gene expression data. J Am Stat Assoc 97:77–87

Fu A, Yan H (1997) A new probabilistic relaxation method based on probabilistic space partition. Pattern Recogn 30:1905–1917

Gan X, Liew AWC, Yan H (2008) Discovering biclusters in gene expression data based on high-dimensional linear geometries. BMC Bioinformatics 9:209

Goldenshluger A, Zeevi A (2004) The hough transform estimator. Ann. Stat. 32:1908–1932.

Golub TR, Slonim DK, Tamayo P et al (1999) Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. Science 286:531–537

Hartigan JA (1972) Direct clustering of a data matrix. J Am Stat Assoc 67:123–129

Ihmels J, Friedlander G, Bergmann S et al (2002) Revealing modular organization in the yeast transcriptional network. Nat Genet 31:370–377

Ihmels J, Bergmann S, Barkai N (2004) Defining transcription modules using large-scale gene expression data. Bioinformatics 20:1993–2003

Illingworth J, Kittler J (1988) A survey of the hough transform. Comput Vis Graph Image Process 44:87–116

Kittler J (2000) Probabilistic relaxation and the Hough transform. Pattern Recogn 33:705–714

Kittler J, Illingworh J (1985) A review of relaxation labeling, algorithm. Image Vis Comput 3:158–189

Lam B, Yan H (2006) Subdimension-based similarity measure for DNA microarray data clustering. Phys Rev E 74:041096

Liew AWC, Yan H, Yang M (2005) Pattern recognition techniques for the emerging field of bioinformatics: A review. Pattern Recogn 38:2055–2073

Liu X, Wang L (2007) Computing the maximum similarity bi-clusters of gene expression data. Bioinformatics 23:50–56

Madeira SC, Oliveira AL (2004) Biclustering algorithms for biological data analysis: a survey. IEEE ACM Trans Comput Biol Bioinformatics 1:24–45

Murli TM, Kasif S (2003) Extracting conserved gene expression motif from gene expression data. In: Proceedings of the 8th Pacific symposium on biocomputing, Lihue, Hawaii, pp 77–88

Ochs MF, Godwin AK (2003) Microarrays in cancer: research and applications. BioTechniques 34:4–15

Prelic A, Bleuler S, Zimmermann P et al (2006) A systematic comparison and evaluation of biclustering methods for gene expression data. Bioinformatics 22:1122–1129

Rosenfeld A, Hummel R, Zucker S (1976) Scene labeling by relaxation operations. IEEE Trans System Man Cybernet 6:420–433

Ross D, Scherf U, Eisen M et al (2000) Systematic variation in gene expression patterns in human cancer cell lines. Nat Genet 24:208–209

Son C, Bilke S, Davis S et al (2005) Database of mRNA gene expression profiles of multiple human organs. Genome Res 15:443–450

Stoughton RB (2005) Applications of DNA microarrays in biology. Annu Rev Biochem 74:53–82

Tanay A, Sharan R, Shamir R (2002) Discovering statistically significant biclusters in gene expression data. Bioinformatics 18:136–144.

Tanay A, Sharan R, Shamir R (2006) Biclustering algorithms: a survey. In: Aluru S (ed) Handbook of computational molecular biology. Chapman & Hall/CRC, Boca Raton, FL

The Gene Ontology Consortium (2000) Gene ontology tool for the unification of biology. Nat Genet 25:25–29

Theis FJ, Georgiev P, Cichocki (2007) A robust sparse component analysis based on a generalized Hough transform. EURASIP J Adv Signal Process 2007:13

Wang L, Montano M, Rarick M et al (2008) Conditional clustering of temporal expression profiles. BMC Bioinformatics 9:147

Wu S, Liew AWC, Yan H et al (2004) Cluster analysis of gene expression data based on self-splitting and merging. IEEE Trans Inf Technol Biomed 8:5–15

Yang J, Wang W, Yu PS (2002) Delta-clusters: capturing subspace correlation in a large data set. In: Proceedings of 18th IEEE international conference on data engineering, 2002, pp 517–528

Yoon S, Nardini C, Benini L et al (2005) Discovering coherent biclusters from gene expression data using zero-suppressed binary decision diagrams. IEEE ACM Trans Comput Biol Bioinformatics 2:339–354

Zhao H, Yan H (2007) HoughFeature: a novel method for assessing drug effects in three-color cDNA microarray experiments. BMC Bioinformatics 8:256

Zhao H, Liew AWC, Xie X et al (2008) A new geometric biclustering algorithm based on the Hough transform for analysis of large-scale microarray data. J Theor Biol 251:264–274

Chapter 3
Statistical Analysis on Microarray Data: Selection of Gene Prognosis Signatures

Kim-Anh Lê Cao and Geoffrey J. McLachlan

Abstract Microarrays are being increasingly used in cancer research for a better understanding of the molecular variations among tumours or other biological conditions. They allow for the measurement of tens of thousands of transcripts simultaneously in one single experiment. The problem of analysing these data sets becomes non-standard and represents a challenge for both statisticians and biologists, as the dimension of the feature space (the number of genes or transcripts) is much greater than the number of tissues. Therefore, the selection of marker genes among thousands to diagnose a cancer type is of crucial importance and can help clinicians to develop gene-expression-based diagnostic tests to guide therapy in cancer patients. In this chapter, we focus on the classification and the prediction of a sample given some carefully chosen gene expression profiles. We review some state-of-the-art machine learning approaches to perform gene selection: recursive feature elimination, nearest-shrunken centroids and random forests. We discuss the difficulties that can be encountered when dealing with microarray data, such as selection bias, multiclass and unbalanced problems. The three approaches are then applied and compared on a typical cancer gene expression study.

3.1 Introduction

Microarray data allow the measurement of expression levels of tens of thousands of genes simultaneously on a single experiment. The biological aim of these experiments is to better understand interactions and regulations between genes, which are spotted on the array in some given conditions. For example, in the context of cancer data, there are several types of statistical problems that can be considered:

– To identify new tumour classes using gene expression signatures (e.g. cluster analysis, unsupervised learning)

G.J. McLachlan (✉)
Department of Mathematics and Institute for Molecular Bioscience, University of Queensland, 4072 St Lucia, Queensland, Australia
e-mail: g.mclachlan@uq.edu.au

T. Pham (ed.), *Computational Biology: Issues and Applications in Oncology*,
Applied Bioinformatics and Biostatistics in Cancer Research,
DOI 10.1007/978-1-4419-0811-7_3,

– To classify samples into known cancer classes (e.g. discriminant analysis, supervised learning)
– To identify marker genes that characterize one or several cancer types (i.e. feature selection)

Considering this last point, feature selection or *gene selection* may allow for the development of diagnostic tests to detect diseases and, in the particular case of cancer data, the selected genes can give more insight into the tumours characteristics. These genes are called *prognosis signatures* or *gene signatures*.

From a statistical point of view, the number of genes is usually often greater than the number of arrays, which renders the problem non-standard to solve. The selection of a relevant subset of genes enables one to improve the prediction performance of classification methods and to circumvent the curse of dimensionality. It also enables one to reduce computation time and allows for an understanding of the underlying biological process that generated these data.

Statistical analysis of microarray data involves several important steps, such as normalization and pre-processing; see McLachlan et al. (2004), Chap. 2 and Li et al. (2003). In this chapter, the focus is solely on the analysis of microarray data and the selection of genes using classification methods.

3.1.1 Notation

In this chapter, we will adopt the following notation. A microarray data set consists of the quantitative measurements of p genes (called *predictor variables*) on n tissues (called *samples*). These data are summarized in a p*n matrix $\boldsymbol{X} = x_{ij}$, where x_{ij} is the expression of gene i in the jth microarray ($i = 1, \ldots, p$; $j = 1, \ldots, n$)

In the context of classification, we can represent the a p*n matrix $\boldsymbol{X}$ of gene expressions as

$$\boldsymbol{X} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n),$$

where the feature vector x_j (the expression signature) contains the expression levels of the p genes in the jth tissue sample ($j = 1, \ldots, n$).

3.2 Supervised Classification

In the context of supervised classification, each tissue belongs to a known biological class k, $k = 1, \ldots, g$. In the following, we let $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$ denote the feature vectors and $z_1, \ldots, z_n$ the corresponding vectors of zero-one indicator variables defining the known class of each sample. The collection of the data

$$\boldsymbol{t} = \left(\boldsymbol{x}_1^{\mathrm{T}}, \boldsymbol{x}_1^{\mathrm{T}}, \ldots \boldsymbol{x}_{\mathrm{n}}^{\mathrm{T}}, \boldsymbol{x}_{\mathrm{n}}^{\mathrm{T}}\right)^{\mathrm{T}}$$

will be referred to as the *training* data.

In supervised classification, the aim is to construct a rule $r(\boldsymbol{x};\boldsymbol{t})$ based on the training data $\boldsymbol{t}$ with feature vectors for which the true class is known. On the basis of this rule, the final aim of supervised classification approaches is to predict the class label of a new tissue sample.

Such problems are ubiquitous and, as a consequence, have been tackled in several different research areas. As a result, a tremendous variety of algorithms and models have been developed for the construction of such rules. In the sequel, we will describe some classification methods, such as Support Vector Machines (SVMs), Shrunken centroids and classification trees. We will show that these classifiers can be included in some machine learning approaches to perform variable selection.

3.2.1 Linear Classifier

As an introduction to prediction rules, we first consider the basic linear function in the case where $g = 2$ (binary problem). For any feature vector, here denoted $\boldsymbol{x}$, its label is assigned to class 1 or class 2 if

$$\begin{aligned} r(\boldsymbol{x};\boldsymbol{t}) &= 1, \quad \text{if} \quad c(\boldsymbol{x};\boldsymbol{t}) > 0, \\ &= 2, \quad \text{if} \quad c(\boldsymbol{x};\boldsymbol{t}) < 0, \end{aligned}$$

where $c(\boldsymbol{x};\boldsymbol{t}) = \beta_0 + \beta^{\mathrm{T}}\boldsymbol{x} = \beta_0 + \beta_1(\boldsymbol{x})_1 + \cdots + \beta_i(\boldsymbol{x}) + \cdots + \beta_p(\boldsymbol{x})_p$, and $(\boldsymbol{x})_i$ denotes the ith element of the feature vector $\boldsymbol{x}(i = 1, \ldots, p)$.

The function $c(\boldsymbol{x};\boldsymbol{t})$ is a linear combination of the features (or genes) with different weights β_i $(i = 1, \ldots, p)$. Once the rule $r(\boldsymbol{x};\boldsymbol{t})$ is constructed on the training data $\boldsymbol{t}$, we can use it to predict the class label of a new feature vector.

3.2.2 Support Vector Machines

The SVM is a powerful machine learning tool that has often been applied to microarray data (Vapnik 2000). We briefly describe the formulation of a soft margin SVM, that is, when classes are linearly non-separable. In this section, we assign a label $y_j \in \{1, \ldots, g\}$ for $j = 1, \ldots, n$ to each tissue sample to indicate the known class of each sample.

In the case where $g = 2$, the SVM learning algorithm with a linear kernel aims to find the separating hyperplane

$$\beta^{\mathrm{T}}\boldsymbol{x} + \beta_0 = 0,$$

that is maximally equidistant from the training data of the two classes. In this case of $g = 2$, it is convenient if we let the class label y_j be 1 or -1 to denote membership of class 1 or class 2. When the classes are linearly separable, the hyperplane is

located so that there is maximal distance between the hyperplane and the nearest point in any of the classes. This distance is called the margin and is equal to $2/\boldsymbol{\beta}^{\mathrm{T}}\boldsymbol{\beta}$. The aim is to maximize this margin, that is, to minimize $\boldsymbol{\beta}^{\mathrm{T}}\boldsymbol{\beta}$.

When the data are not separable, the margin is maximized so that some classification errors are allowed. In that case, the so-called *slack* variables ξ_j are used ($j = 1, \ldots, n$).

The quadratic optimization problem to solve is:

$$\min_{\beta,\beta_0 \xi \in \mathrm{R}^{p+1+n}} \boldsymbol{\beta}^{\mathrm{T}}\boldsymbol{\beta}, \tag{3.1}$$

subject to

$$y_j\left(\boldsymbol{\beta}^{\mathrm{T}}\boldsymbol{x}_j + \boldsymbol{\beta}_0\right) \leq 1 - \xi_j, \tag{3.2}$$

where $\xi = (\xi_1, ..., \xi_n)^{\mathrm{T}}$ is the vector of so-called slack variables.

The cases $\boldsymbol{\beta}^{\mathrm{T}}x + \boldsymbol{\beta}_0 = \pm\left(1 - \xi_j\right)$ are the *support vectors* which define the solution. The Lagrangian dual formulation is finally

$$\begin{aligned} &\min \frac{1}{2}\textstyle\sum_{j,k}^{n} \alpha_j \alpha_k y_j y_k \boldsymbol{x}_j \cdot \boldsymbol{x}_k - \sum_j^n \alpha_j, \\ &\text{subject to} \qquad 0 \leq \alpha_j \leq \mathrm{C} \quad \text{and} \quad \textstyle\sum_j \alpha_j y_j = 0, \end{aligned} \tag{3.3}$$

where C corresponds to a penalty for misclassified cases and the $\alpha_j\,(j = 1, \ldots, n)$ are the Lagrange multipliers corresponding to the constraints (3.2). We call the *support vectors* the cases where $\alpha_j \neq 0$. The use of this 'soft' margin enables the misclassification of outliers during training and avoids overfitting.

Let S be the set of indices of the Support Vectors and $\boldsymbol{x}_s$ any Support Vector case, then given the solution to the problem (3.1), the corresponding discriminant rule is

$$r\,(\boldsymbol{x};\boldsymbol{t}) = \operatorname{sign}\left(y_s \sum\nolimits_{j \in s} \alpha_m y_m \boldsymbol{x}_m \cdot \boldsymbol{x}_s + \boldsymbol{\beta}_0\right).$$

By using the 'kernel trick' and the scalar product in the Lagrangian formulation (3.3), this standard SVM can be extended to nonlinear decision functions to map the data into a higher, possibly infinite, dimensional space. The user will then need to specify the kernel function to use. More details about the SVM methodology can be found in the tutorial of Burges (1998) and Cristianini and Shawe-Taylor (1999).

3.2.3 Nearest Centroid

The nearest centroid rule assigns the feature vector $\boldsymbol{x}$ to the class whose mean centroid is closest in Euclidian distance. For the classes $k = 1, \ldots, g$, let C_k be the indices of the n_k samples in class k. The ith component of the centroid for class k is $\overline{\mathbf{X}}_{ik} = \sum_{j \in c_k}^{n} X_{ij}/n_k$, which is the mean expression value in class k for the gene i. The ith component of the overall centroid is $\overline{\mathbf{X}}_i = \sum_{i=1}^{n} \mathrm{X}_{ij}/n$.

Nearest centroid classification takes the gene expression profile of a new sample, and compares it to each of these class centroids. The class whose centroid that it is closest to, in squared distance, is the predicted class for that new sample.

Note that in contrary to SVM, nearest centroid classifiers can be naturally generalized to multiclass problems ($g > 2$).

In the case of high dimensional microarray data, Tibshirani et al. (2002) proposed the 'nearest-shrunken centroid' rule that 'shrinks' each of the class centroids towards the overall centroid for all classes by moving the centroid towards zero by a *threshold*. It also takes into account the different gene variances. This approach has two advantages: (1) it can make the classifier more accurate by reducing the effect of noisy genes; and (2) it performs automatic gene selection (see Sect. 3.3).

3.2.4 Classification and Regression Trees

Tree-based methods such as Classification and Regression trees (CART, Breiman et al. 1984) are conceptually simple and easy to interpret. In our case, we will focus on binary classification trees only, that is, when a binary split is performed at each node of the tree.

The construction of CART requires one to choose:

1. The best split for each node, i.e. the best predictor (gene) to split the node and the best threshold among this predictor
2. A rule to declare a node 'terminal', i.e. when to stop splitting
3. A rule to affect a class to each terminal node

The best split criterion (1) relies on a *heterogeneity* function, so that the cases or samples that belong to the same class land in the same node. *Gini* index or *entropy* index is an example of such heterogeneity functions; see Breiman et al. (1984).

When applying classification trees to noisy data like microarray data, a major issue concerns the decision when to stop splitting (2). For example, if nine splits are performed (i.e. with AND/OR rules for each split) with only 10 observations, then it is easy to perfectly predict every single case. However, if new cases run this tree, it is highly likely that these cases will land in a terminal node with a wrong predicted class. This issue is called 'overfitting', that is, the model applied on the data does not generalize well to new data because of random noise or variation. The way to address this issue with CART is to stop generating new split nodes when subsequent splits only result in very little overall improvement of the prediction. This is called 'pruning'. The tree is first fully grown and the bottom nodes are then recombined or pruned upward to give the final tree, where the degree of pruning is determined by cross-validation (see Sect. 3.2.5.2) using a cost complexity function.

The class of the terminal node (3) is determined as the majority class of the cases that land in the same terminal node. Details of the CART methodology can be found in Breiman et al. (1984).

Trees are different from other previously considered classification methods as they are learning and selecting features simultaneously (embedded approach, see Sect. 3.1). However, one of the major problems with trees is their high variance. Indeed, a small change in the data can result in a very different series of splits and hence a different prediction for each terminal node. A solution to reduce the variance is to consider bagging (Breiman 1996) as was done in Random Forests, see Sect. 3.4.

3.2.5 Error Rate Estimation

Given a discriminant rule $r(\boldsymbol{x};\boldsymbol{t})$ constructed on some training data $\boldsymbol{t}$, we now describe some techniques to estimate the error rates associated with this rule.

3.2.5.1 Apparent Error Rate

The apparent error rate, also called *resubstitution* error rate, is simply the proportion of the samples in $\boldsymbol{t}$ that are misallocated by the rule $r(\boldsymbol{x};\boldsymbol{t})$. Therefore, this rate is obtained by applying the rule to the same data from which it has been learnt. As mentioned by several authors (McLachlan 1992, Chap. 10), it provides an overly optimistic assessment of the true error rate and would need some bias correction. To avoid this bias, the rule should be tested on an independent test set or a hold out test set from which the rule has not been formed. We next present some estimation methods to avoid this bias.

3.2.5.2 Cross-Validation

To almost eliminate the bias in the apparent error rate, one solution is to perform leave-one-out cross-validation (LOO-CV) or V-fold cross-validation (CV). Cross-validation consists in partitioning the data set into V subsets of roughly the same size, such that the learning of the rule $r(\boldsymbol{x};\boldsymbol{t})$ is performed on the whole subsets minus the vth subset, and tested on the vth subset, $v = 1,\ldots,V$. This is performed V times, such that each sample is tested once and the V subsequent error rates are then averaged.

In the case of LOO-CV, $V = n$ and therefore, the rule is tested on only one sample point for each fold. LOO-CV may seem to require considerable amount of computing, as the rule has to be formed n times to estimate the error rate. Furthermore, this estimate may yield a too high a variance. A bootstrap approach was then proposed in an attempt to avoid these limitations.

3.2.5.3 Bootstrap Approach

Efron (1979, 1983) showed that suitably defined bootstrap procedures can reduce the variability of the leave-one-out error in addition to providing a direct assessment

of variability for estimated parameters in the discriminant rule. Furthermore, if the number of bootstrap replications is less than n, it will result in some saving in computation time relative to LOO-CV computation.

Let E denote the error computed on the cases that were not drawn in the bootstrap sample, Efron (1983) proposed the B^{632} estimator to correct some upward bias in the error E with the downwardly biased apparent error A:

$$B^{632} = 0.368\, A + 0.632\, E.$$

Previously, McLachlan (1977) had derived an estimator similar to B^{632} in the special case of two classes with normal homocedastic distributions.

When the number of variables is much larger than the number of samples, the prediction rule $r(\boldsymbol{x};\boldsymbol{t})$ usually overfits, that is, A often equals 0. Efron and Tibshirani (1997) then proposed the B^{632+} estimate,

$$B^{632+} = (1-w)A + wE,$$

where

$$w = \frac{0.632}{1 - 0.368r}, \quad r = \frac{E - A}{\min(E, \gamma) - A} \quad \text{and} \quad \gamma = \sum_{k=1}^{g} p_k\,(1 - q_k)$$

r is an overfitting rate and γ is the no-information error rate, p_k is the proportion of samples from class C_k, q_k is the proportion of samples assigned to class C_k with the prediction rule $r(\boldsymbol{x};\boldsymbol{t})$ $(k = 1, \ldots, g)$.

3.3 Variable Selection

The so-called "large p small n problem" poses analytic and computational challenges. It motivates the use of variable selection approaches, not only to infer reliable statistical results and to avoid the curse of dimensionality, but also to select relevant and potential gene signature related to the tissue characteristics.

In the machine learning literature, there exists three types of classification and feature selection methods (Kohavi and John 1997; Guyon and Elisseeff 2003): the *filter* methods, the *wrapper* methods and the *embedded* methods. We first describe the particularities of these approaches, before detailing some useful wrapper and embedded methods to perform gene selection: Recursive Feature Elimination (RFE) (Guyon et al. 2002), Nearest Shrunken Centroids (Tibshirani et al. 2002) and Random Forests (Breiman 2001), that will be applied on one well-known microarray data set from Golub et al. (1999).

3.3.1 Filter, Wrapper and Embedded Approaches

The *filter methods* are often considered as a pre-processing step to select differentially expressed genes. The principle of this method is to independently test each gene and to order the genes according to a criterion, for example a p-value. The t- and F-tests are often used for microarray data. In one of the first comparative studies of classification methods in the context of microarray data, Dudoit and Fridlyand (2002) proposed to pre-process the genes based on the ratio of their between-groups to within-groups sum of squares:

$$\frac{\text{BBS}(i)}{\text{WSS}\,(i)} = \frac{\sum_{j,k} I_{yj=k}\,(\bar{\boldsymbol{x}}_{ik} - \bar{\boldsymbol{x}}_i)^2}{\sum_{j,k} I_{yj=k}\left(\boldsymbol{x}_{ij} - \bar{\boldsymbol{x}}_{ik}\right)^2},$$

where $\bar{\boldsymbol{x}}_i$ is the average expression level of gene i across all samples and $\bar{\boldsymbol{x}}_{ik}$ is the average expression level of the gene i across the samples that belong to class k.

They compared the performance of some classification methods, such as the k nearest neighbours (k-NN), CART and Linear Discriminant Analysis on a selection of 30–50 genes.

The main advantages of the filter methods are their computational efficiency and their robustness against overfitting. However, these methods do not take into account the interactions between genes, and they tend to select variables with redundant rather than complementary information (Guyon and Elisseeff 2003). Furthermore, the gene selection that is performed in the first step of the analysis does not take into account the performance of the classification methods that are applied in the second step of the analysis (Kohavi and John 1997).

The *wrapper* terminology was introduced by John et al. (1994). These methods involve successive evaluation of the performance of a gene subset and therefore, take the interactions between variables into account. The selection algorithm wraps the classification method, also named *classifier*, which evaluates the performance. The search for the optimal gene subset requires one to define (1) how to search the space of all possible variable subsets, (2) how to assess the prediction performance of a learning machine to guide the search and (3) how to halt it. Of course, an exhaustive search is an NP-hard problem and when p is large, the problem is intractable and requires stochastic approximations. Furthermore, there is a risk of overfitting if the number of cases n is small. The number of variables to select must be fixed by the user, or chosen according to a criterion, such as the classification error rate. One of the main disadvantages of these methods is their computational cost that increases with p. Nonetheless, the wrapper strategy might be superior to the filter strategy in terms of classification performance, as was first shown by Aha and Bankert (1995) and John et al. (1994) in an empirical manner.

The *embedded methods* include variable selection during the learning process, without the validation step, to maximize the goodness-of-fit and minimize the number of variables that are used in the model. A well-known example is CART, where the selected variables split each node of the tree. Other approaches include greedy

types of strategies, such as forward selection or backward elimination, that result in nested variable subsets. In a forward selection, variables are progressively included in larger and larger variable subsets, whereas the backward elimination strategy begins with all original variables and progressively discards the less relevant variables. According to the review of Guyon and Elisseeff (2003), these approaches are more advantageous in terms of computation time than wrapper methods, and should be robust against overfitting. The forward selection seems computationally more efficient than the backward elimination to generate nested variable subsets. However, the forward selection may select variable subsets that are not relevant, as the variable importance is not assessed with respect to the other variables, which are not included yet in the model. As opposed to wrapper methods, the embedded methods define the size of the variable selection, which is often very small.

3.3.2 Recursive Feature Elimination

RFE (Guyon et al. 2002) is an embedded method, which is based on a backward elimination and applies SVM to select an optimal non-redundant gene subset. The method relies on the fact that variables can be ranked on the basis of the magnitude of the coefficient β_i of each variable i when using a linear kernel SVM. In fact, each element β_i of the weight vector $\boldsymbol{\beta}$ is a linear combination of the cases, and most α_j are null, except for the *support* cases in the optimization problem (3.3). Consequently, the values in $\boldsymbol{\beta}$ can be directly interpreted as an importance measure of the variables in the SVM model. The variables i with the smallest weights $|\beta_i|$ will then be progressively discarded (recursive elimination) in the RFE procedure.

To speed up the computations, Guyon et al. (2002) proposed to discard several variables at a time in the algorithm, in spite of the fact that the classification performance may be altered. In this case, we obtain a ranking criterion on the variable subsets that are nested in each other, rather than a rank criterion on each variable. It is advised to discard variable subsets of various sizes, for example half of the variables in remaining set, to obtain sufficient density of information for the genes eliminated last. These latter will be ranked as first in the selection. Note that in any case, the user can actually choose the number of variables to select if desired.

RFE was first applied to microarray data (Ramaswamy et al. 2001) and was followed by numerous variants. SVM-RFE-annealing from Ding and Wilkins (2006) is based on a simulated annealing method and discards a large number of variables during the first iterations, and then reduces the number of discarded variables. This enables a significant reduction in computation time. This method, which is very similar to RFE, requires a choice of the number of variables to select. Another example is SVM-RCE for Recursive Cluster Elimination (Yousef et al. 2007), to select correlated gene subsets and avoid missing important genes with small weights as they were correlated with some dominant genes. This stresses the issue of correlated genes that bring redundant information. Should they all be in the selection even if genes with complementary information may get a lower rank? Or should the selection be larger?

Other variants were also proposed and the reader can refer to Tang et al. (2007), Mundra and Rajapakse (2007) or Zhou and Tuck (2007) for the MSVM-RFE for multi-class case. The abundant literature on this method shows the popularity of RFE for analysing microarray data.

3.3.3 Nearest Shrunken Centroids

Tibshirani et al. (2002) proposed a 'de-noised' version of the nearest-centroid rule defined in Sect. 3.2.3. The idea is to shrink the class centroids towards the overall centroids after standardizing by the within-class standard deviation for each gene. This gives higher weight to genes whose expression is stable within samples of the same class.

In the definition of the nearest centroid rule, the sample mean of the ith gene in class k $\bar{x}_{ik}$ is replaced by the shrunken estimate $\bar{x}^*_{ik} = \bar{x}_{ik} + m_k(s_i + s_0)d^*_{ik}$

With the following notations:

$$d^*_{ik} = \text{sign}\,(d_{ik})\,(|d_{ik}| - \Delta)_+\,, \tag{3.4}$$

where the $+$ means positive part and zero otherwise, and $d_{ik} = \frac{\bar{x}_{ik} - \bar{x}_i}{m_k(s_i + s_0)}$; s_i is the pooled within-class standard deviation for gene i and s_0 is the median value of the s_i over the set of genes;.$m_k = \sqrt{1/n_k - 1/n}$.

This shrinkage when computing d^*_{ik} is called soft thresholding, as the absolute value of d_{ik} is reduced by an amount of Δ and is set to zero if the result of (3.4) is less than zero. This allows variable selection and when Δ increases, many genes are eliminated from the class prediction and do not contribute to the nearest centroid computation. The shrinkage parameter Δ can be chosen by cross-validation.

Guo et al. (2007) then generalized the idea of Nearest Shrunken Centroids (NSC) with Shrunken Centroids Regularized Discriminant Analysis (SCRDA). Other variants of the NSC have been proposed in the literature for microarray data; see for example Dabney and Storey (2005) or Wang and Zhu (2007).

NSC are implemented in the *pamr* R package.

3.3.4 Random Forests

Some classification methods are sensitive to small perturbations in the initial data set. For example, the construction of CART can dramatically vary if some values are modified in the data set. If a model does not generalize well, i.e. if its variance is large, a solution proposed by Breiman (1996) is to aggregate classifiers. The variance is consequently reduced, but the classifier interpretation becomes more difficult. This is why these techniques are sometimes called 'black box'. Breiman (1996) proposed to aggregate CART to reduce their variance by estimating

each tree on a bootstrap sample. He introduced the bagging methodology for '*bootstrap aggregating*', by creating perturbed learning sets of the same size as the original sample. We describe a variant called random forests (Breiman 2001) where an additional perturbation is introduced when splitting the nodes of each tree to introduce more 'independence' between each tree.

Random forests is a wrapper method that became very popular for classification and feature selection in several contexts: Izmirlian (2004), Bureau et al. (2005), Diaz-Uriarte (2007) applied it with biological data; Svetnik et al. (2003) with QSAR data; Prasad et al. (2006) with ecological data, etc. This approach, which at first seemed empirical, is now theoretically studied, for example Biau et al. (2008) established some results regarding the consistency of aggregated trees.

The surprising paradox of random forests is that it benefits from the great instability of the classifiers CART by aggregating them. This approach combines two sources of randomness that largely improve the prediction accuracy: the bagging and the random feature selection to split each node of the tree. This results in low bias and low variance in the model.

Each tree (classification or regression) is constructed as follows:

1. B bootstrap samples $\{B_1, \ldots, B_B\}$ are drawn from the original data.
2. Each sample B_b $(b = 1, \ldots, B)$ is used as a training set to construct an unpruned tree T_b. Let p be the number of input variables of the tree, for each node of T_b, m variables are randomly selected $(m << p)$ to determine the decision at the node, where m is constant during the forest growing. Then, the best split among these m predictors is chosen to split the node.

The predictions of the B trees are then aggregated to predict new data either by majority vote for classification or by average for regression.

Random forests also generate an internal estimation of the generalisation error by computing the out-of-bag error rate for each bootstrap sample. However, this error rate, which seems accurate and unbiased, cannot be used to evaluate the performance of the variable selection. Indeed, Svetnik et al. (2003) showed that the OOB error estimate tends to overfit since the evaluation is not performed on an external test set. Instead, the variable selection should be evaluated on a test set sample. We will come back to the bias on the variable selection evaluation in Sect. 3.6.

The choice of the m randomly selected variables to split each node can be fixed by default to $\sqrt{p}$ (Liaw and Wiener 2003). However, the number of trees B must be chosen by the user. To obtain stable results, in particular when the number of cases is small, we strongly advise to set a large number of trees to be large, i.e. $\geq$5,000.

Two internal measures of variable importance are proposed in random forests, which allow for feature selection. These are called *Mean Decrease Accuracy* and *Mean Decrease Gini*. Both importance measures are described in Liaw and Wiener (2003) and in the *RandomForests* R package. Note that these measures can lead to different results if the data set contains a very small number of cases, or if some of the classes share similar (biological) characteristics.

3.3.5 Extension to Multiclass

3.3.5.1 Division into Binary Problems

Multiclass problems could make feature selection easier than binary problems, as the more classes, the better the gene subset for a perfect classification task (Guyon and Elisseeff 2003). But in practice, the multiclass case is difficult to deal with. Indeed, in the context of high dimensionality, the number of cases per class is usually smaller than in the binary problem due to experimental costs. This degrades the prediction accuracy when there are numerous classes. Furthermore, some authors noticed that most of the classification errors were due to cases belonging to very similar classes, rather than being outliers (Yeang et al. 2001).

Some binary classification methods are naturally adapted to multiclass problems. This is the case for example for Linear Discriminant Analysis, CART or Nearest Centroid. Other methods require the decomposition of the multiclass problem into several binary problems, such as one class against the other (*1 vs. 1*) or one class against the rest (*1 vs. rest*). Another solution is to define multiclass objective functions. This solution was often addressed with SVMs. For example, Weston and Watkins (1999) and Lee and Lee (2003) proposed to solve the multiclass optimization quadratic problem directly into the SVM, rather than aggregating binary SVMs. The authors concluded that there were a smaller number of support vectors by directly solving the multiclass case than by aggregating binary SVMs. However, it is still less costly to solve several small binary problems rather than a big complex multiclass problem.

Dividing a multiclass problem into several binary problems requires one to choose the appropriate aggregation method. For example with SVM, one could choose majority vote, least square estimation based on weighting that involves weighting each SVM, or double layer hierarchical combining that aggregates SVMs outputs into another SVM (Kim et al. 2003). The type of binary classifier must also been chosen. Lee and Lee (2003) showed that the 1 vs. rest SVM can give bad results if several classes are similar, and that the 1 vs. 1 SVM may contain a high variance, as each binary classifier is computed on a very small subset of cases with only one misclassifying cost for all classes. This latter problem is partly due to unbalanced classes.

A comparative study of several multiclass SVM approaches such as the Weston and Watkins (1999) or Lee and Lee (2003) approaches, 1 vs. rest, and 1 vs. 1 was presented in Statnikov et al. (2005) for microarray data, with first an initial preprocessing step with a filter method.

3.3.5.2 Unbalanced Multiclass Problems

In addition to numerous classes in microarray data, one often faces unbalanced classes. The main reason is that the class of interest is the rare one where data are difficult to obtain. There has been little attention given to the problem of unbalanced

multiclasses in the context of microarray data, although Eitrich and Lang (2006) and Qiao and Liu (2008) recently address this issue for general classification purposes.

The main concern when performing feature selection in a classification context is that a classifier aims at minimizing the overall classification error rate. It thus minimizes the classification error rate of the majority classes, to the detriment of the minority classes. This type of approach has a serious drawback when performing feature selection, as the selected genes will mainly discriminate against the majority classes which are not necessarily the most biologically relevant.

In the case of random forests, Chen et al. (2004) proposed two approaches to balance the classes and to introduce a higher penalty when a minority class is misclassified. The first approach, called *Balanced Random Forests* (BRF), is based on a re-sampling technique. Each tree is constructed on the same number of cases in the majority and minority classes (sampling with replacement). The second approach, called *Weighted Random Forests* (WRF, currently implemented in the *RandomForests* R package), is based on cost sensitive learning. Weights are introduced in the RF algorithm, first during the tree construction, where class weights are used when splitting the nodes with the Gini criterion, and second when assigning the class of the terminal node.

However, BRF risks overfitting the data if the number of cases in the minoritary class is very low, as this down sampling approach does not use many cases in the majoritary classes. The inclusion of weights into the feature selection algorithm seems a better approach and was proposed in Lê Cao et al. (2009) in a stochastic wrapper algorithm.

For the SVM case, Qiao and Liu (2008) recently proposed an adaptive weighted learning procedure in the multiclass quadratic formulation of Lee and Lee (2003) to optimally weight each class.

3.3.6 Selection Bias and Performance Assessment

In the classification context, the performance assessment of a variable selection remains difficult due to the small number of samples. As Dudoit and Fridlyand (2002) underlined, more cases would be needed to compute an accurate classification error rate. It is often unfeasible to obtain an external test set and the performance evaluation must often be computed on the learning set. Furthermore, several authors warn of the selection bias problem (Ambroise and McLachlan 2002; Reunanen 2003). Indeed, some articles presented extremely optimistic results as the classification error rate estimation *and* the variable selection were both performed on the learning set. Therefore, to correct for this selection bias, it is essential that cross-validation or the bootstrap be used external to the gene selection process.

In the present context where feature selection is used in training the prediction rule $r(\boldsymbol{x};\boldsymbol{t})$ from the full training set, the same feature selection method must be implemented in training the rule on the $V-1$ subsets combined at each stage of an cross-validation of $r(\boldsymbol{x};\boldsymbol{t})$ for the selected subset of genes. Of course, there is

no guarantee that the same subset of genes will be obtained as during the original training of the rule (on all the training observations). Indeed, with the huge number of genes available, it generally will yield a subset of genes that has at most only a few genes in common with the subset selected during the original training of the rule.

In the case where the final version of the discriminant rule is based on a small subset selected in some optimal way from a much larger set of variables (genes), it is important that cross-validation is undertaken as described earlier, as otherwise a large selection bias can result; see Ambroise and McLachlan (2002), McLachlan et al. (2008), Wood et al. (2007), Zhu et al. (2008).

3.3.7 Optimal Size of the Selection

Choosing the optimal size of the selection is a difficult question as the small number of samples does not allow for an accurate estimate of the classification error. A naïve choice would be to select a number of genes that gives the lowest error rate. However, McLachlan et al. (2004, Chap. 7) showed on the van't Veer et al. (2002) study that the estimated error rate needed to be corrected for bias. The authors showed that the minimum error rate was attained for approximately 256 genes when evaluating the gene selection with bias correction on the whole data set, instead of only 70 genes as originally proposed in this study.

A solution to choose the optimal set of genes would be to select the genes which give a stabilized error rate and, therefore, consistent predictive results.

3.4 Illustrative Example with the Golub Data Set

3.4.1 Performance of the Three Feature Selection Methods

As an illustrative example, we considered the well known leukaemia data set (Golub et al. 1999), where Affymetrix oligonucleotide arrays were used to measure gene expressions in two types of acute leukaemias: acute lymphoblastic leukaemia (ALL) and acute myeloid leukaemia (AML). The entire data set consists in 72 tissue samples, among which 47 are ALL cases and 25 are AML, and the measurement of 7,129 genes. The data set was pre-processed as in Dudoit and Fridlyand (2002) by filtering and log transforming the data. The final data set comprises 3,731 genes.

We performed external tenfold cross validation $A^{(CV10E)}$ as used by Ambroise and McLachlan (2002) for different sizes of selected subsets of genes to evaluate the performance of RFE, Nearest Shrunken Centroids (NCS) and Random Forests (RF). For the tenfolds, we divided the 72 tissues into balanced training and test sets such that approximately 42 ALL and 22 AML were used for training, 5 ALL, and 3 AML were used for testing in the binary problem. We calculated the 10-CV estimated error rates over 50 random splits.

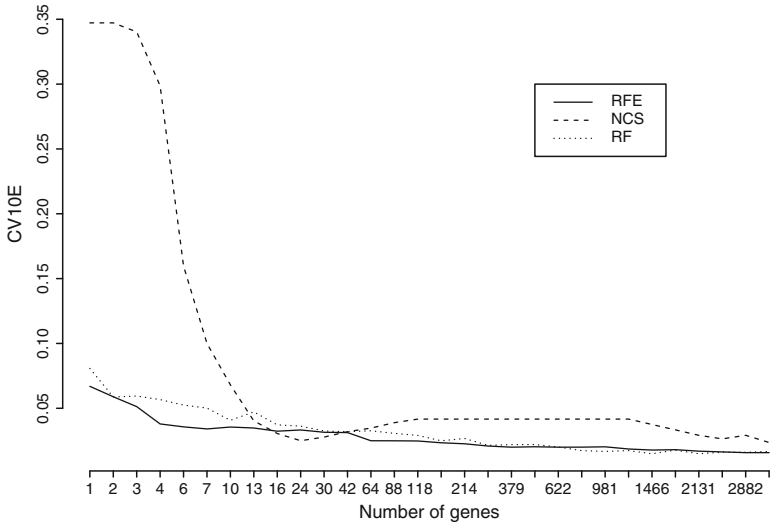

Fig. 3.1 Estimation of the classification error rate for each method with external tenfold cross-validation (repeated 50 times) with respect to the number of genes selected, for the binary problem

The averaged values of these estimates are plotted in Fig. 3.1. It can be seen from this figure that all three wrapper methods perform similarly, except for NCS that requires a larger selection of genes to be competitive with the other approaches. $A^{(CV10E)}$ was found to have little bias when estimating the error rate (Ambroise and McLachlan 2002). However, the conclusion about this graph should be taken with caution, as the error rate should be corrected for bias.

As an illustrative example, this data set is of interest as the ALL cases can be divided into two subclasses, called ALL-B cells (38 samples) and ALL-T cells (9 samples). We are here in the typical case of unbalanced multiclass data set, where the ALL-T class is the minority class. Therefore, when performing external balanced tenfold cross-validation, 34 ALL-B, 8 ALL-T and 22 AML were used for training and approximately 4 ALL-B, 1 ALL-T and 3 AML were used for testing. The fact that there is only one ALL-T sample in the test set may severely affect the estimation of a too optimistic error rate. Indeed, as mentioned in Sect. 3.3, when computing $A^{(CV10E)}$, we tend to neglect misclassified cases from of the minority class.

The averaged values of the $A^{(CV10E)}$ estimates are plotted in Fig. 3.2 over 50 random splits. In this multiclass problem, the estimated error rate is higher than in the binary case presented above where the gene selection is small. Therefore, a larger selection of genes might be advisable for further biological validation. Interestingly, the stabilized error rate seems to be similar to the one obtained in the binary case (around 5%). This may be due to the fact that there is only one ALL-T sample in the test set that can be misclassified. This may result in a too optimistic estimation of the error rate. Indeed (not shown), the error rate for RFE was between 0.1 and 0.2 for the ALL-T minority class, and between 0.01 and 0.02 for the two other

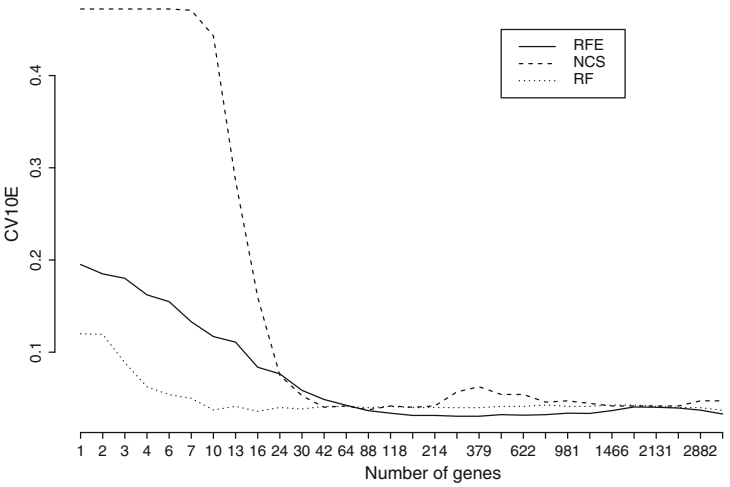

Fig. 3.2 Estimation of the classification error rate for each method with external tenfold cross-validation (repeated 50 times) with respect to the number of genes selected, for the unbalanced multiclass problem

classes. Since in this last example the classes are strongly unbalanced, a better way to take into account the minority class would be to weight the error rate estimation according to the proportion of samples in each class, as was proposed in Lê Cao et al. (2009).

3.4.2 Comparison of the Gene Selections

We arbitrarily chose a selection size of 50 genes and compared the overlap between the selected genes resulting with each approach, for the binary and the 3-class cases (Fig. 3.3). Note that the same trend was observed when the selection size was increased.

It is interesting to see that although each approach uses a different classifier, a fair amount of genes are commonly selected by the three methods (20 and 15 genes for the binary and the multiclass problems). Therefore, these approaches have the ability to select (the same) discriminative genes and these discriminative genes may be of potential relevance for the biological experiment.

Half of the genes selected with RFE differed from those selected with RF and NSC. This difference might be due to the fact that as a backward technique, RFE tends to select non-redundant and non-correlated genes (Yousef et al. 2007), whereas NSC and RF can highlight correlated genes in their selections.

As expected, when the number of classes increased from $g = 2$ to $g = 3$, the overlap between all three methods became smaller. This can be explained by the increasing complexity of the data set, where numerous subsets can lead to a good classification of the samples.

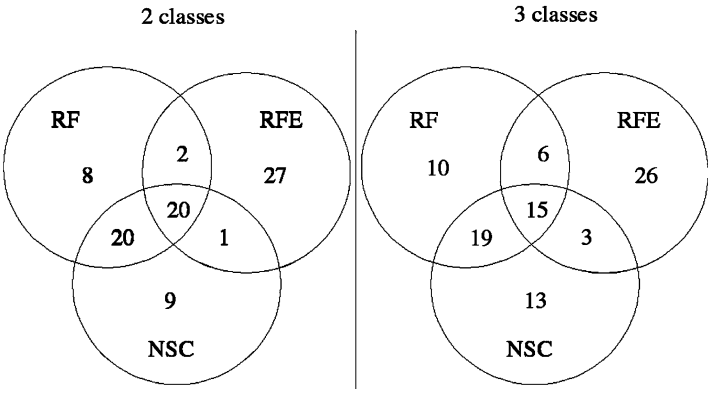

Fig. 3.3 Venn diagrams. Overlap between the gene lists selected with Random Forests, Recursive Feature Elimination, and Nearest Shrunken Centroids (selection of 50 genes for each method)

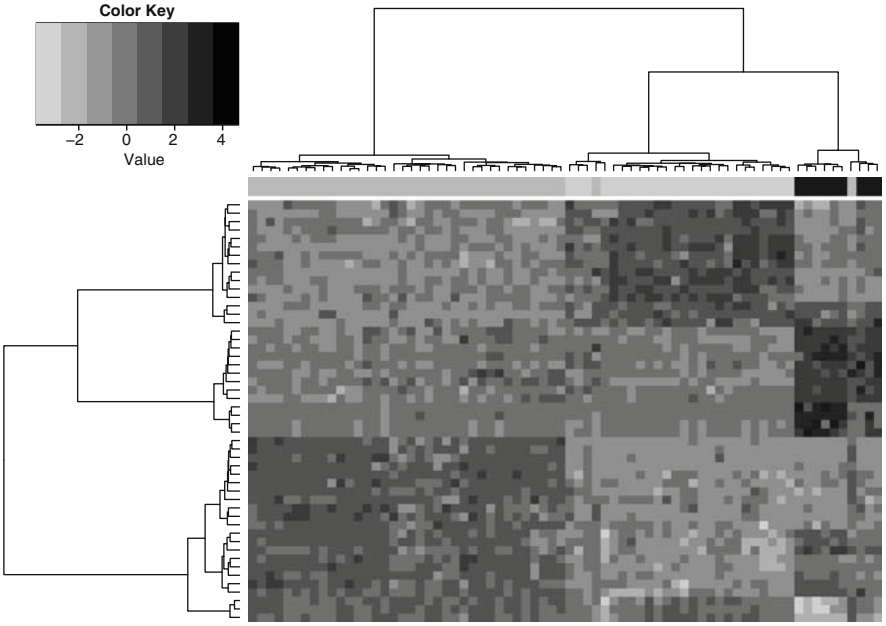

Fig. 3.4 Heat map for the 50 genes selected with Nearest Shrunken Centroids for the unbalanced multiclass problem. *Rows* (genes) and *columns* (tissues) are arranged according to a hierarchical clustering method. Tissue classes are indicated by color bars on the upper dendogram (*black*: ALL-T, *grey*: AML, and *light grey*: ALL-B)

As an illustrative example, Figure 3.4 displays the heat map of the 50 genes selected with NSC for the multiclass case, with Euclidian distance and Ward aggregation method. This type of unsupervised clustering enables a global overview of the genes that were selected with respect to each tissue sample.

For the binary problem (not shown), it was surprising to see that although RFE selected genes with a poor contrast (mostly under-expressed genes), it allowed for a perfect classification of the tissue samples, whereas RF and NSC seemed to select interesting and contrasted gene clusters, but with one misclassified sample.

The same trend could be observed for the 3-class case, where contrasted gene clusters were obtained with RF and NSC (Fig. 3.4). One would expect the ALL-T and ALL-B to share the same dendogram as they are a subclass of ALL. In fact it is the AML samples that seem to share similarities with ALL-B with these gene selections.

3.4.3 Choice of Method

The large difference between the three feature selection methods, and therefore the three gene selections did not really appear when estimating the classification error rate (Figs. 3.1 and 3.2). It is highly probable that different gene subsets can lead to the same classification performance of a given classifier. Nevertheless, some genes were commonly selected by all three approaches, despite the fact that these statistical approaches differ in their construction and the classifiers they use. It is therefore difficult to choose the appropriate statistical method to perform variable selection and we cannot have a definitive answer for this question. Microarray data are very complex and the statistical outcome highly depends on the biological experiment, design and the quality of the data. Furthermore, some statistical approaches might be appropriate in one study, but not in another. Therefore, one has to take into account different criteria as proposed in this illustrative section, compare several statistical approaches, as well as to investigate the biological relevance of the selected genes related to the biological experiment.

3.5 Validation

Validation of the results has been often discussed in the literature. Once the gene signatures have been selected, their clinical utility must be established. For example, they must prove to reliably identify patients with poor or good prognosis. The first step consists in validating the microarray experiment, while the second part consists in an independent validation using these gene signatures.

3.5.1 Biological Interpretation

Once the gene signatures have been selected using a statistical approach, it is of tradition to validate the results and look for false positive by using the same samples,

but on different mRNA measurement procedure, such as reverse-transcriptase PCR. This may highlight erroneous inferences due to poor measurement quality. However, repeating measurement on the same biological samples but with a different measurement technique is a highly debatable practice to validate the microarray experiment.

Post hoc analysis is then required to assess the biological relevance of the gene list. For example, pathway analysis, using softwares such as DAVID (Dennis et al. 2003), Panther (Mi et al. 2005), FatiGO (Al-Shahrour et al. 2004)-to cite a few, can be performed on the gene selection to identify biological functions and networks; see also Lê Cao et al. (2007) for an example of such analysis. This type of analysis also enables one to highlight other genes that are strongly correlated to the selected genes and interact with these genes in biological pathways, but might not be spotted on the microarray, or were discarded during the pre-processing step because of poor quality spots.

3.5.2 Independent Test Set

The gene signatures then need to be proven that they provide additional information to the clinicopathologic risk criteria that are currently used in the clinic. The validation must hence be performed completely independently, not only on a new batch of patients, but also by external institutions to the study. In addition, it should also be applied to a prospective study, rather than using retrospective data of patient that may not be representative of the nowadays breast cancer population.

Buyse et al. (2006) performed this type of analysis, using independent statisticians and multinational collaborations to assess the usefulness of the 70-gene signature in breast cancer on a retrospective study. They showed that this set of gene had reproducible prognostic value across different patient populations, laboratories and biostatistical facilities. However, many questions remain, such as the lack of gene overlap among different studies (Michiels et al. 2005). Some authors argue that these different gene selection that predict the same outcome might be the result of differences among microarray platforms, but also the differences among the genes spotted on the array or the different experimental conditions. Others state that the resulting lists of genes are highly unstable as it depends on the patients on the training set.

3.6 Conclusion

Microarray technology is a promising and a powerful high-throughput tool for researchers in many fields of biology and medicine. Microarray analysis has the potential to refine cancer prognosis, well beyond the currently used clinical parameters to predict disease outcome. Diagnostic assays developed on gene expression profiling studies will therefore benefit to oncology and other areas of medicine.

Many studies showed that supervised classification methods appear to be one of the best approaches to identify prognostic and predictive profiles (Golub et al. 1999; van't Veer et al. 2002; Nuyten and van de Vijver 2008). Further studies are required to check the consistency of the results obtained with these sophisticated statistical approaches before they can replace the current clinical and pathological indicators and be made available to patients.

It would be interesting to further investigate the integration of clinical data and microarray data to improve the prediction performance of the classification methods. Gevaert et al. (2006) and McLachlan and Ng (2008) have shown that a significant improvement could be achieved by using Bayesian networks or expert networks to integrate both discrete and continuous data on the van't Veer breast data set. Clinical variables are often under-used when analysing microarray data. Combined with often noisy gene expresion data, they would allow for a better cancer prognosis as they have a very low noise level (Gevaert et al. 2006).

References

Aha DW and Bankert RL (1995) A comparative evaluation of sequential feature selection algorithms. In: Learning from data: artificial intelligence and statistics V. Springer, New York, pp 199–206

Al-Shahrour F, Diaz-Uriarte R, Dopazo J (2004) FatiGO: a web tool for finding significant associations of Gene Ontology terms with groups of genes. Bioinformatics 20:578–580

Ambroise C, McLachlan G (2002) Selection bias in gene extraction on the basis of microarray gene-expression data. Proc Natl Acad Sci 99:6562–6566

Biau G, Devroye L, Lugosi G (2008) Consistency of random forests and other averaging classifiers. J Mach Learn Res 9:2015–2033

Breiman L (1996) Bagging predictors. Mach Learn 24:123–140

Breiman L (2001) Random forests. Mach Learn 45:5–32

Breiman L, Friedman J, Olshen R, Stone C (1984) Classification and regression trees, The Wadsworth statistics/probability series, Belmont, CA

Bureau A, Dupuis J, Falls K, Lunetta K, Hayward B, Keith T, Van Eerdewegh P (2005) Identifying SNPs predictive of phenotype using random forests. Genet Epidemiol 28:171–182

Burges C (1998) A tutorial on support vector machines for pattern recognition. Data Min Knowl Discov 2:121–167

Buyse M, Loi S, van't Veer L, Viale G, Delorenzi M, Glas A, Saghatchian d'Assignies M, Bergh J, Lidereau R, Ellis P (2006) Validation and clinical utility of a 70-gene prognostic signature for women with node-negative breast cancer. J Natl Cancer Inst 98:1183–1192

Chen C, Liaw A, Breiman L (2004) Using random forests to learn unbalanced data, Department of Statistics, University of Berkeley

Cristianini N, Shawe-Taylor J (1999) An introduction to support vector machines: and other kernel-based learning methods, Cambridge University Press, New York

Dabney A, Storey J (2005) Optimal feature selection for nearest centroid classifiers, with applications to gene expression microarrays. UW Biostatistics Working Paper Series, Article 267

Dennis G Jr, Sherman B, Hosack D, Yang J, Gao W, Lane H, Lempicki R (2003) DAVID: database for annotation, visualization, and integrated discovery. Genome Biol 4:Article R60

Diaz-Uriarte R (2007) GeneSrF and varSelRF: a web-based tool and R package for gene selection and classification using random forest. BMC Bioinformatics 7:Article 328

Ding Y, Wilkins D (2006) Improving the performance of SVM-RFE to select genes in microarray data. BMC Bioinformatics 7:Article S12

Dudoit S, Fridlyand J (2002) Comparison of discrimination methods for the classification of tumors using gene expression data. J Am Stat Assoc 97:77–87

Efron B (1979) Bootstrapping methods: another look at the jackknife. Ann Stat 7:1–26

Efron B (1983) Estimating the error rate of a prediction rule: improvement on cross-validation. J Am Stat Assoc 78:316–331

Efron B, Tibshirani R (1997) Improvements on cross-validation: the. 632 + bootstrap method. J Am Stat Assoc 92:548–560

Eitrich T, Lang B (2006) Efficient optimization of support vector machine learning parameters for unbalanced datasets. J Comput Appl Math 196: 425–436

Gevaert O, Smet F, Timmerman D, Moreau Y, Moor B (2006) Predicting the prognosis of breast cancer by integrating clinical and microarray data with Bayesian networks. Bioinformatics 22:184–190

Golub T, Slonim D, Tamayo P, Huard C, Gaasenbeek M, Mesirov J, Coller H, Loh M, Downing J, Caligiuri M (1999) Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. Science 286:531–537

Guo Y, Hastie T, Tibshirani R (2007) Regularized linear discriminant analysis and its application in microarrays. Biostatistics 8:86–100

Guyon I, Elisseeff A (2003) An introduction to variable and feature selection. J Mach Learn Res 3:1157–1182

Guyon I, Weston J, Barnhill S, Vapnik V (2002) Support vector machine with recursive feature selection. Mach Learn 46:389–422

Izmirlian G (2004) Application of the random forest classification algorithm to a SELDI-TOF proteomics study in the setting of a cancer prevention trial. Ann New York Acad Sci 1020: 154–174

John G, Kohavi R, Pfleger K (1994) Irrelevant features and the subset selection problem. In: Proceedings of the Eleventh International Conference on Machine Learning, New Brunswick, NJ, USA, Morgan Kaufmann

Kim H, Pang S, Je H, Kim D, Yang Bang S (2003) Constructing support vector machine ensemble. Pattern Recogn 36:2757–2767

Kohavi R, John G (1997) Wrappers for feature subset selection. Artif Intell 97:273–324

Lê Cao K-A, Bonnet A, Gadat S (2009) Multiclass classification and gene selection with a stochastic algorithm. Comput Stat Data Anal 53:3601–3615

Lê Cao K-A, Goncalves O, Besse P, Gadat S (2007) Selection of biologically relevant genes with a wrapper stochastic algorithm. Stat Appl Genetics Mol Biol 6:Article 29

Lee Y, Lee C (2003) Classification of multiple cancer types by multicategory support vector machines using gene expression data. Bioinformatics 19:1132–1139

Li C, Tseng G, Wong W (2003) Model-based analysis of oligonucleotide arrays and issues in cDNA microarray analysis. In: Speed T (ed) Statistical analysis of gene expression microarray data. Chapman & Hall, New York, pp 1–34

Liaw A, Wiener M (2003) Classification and regression by randomForest. R News 2/3:18–22

McLachlan G (1977) A note on the choice of a weighting function to give an efficient method for estimating the probability of misclassification. Pattern Recogn 9:147–149

McLachlan G (1992) Discriminant analysis and statistical pattern recognition. Wiley, New York

McLachlan G, Chevelu J, Zhu J (2008) Correcting for selection bias via cross-validation in the classification of microarray data. In: Balakrishnan N, Pena E, Silvapulle MJ (eds) Beyond parametrics in Interdisciplinary research: Festschrift in Honor of Professor Paranab K. Sen. Hayward, Vol 1. IMS Collections, California, pp 364–376

McLachlan G, Do K, Ambroise C (2004) Analyzing microarray gene expression data. Wiley-Interscience, New York

McLachlan G, Ng S-K (2008) Expert networks with mixed continuous and categorical feature variables: a location modeling approach. In: Peters H, Vogel M (eds) Machine learning research progress. Hauppauge, New York, pp 1–14

Mi H, Lazareva-Ulitsky B, Loo R, Kejariwal A, Vandergriff J, Rabkin S, Guo N, Muruganujan A, Doremieux O, Campbell M (2005) The PANTHER database of protein families, subfamilies, functions and pathways. Nucleic Acids Res 33:284–288

Michiels S, Koscielny S, Hill C (2005) Prediction of cancer outcome with microarrays: a multiple random validation strategy. The Lancet 365:488–492

Mundra P, Rajapakse J (2007) SVM-RFE with relevancy and redundancy criteria for gene selection. Lect Notes Comp Sci 4774:242–252

Nuyten D, van de Vijver M (2008) Using microarray analysis as a prognostic and predictive tool in oncology: focus on breast cancer and normal tissue toxicity. In: Seminars in radiation oncology, pp 105–114

Prasad A, Iverson L, Liaw A (2006) Newer classification and regression tree techniques: bagging and random forests for ecological prediction. Ecosystems 9:181–199

Qiao X, Liu Y (2008) Adaptive weighted learning for unbalanced multicategory classification. Biometrics (in press)

Ramaswamy S, Tamayo P, Rifkin R, Mukherjee S, Yeang C, Angelo M, Ladd C, Reich M, Latulippe E, Mesirov J (2001) Multiclass cancer diagnosis using tumor gene expression signatures. Proc Natl Acad Sci 98:15149–15154

Reunanen J (2003) Overfitting in making comparisons between variable selection methods. J Mach Learn Res 3:1371–1382

Statnikov A, Aliferis C, Tsamardinos I, Hardin D, Levy S (2005) A comprehensive evaluation of multicategory classification methods for microarray gene expression cancer diagnosis. Bioinformatics 21:631–643

Svetnik V, Liaw A, Tong C, Culberson J, Sheridan R, Feuston B (2003) Random forest: a classification and regression tool for compound classification and QSAR modeling. J Chem Inform Comp Sci 43:1947–1958

Tang Y, Zhang Y, Huang Z (2007) Development of two-stage SVM-RFE gene selection strategy for microarray expression data analysis. IEEE ACM Trans Comput Biol Bioinformatics 4:365–389

Tibshirani R, Hastie T, Narasimhan B, Chu G (2002) Diagnosis of multiple cancer types by shrunken centroids of gene expression. Proc Natl Acad Sci 99:6567–6572

van't Veer L, Dai H, Van de Vijver M, He Y, Hart A, Mao M, Peterse H, Van der Kooy K, Marton M, Witteveen A (2002) Gene expression profiling predicts clinical outcome of breast cancer. Nature 415:530–536

Vapnik V (2000) The nature of statistical learning theory, Springer, New York

Wang S, Zhu J (2007) Improved centroids estimation for the nearest shrunken centroid classifier. Bioinformatics 23:972–979

Weston J, Watkins C (1999) Multi-class support vector machines. In: Proceedings ESANN, Brussels, Belgium

Wood I, Visscher P, Mengersen K (2007) Classification based upon gene expression data: bias and precision of error rates. Bioinformatics 23:1363–1370

Yeang C, Ramaswamy S, Tamayo P, Mukherjee S, Rifkin R, Angelo M, Reich M, Lander E, Mesirov J, Golub T (2001) Molecular classification of multiple tumor types. Bioinformatics 17:316–322

Yousef M, Jung S, Showe L, Showe M (2007) Recursive cluster elimination (RCE) for classification and feature selection from gene expression data. BMC Bioinformatics 8:144

Zhou X, Tuck D (2007) MSVM-RFE: extensions of SVM-RFE for multiclass gene selection on DNA microarray data. Bioinformatics 23:1106–1114

Zhu J, McLachlan G, Ben-Tovim Jones L, Wood I (2008) On selection biases with prediction rules formed from gene expression data. J Stat Plann Infer 138:374–386

Chapter 4
Agent-Based Modeling of Ductal Carcinoma In Situ: Application to Patient-Specific Breast Cancer Modeling

Paul Macklin, Jahun Kim, Giovanna Tomaiuolo, Mary E. Edgerton, and Vittorio Cristini

Abstract Ductal carcinoma in situ (DCIS) of the breast is the most common precursor to invasive carcinoma (IC), the second-leading cause of death in women in USA. There has been great progress in modeling DCIS at both the cellular scale (e.g., using cellular automata and agent-based models) and the population scale (e.g., using partial differential equations or systems of ordinary differential equations), but these past efforts have been difficult to calibrate with patient-specific molecular and cellular measurements. We develop a biophysically justified, agent-based cellular model of DCIS that is well-suited to patient-specific calibration. The model is modular in nature and can thus be readily extended to incorporate more advanced biology. We give an example of recently developed, patient-specific calibration of the model and conduct parameter studies that generate testable biological hypotheses.

4.1 Introduction

Ductal carcinoma in situ (DCIS) is the most prevalent precursor to invasive breast cancer (IC), the second-leading cause of death in women in USA. The American Cancer Society predicted that 50,000 new cases of DCIS alone (excluding lobular carcinoma in situ) and 180,000 new cases of IC would be diagnosed in 2007 (Jemal et al. 2007; American Cancer Society 2007). Coexisting DCIS is expected in 80% of IC, or 144,000 cases (Lampejo et al. 1994). Because DCIS is a known precursor to IC, this leads us to hypothesize that up to 75% of DCIS cases progress to invasion prior to detection by screening mammography. While DCIS itself is not a life-threatening disease, it is a very important precursor to invasive breast cancer because (1) it can be treated and (2) if left untreated, it is likely to progress to IC, which is a deadly disease (Page et al. 1982; Kerlikowske et al. 2003; Sanders et al. 2005; Collins et al. 2005).

P. Macklin (✉)
Assistant Professor of Health Informatics, School of Health Information Sciences, University of Texas Health Science Center, Houston, TX, USA
e-mail: Paul.T.Macklin@uth.tmc.edu

T. Pham (ed.), *Computational Biology: Issues and Applications in Oncology*, Applied Bioinformatics and Biostatistics in Cancer Research,
DOI 10.1007/978-1-4419-0811-7_4,

Women prefer breast conserving surgery (BCS), also known as lumpectomy, vs. complete mastectomy to treat DCIS (Silverstein 1997b); in USA today, approximately two-thirds of women diagnosed with DCIS will opt for BCS over mastectomy. Women who undergo BCS face two problems. First, an estimated 38–72% of women seeking BCS will not have their entire tumor removed in one surgery and may require up to three surgeries (called re-excisions) for complete removal of the DCIS (Cheng et al. 1997; Cabioglu et al. 2007; Dillon et al. 2007). Second, DCIS recurs at the same location greater than 20% of the time in patients who undergo BCS alone (Patani et al. 2008). To combat this recurrence, women are advised to undergo radiation therapy to the breast, which induces residual cells of DCIS to apoptose. Even in women who have been treated with surgery and radiation, DCIS recurs approximately 10% of the time (Patani et al. 2008). Half of these recurrences already show progression to invasive cancer (IC). The single most important underlying problem that contributes to both re-excisions and recurrences is DCIS that is left inside the breast (Silverstein 1997a).

Hence, predicting the size and shape of DCIS is critical to successfully eradicating the disease in patients and preventing recurrences that often progress to deadlier invasive carcinoma. In addition, understanding the progression from DCIS to IC is key to developing future treatments to improve patient survival. Mathematical modeling can play a role in both these tasks. In this chapter, we introduce an agent-based model of DCIS that is well-suited to patient-specific calibration, can be modularly extended to focus attention on specific aspects of biological interest, and can be used for generating testable scientific hypotheses. In ongoing and future work, we shall incorporate the model developed here into a broader, multiscale framework capable of making patient-specific, clinical predictions of DCIS outcome (Edgerton et al. 2008, in preparation-a; Chuang et al. in preparation; Macklin et al. in preparation; Cristini and Lowengrub in preparation).

4.1.1 Biology of Breast Duct Epithelium

As an organ, the breast is organized as a system of 12–15 independent, largely parallel duct systems: clusters of milk-producing lobules that feed into a branched duct system that terminates at the nipple (Wellings et al. 1975; Moffat and Going 1996; Ohtake et al. 2001; Going and Mohun 2006). The duct systems are separated by supporting ligaments and fatty tissue and drained by the lymphatic system (Tannis et al. 2001). The ducts have a well-characterized microarchitecture: each duct is a tubular arrangement of epithelial cells, surrounded by myoepithelial cells (epithelial cells with muscle-like properties, such as contracting the duct to transport milk) and a basement membrane (hereafter BM). The center of the duct, known as the lumen, is filled with either milk (during lactation) or fluid (see Fig. 4.1, top left). Surrounding and supporting the duct is the stroma: a scaffolding of collagen and other fibers (collectively called the extracellular matrix, or ECM) that is secreted and maintained by fibroblast cells. The stroma also contains blood vessels that supply oxygen, glucose,

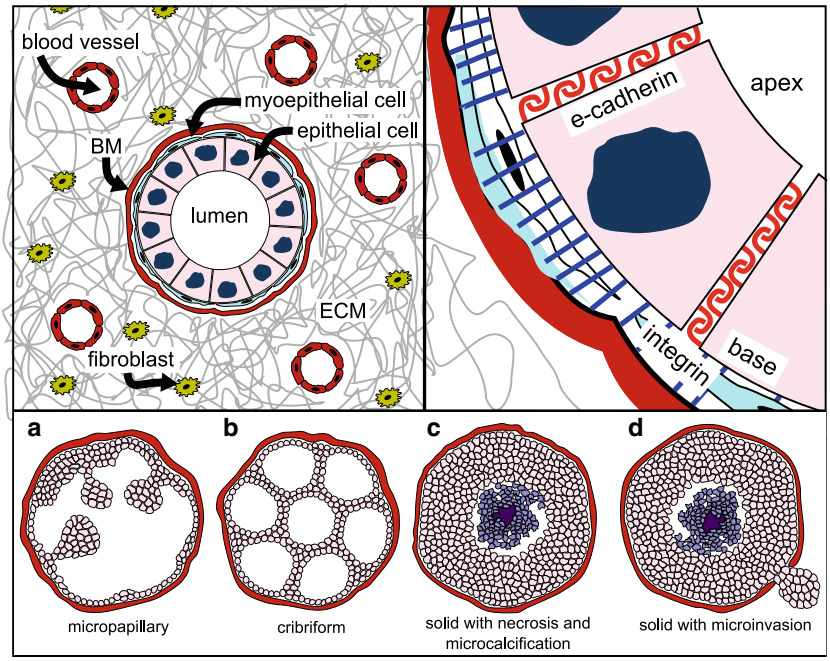

Fig. 4.1 *Top left*: Typical breast duct microarchitecture. *Top right*: Breast duct epithelial cell polarization. *Bottom*: Major DCIS types and IDC. Reprinted with permission from Macklin et al. (in preparation)

and growth factors to the tissue. A key aspect of this architecture is the cells in the breast duct have no direct access to nutrients; instead, these must diffuse into the duct through the BM.

The arrangement of the epithelial cells in the duct depends upon the polarization of the cells and the anisotropic distribution of different surface adhesion molecules. Integrins line the cell base and adhere to several ligands (generally laminin and fibronectin) on the basement membrane; E-cadherin molecules cover the cell surface between the base and apex and adhere to E-cadherin molecules on neighboring cells (Butler et al. 2008) (see Fig. 4.1, top right). The careful orchestration of integrin-mediated cell–BM adhesion and E-cadherin-mediated cell–cell adhesion helps determine the geometry of the duct (Hansen and Bissell 2000; Wei et al. 2007). While the epithelial cell population oscillates with the menstrual cycle (Khan et al. 1998, 1999), on average the duct tissue is maintained in homeostasis by carefully controlling the balance of cell proliferation and apoptosis (programmed cell death). In particular, microenvironmental changes can trigger internal signaling responses in the epithelial cells that lead to either proliferation or apoptosis as warranted by the proper maintenance of the tissue architecture. After apoptotic cells disintegrate into apoptotic bodies, they are either absorbed by surrounding epithelial cells or digested by macrophages that can travel through and along the BM (Kerr et al. 1994; Krysko et al. 2008).

The integrin signaling pathway provides a method for cells to detect detachment from the basement membrane: when integrins are actively adhered to their ligands on the BM, they send signals within the cell that eventually trigger the production of survival proteins (e.g., FAK) that inhibit p53-mediated apoptosis (Ilic et al. 1998; Wang et al. 2005). Loss of attachment to the BM therefore allows apoptosis to occur (referred to as anoikis in this context), thus preventing overgrowth of cells into the lumen (Danes et al. 2008). E-cadherin signaling helps the cells to detect the presence or absence of neighbors: the attachment of E-cadherin molecules to like E-cadherins on neighboring cells results in the formation of E-cadherin/β-catenin complexes, thus preventing β-catenin from triggering the transcription of Cyclin D1, c-myc, and Axin2; as a result, cell cycling is inhibited (Bienz and Clevers 2000; Seidensticker and Behrens 2000; Lustig et al. 2002; Hino et al. 2005). When a neighboring cell dies, this E-cadherin signaling is reduced, thereby allowing the cell cycle to progress. This results in the production of a new daughter cell to fill in the gap in the duct epithelium. The epithelial cells also respond to hormones (intercellular signaling molecules) when they bind to surface receptors. In particular, estrogen, progesterone, androgen, prolactin, and epidermal growth factor all affect epithelial cell proliferation and apoptosis decisions, such as increased proliferation prior to lactation (to enlarge the breast duct system and prepare the lobules (Anderson 2004)) and increased apoptosis during breast involution (the "shutdown" process after lactation (Baxter et al. 2007)).

4.1.2 Pathobiology of DCIS

The overexpression of oncogenes (growth-promoting genes) and underexpression of tumor suppressor genes (growth-inhibiting and DNA repair genes) can disrupt the balance of epithelial cell proliferation and apoptosis, leading to cell overproliferation. This can occur either by the accumulation of DNA mutations (genetic damage) (Simpson et al. 2005) or epigenetic anomalies (e.g., alterations in heritable CH_3 methyl groups that suppress key oncogenes (Ai et al. 2006)). The transformation from regular breast epithelium to carcinoma is thought to occur in stages. For simplicity, we neglect the benign, precursor transformations (e.g., atypical ductal hyperplasia or ADH (Simpson et al. 2005)) and focus on DCIS.

In the most well-differentiated classes of DCIS, the epithelial cells maintain their basic polarity and anisotropic distribution of surface adhesion receptors, resulting in partial recapitulation of the nonpathological duct structure within the lumen. These demonstrate either finger-like growths into the lumen (micropapillary, as in Fig. 4.1a (bottom)) or arrangements of duct-like structures within the duct (cribriform, as in Fig. 4.1b (bottom)) (Silverstein 2000). In solid-type DCIS, the cells lack polarity and the microstructures just described disappear. Instead, the cells proliferate until they fill the entire lumen (solid type, as in Fig. 4.1c (bottom)) (Danes et al. 2008). The proliferating cells uptake nutrients as it diffuses into the duct through the basement membrane, leading to the development of oxygen, glucose, and growth factor

gradients (decreasing nutrient concentrations with distance from the BM). If the central oxygen level is sufficiently depleted, the interior tumor cells die (necrose), leading to the formation of a necrotic core of cellular debris (comedo-type solid DCIS, as in Fig. 4.1c (bottom)) (Silverstein 2000). (Note that while we regard comedo-type DCIS as "solid with a necrotic core," some pathological classifications regard comedo as separate from solid. In such classifications, solid-type is rare compared to comedo-type (Jha et al. 2001), demonstrating the high probability of hypoxia and necrosis in nonpapillary, non-cribriform DCIS.) Because these dead cells are neither in close proximity to non-apoptotic epithelial cells nor reachable by macrophages, they are not removed from the lumen. Instead, they swell and eventually burst (Barros et al. 2001), and their solid (i.e., non-water) components are gradually calcified (Stomper and Margolin 1994; Cotran et al. 1994). It is these calcifications that are generally detected by mammograms (Ciatto et al. 1994).

DCIS is a premalignant cancer because it is contained within the duct system by the basement membrane, preventing metastasis. However, it is regarded as an important precursor stage of invasive ductal carcinoma (IDC), where further mutations or changes in gene expression lead to tumor cell motility along the basement membrane, secretion of matrix-degrading enzymes (matrix metalloproteinases, or MMPs) that degrade the BM, and subsequent invasion into the surrounding stroma (Fig. 4.1d, bottom) (Silver and Tavassoli 1998; Adamovich and Simmons 2003). Because DCIS can progress from an undetectable state to filling an entire duct system in a matter of months (Edgerton et al. 2008), there is substantial risk of progressing from an undetected precursor of DCIS (e.g., ADH) to IDC between annual mammograms. Indeed, it is estimated that over three-fourths of all the detected cases of DCIS are invasive (Lampejo et al. 1994; Jemal et al. 2007; American Cancer Society 2007). Hence, predicting the behavior of DCIS is important to understanding and hopefully preventing the progression to IDC.

4.1.3 A Mini-Review of DCIS Modeling

There has been little work to date in combining and applying the great success in modeling the many individual aspects of cancer (e.g., growth and stability as in the classic work by Greenspan (1976), cancer cell population dynamics as in Shuryak et al. (2006), tumor morphology as in Frieboes et al. (2007), tumor–microenvironment interactions as in Anderson et al. (2006) and Macklin and Lowengrub (2007), and tumor response to chemotherapy in Frieboes et al. (2009)) to DCIS (Byrne et al. 2006). Kopans et al. (2003) recently attempted to explain the clinical distribution of tumor sizes by exploring the impact of slow, intermediate, and fast growth rates in a linear growth model. More recently, Sontag and Axelrod (2005) used a compartmental model (a "population dynamics" model based upon a system of ordinary differential equations governing transitions between subpopulations) of DCIS to analyze the mutation pathways that transform normal breast epithelial cells (BECs) to DCIS and later to IDC, and applied machine learning

techniques to fit the model's predicted distribution of cancer types to clinically observed frequencies of DCIS and IDC grades. The work was not able to fully match the clinically observed distribution without hypothesizing an as-yet unobserved common progenitor to DCIS and IDC. Both these works were early, notable attempts to use mathematical modeling to explain clinical observations. However, because they lacked biologically grounded, mechanistic models of the fundamental biophysical processes (the spatiotemporal interaction of proliferation, apoptosis, adhesion, motility, and microenvironment), they cannot provide patient-specific predictions, nor can they readily incorporate the growing body of molecular cancer biology data to provide new, testable hypotheses on DCIS and the progression to IDC.

Other recent models have applied physical conservation laws (mass, momentum, and energy) to DCIS, resulting in continuum models of the spatiotemporal dynamics of tumor cell density, nutrient distribution, and other key microenvironmental quantities at the tissue scale. In some of the most rigorous modeling to date, Franks et al. (2003a) applied a continuum model to tumor propagation in a single duct. One of their predictions was that strong cell–cell and cell–BM adhesive forces, in combination with the difference in viscosity between the tumor and the surrounding fluid in the duct lumen, would lead to an increasingly convex shape of the leading edge of the tumor front. They did not include a model of migration of tumor cells at the surface of the BM. Subsequent work (Franks et al. 2003b, 2005) extended the model to include expansion of the duct due to outward pressure exerted by the DCIS with the objective of predicting evolution to IC, as well as central comedo-type necrosis. While continuum models bring the power of high-performance scientific computing to biology (such as done recently by Frieboes et al. (2007), Macklin and Lowengrub (2008), and Macklin et al. (2009) in applying tissue-scale continuum models to brain and other cancers), they share a common weakness: the key physical effects are lumped together in nonphysical parameters that are not well suited to calibration by molecular and cellular data.

Others have focused on modeling individual cells and how their interactions lead to the emergent properties of DCIS. Edgerton, Mannes, and others used a cellular automata (grid-aligned) model of tumor proliferation and motility to recapitulate the phenomenon known as "pagetoid spread," which describes the migration of tumor cells along the BM away from the "leading edge" or "tumor front" of the bulk DCIS tumor (Mannes et al. 2002; Edgerton et al. in preparation-b). Bankhead et al. (2007) used a cellular automata model to examine the 3-D population dynamics of myoepithelial and breast epithelial cells and their associated progenitor cells, each endowed with a virtual pseudogenome consisting of four lumped "genes," whose mutations led to virtual DCIS. However, the model used *ad hoc*, nonmechanistic rules to govern the cells. Also, the grid-constrained cell arrangement precluded a realistic treatment cell–cell and cell–BM adhesion, stress, and their impact on cell arrangement. Rejniak, Hillen, and others used continuum fluid mechanics methods to model the shape of individual cell membranes interacting under cell–cell and cell–BM adhesion, and proliferation, without restricting cell positions to a lattice (Rejniak and Dillon 2007; Dillon et al. 2008). Their detailed models were able to

partially recapitulate the microstructure of DCIS, particularly due to the polarized arrangement of cell adhesion molecules for well-differentiated epithelial cells. However, both these cell-based approaches neglected motility and were not rigorously calibrated to molecular and cellular data.

4.1.4 Why Agent-Based Modeling?

While the aforementioned modeling attempts have provided a wealth of insight on the mechanisms of DCIS, they have shortcomings as well. Continuum modeling is too coarse-scaled, particularly because it cannot capture the spatial intricacy of DCIS cell patterning (particularly in micropapillary and cribriform types) or cell motility of a single layer of cells along the BM. Furthermore, continuum models tend to lump multiple physical properties into a few parameters. For example, models such as Macklin and Lowengrub (2007) and Frieboes et al. (2007) lump cell–cell, cell–BM, and cell–ECM adhesion, motility, and properties of the ECM into a single "mobility" parameter. This is advantageous for studying the mathematical properties of the physical systems, but makes it difficult to directly match the model parameters to physical measurements. Indeed, many of the key patient-specific measurements occur at the molecular (immunohistochemistry or IHC) and cellular (cell patterning) scales, i.e., at a finer scale than continuum models.

In cellular automata (CA) models, cells occupy a rectangular grid of finite-sized locations (e.g., $10\,\mu m^2$) and change state and/or location (by moving along four or eight allowed, grid-aligned directions) based upon a set of biological rules. This modeling technique has several key advantages, chiefly that they are simple to program and hence accessible to many biologists and mathematicians without formal programming experience, they are simple to modify with new biological rules, and they are discrete and cell-based, and thus well-suited to modeling stochastic cell behavior.

However, CA methods have several scientific drawbacks, particularly when applied to DCIS. Because they are grid aligned, they cannot capture the nonlattice cell patternings of cribriform and papillary DCIS, and their rectangular cell arrangements are generally insufficient to describe the often nonlattice cell arrangement (e.g., hexagonal) found in solid DCIS as well. Indeed, lattice-based methods impose constraints on cell locations and motion that can introduce artificial biases and artifacts into the cell arrangements. Because the cells only have four possible orientations, CA models can only crudely treat cell polarization. The artificial lattice cell arrangement makes it at best difficult to treat mechanical stresses; this limits the ability of CA models to rigorously explore the balance of cell–cell adhesion, cell–BM adhesion, and cellular mechanical properties (incompressibility, deformability, etc.).

Agent-based models (also referred to as individual-based models) are a natural extension of CA methods. Each cell is an object or agent, endowed with as much or as little detail as is necessary to the scientific problem. One can choose

the complexity of the cell's morphology, ranging from treating the cells as points in space, spherical, or even of variable morphology as in recent work by Rejniak and Dillon (2007) and Dillon et al. (2008). Their locations are not necessarily lattice-constrained. They are well-suited to multiscale modeling of cells: each cell agent can be given a subcellular scale, such as a protein signaling model. Zhang et al. (2007) have been very successful in tying intracellular, protein-gene signaling models to individual cell phenotype and motility, and they have used it to simulate small numbers of interacting cells in brain cancer. Their work also made key advances in using molecular (e.g., microarray, Western blot) and cellular (e.g., chemotaxis assay) data to inform and calibrate the cell-scale model. Several groups (Zaman et al. 2005) have also made considerable advances at linking molecular- and cell-scale models, often with calibration to data at the appropriate scales. Indeed, the direct relationship between molecular- and cell-scale experimental measurements and the corresponding components in well-formulated agent-based models makes them ideal for integrative biology:

1. Experimental biology drives the formulation of hypotheses that are encapsulated in the mathematical model equations.
2. Numerical and mathematical analyses of the (calibrated) mathematical model help elucidate the dynamics of the system and suggest new, testable hypotheses to explain the observed biological behavior.
3. New experiments are conducted to test the hypotheses, validate the model, and suggest new model refinements. Return to 1.

For example, see the work by Frieboes et al. (2006) and Thorne et al. (2007), which include an excellent descriptions of integrative modeling.

In our approach, we leverage these strengths to design a biophysically justified model of DCIS that is built upon the balance of the basic forces acting on each cell, can be calibrated to molecular and cellular measurements from patient biopsies, and is modular in (software/modeling) architecture, allowing for the inclusion of more advanced biology as necessary.

4.2 Agent-Based Model of DCIS

We now develop an integrative, agent-based model that is designed with the eventual goal of patient-specific calibration with formalin-fixed, paraffin-embedded (FFPE) patient tissue. The general philosophy of the model is to treat each cell as a physical object subject to classical conservations laws, particularly conservation of momentum by Newton's second law. We then incorporate biology as time- and space-dependent coefficient functions, through the identification of the forces acting on the cells, and via constituent relations. Below is a summary of the continuing agent-based DCIS modeling by a team at the University of Texas Health Science Center in Houston and the neighboring M. D. Anderson Cancer Center; full details are published in Macklin et al. (in preparation). In this chapter, we focus on

modeling solid-type DCIS with and without central (comedo) necrosis, which together constitute the majority of all cases (Jha et al. 2001). Cells are not polarized, and in particular, we assume an isotropic distribution of cell surface receptors. We currently neglect the myoepithelial cells and treat epithelial cells as directly adhered to the BM. Also, because our immunohistochemistry measurements cannot differentiate stem cells, we do not include stem cell and progenitor cell dynamics in our model. We treat the cells as mostly rigid spheres, and we model motion in (currently 2-D) ducts near a terminal lobule, which is approximated as a semicircular "endcap" to the duct. The work can readily be extended to 3-D.

Each cell i is an agent endowed with physical quantities (mass m_i, radius r_i, volume V_i, solid volume $V_{S,i}$, position $\mathbf{x}_i$, and velocity $\mathbf{v}_i$) and phenotypic properties (cell state $\mathcal{S}_i$, internal protein signaling state, surface receptor distribution, calcification, and parameters governing the "genetics" of the cell) (see Fig. 4.2). We model the transitions between cell states as stochastic processes (with parameters

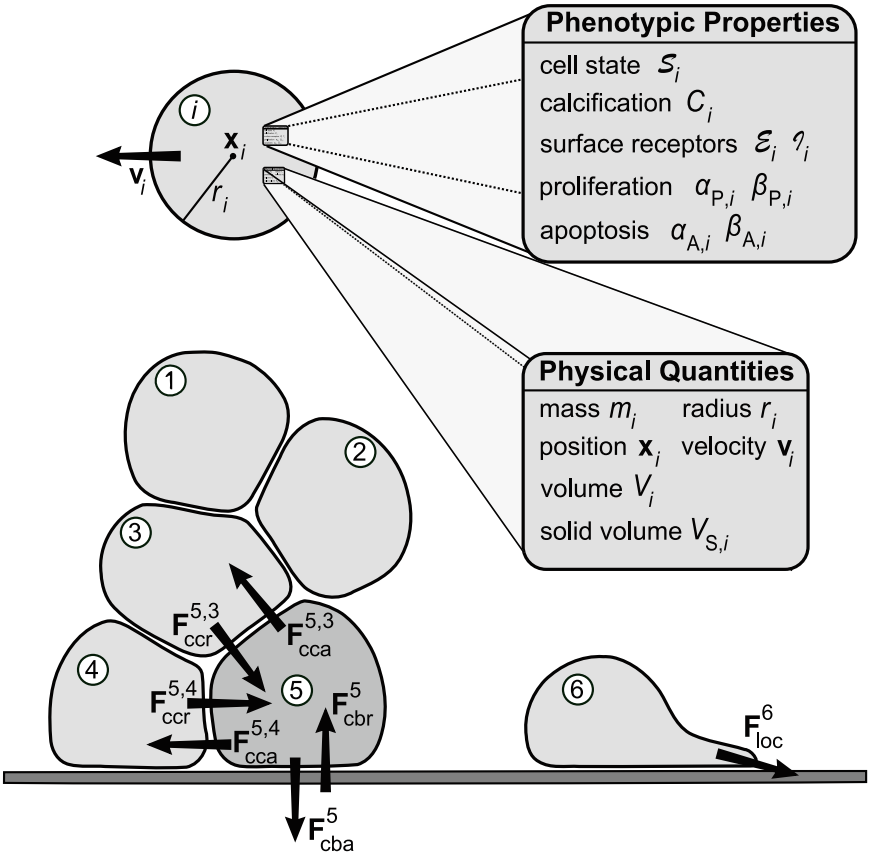

Fig. 4.2 Basic schematic of the model. Key forces acting on cell 5 are labeled. Reprinted with permission from Macklin et al. (in preparation)

that depend upon the microenvironment and cell properties) and determine the cell velocity based upon the balance of the physical forces acting upon it. Lastly, we note that we use the same model for both cancerous and noncancerous duct epithelial cells; instead, the cells differ primarily in the values of their proliferation, apoptosis, and motility coefficients, which is analogous to modeling alterations in the cancerous cells' oncogenes and tumor suppressor genes (Hanahan and Weinberg 2000).

4.2.1 A Brief Review of Exponential Random Variables and Poisson Processes

Because we model transitions between cell states as stochastic processes, we begin with a brief review of the necessary preliminaries. We note that this discussion is necessarily sparse, introducing only the key concepts and not the full richness of measure theory-based probability and stochastic processes. The interested reader can find more information in widespread references, such as Shiryaev (1995) and Øksendal (2007).

A random variable T is *exponentially distributed* with parameter α if for any $t > 0$, the probability $\Pr(T < t)$ is given by

$$\Pr(T < t) = 1 - \mathrm{e}^{-\alpha t}. \tag{4.1}$$

Also, T has expected value $\mathrm{Ex}[T] = \frac{1}{\alpha}$ (i.e., the mean $\langle T \rangle$ is $1/\alpha$) and variance $\mathrm{Var}[T] = \frac{1}{\alpha^2}$. The simple relationship between the mean $\langle T \rangle$ and the parameter α makes it particularly useful for calibration by limited data. Exponential random variables have the important property of being *memoryless*. For any $0 \leq t, \Delta t$, the probability that $T > t + \Delta t$ given that $T > t$ is

$$\Pr(T > t + \Delta t | T > t) = \Pr(T > \Delta t), \tag{4.2}$$

i.e., if the event T has not occurred by time t, then the probability of that event occurring by an additional Δt units of time is unaffected by the earlier times, and so we can "reset the clock" at time t. This property is useful for modeling cell decision processes that are assumed to depend upon the current subcellular and microenvironmental state, and not on previous times. Even if the current cell decisions do depend upon past times, that information can be built into the time evolution of the internal cell state.

A stochastic process N_t is a series of random variables indexed by the "time" t. In particular, N_t is a *counting process* if:

1. $N_0 \geq 0$. (The initial number of events N_0 is at least zero).
2. $N_t \in \mathbb{Z}$ for all $t \geq 0$. (The current number of events N_t is an integer).

3. If $s < t$, then $N_t - N_s \geq 0$. (We count the cumulative number of events, and $N_t - N_s$ gives the number of events occurring in the time interval $(s, t]$).

A *Poisson process* P_t is a particular type of counting process (with parameter α) satisfying:

1. $P_0 = 0$. (The initial count is 0).
2. If $[s, s + \Delta s]$ and $[t, t + \Delta t]$ are nonoverlapping intervals, then $P_{s+\Delta s} - P_s$ and $P_{t+\Delta t} - P_t$ are independent random variables. (What happens in the interval $(t, t + \Delta t)$ is independent of what happened in $(s, s + \Delta s)$).
3. For any $0 \leq s < t$, the distribution of $P_t - P_s$ only depends upon the length of the interval (s, t) (stationary increments), and in particular, if $n \in \mathbb{N} = \mathbb{Z} \cap [0, \infty)$,

$$\Pr(P_t - P_s = n) = \frac{\mathrm{e}^{-\alpha(t-s)}\alpha(t-s)}{n!}. \tag{4.3}$$

Poisson processes have a useful property that we shall rely upon in the model. If T_{n+1} is the *interarrival time* between the events $\inf\{t : P_t = n\}$ (the first time at which $P_t = n$) and $\inf\{t : P_t = n + 1\}$ for $n \in \mathbb{N}$, then T_n is an exponentially distributed random variable with parameter α, and in particular,

$$\Pr(P_{t+\Delta t}) - P_t \geq 1 = \Pr(T_n < \Delta t) = 1 - \mathrm{e}^{-\alpha \Delta t}. \tag{4.4}$$

Lastly, we note that if $\alpha = \alpha(t)$ varies in time, then P_t is a *non-homogeneous* Poisson process with interarrival times given by

$$\begin{aligned}
\Pr(P_{t+\Delta t} - P_t = n) &= \frac{\mathrm{e}^{-\int_t^{t+\Delta t} \alpha(s)\mathrm{d}s} \left(\int_t^{t+\Delta t} \alpha(s)\mathrm{d}s\right)^n}{n!} \\
\Pr(P_{t+\Delta t} - P_{t \geq 1}) &= \Pr(T_n < \Delta t) \\
&= 1 - \mathrm{e}^{-\int_0^{t+\Delta t} \alpha(s)\mathrm{d}s} \\
&\approx 1 - \mathrm{e}^{-\alpha(t)\Delta t}, \quad \Delta t \downarrow 0.
\end{aligned}$$

In our work, the Poisson processes are nonhomogeneous due to their dependencies upon microenvironmental and intracellular variables that vary in time. However, these can be approximated by homogeneous processes on small time intervals $[t, t + \Delta t]$ as above (Macklin et al. in preparation).

4.2.2 A Family of Potential Functions

We introduce a family of interaction potential functions $\varphi(\mathbf{x}; R, n)$, parameterized by R and n, satisfying

$$\varphi(\mathbf{x}; R, n) = \begin{cases} -\frac{R}{n+2}\left(1 - \frac{|\mathbf{x}|}{R}\right)^{n+2} & \text{if } |\mathbf{x}| < R \\ 0 & \text{else,} \end{cases} \tag{4.5}$$

$$\varphi'(x; R, n) = \begin{cases} \left(1 - \frac{x}{R}\right)^{n+1} & \text{if } x < R \\ 0 & \text{else,} \end{cases} \tag{4.6}$$

$$\nabla\varphi(\mathbf{x}; R, n) = \begin{cases} \left(1 - \frac{|\mathbf{x}|}{R}\right)^{n+1} \frac{\mathbf{x}}{|\mathbf{x}|} & \text{if } |\mathbf{x}| < R \\ \mathbf{0} & \text{else,} \end{cases} \tag{4.7}$$

where R is the *maximum interaction distance* of φ, and n is the *power* of the interaction potential. We use this particular form of potential function because it satisfies the following:

1. The potential (and its derivatives) has *compact support*: it is zero outside a closed bounded set (in this case, the closed ball $\overline{B}(\mathbf{0}, R)$). This is a more realistic depiction of cell–cell and cell–BM interactions than exponential decay, for example, which nonphysically gives interaction over infinite distances.
2. For any R and n, and for any $0 < |\mathbf{x}| < R$, we have

$$0 = \varphi'(R; R, n) < \varphi'(|\mathbf{x}|; R, n) < \varphi'(0; R, n) = 1. \tag{4.8}$$

   The baseline case $n = 0$ is a linear ramping, and for higher n, the function tapers off to zero gradient smoothly.

A good example and discussion of the use of potential functions to mediate cell–cell adhesion and interaction for individual-based models can be found in Byrne and Drasdo (2009).

4.2.3 Cell States

To emulate the biological function of cells, we endow each cell agent with a state $\mathcal{S}(t) \in \{\mathcal{Q}, \mathcal{P}, \mathcal{A}, \mathcal{N}, \mathcal{C}\}$.

4.2.3.1 Quiescent Cells ($\mathcal{Q}$)

Quiescent cells are not actively cycling but are instead in a "resting state" (G0, in terms of the cell cycle); this is the "default" cell state in the agent framework. We model the transitions between cell states as stochastic events governed by exponentially distributed random variables. (That is, the transition events are interarrival times modeling the elapsed time between proliferation and apoptosis events. A fuller discussion of the mathematical theory of this modeling construct can be found in

Macklin et al. (in preparation).) The subcellular scale is built into this framework by making the mean of these random exponential variables depend upon the microenvironment and the cell's internal properties.

4.2.3.2 Proliferation ($\mathcal{P}$)

Cells transition from the quiescent state $\mathcal{Q}$ to the proliferative state $\mathcal{P}$ with a probability that depends upon the microenvironment. For a cell in the quiescent state at time t, the probability that the cell enters the proliferative state during the interval $(t, t + \Delta t]$ is modeled as an exponential interarrival time with parameter $\alpha_{\mathrm{P}}(\mathcal{S}, \circ, \bullet)$, where $\circ$ represents the microenvironment and $\bullet$ is the cell's internal state. Hence,

$$\Pr\left(\mathcal{S}(t + \Delta t) = \mathcal{P} | \mathcal{S}(t) = \mathcal{Q}\right) = 1 - \mathrm{e}^{-\alpha_{\mathrm{P}} \Delta t} \approx \alpha_{\mathrm{P}} \Delta t.^{1} \tag{4.9}$$

By our preliminary IHC results, there is a correlation between the microenvironmental oxygen level σ (nondimensionalized by σ_∞, the far-field oxygen level in nondiseased, normoxic breast tissue, i.e., "well-oxygenated" tissue) and the proliferative index PI (see Fig. 4.3, bottom left). Thus, we expect α_{P} to be an increasing function of σ, which we model by

$$\alpha_{\mathrm{P}} = \alpha_{\mathrm{P}}(\mathcal{S}(t), \sigma, \bullet) = \begin{cases} \overline{\alpha}_{\mathrm{P}}(\bullet) \frac{\sigma - \sigma_{[\mathrm{H}]}}{1 - \sigma_{\mathrm{H}}} & \text{if } \mathcal{S}(t) = \mathcal{Q} \\ 0 & \text{else,} \end{cases} \tag{4.10}$$

where σ_{H} is a threshold oxygen value at which cells become hypoxic, and $\overline{\alpha}_{\mathrm{P}}(\bullet)$ is the cell's $\mathcal{Q} \to \mathcal{P}$ transition rate when $\sigma = 1$ (i.e., in "well-oxygenated" tissue), which depends upon the cell's genetic profile $\bullet$. For now, we shall model $\overline{\alpha}_{\mathrm{P}}$ as constant for and specific to each cell type (tumor cells or noncancerous epithelial cells).

Once a cell has entered the proliferative state $\mathcal{P}$, it remains in that state for time β_{P}^{-1}, which may generally depend upon the microenvironment and the cell's internal state, but which we currently model as fixed for both tumor and epithelial cells (with the same value). This models our assumption that both tumor cells and (noncancerous) epithelial cells use the same basic cellular machinery to proliferate, but with differing frequency due to differing expressions of oncogenes (Hanahan and Weinberg 2000). Once the cell exits the proliferative state, we replace it with two identical daughter cells, both with phenotypic properties inherited from the parent cell, and initially placed in the "default" quiescent state. We position the daughter cells adjacent to one another with center of mass equal to the position of the parent cell.

[1] While using the interarrival time ordinarily gives probability of having at least one proliferation event (rather than precisely one) in the interval $(t, t + \Delta t]$, our form of α_{P} in (4.10) precludes this, because α_{P} decreases to zero until completing proliferation.

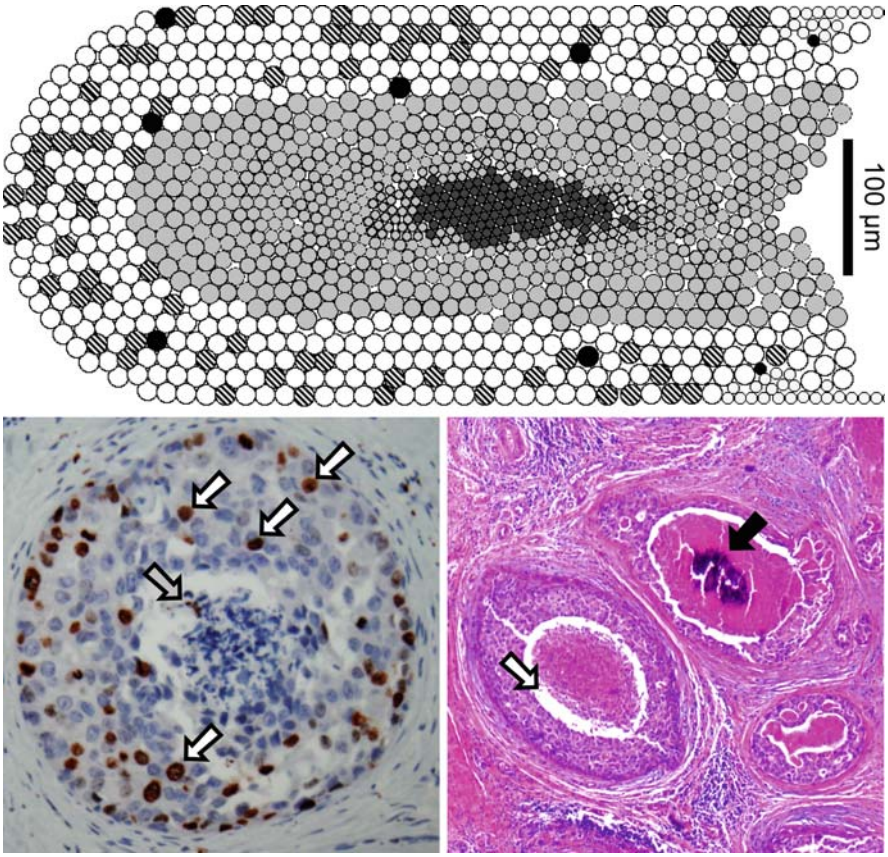

Fig. 4.3 Verification of the morphological features of the calibrated simulation. *Top*: Simulation at time 30 days. *White* cells are quiescent, striped cells that are proliferating (virtual Ki-67), *black* cells are apoptotic (virtual cleaved Caspase-3), medium *gray* cells are necrotic, and central *dark gray* cells are calcified debris. Small cells along the BM are noncancerous epithelium. *Bottom left*: Ki-67 immunohistochemistry of a duct cross section. *White arrows* show Ki-67 positive nuclei. The *gray arrow* shows necrotic debris. *Bottom right*: H&E staining showing calcifications (*black arrow*) and the gap between the viable rim and necrotic core (*white arrow*). Reprinted with permission from Macklin et al. (in preparation)

4.2.3.3 Apoptosis ($\mathcal{A}$)

Apoptotic cells are undergoing "programmed" cell death in response to internal protein signaling. As with proliferation, we model the transition as an exponential interarrival time with parameter $\alpha_A(\mathcal{S}, \circ, \bullet)$. In our preliminary IHC, apoptotic cells are sporadic within the tissue, with no apparent correlation with oxygen (Edgerton et al. in preparation-a). We thus model $\alpha_A(\mathcal{S}, \bullet)$ as fixed for each cell population. Hence,

$$\Pr\left(\mathcal{S}(t + \Delta t) = \mathcal{A} | \mathcal{S}(t) = \mathcal{Q}\right) = 1 - \mathrm{e}^{-\alpha_A \Delta t}, \tag{4.11}$$

where

$$\alpha_{\mathrm{A}} = \alpha_{\mathrm{A}}(\mathcal{S}(t), \bullet) = \begin{cases} \overline{\alpha}_{\mathrm{A}}(\bullet) & \text{if } \mathcal{S}(t) = \mathcal{Q} \\ 0 & \text{else.} \end{cases} \tag{4.12}$$

Cells remain in the apoptotic state for a fixed amount of time β_{A}^{-1}, similarly to proliferation. Upon leaving the apoptotic state, cells are deleted from the simulation to model the removal of their solid content (encapsulated as apoptotic bodies) by surrounding epithelial cells, while their previously occupied volume is made available to the surrounding cells to model the release of the cells' water content after lysis.

4.2.3.4 Necrosis ($\mathcal{N}$)

Necrosis can be described as unplanned cell death, either due to injury or due to severe nutrient deprivation, leading to internal energy (ATP) depletion (Scarlett et al. 2000; Krysko et al. 2008). In this model in particular, cells become necrotic when the oxygen level σ dips below a threshold value σ_{N} where the cell can no longer sustain metabolism and hence depletes its ATP. For simplicity, we set $\sigma_{\mathrm{N}} = \sigma_{\mathrm{H}}$. Any state in $\{\mathcal{Q}, \mathcal{A}, \mathcal{P}\}$ irreversibly and deterministically transitions to the necrotic state $\mathcal{N}$ when $\sigma \leq \sigma_{\mathrm{N}}$. We note that cells should be able to survive for a short time span in a hypoxic state before irreversibly transitioning to the necrotic state. However, in our testing of this model, we have found that introducing such a "hypoxic distress" state does not have any significant impact on the model dynamics (Macklin et al. in preparation).

We assume that cells remain in the necrotic state for a fixed time β_{N}^{-1}, during which time they gradually degrade and calcify. In particular, we assume that the surface receptors degrade, the cell loses its water content (a simplification of cell swelling and subsequent bursting early in necrosis), and the solid content is gradually calcified. In this early model of the necrosis state, we approximate all three of these processes as occurring simultaneously at the same rate β_{N}. We discuss the implications of this assumption in Sect. 4.5.4.1, with future improvements in Sect. 4.6.

We model the degradation of the surface receptors S (where S can be the nondimensionalized E-cadherin or integrin expression (between 0 and 1), and τ is time since entering the necrotic state) by

$$\frac{\mathrm{d}S}{\mathrm{d}\tau} = -\left(\beta_{\mathrm{N}} \log 100\right) S, \tag{4.13}$$

$$S(0) = 1, \tag{4.14}$$

where the coefficient of the ODE was chosen such that $S(\beta_{\mathrm{N}}^{-1}) = 0.01\ S(0)$. This models the assumption that virtually all of the surface receptor S is degraded by the end of necrosis at time $\tau = \beta_{\mathrm{N}}^{-1}$.

We assume that water loss is proportional to surface area of the necrotic cells, as well as the remaining water fraction $((V - V_{\mathrm{S}})/V$, where V_{S} is the solid volume of the cell). Hence, we model

$$\frac{\mathrm{d}V}{\mathrm{d}\tau} = -(2\beta_{\mathrm{N}} \log 100)(4\pi r^2(\tau))\frac{V - V_{\mathrm{S}}}{V}, \tag{4.15}$$

where the coefficient was chosen to make this nonlinear ODE satisfy $V(\beta_{\mathrm{N}}^{-1}) \approx V_{\mathrm{S}}$ (Macklin et al. in preparation), i.e., water loss is mostly complete by the end of necrosis.

Lastly, we assume a constant rate of cell calcification, with the necrotic cell 100% calcified by time β_{N}^{-1}. If C is the nondimensional degree of calcification, then

$$C(t) = \beta_{\mathrm{N}}\tau. \tag{4.16}$$

4.2.3.5 Calcified Debris ($\mathcal{C}$)

Once cells leave the necrotic state $\mathcal{N}$, they irreversibly enter the calcified debris state $\mathcal{C}$. These cells are assumed to have zero surface integrin and E-cadherin adhesion receptors, and hence they can only adhere to other debris particles; this is a simplified model of the (crystalline) chemical bond between calcium phosphate and/or calcium oxalate molecules that comprise the microcalcification.

4.2.4 Cell Motion Based upon the Balance of Forces

Each epithelial cell is subject to competing forces that determine its motion within the duct. Cells adhere to other cells ($\mathbf{F}_{\mathrm{cca}}$) and the basement membrane ($\mathbf{F}_{\mathrm{cba}}$), calcified debris particles adhere to other calcified debris particles ($\mathbf{F}_{\mathrm{dda}}$), cells and calcified debris particles resist compression by other cells and debris particles ($\mathbf{F}_{\mathrm{ccr}}$), and the basement membrane resists its penetration and deformation by cells and debris particles ($\mathbf{F}_{\mathrm{cbr}}$). See Fig. 4.2, where we show the forces acting on cell 5. In addition, moving cells and debris particles experience a drag force by the luminal and interstitial fluids ($\mathbf{F}_{\mathrm{drag}}$). We express this balance by Newton's second law, acting on cell i:

$$m_i \dot{\mathbf{v}}_i = \sum \mathbf{F} = \sum_j \mathbf{F}^{ij}_{\mathrm{cca}} + \mathbf{F}^{i}_{\mathrm{cba}} + \sum_j \mathbf{F}^{ij}_{\mathrm{dda}} + \sum_j \mathbf{F}^{ij}_{\mathrm{ccr}} + \mathbf{F}^{i}_{\mathrm{cbr}} + \mathbf{F}^{i}_{\mathrm{drag}}. \tag{4.17}$$

We model the drag force by $\mathbf{F}^{i}_{\mathrm{drag}} = -\nu\mathbf{v}_i$, and we make the "inertialess" assumption that the forces equilibrate quickly, and so $|m_i\dot{\mathbf{v}}_i| \approx 0$. Hence, we approximate $\sum \mathbf{F} = \mathbf{0}$ and solve for the cell velocity:

$$\mathbf{v}_i = \frac{1}{\nu}\left(\sum_j \mathbf{F}^{ij}_{\mathrm{cca}} + \mathbf{F}^{i}_{\mathrm{cba}} + \sum_j \mathbf{F}^{ij}_{\mathrm{dda}} + \sum_j \mathbf{F}^{ij}_{\mathrm{ccr}} + \mathbf{F}^{i}_{\mathrm{cbr}}\right). \tag{4.18}$$

This formulation has a convenient interpretation: each term $\frac{1}{\nu}\mathbf{F}_{\square}$ is the "terminal" (equilibrium) velocity of the cell when fluid drag and $\mathbf{F}_{\square}$ are the only forces acting on the cell. This will be particularly useful when calibrating cell motility in future work, as motility is generally measured as a cell velocity (Harms et al. 2005). We now detail the biological assumptions of the remaining forces.

4.2.4.1 Cell–Cell Adhesion ($\mathbf{F}_{\mathbf{cca}}$)

E-cadherin molecules on the cell surface form homophilic bonds with E-cadherin molecules on neighboring cells (Panorchan et al. 2006). Hence, the strength of the cell–cell adhesive force between neighboring cells is proportional to the product of their respective e-cadherin surface receptor expressions. Furthermore, the strength of the adhesion increases as the cells are drawn more closely together, bringing more surface area and hence more surface receptors into direct contact. We model the adhesive force on cell i resulting from adhesion between cells i and j by

$$\frac{1}{\nu}\mathbf{F}^{ij}_{\text{cca}} = \alpha_{\text{cca}}\mathcal{E}_i\mathcal{E}_j\nabla\varphi\left(\mathbf{x}_j - \mathbf{x}_i;\ R^i_{\text{cca}} + R^j_{\text{cca}}, n_{\text{cca}}\right), \tag{4.19}$$

where $\mathcal{E}_i$ and R^i_{cca} are cell i's (nondimensionalized) surface e-cadherin receptor expression and maximum cell–cell adhesion interaction distance, respectively, and n_{cca} is the cell–cell adhesion power introduced with our potential function family in Sect. 4.2.2. We typically set $R^i_{\text{cca}} > r_i$, to approximate the ability of cells to deform before breaking all adhesive bonds, with the strength of force decreasing as the separation between the cells increases. The α_{cca} can be interpreted as the force per (nondimensionalized) e-cadherin bond.

4.2.4.2 Cell–BM Adhesion ($\mathbf{F}_{\mathbf{cba}}$)

Integrin molecules on the cell surface form heterophilic bonds with specific ligands (generally laminin and fibronectin (Butler et al. 2008)) on the basement membrane, here assumed to be constant. Hence, the strength of the cell–BM adhesive force is proportional to its integrin surface receptor expression. Furthermore, the strength of the adhesion increases as the cells approaches the BM, bringing more cell surface adhesion receptors in contact with their respective ligands on the BM. We model the adhesive force on cell i resulting from adhesion to the BM by

$$\frac{1}{\nu}\mathbf{F}^{i}_{\text{cba}} = \alpha_{\text{cba}}\mathcal{I}_i\nabla\varphi\left(d(\mathbf{x}_i);\ R^i_{\text{cba}}, n_{\text{cba}}\right), \tag{4.20}$$

where d is the distance to the basement membrane, $\mathcal{I}_i$ and R^i_{cba} are cell i's (nondimensionalized) surface integrin receptor expression and maximum cell–BM adhesion interaction distance, respectively, and n_{cba} is the cell–BM adhesion power

introduced with our potential function family in Sect. 4.2.2. As with cell–cell adhesion, we typically set $R^i_{\text{cba}} > r_i$ to approximate the cell's limited capacity to deform before breaking all its adhesive bonds. The α_{cba} can be interpreted as the force per (nondimensionalized) integrin bond.

4.2.4.3 (Calcified) Debris–(Calcified) Debris Adhesion ($\mathbf{F_{dda}}$)

We model adhesion between calcified debris particles similarly to e-cadherin-mediated cell–cell adhesion. We assume that calcium phosphate and/or calcium oxalate crystals in the interacting calcified debris particles remain strongly bonded as part of an overall crystalized structure – the microcalcification. We model the adhesive force on calcified debris particle i resulting from adhesion between calcified debris particles i and j by

$$\frac{1}{\nu}\mathbf{F}^{ij}_{\text{dda}} = \alpha_{\text{dda}} C_i C_j \nabla\varphi\left(\mathbf{x}_j - \mathbf{x}_i;\ R^i_{\text{dda}} + R^j_{\text{dda}}, n_{\text{dda}}\right), \tag{4.21}$$

where C_i and R^i_{dda} are cell i's (nondimensionalized) degree of calcification and maximum debris–debris adhesion interaction distance, respectively, and n_{dda} is the cell–cell adhesion power introduced with our potential function family in Sect. 4.2.2. The α_{dda} can be interpreted as the adhesive force between two fully calcified debris particles.

4.2.4.4 Cell–Cell Repulsion (Including Calcified Debris) ($\mathbf{F_{ccr}}$)

Cells resist compression by other cells due to the internal structure of their cytoskeletons, the incompressibility of their internal cytoplasm (fluid), and the surface tension of their membranes. We thus introduce a cell–cell repulsive force that is zero when cells are just touching, and then increases rapidly as the cells are pressed together. However, cells do have a capacity deform in response to pressure; we approximate this by allowing some overlap between cells. We model the force by

$$\mathbf{F}^{ij}_{\text{ccr}} = -\alpha_{\text{ccr}} \nabla\varphi\left(\mathbf{x}_j - \mathbf{x}_i;\ r_i + r_j, n_{\text{ccr}}\right), \tag{4.22}$$

where n_{ccr} is the cell–cell repulsion power (Sect. 4.2.2) and α_{ccr} is the maximum repulsive force when the cells are completely overlapping.

4.2.4.5 Cell–BM Repulsion (Including Debris) ($\mathbf{F_{cbr}}$)

We model the basement membrane as rigid and nondeformable due to its relative stiffness and strength. Hence, it resists deformation and penetration by the cells and debris particles. We model the force by

$$\mathbf{F}^i_{\text{cbr}} = -\alpha_{\text{cbr}} \nabla \varphi(d(\mathbf{x}_i);\ r_i, n_{\text{cbr}}), \tag{4.23}$$

where n_{cbr} is the cell–BM repulsion power (Sect. 4.2.2) and α_{cbr} is the maximum repulsive force when the cell's center is embedded in the basement membrane.

4.2.5 Duct Geometry

We denote the duct lumen by Ω and the duct boundary (BM) by $\partial\Omega$. In this chapter, we treat the duct as a rectangular region (a longitudinal cross section of a cylinder) of radius r_{duct} and length ℓ_{duct}. We terminate the left side of the duct with a semicircle, as an initial approximation to a lobule. (See Fig. 4.3 for a typical simulation view.)

While the ducts we simulate in this chapter are relatively simple, we introduce a framework that allows us to simulate DCIS growth in arbitrary duct geometries, such as near a branch in the duct tree. We represent the duct wall implicitly by introducing an auxiliary signed distance function d (a *level set function*) satisfying

$$\begin{cases} d(\mathbf{x}) > 0 & \mathbf{x} \in \Omega \\ d(\mathbf{x}) = 0 & \mathbf{x} \in \partial\Omega \\ d(\mathbf{x}) < 0 & \mathbf{x} \notin \overline{\Omega} = \Omega \cup \partial\Omega \\ |\nabla d(\mathbf{x})| \equiv 1. \end{cases} \tag{4.24}$$

The gradient of the distance function ∇d yields the normal vector $\mathbf{n}$ (oriented from the BM into the lumen) to the interior duct surface. We have adapted this technique to represent the morphology of moving tumor boundaries, and it is well suited to the eventual description of duct expansion (Macklin and Lowengrub 2005, 2006, 2007, 2008; Frieboes et al. 2007; Macklin et al. 2009).

Level set methods were first developed by Osher and Sethian (1988) and have been used to study the evolution of moving surfaces that exhibit frequent topology changes (e.g., merger of regions and fragmentation), particularly in the contexts of fluid mechanics and computer graphics. (See the books by Sethian (1999) and Osher and Fedkiw (2002) and the references by Osher and Sethian (1988), Osher and Fedkiw (2001), and Sethian and Smereka (2003).) For more information on the level set method and applications, please see Osher and Sethian (1988), Sussman et al. (1994), Malladi et al. (1995, 1996), Adalsteinsson and Sethian (1999), Sethian (1999), Osher and Fedkiw (2001, 2002), and Sethian and Smereka (2003).

4.2.6 Intraductal Oxygen Diffusion

We model the release of oxygen by blood vessels outside the duct, its diffusion through the duct wall $\partial\Omega$ and throughout the duct lumen Ω, and its uptake by

epithelial cells and its decay (e.g., by reacting with molecules in the interstitial and luminal fluid) by

$$\begin{cases} \dfrac{\partial \sigma}{\partial t} = D\nabla^2\sigma - \lambda\sigma & \text{if } \mathbf{x} \in \Omega \\ \sigma = \sigma_{\mathrm{B}} & \text{if } \mathbf{x} \in \partial\Omega, \end{cases} \tag{4.25}$$

where σ is the nondimensional oxygen level (scaled by the oxygen concentration in well-oxygenated tissue near the blood vessels in the stroma), D is the oxygen diffusion coefficient, λ is the cellular oxygen uptake rate (generally 0.1 min^{-1}, currently assumed equal for all cell types for simplicity), and σ_{B} is the (nondimensional) oxygen level on the basement membrane.

The oxygen diffusion equation admits an intrinsic diffusion length scale L that we use to nondimensionalize space in (4.25):

$$L = \sqrt{\frac{D}{\lambda}}. \tag{4.26}$$

By the literature, we have $L \approx 100\,\mu\mathrm{m}$ (Owen et al. 2004).

4.3 Numerical Technique

While the cells' positions are not lattice constrained, we introduce several independent computational meshes for the field variables. In particular, we introduce an oxygen mesh that discretizes the duct lumen with spacing $\Delta x = \Delta y = 0.1$ (approximately 10-μm spacing in dimensional space) to resolve oxygen gradients. We also introduce a cell interaction mesh (see Sect. 4.3.1) with 1-μm spacing; this construct allows us to avoid directly testing each cell for interaction with every other cell, hence avoiding $\mathcal{O}\left(\#\ \mathrm{cells}^2\right)$ computational cost.

The cells are implemented in an object-oriented C++ framework, where each cell is an instance of a `Cell` class and endowed with instances of `Cell_properties` (proliferation and apoptosis parameters, initial radius and volume, etc.) and `Cell_state` (cell state, position, velocity) classes. We use a doubly linked list structure to conveniently order the cells, which allows us to easily delete apoptosed cells and insert new daughter cells.

To update our agent-based model at time t to the next simulation time $t + \Delta t$, we:

1. Update the oxygen solution on the oxygen mesh. This is achieved using standard explicit forward Euler methods. In particular, the oxygen $\sigma(x_i, y_j, t + \Delta t)$ is given by

$$\begin{aligned}&\sigma(x_i, y_j, t + \Delta t)\\ &= \sigma(x_i, y_j, t) + \Delta t\left(\frac{\sigma(x_i - \Delta x, y_j, t) - 2\sigma(x_i, y_j, t) + \sigma(x_i + \Delta x, y_j, t)}{\Delta x^2}\right.\\ &\left. + \frac{\sigma(x_i, y_j - \Delta y, t) - 2\sigma(x_i, y_j, t) + \sigma(x_i, y_j + \Delta y, t)}{\Delta y^2} - \sigma(x_i, y_j, t)\right).\end{aligned} \tag{4.27}$$

We note that in practice, we may need to iterate a few times with a smaller $\Delta\tau$ chosen to satisfy a CFL stability condition. Fuller details can be found in Macklin et al. (in preparation).

2. Iterate through all the cells to update the interaction mesh (see Sect. 4.3.1).
3. Iterate through all the cells to update their states according to the transition models in Sect. 4.2.3. Update the necrosing cells' radii, volumes, and calcification as described.
4. Iterate through all the cells to update their velocities as described above.
5. Iterate through all the cells to determine $\max|\mathbf{v}_i|$. Use this to determine the new Δt using the stability criterion $\Delta t < 1/\max|\mathbf{v}_i|$.
6. Iterate through all the cells to update their positions according to their new velocities. We use forward Euler stepping ($\mathbf{x}_i(t + \Delta t) = \mathbf{x}_i(t) + \Delta t\mathbf{v}_i(t)$), although improvements to higher-order Runge–Kutta methods are straightforward.

Notice that all of these steps require at most cycling through all the cells. So long as interaction testing also involves at most cycling through all the cells, the overall simulation requires computational effort that is linear in the number of cells. This is an improvement over most discrete models, where cellular interactions require quadratic effort in the number of cells.

4.3.1 Efficient Interaction Testing

With spatial resolution given by the interaction mesh (1-μm micron spacing), we create an array of linked lists of interactions as follows:

1. Let $R = 2\max_i\left\{r^i_{\text{cca}}\right\}$
2. Initialize the array such that each pointer is NULL.
3. For each cell i, append its memory address to the list for each mesh point within a distance R of its center $\mathbf{x}_i$.

Once complete, at any mesh point (i, j), we have a linked list of cells which are allowed to interact with a cell centered at or near (x_i, y_j).

We use this list whenever we compute a quantity of the form

$$\sum_j f\left(\text{cell}_i, \text{cell}_j\right)(x_k, y_\ell) \tag{4.28}$$

by contracting the sum to the members of the linked list at (x_k, y_ℓ). Because the number of points written to the array is fixed for each cell, this reduces the computational cost of cell–cell interaction testing to $\mathcal{O}$ (# cells), rather than the more typical $\mathcal{O}\left(\#\text{ cells}^2\right)$. Furthermore, writing this interaction data structure still allows us to use arbitrary cell–cell interactions. Notice that this computational gain relies upon the fact that cells can only interact over finite distances.

4.4 Estimating Key Parameters

To make the model *predictive* we must constrain the nonpatient-specific parameters as much as possible. We now summarize key parameter estimates made in Macklin et al. (in preparation).

4.4.1 Cell Cycle and Apoptosis Time

We estimate that the cell cycle time, β_{P}^{-1}, is 18 h by the modeling literature (e.g., see Owen et al. (2004)). The time to complete apoptosis, β_{A}^{-1}, is generally difficult to determine. Experimental and theoretical estimates, however, are on the order of hours (e.g., see Kerr et al. (1994), Hu et al. (1997), and Scarlett et al. (2000)). We estimated β_{A} for BECs by:

1. Assuming that cancerous and noncancerous BECs use the same subcellular machinery to complete apoptosis, and hence β_{A}^{-1} can be estimated using noncancerous cells (Hanahan and Weinberg 2000)
2. Assuming similarly that $\beta_{\mathrm{P}}^{-1} = 18\,\mathrm{h}$ for both cancerous and noncancerous cells
3. Assuming that on average, the noncancerous BEC population is maintained in homeostasis in nonlactating, noninvoluting women (Anderson 2004).

Putting these assumptions together, we find that a homeostatic epithelial breast cell population satisfies

$$\beta_{\mathrm{A}}\mathrm{AI} = \frac{1}{18\,\mathrm{h}}\mathrm{PI}, \tag{4.29}$$

where AI is the apoptotic index (the percentage of cells in the apoptotic state $\mathcal{A}$) and PI is the proliferative index (the percentage of cells in the proliferative state $\mathcal{P}$). Using published statistics for AI and PI in noncancerous breast duct epithelium for a large group of pre- and postmenopausal women (Lee et al. 2006b) with the above, applied separately to the pre- and postmenopausal women, we estimated (after accounting for the fraction of apoptotic cells not imaged due to limitations of TUNEL assays (Scarlett et al. 2000; Macklin et al. in preparation)) that $\beta_{\mathrm{A}}^{-1} = 8.6\,\mathrm{h}$ for both groups, consistent with the order-of-magnitude estimate above. See Macklin et al. (in preparation) for a more detailed derivation of this estimate. We point out that the good agreement in the premenopausal and postmenopausal β_{A}

estimates, in spite of very different hormonal conditions, gives further credibility to our assumption that cell apoptosis uses the same machinery with differing frequency in cancerous breast epithelium.

4.4.2 Oxygen Parameters

By the literature, the cellular oxygen uptake rate is $\lambda = 0.1\ \text{min}^{-1}$, the diffusion length scale is $L = 100\,\mu\text{m}$, and the hypoxic oxygen threshold is $\sigma_{\text{H}} = 0.2$ (Ward and King 1997; Franks et al. 2003a).

4.4.3 Cell Mechanics

We estimate the solid volume fraction of cells (V_{S}/V) at approximately 10%, based upon the published data of Macknight et al. (1971), combined with the assumption that the solid cell component is 1–10 times denser than water (Macklin et al. in preparation).

We estimate the maximum cell–cell and cell–BM interaction distances R_{cca} and R_{cba} by using published measurements of breast cancer cell deformations. Byers et al. (1995) measured the deformation of MCF-7 (an adhesive, fairly nonaggressive breast cancer cell line) and MCF-10A (a benign cell line) BECs in shear flow conditions, and found the deformations to be bounded around 50–70% of the cell radius; this is an upper bound on R_{cca} and R_{cba}. Guck et al. (2005) measured BEC deformability (defined as additional stretched length over relaxed length) after 60 s of stress using an optical technique. The deformability was found to increase with malignant transformation: MCF10 measured 10.5% deformation, MCF7 measured 21.4%, MCF7 modified with a weaker cytoskeleton measured 30.4%, and MDA-MB-231 (an aggressive cancer cell line) measured 33.7% deformability. Because DCIS is moderately aggressive, we use the MCF7 estimate of 21.4% deformability, and thus set $R^i_{\text{cca}} = R^i_{\text{cba}} = 1.214 r_i$. It is likely that the cell–cell and cell–BM adhesive forces decrease rapidly with distance, and so we used the lowest (simplest) adhesion powers to capture a smooth decrease at the maximum interaction distances, setting $n_{\text{cca}} = n_{\text{cba}} = 1$. For simplicity, we set the repulsion powers to 1 as well.

4.5 Application to Patient-Specific Modeling

We now demonstrate calibration of the model to patient-specific data. After verifying that we have successfully calibrated the model, we give examples of its use for investigating open scientific and clinical questions, as well as its ability to generate hypotheses for future investigation.

4.5.1 Data Sources and Processing

Under the leadership of Mary Edgerton, we created a series of 13 index cases of DCIS from past M. D. Anderson Cancer Center patients who had undergone full mastectomies, rather than breast-conserving surgery. In the work below, we focus on calibrating the model to one of these 13 index cases. We now present a brief overview of the (deidentified) data we use for calibration. Full details are available in Edgerton et al. (2008, in preparation-a) and Chuang et al. (in preparation), including an in-depth analysis of the predictivity of a patient-calibrated, multiscale model of DCIS, of which this agent-based model is a component.

We selected a minimum of two and maximum of three formalin-fixed paraffin embedded (FFPE) blocks of tumor tissue. We selected blocks that contained the highest density of DCIS. Each block had a minimum of 1-cm^2 surface area of breast tissue for examination. Six μm-thick sections were cut for staining either with Hematoxylin and Eosin (H&E) to visualize the tumor, or with IHC for a specific antibody to identify and quantify a particular antigen and its subcellular location at a minimum resolution of 2 μm (the spatial resolution equivalent to the cells' nuclear half-width). To measure the proliferative index PI, we stained for Ki-67, the current "gold standard" proliferation marker that is present throughout the cell cycle, except portions of G1 (Gerdes et al. 1984). To measure the apoptotic index AI, we stained for cleaved Caspase-3, a key enzyme used throughout the apoptosis cycle to degrade subcellular structures (Duan et al. 2003).

To quantify the AI and PI on IHC-stained sections, we imaged multiple areas on the sections at 100× and 200× and analyzed these images using a custom-built Visual Basic plug-in for Image Pro Plus 4.5 to count the total number of Ki-67 and cleaved Caspase-3 positive nuclei (numerators for the PI and AI, respectively) and the total number of tumor cell nuclei (denominator for the PI and AI) within the tumor viable rims (see the solid-type DCIS in Fig. 4.1, bottom row). Due to the low number of apoptotic cells and low staining intensity, we counted the AI manually. We measured the duct radius and intraductal viable rim size using a calibrated scale embedded in the images. We calculated the cell density in the viable rim by dividing the total number of tumor cells by the area (in μm^2) of the viable rim. The measurement errors were estimated by recording the sample distributions (means and standard deviations).

A summary of the most important measurements for a (deidentified) case is given in Table 4.1.

4.5.2 Patient-Specific Calibration

Using this information and the protocols detailed in Macklin et al. (in preparation), we now calibrate the model to a specific case. Because the cells are essentially confluent in the viable rim, we estimate the tumor cell radius by $r_{\text{cell}} = \sqrt{1/(\rho\pi)} \approx 9.749\,\mu\text{m}$. Thus, by the earlier estimates of cell deformability, we set $r_{\text{cca}} = r_{\text{cba}} = 12\,\mu\text{m}$.

Table 4.1 Key data for a deidentified patient. Reprinted with permission from Macklin et al. (in preparation)

Quantity	Measured mean	Units
Duct radius r_{duct}	158.737	μm
Viable rim thickness T	78.873	μm
PI	17.429	%
Raw AI	0.638	%
Corrected AI	0.831	%
Cell density ρ	0.003217	cells per μm^2

We estimate the oxygen boundary condition σ_{B} by analytically solving the oxygen model to steady state in a rectangular domain, assuming the solution only varies with distance r from the duct center, that $\sigma = \sigma_{\text{H}} = 0.2$ at the interior boundary of the viable rim ($r_{\text{duct}} - T$), and $\sigma'(0) = 0$ (Macklin et al. in preparation). By this method, we estimate $\sigma_{\text{B}} \approx 0.3813$.

We use our measurements of AI and PI, along with our estimates of β_{A} and β_{P}, to uniquely determine $\langle \alpha_{\text{A}} \rangle = \alpha_{\text{A}}$ and $\langle \alpha_{\text{P}} \rangle = \overline{\alpha}_{\text{P}}(\langle \sigma \rangle - \sigma_{\text{H}})/(1 - \sigma_{\text{H}})$. (We estimate $\langle \sigma \rangle$ by integrating the analytical solution.) By the analysis in Macklin et al. (in preparation),

$$0 = \langle \alpha_{\text{P}} \rangle (1 - \text{AI} - \text{PI}) - \beta_{\text{P}}(\text{PI} + \text{PI}^2) + \beta_{\text{A}} \text{AI} \cdot \text{PI}, \tag{4.30}$$

$$0 = \alpha_{\text{A}}(1 - \text{AI} - \text{PI}) - \beta_{\text{A}}(\text{AI} - \text{AI}^2) - \beta_{\text{P}} \text{AI} \cdot \text{PI}. \tag{4.31}$$

Solving this and calculating $\overline{\alpha}_{\text{P}}$, we get $\alpha_{\text{A}}^{-1} = 40051.785$ min and $\overline{\alpha}_{\text{P}}^{-1} = 418.903$ min.

To calibrate the mechanical parameters, we balance the forces of cell–cell adhesion and cell–cell repulsion to enforce the measured cell density. If we approximate the cell arrangement as two-dimensional hexagonal circle packing (the optimal arrangement of circles in two dimensions), then the density $\rho = 0.003217$ cells per μm^2 is equivalent to a spacing of $s = \sqrt{2/(\sqrt{3}\rho)} \approx 18.946\,\mu$m between the cell centers. We set the adhesive and repulsive forces equal at this distance and solve for the ratio of the forces' parameters, yielding:

$$\frac{\alpha_{\text{cca}}}{\alpha_{\text{ccr}}} = 0.01810. \tag{4.32}$$

Notice that this does not fully constrain the magnitude of the forces; we are free to vary the magnitude of both forces so long as this ratio is maintained. This is equivalent to varying how strongly we enforce the mean density, and in the future we will constrain this number by attempting to match the standard deviation of the density as well. In our preliminary testing, we have found that setting $\alpha_{\text{cca}} = 0.1448$ and $\alpha_{\text{ccr}} = 8$, with $\alpha_{\text{cba}} = 0.1448$ and $\alpha_{\text{cbr}} = 5$ sufficiently enforces the density (Macklin et al. in preparation).

Table 4.2 Verification of the patient-specific calibration. Note that there is no standard deviation for the simulated cell density because it was calculated over the entire viable rim. Reprinted with permission from Macklin et al. (in preparation)

All figures given as mean ± standard deviation		
Quantity	Patient data	Simulated
PI (%)	17.429 ± 9.996	17.193 ± 7.216
AI (%)	0.8813 ± 0.5798	1.447 ± 3.680
Viable rim thickness (μm)	78.873 ± 13.538	80.615 ± 4.454
Cell density (cells/μm^2)	0.003217 ± 6.440e-4	0.003336

4.5.3 Verification of Calibration

We verified the calibration by checking the model's predictions of AI, PI, viable rim thickness, and density in the viable rim. We did this by slicing the computational domain at time $t = 30$ days into 6-μm-thick slices and performing virtual immunohistochemistry on those slices. We also calculated the viable rim thickness in each slice and the average cell density over the entire viable tumor region. The results can be found in Table 4.2. As we can see, the proliferative index (PI) matches extremely well. The apoptotic index (AI) is within error tolerances, and because apoptosis is a rare stochastic event (<1%) in a region containing fewer than 500 cells, considerable noise is anticipated, as can indeed be seen in the patient AI data as well. The viable rim thickness matches within the error bounds, and the cell density is in excellent agreement. Because we have attained all the numerical targets outlined in Sect. 4.5.2, the calibration is considered a success.

We also compared the general tumor morphology to H&E stains (bottom right in Fig. 4.3) and the spatial distribution of proliferating cells to Ki-67 immunostains (bottom left in Fig. 4.3, Ki-67 positive nuclei indicated with white arrows) from the patient. The virtual DCIS reproduced the expected tumor microarchitecture: a viable rim closest to the duct wall, an interior necrotic core, and sporadic interior microcalcifications. (There will be more on the matching of the calcified core in Sect. 4.5.4.) The simulation also recapitulated the general distribution of proliferating cells across the viable rim: in both the simulation and the Ki-67 imaging, cycling tumor cells were observed most frequently along the duct wall where oxygen is most plentiful, and almost never at the peri-necrotic boundary where substrate levels are lowest. This gives evidence to support our model of α_P depending upon σ.

4.5.4 Sample Applications of the Calibrated Model

4.5.4.1 Parameter Study: Necrosis and Calcification Time

There are little-to-no literature data available on the time to complete necrosis and calcify the breast tumor cells. The best available experimental data are generally

Table 4.3 Parameter study on the necrosis/calcification time. Reprinted with permission from Macklin et al. (in preparation)

β_N^{-1}	12 h	1 day	5 days	15 days	30 days
Percentage of core calcified	94.0%	83.7%	51.1%	6.9%	0%

animal and *ex vivo* time course studies of arterial calcification; we use these to estimate the order of magnitude of β_N^{-1}. Time course studies on *ex vivo* cardiac valves by Jian et al. (2003) observed significant tissue calcification between 7 days (10% increase in Ca incorporation) and 14 days (40% increase) after injection by TGF-β1. Lee et al. (2006a) examined a related process (elastin calcification) using a rat subdermal model, demonstrating calcification to occur gradually over the course of 2–3 weeks.

Gadeau et al. (2001) measured calcium accumulation in rabbit aortas following oversized balloon angioplasty injury. Calcified deposits appeared as soon as 2–4 days after the injury, increased over the course of 8 days, and approached a steady state between 8 and 30 days. The authors hypothesized based upon their work that necrotic cells nucleated the calcium crystals. Hence, we estimate β_N^{-1} on the order of days to weeks. To sharpen our estimate, we conducted a parameter study on the necrosis/calcification time parameter β_N^{-1}. We varied β_N^{-1} from 12 h to 30 days and simulated our calibrated DCIS model time to 30 days; the results are given in Table 4.3. We found that calcification times under 15 days lead to necrotic cores that were nearly entirely calcified; this is not observed in H&E image data (see Fig. 4.3, bottom right, black arrow). On the other hand, the 30-day calcification time lead (as expected) to a complete absence of microcalcifications in the core at time 30 days. Because DCIS tumors are hypothesized to grow to steady state in as little as 2–3 months (Edgerton et al. 2008, in preparation-a; Chuang et al. in preparation), we expect the presence of microcalcifications by this time. Hence, our sharpened estimate of β_N^{-1} is 15 days, consistent with the literature. Parameter studies such as these are significant, because they allow us to estimate physical quantities that are difficult or impossible to determine experimentally.

Using Model Shortcomings to Better Understand DCIS Necrosis and Calcification

This work also points out the importance of extracting as much information from the H&E and immunohistochemical stains as possible, both to further constrain the model and to suggest scientific hypotheses and model refinements. Notice that the simulated tumor morphology (Fig. 4.3, top) does not recapitulate the observed "gap" between the viable rim and the necrotic core observed in both the H&E stain (Fig. 4.3, bottom right, white arrow) and immunohistochemistry (Fig. 4.3, bottom left). We hypothesize that the discrepancy is caused by the combination of two effects:

1. *Separation of time scales*. In our necrosis model, cell water loss, degradation of surface (adhesion) receptors, and cell calcification occur at the same, slow rate.

However, necrotic cells are known to swell (due to failure of ion pumps, the accumulation of sodium ions, and subsequent osmosis) and burst relatively early in the necrosis process. Thus, we expect volume loss to occur more rapidly than receptor degradation and calcification.

2. *Differential adhesion*. In our model, the necrotic cells initially have the same E-cadherin level, which then decays over the time course of necrosis. However, due to the relatively rapid cell rupture, the cells should initially concentrate the same number of E-cadherin molecules over a smaller surface area, creating a relatively high-adhesion species (necrotic cells) relative to the viable tumor cells. Due to the differential adhesive forces, we anticipate cell sorting behavior and, in particular, compaction of the necrotic cells. Indeed, similar cell sorting behavior was observed in the classic experiment by Armstrong (1971), where two cell populations (low adhesive strength and high adhesive strength) were mixed in chicken embryos; cell sorting was observed, where the high-adhesion cells aggregated as a spheroid surrounded by the low-adhesion cells.

Indeed, a closer examination of our data supports these ideas:

1. The prominence and relatively high density of the cell nuclei in the necrotic debris (Fig. 4.3, bottom left, gray arrow) collectively indicate that the cells burst more quickly than they degrade and calcify.
2. Preliminary E-cadherin immunostaining demonstrates relatively high concentrations of E-cadherin on lysed necrotic debris. This lends support to the high-adhesion species idea, as well as the separation of timescales between the degradation of surface receptors (slow timescale) and cell lysis (fast timescale).

These hypotheses can be tested by modifying the necrosis model and comparing the results. Recapitulating the "gap" between the viable rim and necrotic core would be evidence in support of the hypotheses.

Predictions of Tumor Growth vs. Oxygen Availability

If we volume-average the model behavior throughout the viable rim as in Macklin et al. (in preparation), we obtain a system of nonlinear ODEs governing the AI and PI:

$$\dot{\mathrm{PI}} = \langle \alpha_{\mathrm{P}} \rangle (1 - \mathrm{AI} - \mathrm{PI}) - \beta_{\mathrm{P}} \left(\mathrm{PI} + \mathrm{PI}^2 \right) + \beta_{\mathrm{A}} \mathrm{AI} \cdot \mathrm{PI} \tag{4.33}$$

$$\dot{\mathrm{AI}} = \alpha_{\mathrm{A}} (1 - \mathrm{AI} - \mathrm{PI}) - \beta_{\mathrm{A}} \left(\mathrm{AI} - \mathrm{AI}^2 \right) - \beta_{\mathrm{P}} \mathrm{AI} \cdot \mathrm{PI}. \tag{4.34}$$

If, instead, we replace $\langle \alpha_{\mathrm{P}} \rangle$ with $\alpha_{\mathrm{P}}(\mathcal{S}, \sigma, \bullet)$, we get the local evolution of AI and PI as a function of oxygen. Assuming that a local equilibrium emerges in the population dynamics (i.e., $\dot{\mathrm{AI}} \approx 0$ and $\dot{\mathrm{PI}} \approx 0$, even if $\dot{P}$ and $\dot{A}$ are nonzero (Macklin and Lowengrub 2007)), we can investigate the model-predicted relationship between PI and oxygen by solving the ODE system to steady state with our calibrated α_{P} and α_{A}.

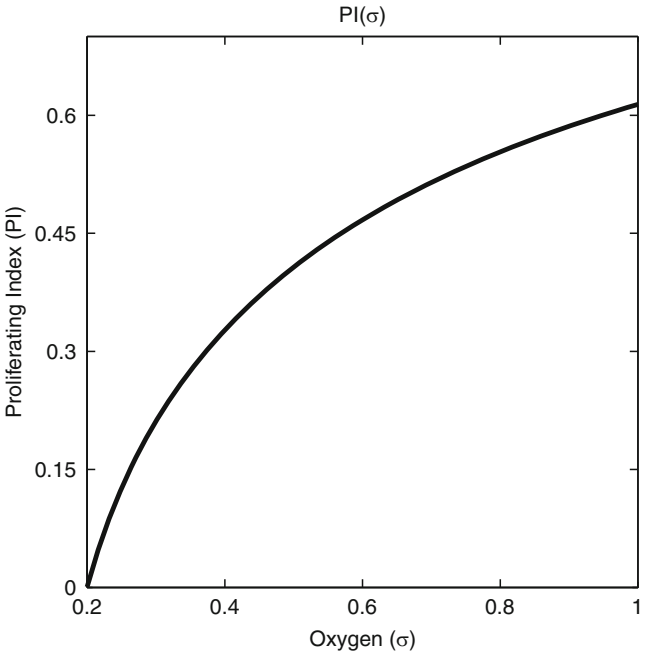

Fig. 4.4 Model-predicted relationship between PI and oxygen.

The model predicts a Michaelis–Menton-type curve, where oxygen ceases to be the primary growth-limiting factor for sufficient tissue oxygenation (Fig. 4.4). Because the rate of tumor growth is proportional to PI (Chuang et al. in preparation; Edgerton et al. in preparation-a), the model predicts a Michaelis–Menton tumor proliferation growth term of the form $\lambda(\sigma^m/(c+\sigma^m))$.

This is significant for several reasons. First, there is some variety in tumor growth models, with some choosing Michaelis–Menton-type growth terms (Ward and King 1997, 1999; Xu and Gilbert 2009) and others using linear growth terms of the form $\lambda\sigma$ (Cristini et al. 2003). Examining Fig. 4.4, we see that the linear model is accurate for moderate oxygen conditions, whereas the Michaelis–Menton-type model is likely more accurate overall. Choosing a linear model subject to the constraint that proliferation is zero when $\sigma = 0$ can likely be improved with a careful least-squares fit to the model-predicted growth curve.

The predicted growth curve also has interesting biological and clinical implications. The saturation of the curve is evidence that for this particular patient, tumor growth is limited primarily by factors other than oxygen availability, such as gene expression, receptor levels, or overcrowding. Had the growth curve been linear, then we would have instead concluded that oxygen was the principal growth-limiting factor, rather than genetic profile.

Perhaps more importantly, because the model is built upon only simple, mechanistic assumptions and no particular growth curve is expected *a priori*, it is encouraging to see such properties demonstrated as emergent behavior of the model. Indeed, this demonstrates the power of using simple, mechanistic models of tumor growth as tools to generate testable scientific hypotheses. In this instance, we plan to return to the IHC data and attempt to correlate the PI measurements with the estimated oxygen profiles, to see if it, too, can predict a Michaelis–Menton-type curve. The hypothesis could also be tested with appropriate *in vitro* experiments.

4.6 Ongoing and Future Work

The model and applications we presented in this chapter represent a "snapshot" of the ongoing work by our group. The modular nature of agent-based models will enable some exciting future extensions, including (1) an improved necrosis model that includes cell swelling and bursting early in the process, increased cell receptor expression per surface area, and gradual degradation and calcification of the remaining solid component; (2) models of polarized cell adhesion, by modifying the potential functions and adding the balance of torque; (3) an intracellular protein signaling network component, the state of which alters the α_P and α_A parameters; (4) improved calibration of the model by incorporating additional biomarkers to better constrain the model, particularly when joined with the protein signaling model; (5) a "motile" state, with a corresponding α_M depending upon proximity to the basement membrane and the internal protein signaling network, with rate and direction of travel dependent upon sampling of the microenvironment (e.g., hormone gradients and BM structure); (6) secretion of matrix metalloproteinases by motile cells that degrade the basement membrane, alongside expansion of the duct caused by proliferating cells; and (7) microinvasion of tumor cells through the degraded BM, as a first model of progression to invasive carcinoma.

In recent work, we have integrated this agent-based cell model into a broader, multiscale model of DCIS being developed by our group. Once calibrated, the multiscale model is capable of making patient-specific predictions at the tissue scale of the required excision volume to remove DCIS surgically (Edgerton et al. 2008, in preparation-a; Chuang et al. in preparation). Further information on the multiscale model and its potential clinical applications can be found in Cristini and Lowengrub (in preparation).

Acknowledgments This work was partially funded by a generous grant from the Cullen Trust for Health Care (VC) and by the National Science Foundation (VC). We thank Yao-Li Chuang and Sandeep Sanga at the University of Texas Health Science Center-Houston for insightful discussions on tumor growth and protein signaling modeling.

References

Adalsteinsson D, Sethian JA (1999) The fast construction of extension velocities in level set methods. J Comput Phys 148(1):2–22. doi:10.1006/jcph.1998.6090

Adamovich TL, Simmons RM (2003) Ductal carcinoma in situ with microinvasion. Am J Surg 186(2):112–116. doi:10.1016/S0002-9610(03)00166-1

Ai L, Kim WJ, Kim TY, Fields CR, Massoll NA, Robertson KD, Brown KD (2006) Epigenetic silencing of the tumor suppressor cystatin m occurs during breast cancer progression. Cancer Res 66:7899–7909

American Cancer Society (2007) American cancer society breast cancer facts and figures 2007–2008. American Cancer Society, Inc., Atlanta. http://www.cancer.org/downloads/STT/BCFF-Final.pdf

Anderson E (2004) Cellular homeostasis and the breast. Maturitas 48(S1):13–17. doi:10.1016/j.maturitas.2004.02.010

Anderson ARA, Weaver AM, Cummings PT, Quaranta V (2006) Tumor morphology and phenotypic evolution driven by selective pressure from the microenvironment. Cell 127(5):905–915. doi:10.1016/j.cell.2006.09.042

Armstrong PB (1971) Light and electron microscope studies of cell sorting in combinations of chick embryo neural retina and retinal pigment epithelium. Wilhelm Roux' Arch 168:125–141

Bankhead III A, Magnuson NS, Heckendorn RB (2007) Cellular automaton simulation examining progenitor heirarchy structure effects on mammary ductal carcinoma in situ. J Theor Biol 246(3):491–498. doi:10.1016/j.jtbi.2007.01.011

Barros LF, Hermosilla T, Castro J (2001) Necrotic volume increase and the early physiology of necrosis. Comp Biochem Physiol A Mol Integr Physiol 130:401–409

Baxter FO, Neoh K, Tevendale MC (2007) The beginning of the end: Death signaling in early involution. J Mamm Gland Biol Neoplasia 12(1):3–13. doi:10.1007/s10911-007-9033-9

Bienz M, Clevers H (2000) Linking colorectal cancer to Wnt signaling. Cell 103:311–320

Butler LM, Khan S, Rainger GE, Nash GB (2008) Effects of endothelial basement membrane on neutrophil adhesion and migration. Cell Immunol 251:56–61. doi:10.1016/j.cellimm.2008.04.004

Byers SM, Sommers CL, Hoxter B, Mercurio AM, Tozeren A (1995) Role of e-cadherin in the response of tumor cell aggregates to lymphatic, venous, and arterial flow: measurement of cell–cell adhesion strength. J Cell Sci 108(5):2053–2064

Byrne H, Drasdo D (2009) Individual-based and continuum models of growing cell populations: a comparison. J Math Biol 58(4–5):657–687. doi:10.1007/s00285-008-0212-0

Byrne HM, Alarcon T, Owen MR, Webb SD, Maini PK (2006) Modelling cancer dynamics: a review. Philos Trans R Soc A 364:1563–1578. doi:10.1098/rsta.2006.1786

Cabioglu N, Hunt KK, Sahin AA, Kuerer HM, Babiera GV, Singletary SE, Whitman GJ, Ross MI, Ames FC, Feig BW, Buchholz TA, Meric-Bernstam F (2007) Role for intraoperative margin assessment in patients undergoing breast-conserving surgery. Ann Surg Oncol 14(4):1458–1471

Cheng L, Al-Kaisi NK, Gordon NH, Liu AY, Gebrail F, Shenk RR (1997) Relationship between the size and margin status of ductal carcinoma in situ of the breast and residual disease. J Natl Cancer Inst 89(18):1356–1360

Chuang YL, Edgerton ME, Macklin P, Wise S, Lowengrub JS, Cristini V (in preparation) Clinical predictions of bulk DCIS properties based on a duct-scale mixture model

Ciatto S, Bianchi S, Vezzosi V (1994) Mammographic appearance of calcifications as a predictor of intraductal carcinoma histologic subtype. Eur Radiol 4(1):23–26. doi:10.1007/BF00177382

Collins LC, Tamimi RM, Baer HJ, Connolly JL, Colditz GA, Schnitt SJ (2005) Outcome of patients with ductal carcinoma in situ untreated after diagnostic biopsy: results from the Nurses' Health Study. Cancer 103(9):1778–1784

Cotran RS, Kumar V, Robbins SL (1994) Robbins Pathologic Basis of Disease, 5th edn. W.B. Saunders, Philadelphia, PA

Cristini V, Lowengrub JS (in press) Multiscale Modeling of Cancer. Cambridge University Press, New York, NY

Cristini V, Lowengrub JS, Nie Q (2003) Nonlinear simulation of tumor growth. J Math Biol 46:191–224. doi:10.1007/s00285-002-0174-6

Danes CG, Wyszomierski SL, Lu J, Neal CL, Yang W, Yu D (2008) 14-3-3ζ down-regulates p53 in mammary epithelial cells and confers luminal filling. Cancer Res 68:1760–1767. doi:10.1158/0008-5472.CAN-07-3177

Dillon MF, McDermott EW, O'Doherty A, Quinn CM, Hill AD, O'Higgins N (2007) Factors affecting successful breast conservation for ductal carcinoma in situ. Ann Surg Oncol 14(5):1618–1628

Dillon R, Owen M, Painter K (2008) A single-cell based model of multicellular growth using the immersed boundary method. In: Cheong K, Li Z, Lin P (eds) Moving interface problems and applications in fluid dynamics. AMS Contemporary Mathematics. American Mathematical Society, Providence, RI

Duan WR, Garner DS, Williams SD, Funckes-Shippy CL, Spath IS, Blomme EAG (2003) Comparison of immunohistochemistry for activated caspase-3 and cleaved cytokeratin 18 with the TUNEL method for quantification of apoptosis in histological sections of PC-3 subcutaneous xenografts. J Pathol 199(2):221–228

Edgerton M, Chuang YL, Kim J, Tomaiuolo G, Macklin P, Sanga S, Yang W, Broom A, Do KA, Cristini V (2008) Using mathematical models to understand the time dependence of the growth of ductal carcinoma in situ. In: 31st Annual San Antonio breast cancer symposium supplement to volume 68(24), Abstract 1165

Edgerton ME, Macklin P, Chuang YL, Tomaiuolo G, Kim J, Sanga S, Broom A, Yang W, Do KA, Cristini V (in review) A Multiscale Mathematical Models for Improved Pre-Operative Estimates of Ductal Carcinoma in Situ Size, Canc. Res.

Edgerton ME, Mannes K, Dudek S, Jensen R, Page D (in preparation-b) Pagetoid spread of ductal carcinoma in situ

Franks SJ, Byrne HM, King JR, Underwood JCE, Lewis CE (2003a) Modelling the early growth of ductal carcinoma in situ of the breast. J Math Biol 47:424–452. doi:10.1007/s00285-003-0214-x

Franks SJ, Byrne HM, Mudhar H, Underwood JCE, Lewis CE (2003b) Modelling the growth of comedo ductal carcinoma in situ. Math Med Biol 20:277–308

Franks SJ, Byrne HM, Underwood JCE, Lewis CE (2005) Biological inferences from a mathematical model of comedo ductal carcinoma in situ of the breast. J Theor Biol 232(4):523–543. doi:10.1016/j.jtbi.2004.08.032

Frieboes HB, Zheng X, Sun CH, Tromberg B, Gatenby R, Cristini V (2006) An integrated computational/experimental model of tumor invasion. Cancer Res 66(3):1597–1604

Frieboes HB, Lowengrub JS, Wise S, Zheng X, Macklin P, Cristini V (2007) Computer simulations of glioma growth and morphology. NeuroImage 37(S1):S59–S70. doi:10.1016/j.neuroimage.2007.03.008

Frieboes HB, Edgerton ME, Fruehauf JP, Rose FRAJ, Worrall LK, Gatenby RA, Ferrari M, Cristini V (2009) Prediction of drug response in breast cancer using integrative experimental/computational modeling. Cancer Res 69(10):4484–4492

Gadeau AP, Chaulet H, Daret D, Kockx M, Daniel-Lamaziè JM, Desgranges C (2001) Time course of osteopontin, osteocalcin, and osteonectin accumulation and calcification after acute vessel wall injury. J Histochem Cytochem 49:79–86

Gerdes J, Lemke H, Baisch H, Wacker HH, Schwab U, Stein H (1984) Cell cycle analysis of a cell proliferation-associated human nuclear antigen defined by the monoclonal antibody Ki-67. J Immunol 133(4):1710–1715

Going JJ, Mohun TJ (2006) Human breast duct anatomy, the sick lobe hypothesis and intraductal approaches to breast cancer. Breast Cancer Res Treat 97(3):285–291. http://dx.doi.org/10.1007/s10549-005-9122-7

Greenspan HP (1976) On the growth and stability of cell cultures and solid tumors J Theor Biol 56(1):229–242. doi:10.1016/S0022-5193(76)80054-9

Guck J, Schinkinger S, Lincoln B, Wottawah F, Ebert S, Romeyke M, Lenz D, Erickson HM, Ananthakrishnan R, Mitchell D, Käs J, Ulvick S, Bilby C (2005) Optical deformability as an inherent cell marker for testing malignant transformation and metastatic competence. Biophys J 88(5):3689–3698. doi:10.1529/biophysj.104.045476

Hanahan D, Weinberg RA (2000) The hallmarks of cancer. Cell 100(1):57–70. doi:10.1016/S0092-8674(00)81683-9

Hansen RK, Bissell MJ (2000) Tissue architecture and breast cancer: the role of extracellular matrix and steroid hormones. Endocr Relat Cancer 7(2):95–113. doi:10.1677/erc.0.0070095

Harms BD, Bassi GM, Horwitz AR, Lauffenburger DA (2005) Directional persistence of EGF-induced cell migration is associated with stabilization of lamellipodial protrusions. Biophys J 88(2):1479–1488

Hino Si, Tanji C, Nakayama KI, Kikuchi A (2005) Phosphorylation of β-catenin by cyclic AMP-dependent protein kinase stabilizes β-catenin through inhibition of its ubiquitination. Mol Cell Biol 25(20):9063–9072. doi:10.1128/MCB.25.20.9063-9072.2005

Hu Z, Yuri K, Ozawa H, Lu H, Kawata M (1997) The in vivo time course for elimination of adrenalectomy-induced apoptotic profiles from the granule cell layer of the rat hippocampus. J Neurosci 17(11):3981–3989

Ilic D, Almeida EA, Schlaepfer DD, Dazin P, Aizawa S, Damsky CH (1998) Extracellular matrix survival signals transduced by focal adhesion kinase suppress p53-mediated apoptosis. J Cell Biol 143:547–560

Jemal A, Siegel R, Ward E, Murray T, Xu J, Thun MJ (2007) Cancer statistics, 2007. CA Cancer J Clin 57(1):43–66

Jha MK, Avlontitis VS, Griffith CDM, Lennard TWJ, Wilson RG, McLean LM, Dawes PD, Shrinmankar J (2001) Aggressive local treatment for screen-detected DCIS results in very low rates of recurrence. Eur J Surg Oncol 27(5):454–458. doi:10.1053/ejso.2001.1163

Jian B, Narula N, Li QY, Mohler III ER, Levy RJ (2003) Progression of aortic valve stenosis: TGF-β1 is present in calcified aortic valve cusps and promotes aortic valve interstitial cell calcification via apoptosis. Ann Thoracic Surg 75(2):457–465. doi:10.1016/S0003-4975(02)04312-6

Kerlikowske K, Molinaro A, Cha I, Ljung BM, Ernster VL, Stewart K, Chew K, Moore 2nd DH, Waldman F (2003) Characteristics associated with recurrence among women with ductal carcinoma in situ treated by lumpectomy. J Natl Cancer Inst 95(22):1692–1702

Kerr JF, Winterford CM, Harmon BV (1994) Apoptosis. Its significance in cancer and cancer therapy. Cancer 15(8):2013–2026

Khan S, Rogers M, Khurana K, Meguid M, Numann P (1998) Estrogen receptor expression in benign breast epithelium and breast cancer risk. J Natl Cancer Inst 90:37–42

Khan S, Sachdeva A, Naim S, Meguid M, Marx W, Simon H, et al (1999) The normal breast epithelium of women with breast cancer displays an aberrant response to estradiol. Cancer Epidemiol Biomarkers Prev 8:867–872

Kopans DB, Rafferty E, Georgian-Smith D, Yeh E, D'Alessandro H, Hughes K, Halpern E (2003) A simple model of breast carcinoma growth may provide explanations for observations of apparently complex phenomena. Cancer 97(12):2951–2959

Krysko DV, Berghe TV, D'Herde K, Vandenabeele P (2008) Apoptosis and necrosis: Detection, discrimination and phagocytosis. Methods 44:205–221. doi:10.1016/j.ymeth.2007.12.001

Lampejo OT, Barnes DM, Smith P, Millis RR (1994) Evaluation of infiltrating ductal carcinomas with a DCIS component: correlation of the histologic type of the in situ component with grade of the infiltrating component. Semin Diagn Pathol 11(3):215–222

Lee JS, Basalyga DM, Simionescu A, Isenburg JC, Sinionescu DT, Vyavahare NR (2006a) Elastin calcification in the rate subdermal model is accompanied by up-regulation of degradative and osteogenic cellular responses. Am J Pathol 168:490–498. doi:10.2353/ajpath.2006.050338

Lee S, Mohsin SK, Mao S, Hilsenbeck SG, Medina D, Allred DC (2006b) Hormones, receptors, and growth in hyperplastic enlarged lobular units: early potential precursors of breast cancer. Breast Cancer Res 8(1):R6

Lustig B, Jerchow B, Sachs M, Wiler S, Pietsch T, Karsten U, van de Wetering M, Clevers H, Schlag PM, Birchmeier W, Behrens J (2002) Negative feedback loop of Wnt signaling through upregulation of conductin/axin2 in colorectal and liver tumors. Mol Cell Biol 22:1184–1193

Macklin P, Lowengrub JS (2005) Evolving interfaces via gradients of geometry-dependent interior Poisson problems: application to tumor growth 203(1):191–220. doi:10.1016/j.jcp.2004.08.010

Macklin P, Lowengrub JS (2006) An improved geometry-aware curvature discretization for level set methods: application to tumor growth J Comput Phys 215(2):392–401. doi:10.1016/j.jcp.2005.11.016

Macklin P, Lowengrub JS (2007) Nonlinear simulation of the effect of microenvironment on tumor growth. J Theor Biol 245(4):677–704. doi:10.1016/j.jtbi.2006.12.004

Macklin P, Lowengrub JS (2008) A new ghost cell/level set method for moving boundary problems: application to tumor growth. J Sci Comput 35(2–3):266–299. doi:10.1007/s10915-008-9190-z

Macklin P, McDougall SR, Anderson ARA, Chaplain MAJ, Lowengrub J (2009) Multiscale modelling and nonlinear simulation of vascular tumour growth. J Math Biol 58(4–5):765–798. doi:10.1007/s00285-008-0216-9

Macklin P, Kim J, Tomaiuolo G, Edgerton ME, Cristini V (in preparation) An agent-based cell model, with application to patient-specific ductal carcinoma in situ modeling

Macknight ADC, DiBona DR, Leaf A, Mortimer MC (1971) Measurement of the composition of epithelial cells from the toad urinary bladder. J Membr Biol 6(2):108–126. doi:10.1007/BF01873458

Malladi R, Sethian JA, Vemuri BC (1995) Shape modeling with front propagation: a level set approach. IEEE Trans Pattern Anal Mach Intell 17(2):158–175. doi:10.1109/34.368173

Malladi R, Sethian JA, Vemuri BC (1996) A fast level set based algorithm for topology-independent shape modeling. J Math Imaging Vis 6(2–3):269–289. doi:10.1007/BF00119843

Mannes KD, Edgerton ME, Simpson JF, Jenson RA, Page DL (2002) Pagetoid spread in ductal carcinoma in situ: characterization and computer simulation. In: United States and Canadian Academy of Pathology (USCAP) annual meeting 2002, Chicago, IL

Moffat DF, Going JJ (1996) Three dimensional anatomy of complete duct systems in human breast: pathological and developmental implications. J Clin Pathol 49:48–52. doi:10.1136/jcp.49.1.48

Ohtake T, Kimijima I, Fukushima T, Yasuda M, Sekikawa K, Takenoshita S, Abe R (2001) Computer-assisted complete three-dimensional reconstruction of the mammary ductal/lobular systems. Cancer 91:2263–2272

Øksendal B (2007) Stochastic differential equations: an introduction with applications, 6th edn. Springer, New York

Osher S, Fedkiw R (2001) Level set methods: an overview and some recent results 169(2):463–502. doi:10.1006/jcph.2000.6636

Osher S, Fedkiw R (2002) Level set methods and dynamic implicit surfaces. Springer, New York

Osher S, Sethian JA (1988) Fronts propagating with curvature-dependent speed: algorithms based on Hamilton–Jacobi formulations 79(1):12–49. doi:10.1016/0021-9991(88)90002-2

Owen MR, Byrne HM, Lewis CE (2004) Mathematical modelling of the use of macrophages as vehicles for drug delivery to hypoxic tumour sites. J Theor Biol 226(4):377–391

Page DL, Dupont WD, Rogers LW, Landenberger M (1982) Intraductal carcinoma of the breast: follow-up after biopsy only. Cancer 49(4):751–758

Panorchan P, Thompson MS, Davis KJ, Tseng Y, Konstantopoulos K, Wirtz D (2006) Single-molecule analysis of cadherin-mediated cell-cell adhesion. J Cell Sci 119:66–74. doi:10.1242/jcs.02719

Patani N, Cutuli B, Mokbel K (2008) Current management of DCIS: a review. Breast Cancer Res Treat 111(1):1–10

Rejniak KA, Dillon RH (2007) A single cell-based model of the ductal tumour architecture. Comput Math Methods Med 8(1):51–69

Sanders ME, Schuyler PA, Dupont WD, Page DL (2005) The natural history of low-grade ductal carcinoma in situ of the breast in women treated by biopsy only revealed over 30 years of long-term follow-up. Cancer 103(12):2481–2484

Scarlett JL, Sheard PW, Hughes G, Ledgerwood EC, Ku HH, Murphy MP (2000) Changes in mitochondrial membrane potential during staurosporine-induced apoptosis in jurkat cells. IFEBS Lett 475:267–272

Seidensticker MJ, Behrens J (2000) Biochemical interactions in the wnt pathway. Biochim Biophys Acta 1495:168–182

Sethian JA (1999) Level set methods and fast marching methods. Cambridge University Press, New York

Sethian JA, Smereka P (2003) Level set methods for fluid interfaces. Ann Rev Fluid Mech 35(1):341–372. doi:10.1146/annurev.fluid.35.101101.161105

Shiryaev AN (1995) Probability, 2nd edn. Springer, New York

Shuryak I, Sachs RK, Hlatky L, Little MP, Hahnfeldt P, Brenner D (2006) Radiation-induced leukemia at doses relevant to radiation therapy: modeling mechanisms and estimating risks. J Natl Cancer Inst 98(24):1794–1806. doi:10.1093/jnci/djj497

Silver SA, Tavassoli FA (1998) Ductal carcinoma in situ with microinvasion. Breast J 4(5):344–348

Silverstein MJ (1997a) Predicting residual disease and local recurrence in patients with ductal carcinoma in situ. J Natl Cancer Inst 89(18):1330–1331

Silverstein MJ (1997b) Recent advances: diagnosis and treatment of early breast cancer. Br Med J 314(7096):1736ff

Silverstein MJ (2000) Ductal carcinoma in situ of the breast. Annu Rev Med 51:17–32. doi:10.1146/annurev.med.51.1.17

Simpson PT, Reis-Filho JS, Gale T, Lakhani SR (2005) Molecular evolution of breast cancer. J Pathol 205(2):248–254. doi:10.1002/path.1691

Sontag L, Axelrod DE (2005) J Theor Biol 232(2):179–189. doi:10.1016/j.jtbi.2004.08.002

Stomper PC, Margolin FR (1994) Ductal carcinoma in situ: the mammographer's perspective. Am J Roentgenol 162:585–591

Sussman M, Smereka P, Osher S (1994) A level set approach for computing solutions to incompressible two-phase flow J Comput Phys 114(1):146–159. doi:10.1006/jcph.1994.1155

Tannis PJ, Nieweg OE, Valdés Olmos RA, Kroon BBR (2001) Anatomy and physiology of lymphatic drainage of the breast from the perspective of sentinel node biopsy. J Am Coll Surg 192(3):399–409. doi:10.1016/S1072-7515(00)00776-6

Thorne BC, Bailey AM, Peirce SM (2007) Combining experiments with multi-cell agent-based modeling to study biological tissue patterning. Breif Bioinform 8(4):245–257. doi:10.1093/bib/bbm024

Wang R, Jinming L, Lyte K, Yashpal NK, Fellows F, Goodyer CG (2005) Role for $\beta 1$ integrin and its associated $\alpha 3$, $\alpha 5$, and $\alpha 6$ subunits in development of the human fetal pancreas. Diabetes 54:2080–2089

Ward JP, King JR (1997) Mathematical modelling of avascular-tumour growth. IMA J Math Appl Med Biol 14(1):39–69

Ward JP, King JR (1999) Mathematical modelling of avascular tumour growth II: Modelling growth saturation. IMA J Math Appl Med Biol 16:171–211

Wei C, Larsen M, Hoffman MP, Yamada KM (2007) Self-organization and branching morphogenesis of primary salivary epithelial cells. Tissue Eng 13(4):721–735. doi:10.1089/ten.2006.0123

Wellings SR, Jensen HM, Marcum RG (1975) An atlas of subgross pathology of the human breast with special reference to possible precancerous lesions. J Natl Cancer Inst 55(2):231–273

Xu Y, Gilbert R (2009) Some inverse problems raised from a mathematical model of ductal carcinoma in situ. Math Comput Model 49(3-4):814–828. doi:10.1016/j.mcm.2008.02.014

Zaman MH, Kamm RD, Matsudaira P, Lauffenburger DA (2005) Computational model for cell migration in three-dimensional matrices. Biophys J 89(2):1389–1397

Zhang L, Athale CA, Deisboeck TS (2007) Development of a three-dimensional multiscale agent-based tumor model: simulating gene-protein interaction profiles, cell phenotypes and multicellular patterns in brain cancer. J Theor Biol 244(1):96–107

Chapter 5
Multicluster Class-Based Classification for the Diagnosis of Suspicious Areas in Digital Mammograms

Brijesh Verma

Abstract This chapter presents a multicluster class-based classification approach for the classification of suspicious areas extracted from digital mammograms into benign and malignant classes. The approach creates multiple clusters and selects strong clusters for each class. The created strong clusters are used to form subclasses within benign and malignant classes and training of a classifier. The creation of strong multiple clusters during the classification process can improve the accuracy of the classification system. The experiments using multicluster class-based approach and a standard classifier with a single cluster per class have been conducted on a benchmark database of digital mammograms. The results have shown that the multicluster class-based approach makes a significant impact on improving the classification accuracy.

5.1 Introduction

5.1.1 Background

Breast cancer kills more women in Australia than any other cancer (National Breast Cancer Foundation 2008). Each year more than 11,700 new cases of breast cancer and 2,600 deaths occur in Australia. In the USA, an estimated 1,437,180 new cases of breast cancer occur together with an estimated mortality of 565,550 during 2008 (American Cancer Society 2008; Breast cancer facts and figures 2007–2008). Survival from breast cancer is dependent on the stage at which it is detected and the implementation of appropriate treatment. Early stage detection and treatment results in a 98% survival rate; however, this plummets to 28% if metastases have spread to distant organs (American Cancer Society 2008; Global cancer facts and figures 2007).

B. Verma (✉)
School of Computing Sciences, CQUniversity, Bruce Highway, North Rockhampton, Queensland 4702, Australia
e-mail: b.verma@cqu.edu.au

T. Pham (ed.), *Computational Biology: Issues and Applications in Oncology*, Applied Bioinformatics and Biostatistics in Cancer Research,
DOI 10.1007/978-1-4419-0811-7_5,

Mammography is one of the best methods for early detection of breast cancer. It reduces the mortality rate by as much as 41% (Roder et al. 2008). However, various studies have demonstrated that an estimated 11–25% of breast cancers are missed (Goergen et al. 1997) during screening mammography. In the screening process, radiologists carefully search each mammogram for any visual sign of abnormality. However, in the earliest stage, the visual clues are subtle and varied in appearance, making detection/diagnosis difficult; challenging even for specialists. The abnormalities are often embedded in and camouflaged by various breast tissue structures.

An abnormal growth of tissues in the breast creates lumps or tumors in the breast. Tumors perform no useful body function and grow at the expense of healthy tissues. Mammography is capable of detecting such features that may indicate a potential clinical problem, which include asymmetries between the breasts, architectural distortion, confluent densities associated with benign fibroses, calcifications, and masses. Mammographic abnormalities in breast cancer can be characterized into two major classes; calcifications and masses as shown in Fig. 5.1.

Calcifications are small mineral (calcium) deposits within breast tissues and appear as localized high-intensity regions (bright spots) on mammograms. They can be produced from cell secretion or necrotic cellular debris. There are two types of calcifications: microcalcifications and macrocalcifications. Macrocalcifications are coarse (large) calcium deposits that are often associated with benign (noncancerous) conditions and do not usually require a biopsy. Microcalcifications are tiny calcium specks and may be isolated, appear in cluster, or found embedded in a mass. In general, individual microcalcifications are found in size range of 0.1–1.0 mm with an average diameter of about 0.5 mm. A cluster is typically defined by the presence of at least three microcalcifications within a 1 cm^2 region. Microcalcifications are one of the mammographic hallmarks of early breast cancer. About 25% of all breast

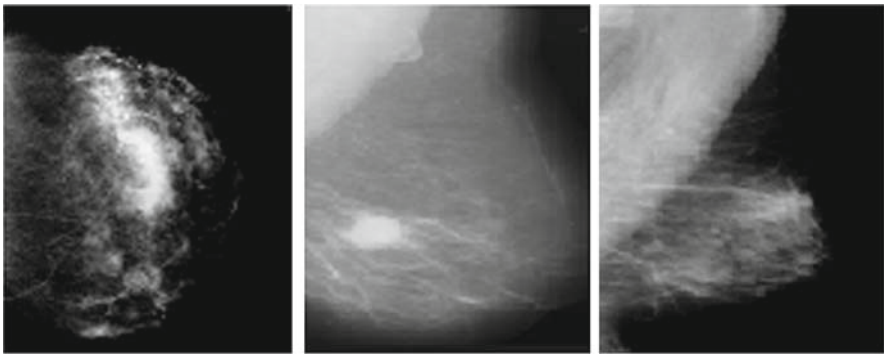

Fig. 5.1 Breast Abnormalities. (**a**) Calcification and (**b**) mass. *Source*: The mammography image analysis society (MIAS 2006) database

cancer is detected by the presence of microcalcification clusters. The majority of ductal carcinoma in situ cancers are associated with microcalcifications.

A breast mass is a localized sign of breast cancer and defined as a space-occupying lesion seen in at least two different projections. A mass may occur with or without associated calcifications. Figure 5.1b shows circumscribed and spiculated masses. The common circumscribed mass is fibroadenoma and is usually found in younger age women. Spiculated lesions having stellate appearance on mammograms are high-probability suspicious indicators of breast cancer. Generally, it has a central tumor mass with spicules extending into surrounding tissues. The subtlety and camouflaged nature of masses make their detection far more challenging than calcifications in mammograms.

Masses are described by their location, size, shape, and margin characteristics. These morphological features are the key factors to be considered by the radiologist when judging the likelihood of cancer being present. A wide range of shapes, sizes, and contrast is found in masses. Generally, benign masses are well circumscribed, compact, and roughly circular or elliptical. Masses having irregular shapes and spiculated or indistinct margins suggest a higher possibility of malignancy.

5.1.2 Review of Existing Techniques

In last few decades, a significant amount of research on detection and classification of suspicious areas has been conducted and many techniques have been developed. It has been shown that Computer-Aided Diagnosis (CAD) systems for breast cancer can improve the detection rate from 4.7% to 19.5% compared to radiologists (Brem 2007; Freer and Ulissey 2001; Dean and Ilvento 2006; Birdwell et al. 2005; Morton et al. 2006). To solve the problems with the diagnosis of breast cancer, various intelligent and statistical techniques have been proposed. A comprehensive review of existing techniques for the detection and classification of masses and microcalcifications in digital mammograms has been recently conducted and presented in Cheng et al. (2006), Verma and Panchal (2006), and Cheng et al. (2003).

Ramirez et al. (2007) used seven Bayesian network classifiers for the diagnosis of breast cancer on two real-world databases. They used the breast lesion cases collected by a single and multiple observers and obtained an accuracy of 93.04% and 83.31%, respectively.

Tourassi et al. (2007) evaluated image similarity measures for detection of masses in mammograms. They used database of 1,820 mammographic regions of interest and compared 8 entropy-based similarity measures. They concluded that a substantial reduction in false positive while maintaining high detection rate for malignant masses was achieved.

Manrique et al. (2006) utilized a genetic algorithm-based radial basis function neural network for classification of masses from a Madrid hospital dataset. They obtained 83% classification accuracy (with 83% specificity and 81% sensitivity). Although their accuracy was not high, their network converged quickly.

Halkiotis et al. (2007) used a multilayered perceptron (MLP)-type neural network on the MIAS (Mammography Image Analysis Society) database. They obtained a good classification rate of 94.7% with an average of 0.27 false positives per image for microcalcifications.

Georgiou et al. (2007) evaluated a mass shape feature with a wide range of linear and nonlinear classifiers, including linear discriminant analysis, least-squares minimum distance, k-nearest neighbor, radial basis function (RBF), MLP neural networks, and support vector machines (SVM). They found that support vector machine produced highest accuracy. They obtained 91.54% classification accuracy on masses.

Brem used the second look CAD system to determine the performance of CAD systems on different-sized lesions (Brem et al. 2004). He achieved an overall sensitivity of 89%. His investigation was to try and determine if lesion size would adversely affect the performance of a CAD system.

Abdalla et al. (2007) used textual features with a support vector machine classifier and they achieved a classification accuracy of 82.5% on mammograms from the Digital Database of Screening Mammography (DDSM) (Heath et al. (2001).

Panchal and Verma (2006) proposed an autoassociator network for feature extraction and combined it with an MLP-based classifier. The network was trained using a back propagation algorithm. They achieved 91% classification accuracy on DDSM.

Massotti (2006) used a support vector machine to classify suspicious areas or regions of interest found on mammograms into cancer and normal tissue. He obtained 90% classification accuracy on DDSM.

Acharya et al. (2008) obtained a sensitivity of 91.67% using an artificial neural network and 95% using a Gausian Mixture Model with 93.33% sensitivity and 96.67% specificity, respectively, on DDSM.

Verma and Zakos (2001) used backpropagation (BP)-type neural network and investigated the significance of microcalcification features by combining them. They presented a number of modified features and reported that the combination of modified features such as entropy, standard deviation, and number of pixels produced the best results. They obtained 88.9% classification accuracy on DDSM.

Verma (2008) proposed a modified MLP-type architecture by adding additional neurons and a new learning algorithm. The additional neurons for benign and malignant classes were used to improve memorization ability without destroying the generalization ability of the network. He used DDSM and obtained classification accuracies of 100% on training set and 94% on test set.

Kumar et al. (2006) applied decision trees for classification of masses in digital mammograms. In their research, they used CART and See5 software packages for conducting experiments with decision trees on DDSM. They obtained classification accuracies of 95% on training set and 91% on test set using CART and classification accuracies of 95% on training set and 89% on test set using See 5.

Mazurowski et al. (2008) investigated BP and Particle Swarm Optimization (PSO) techniques for finding the effect of class imbalance in training data when developing neural network classifiers for breast cancer diagnosis. They showed that

classifier performance deteriorates with even modest class imbalance in the training data. They concluded that BP is generally preferable over PSO for imbalanced training data especially with small data sample and large number of features.

Rangayyan et al. (2007) noted that several CAD techniques have achieved over 85% sensitivity for the identification of masses but also have a high false positive rate. In general, mass identification is a more difficult task than microcalcifications because masses are variable in size, shape and density can exhibit poor image contrast, and can be strongly intertwined with surrounding tissues making detection and classification difficult (Delogu et al. 2007).

As can be seen from the above review that there has been a vast amount of research in particular development of intelligent techniques for the classification of masses in digital mammograms; however, successful commercial systems are not available. The main problem in developing an acceptable CAD system is inconsistent and low classification accuracy. In order to improve the classification accuracy, this chapter presents a novel methodology that uses clustering to create multiple clusters within existing classes (benign and malignant) and incorporates multicluster-based new classes within the training process.

This chapter consists of four sections. Following introduction and literature review, a research methodology is described in Sect. 2. Section 3 presents the experimental results and a comparative analysis of the results with other existing techniques. Section 4 concludes the chapter.

5.2 Research Methodology

The research methodology consists of three major steps: (1) acquiring and processing of digital mammograms, (2) creation of multicluster classes with strong clusters, and (3) classification. The digital mammograms are processed, suspicious areas and features are extracted. A clustering algorithm is used to cluster the feature data into a number of clusters for both benign and malignant classes. Strong clusters are selected and a classifier with input features and strong clusters is used for final classification. The details of research methodology are described below.

5.2.1 Acquiring and Processing of Digital Mammograms

Digital mammograms were acquired from DDSM (Heath et al. 2001). It contains approximately 2,600 high-quality images together with case-related information. The research in this chapter used 200 mammograms comprising an equal number of masses with 100 mammograms being selected for training and 100 for testing. The suspicious areas were extracted using a chain code. The chain code is provided with DDSM for each suspicious area. Six features have been extracted from suspicious areas and they represent four BI-RADS descriptor features together with patient age

and a subtlety value (Heath et al. 2001). All six features are density, mass shape, mass margin, abnormality assessment rank, patient age, and subtlety value. Readers are referred to Verma (2008) for details on features and feature extraction process.

5.2.2 *Creation of Multicluster Classes with Strong Clusters*

The creation of multicluster classes with strong clusters can be done in two different ways. First approach is to cluster whole data into *n* clusters and then select *m* strong clusters as shown in Fig. 5.2. The second approach is to first divide the whole data into two classes (benign and malignant) and then cluster data into *n* clusters for each class. Finally, *m* strong clusters are selected for each class. The whole process for second approach is shown in Fig. 5.3. *K*-means clustering algorithm is used for clustering that is based on evaluating the distance between a point and the cluster centroid. Strong clusters are selected based on threshold that needs to be investigated as there is no magic way of finding the threshold.

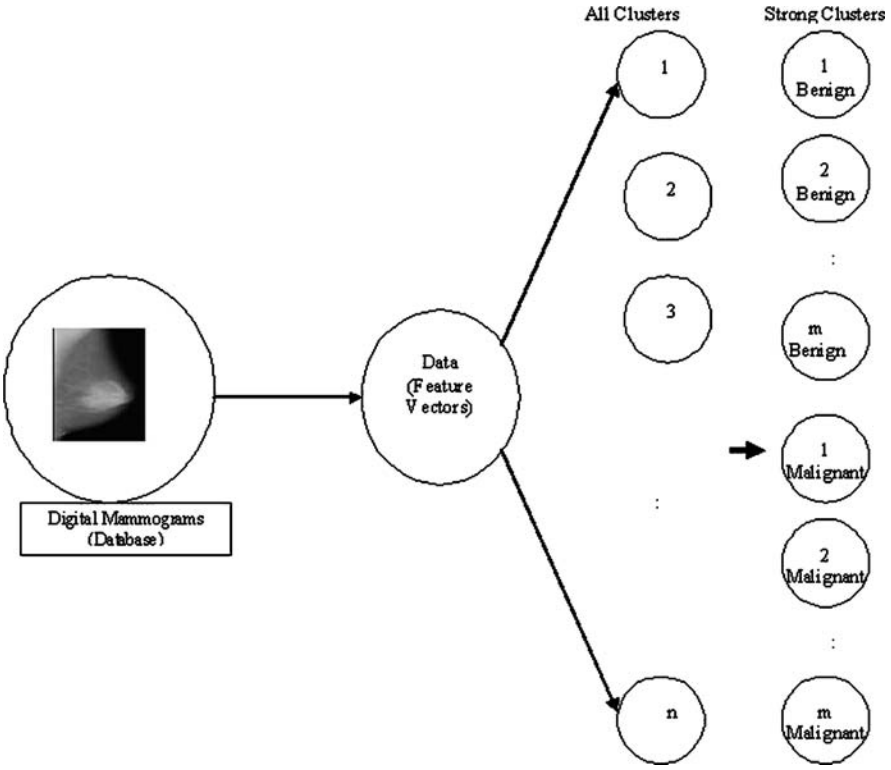

Fig. 5.2 Creation of multiple strong clusters from combined data

5.2.3 *Classification*

The process that takes features as input and outputs the class (benign/malignant) is called a classification process. There are different types of classifiers based on intelligent and statistical techniques. Some classifiers in particular neural network-based classifiers used in this research need to be trained before they are ready to classify. The well-trained classifiers can have good generalization abilities. The generalization means that the classifiers are able to classify correctly the input features which they have never seen before. The neural network-based classifier using multiple strong clusters (Fig. 5.3) used in this research is shown in Fig. 5.4. There are number of ways the inputs, and outputs can be used to train a classifier after the clustering of data (multiple clusters per class as shown in Fig. 5.3).

5.2.3.1 Original Inputs with Multiple Classes

In this process, the original input features for benign and malignant classes are divided into $2m$ (where m is a number of strong clusters) classes using strong clusters

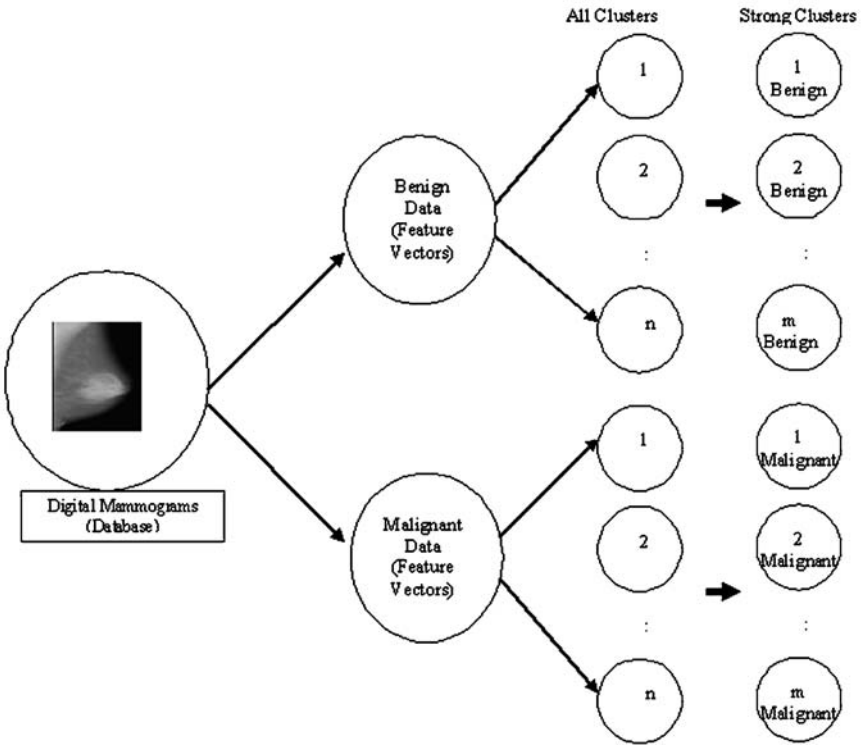

Fig. 5.3 Creation of multiple strong clusters from data for each class

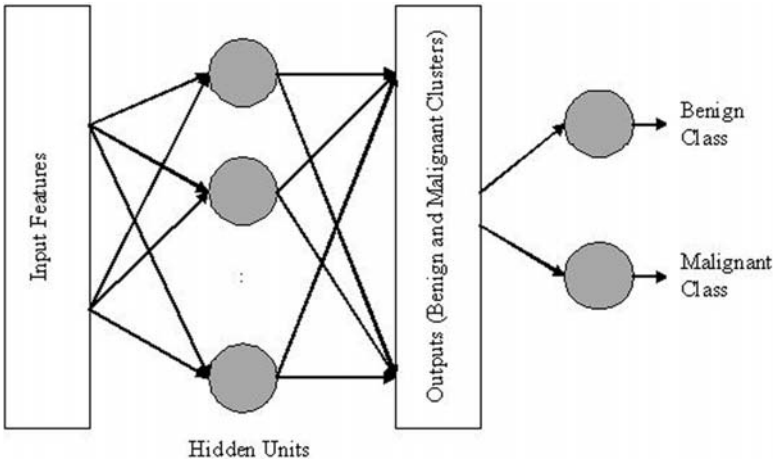

Fig. 5.4 Classification using multicluster classes with strong clusters

as shown in Fig. 5.3. Assume that we have six inputs and three strong clusters for each class. The inputs and outputs for classification process can be written as follows.

Original Input: x1, x2, x3, x4, x5, x6	New Target: y1, y2, y3, y4, y5, y6
0.45 0.8 0.4 0.22 0.9 0.8	0.9 0.1 0.1 0.1 0.1 0.1 (Benign)
0.84 0.6 0.34 0.55 0.4 0.2	0.1 0.1 0.1 0.1 0.1 0.9 (Malignant)

The neural network classifier as shown in Fig. 5.4 is used for classification. The neural network classifier that has been employed in the research presented in this chapter is a MPL. It utilizes six input nodes to represent each input feature. The number of hidden neurons is being determined experimentally to ascertain the optimal configuration. In the proposed approach, the neural network has a variable number of output neurons to represent the number of strong clusters.

5.2.3.2 Cluster Values with Multiple Classes

In this process, the values of strong clusters after feeding the original input features to clustering algorithm are used instead of original input features. In this case, original features are transformed into a new feature space. The new target outputs are created same way as in Sect. 2.3.1.

5.3 Experimental Results and Comparative Analysis

The multicluster-based classification approach described in Sect. 5.2 has been evaluated on a benchmark database. Digital mammograms from the DDSM benchmark

Table 5.1 Classification results

Technique	Clusters	Hidden units	Classification accuracy (%)	
			Training set size: 100 (50 benign and 50 malignant)	Test set size: 100 (50 benign and 50 malignant)
Standard multilayer perceptron (random-MGS training algorithm)	No clustering used	10	86	94
		16	92	93
		27	96	93
		30	96	93
Multicluster-based classification approach (random-MGS training algorithm)	Three clusters for benign and three clusters for malignant	10	82	94
		16	88	95
		24	94	96
		25	94	96
		27	93	96
		30	95	96

Table 5.2 Comparison of classification accuracy

Technique	Database	Best accuracy on test data (%)	Reference
Multicluster class	DDSM	96	Chapter
Genetic algorithm-based radial basis function	Local hospital database	83	Manrique et al. 2006
Multilayer perceptron	MIAS	94.70	Halkiotis et al. 2007
Support vector machine	Local data	91.54	Georgiou et al. 2007
Auto-associator MLP	DDSM	90.90	Panchal and Verma 2006
Support vector machine	DDSM	90	Massotti 2006
Neural network	DDSM	88.9	Verma and Zakos 2001
Neural network	DDSM	94	Verma 2008
Decision trees (CART)	DDSM	91	Kumar et al. 2006
Decision trees (See 5)	DDSM	89	Kumar et al. 2006

described in Sect. 2.1 were used for evaluation. The experiments by varying the number of hidden units have been conducted and the results are presented in Table 5.1.

The proposed multicluster class-based approach has been compared with standard MLP-based classification and other recently published techniques. A comparison between the proposed and published techniques is not an easy task as many factors can affect the classification accuracy of the system.

The classification accuracies obtained using the proposed multicluster-based approach and existing techniques for the diagnosis of breast cancer are presented in Table 5.2. As shown in Table 5.2, the results obtained by using multiple strong clusters are better than the single cluster-/single class-based approach and other recently published techniques.

5.4 Conclusions

This chapter has reviewed and presented a state of the art for the classification of suspicious areas in digital mammograms. It has presented a novel multicluster class-based classification approach. It has shown how multiple clusters can be formed and strong clusters can be incorporated during the training of a classifier. The proposed approach has achieved 96% classification accuracy on test data which is much higher than the standard classifier with a single cluster per class for benign and malignant. The research presented in this chapter shows that the multicluster class approach has a significant impact on improving overall classification accuracy. The results presented in this chapter were obtained with one and three clusters per class only. In our future research, we would like to investigate the cluster size and improved mechanism for selecting strong clusters.

References

Acharya, R., Ng, U., Chang, Y., Yang, J. and Kaw, G., "Computer based identification of breast cancer using digitized mammograms," Journal of Medical Systems, 2008, doi:10.1007/s10916–008–9156–6.

Abdalla, A., Deris, S. and Zaki, N., "Breast cancer detection based on statistical features and support vector machine," 4th International Conference on, Innovation in Information Technology, pp. 728–730, 2007.

American Cancer Society, Breast cancer facts and figures 2007–2008, http://www.cancer.org/ accessed on 22 December 2008.

American Cancer Society, Global cancer facts and figures 2007, http://www.cancer.org/ accessed on 22 December 2008.

Birdwell, R., Bandodkar, P. and Ikeda, D., "Computer-aided detection with screening mammography in a university hospital settings," Radiology, vol. 236, pp. 451–457, 2005.

Brem, R., "Clinical versus research approach to breast cancer detection with CAD: where are we now?" American Journal of Roentology, vol. 188, pp. 234–235, 2007.

Brem, R., Hoffmeister, J., Zisman, G., Simio, M. and Rogers, S., "A computer aided detection system for the evaluation of breast cancer by mammographic appearance and lesion size," American Journal of Roentology, vol. 184, pp. 893–896, 2004.

Cheng, H., Cai, X, Chen, X., Hu, L. and Lou, X., "Computer-aided detection and classification of microcalcifications in Mammograms: a survey," Pattern Recognition, vol. 36, pp. 2967–2991, 2003.

Cheng, H., Shi, X., Min, R., Ju, L., Cai, X. and Du, H., "Approaches for automated detection and classification of masses in mammograms," Pattern Recognition, vol. 39, no. 4, pp. 464–668, 2006.

Dean, J. and Ilvento, V., "Improved cancer detection using computer-aided detection with diagnostic and screening mammography: prospective study of 104 cancers," American Journal of Roentology, vol. 187, pp. 20–28, 2006.

Delogu, P., Fantacci, M., Kasae, P. and Retico, A., "Characterization of mammographic masses using a gradient based segmentation algorithm and a neural classifier," Computers in Biology and Medicine, vol. 37, pp. 1479–1491, 2007.

Freer, T. and Ulissey, M., "Screening mammography with computer-aided detection: prospective study of 12,860 patients in a community breast centre," Radiology, vol. 220, pp. 781–786, 2001.

Georgiou, H., Mavrofarakis, M., Dimitropoulos, N., Cavouras, D. and Theodoridis, S., "Multi-scaled morphological features for the characterization of mammographic masses using statistical classification schemes," Artificial Intelligence in Medicine, vol. 41, pp. 39–55, 2007.

Goergen, S., Evans, J., Cohen, G. and Macmillan, J., "Characteristics of breast carcinomas missed by screening radiologists," Radiology, vol. 204, no. 11, pp. 131–135, 1997.

Heath, M., Bowyer, K., Kopans, D., Moore, R. and Kegelmeyer, P., "The Digital Database for Screening Mammography," IWDM-2000, Medical Physics Publishing, 2001.

Halkiotis, S., Botsis, T. and Rangoussi, M., "Automatic detection of clustered microcalcifications in digital mammograms using mathematical morphology and neural networks," Signal Processing, vol. 87, pp. 1559–1568, 2007.

Kumar, K., Zhang, P. and Verma, B., "Application of decision trees for mass classification in mammography," International Conference on Fuzzy Systems and Knowledge Discovery, FSKD'06, pp. 366–376, China, 2006.

Manrique, D., Rios, J. and Rodriguez-Paton, A., "Evolutionary system for automatically constructing and adapting radial basis function networks," Neurocomputing, vol. 69, pp. 2268–2283, 2006.

Massotti, M., "A ranklet-based image representation for mass classification in digital mammograms," Medical Physics, vol. 33, no. 10, pp. 3951–3961, 2006.

Mazurowski, M., Habas, P., Zurada, J., Lo, J., Baker, J. and Tourassi, G., "Training neural network classifiers for medical decision making: the effects of imbalanced datasets on classification performance," Neural Networks, vol. 21, pp. 427–436, 2008.

Morton, M., Whaley, D., Brandt, K. and Amrami, K., "Screening mammograms: interpretation with computer-aided detection – prospective evaluation," Radiology, vol. 239, pp. 204–212, 2006.

National Breast Cancer Foundation, Fast facts about breast cancer in Australia, http://www.nbcf.org.au/, accessed on 22 December 2008.

Panchal, R. and Verma, B., "Neural classification of mass abnormalities with different types of features in digital mammography," International Journal of Computational Intelligence and Applications, Vol. 6, No. 1, pp. 61–76, 2006.

Ramirez, N., Acosta-Mesa, H., Carillo-Calvert, H., Nava-Fernandez, L. and Barrientos-Martinez, R., "Diagnosis of breast cancer using bayesian networks: a case study," Computers in Biology and Medicine, vol. 37, pp. 1553–1564, 2007.

Rangayyan, R., Ayres, F. and Desautels, L., "A review of computer-aided diagnosis of breast cancer: toward the detection of subtle signs," Journal of the Franklin Institute, Special Issue: Medical Applications of Signal Processing, Part I, vol. 344, pp. 312–348, 2007.

Roder, D., Houssami, N., Farshid, G., Gill, G., Luke, X., Downey, P., Beckmann, K., Iosifidis, P., Grieve, L. and Williamson, L., "Population screening and intensity of screening are associated with reduced breast cancer mortality: evidence of efficacy of mammography screening in Australia," Breast Cancer Research and Treatment, vol. 108, no. 3, pp. 409–416, 2008.

Tourassi, G., Haarawood, B., Singh, S., Lo, J. and Floyd, C., "Evaluation of information-theoretic similarity measures for content based retrieval and detection of masses in mammograms," Medical Physics, vol. 34, pp. 140–150, 2007.

Verma, B. and Panchal, R. "Neural networks for the classification of benign and malignant patterns in digital mammograms," Advances in Applied Artificial Intelligence, Idea Group, Inc., USA, Book Editor: John Fulcher, 2006.

Verma, B. and Zakos, J., "A computer-aided diagnosis system for digital mammograms based on fuzzy-neural and feature extraction techniques," IEEE Transactions on Information Technology in Biomedicine, vol. 5, pp. 46–54, 2001.

Verma, B., "Novel network architecture and learning algorithm for the classification of mass abnormalities in digitized mammograms," Artificial Intelligence in Medicine, vol. 42, no. 1, pp. 67–79, 2008.

Chapter 6
Analysis of Cancer Data Using Evolutionary Computation

Cuong C. To and Tuan Pham

Abstract We present several methods based on evolutionary computation for classification of oncology data. The results in comparisons with other existing techniques show that our evolutionary computation-based methods are superior in most cases. Evolutionary computation is effective in this study because it can offer efficiency in searching in high-dimension space, particularly in nonlinear optimization and hard optimization problems. The first part of this chapter is the review of some previous work on cancer classification. The second part is an overview of evolutionary computation. The third part focuses on methods based on evolutionary computation and their applications on oncology data. Finally, this chapter concludes with some remarks and suggestions for further investigation.

6.1 Introduction

Cancer is a class of diseases in which a group of cells display uncontrolled growth, invasion, and sometimes metastasis. These three malignant properties of cancers differentiate them from benign tumors, which are self-limited, do not invade or metastasize. Modern cancer data are derived from many different sources such as microarray, mass spectrometry, and digital imaging. In this chapter, we focus on mass spectrometry data [ovarian cancer, Petricoin et al. (2002)] and image data [Wisconsin diagnostic breast cancer, Street et al. (1993)]. Revolutionary proteomic technology, which has recently been developed, uses the pattern of proteins observed within a clinical sample as a diagnostic fingerprint to create mass spectrometry cancer data. These so-called proteomic patterns are of time-series data. In its current state, surface-enhanced laser desorption/ionization time-of-flight mass spectrometry (SELDI-TOF MS) is the technology used to acquire the proteomic patterns to be used in the diagnostic setting. The principle of SELDI-TOF is very

C.C. To (✉)
ADFA School of Information Technology and Electrical Engineering, The University of New South Wales, Canberra, ACT 2600, Australia
e-mail: C.To@adfa.edu.au

T. Pham (ed.), *Computational Biology: Issues and Applications in Oncology*,
Applied Bioinformatics and Biostatistics in Cancer Research,
DOI 10.1007/978-1-4419-0811-7_6,

simple; proteins of interest are captured, by adsorption, partition, electrostatic interaction, or affinity chromatography on a stationary-phase and immobilized in an array format on a chip surface. One of the benefits of this process is that raw biofluids, such as urine, serum, and plasma, can be directly applied to the array surface. After a series of binding and washing steps, a matrix is applied to the array surfaces. The species bound to these surfaces can be ionized by matrix-assisted laser desorption/ionization (MALDI) and their mass-to-charge (m/z) ratios measured by TOF-MS. The result is simply a mass spectrum of the species that bound to and subsequently desorbed from the array surface. While the inherent simplicity of the technology has contributed to the enthusiasm generated for this approach, applications of sophisticated bioinformatics methodology have enabled the use of SELDI-TOF MS as a potentially revolutionary diagnostic tool. In Wisconsin diagnostic breast cancer, a small region of each breast Fine Needle Aspirates (FNAs) was digitized, resulting in a 640×400, 8 bits per pixel gray scale image. The image analysis program used a curve fitting program to determine the boundaries of nuclei from initial dots placed near these boundaries by a mouse. Ten features were computed for each nucleus: area, radius, perimeter, symmetry, number and size of concavities, fractal dimension (of the boundary), compactness, smoothness (local variation of radial segments), and texture (variance of gray levels inside the boundary). The mean value, extreme value, and standard error of each of these cellular features are computed, resulting in a total of 30 real valued features for each image. A set of 569 images was processed in the manner described above, yielding a database of 569 thirty-dimensional points.

Proteomics is considered as a mass-screening approach to molecular biology, which aims to document the overall distribution of proteins in cells, identify and characterize individual proteins of interest, and ultimately elucidate their relationships and functional roles. Such direct, protein-level analysis has become necessary because the study of genes, by genomics, cannot adequately predict the structure or dynamics of protein synthesis, since it is at that protein level where most regulatory processes take place, where disease processes primarily occur, and where most drug targets are to be found. Rapidly emerging field of proteomics has now established itself as a credible approach for furthering our understanding of the biology of whole organisms – from simple unicellular organisms to those as complex as human. The readily available experimental tools for measurement of protein expression by two-dimensional gel electrophoresis, and for protein identification and characterization by mass spectrometry-based methods, have already made a significant impact on proteomics.

Functional genomics studies functionality of specific genes, their relations to diseases, their associated proteins, and their participation in biological processes. Genes are fundamental to the life; for each gene, the expression level is different, performing different function. So monitoring genes' expression levels is very important. However, traditional methods in molecular biology generally work on a "one gene in one experiment" basis, which means that the throughput is very limited and the "whole picture" of gene function is hard to obtain. DNA microarray is a novel technology that allows for monitoring of gene expression for thousands of genes in a single experiment and is already producing huge amounts of valuable data.

The principle of both methods is to measure amount of proteins (genes) in cell at definite time point when sample is experimented. If samples are experimented at different time points during a particular process, then we have time series vectors for all detectable proteins (genes) in cell. In other words, each pattern (protein or gene) of proteomics or transcriptomics database is presented as an n-dimensional vector.

The advances of proteomics can be used to diagnose diseases such as cancer. Proteomic pattern analysis that can analyze hundreds of clinical samples a day has the potential being a novel, highly sensitive diagnostic tool for early detection of cancer. The ability to classify patterns generated from healthy persons and those patients affected with cancer is usually accomplished through applications of machine learning and pattern recognition methods.

Street et al. (1993) extracted ten different features from the snake-generated cell nuclei boundaries. All of the features are numerically modeled such that larger values will typically indicate a higher likelihood of malignancy. Mangasarian et al. (1995) used linear programming to discriminate benign from malignant breast lumps. Both methods were applied to breast tumor. Petricoin et al. (2002) developed an analytical tool that combines genetic algorithms and cluster analysis methods. The input data for analysis are proteomic spectra and the output is the best fit subset of amplitudes at defined m/z values that best segregates the preliminary data. Pham (2008) used a combination of linear predictive coding and vector quantization to predict the class of an unknown mass spectrometry data. The proposed method was applied to ovarian cancer dataset.

Cluster analysis is an unsupervised approach. In order to use these methods, we must define a way to measure the similarity between patterns we are comparing (Euclidean distance, correlation coefficient, Manhattan distance, etc.). Patterns are then grouped by using a clustering algorithm such as hierarchical, k-means, self-organization, and hierarchical clustering with partial least squares (LSs). Tumor classification using clustering methods have reported by Golub et al. (1999), Alon et al. (1999), Perou et al. (1999), Nguyen and Rocke (2002), and Bittner et al. (2000).

Linear discrimination analysis is based on linear combinations of the pattern with large ratios of between-group to within-group sums of squares. The k nearest-neighbor is based on a distance function for pairs of patterns, such as the Euclidean distance. For each pattern of test set, k closest patterns of learning set are found, and the class is predicted by the majority vote; that is, a class that is most common among those k neighbors is chosen. Binary tree classifiers are constructed by repeated splits of the subsets of the space of patterns X into two descendant subsets, starting with X itself. Each terminal subset is assigned a class label, and the resulting partition of X corresponds to the structure of the classification tree. Dudoit et al. (2002) presented a comparison of above-mentioned three methods for the classification of tumors using gene expression data. Zhang et al. (2001) introduced a method based on classification trees for tumor classification with gene expression data.

Boosting is one of the most powerful learning ideas introduced in last decade. The motivation for boosting was a procedure that combines the output of many

"weak" classifiers to produce a powerful "committee." In classifying tumor samples with gene expression data, the feature selection was done first using nonparametric scoring method, and then the LogitBoost (LB) was used to classify the samples (Dettling and Buhlmann 2003).

Artificial neural networks (ANN) are machine-learning methods that imitate biological neural networks for learning multiple examples. In order to solve a problem using ANN, we need to determine first the topology of ANN including the number of input neurons, number of output neurons, number of hidden layers, the transfer function for each hidden layer, and the transfer function for output neurons. ANN is then trained to learn features of the training data. Toure and Basu (2001) used backpropagation network to solve binary classification problems. A combination of principal component analysis (PCA) and neural networks was introduced to solve pattern classification problem (Khan et al. 2001). PCA was used to reduce the dimensions of the patterns, and ten dominant PCA components were used for subsequent analysis. Each linear ANN model was then calibrated using 10 PCA components as input variables and four cancer categories as output. ANNs were trained and tested using the small, round blue-cell tumors. In another work by Su et al. (2002), a modular-gating network had three individual expert networks which were trained separately by using original and frequency domain data. Each expert network consisted of three layers. Ten neurons were used in both input and hidden layers. The output layer had only one neuron. The gating network used a majority voting scheme to generate the final output.

Kernel-based methods such as support vector machines (SVM) find a mapping to project objects onto a new space (feature space) so that the problem can be solved more effectively. The most important thing when using kernel-based methods is the determination of a kernel. The kernel is a way of representing data. The kernel is a real-valued "comparison function"; the data set is represented by square matrix of pair wise comparisons. SVM has been widely used in computational biology, including pattern classification using cancer data. Binary SVM with a linear kernel was applied to ovarian and colon tumor data (Furey et al. 2000). Before applying SVMs, feature selection was used. SVMs based on recursive feature elimination were applied to cancer classification using Leukemia data (Guyon et al. 2002). A general multicategory SVM method was introduced and applied to Leukemia data for solving a three-class problem (Lee and Lee 2003).

The following work presented the applications of evolutionary algorithms to oncology data. Genetic programming was used to classify tumours based on H Nuclear Magnetic Resonance spectra of human brain tumour biopsies (Gray et al. 1996). For pattern recognition analysis, spectra were digitised at intervals of 0.010 ppm over the range 4.5 to 0.5 ppm giving 400 variables. Principal component (PC) analysis showed that the first 20 PC's accounted for 99% of the variance. The PC vectors were simplified by Varimax rotation. The GA/SVM algorithm (Liu et al. 2005) consists of three main components: a GA-based gene selector, SVM-based binary classifiers distinguishing between tumor samples and multiclass categorization by an AP/SVM voting strategy.

Having pointed out that and most modern cancer data are complex and high-dimensional, application of evolutionary computation for cancer classification

appears to be appropriate but has been rarely explored. We attempt to introduce evolutionary computation algorithms for studying cancer classification. We selected ovarian cancer and breast cancer data for our study because the ovarian cancer data were generated from a novel proteomic technique in biotechnology, whereas the breast cancer data have been widely investigated and can be considered as good benchmark data.

6.2 Overview of Evolutionary Computation

The basic idea of evolutionary computation is based on the theory of evolution. In other words, evolutionary computation is a computation system that mimics the adaptation and evolution of each individual in its environment. By doing this, it is expected that the next generation of individuals is always better than the previous one. There are different techniques of evolutionary computation: genetic algorithms, genetic programming, evolutionary programming, and evolutionary strategy. In this chapter, only genetic algorithms and genetic programming are discussed.

6.2.1 Genetic Programming

Genetic programming (Koza 1992) handles a population that contains many objects of which one will become the solution of the problem and called a tree or a program. A tree can return no value or more values depending on what we want it to represent. The domain of values returned by the tree is divers; it can be, for instance, a numerical value, a Boolean value, or a symbolic function, etc.

There are two types of nodes in a tree. One is called internal node (or function node) and the others are called leaf nodes (or terminal nodes). Figure 6.1 is an example of tree.

The terminal node has no child node (radiate line), and it usually represents constant value or a variable of the problem. Function node has children nodes (radiate lines), and the number of children nodes of a function node depends on the number of arguments of function that it represents. Any function can be represented by

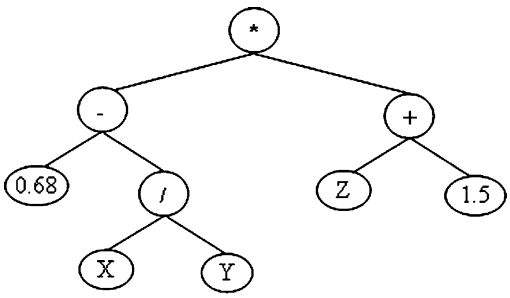

Fig. 6.1 In this tree, function nodes are $\{+, -, *, /\}$ and terminal nodes are $\{X, Y, Z, 0.68, 1.5\}$. This tree is equivalent to the function $f(X, Y, Z) = [0.68 - (X/Y)] \times (Z + 1.5)$

the function node, such as arithmetic operations, mathematical functions, and conditional operation. In order to know how well each tree in the population solves the problem, each tree is measured. This measurement is called a fitness function. There are no common fitness functions for all problems and the fitness function is a crucial condition for getting a good solution. Normally, a fitness function is given in the following equation:

$$\text{Fitness} = \sum_{i=1}^{N} (f_i - v_i)^2 \tag{6.1}$$

where N is the number of fitness cases, v_i is the target value of one fitness case, and f_i is the value returned by a tree.

The scheme of genetic programming (GP) is given by the following steps:

Step 1: A population is randomly created.
Step 2: Compute fitness value of each tree in the population.
Step 3: On the basis of fitness values, generating a new generation using reproduction and crossover operator.
Step 4: Steps 2 and 3 are repeated until specified criteria are satisfied. The criteria can be the maximum number of generations, designated conditions, etc.
Step 5: The solution of problem is the best-so-far tree (the best tree appears in any generation).

6.2.1.1 Operations for Modifying the Tree

Reproduction

The reproduction operation is performed by copying good trees from the current generation to the next generation. There are several different selection methods based on fitness such as roulette-wheel, tournament selection, and greedy overselection to choose good trees for the next generation. In roulette-wheel mechanism, each tree is assigned a roulette-wheel slot whose size is proportional to the ratio of the fitness of this tree divided by the sum of fitness of all trees in population. In tournament selection, k trees are randomly selected from population and then the selected result tree is the one with the highest fitness among k selected trees. In greedy overselection, some best trees are selected.

Crossover

The crossover operates on two parental trees and produces two offspring trees. This operation is done as follows:

- Select two parental trees based on their fitness.
- Randomly select two nodes on two selected parental trees.

- The first offspring is created by copying the first parental tree but eliminating the subtree below the selected node of first parental tree, and then inserting the subtree below the selected node of second parental tree. The second offspring is created in the same way.

6.2.1.2 Control Parameters

In order to use genetic programming for solving a problem, we need to determine the following parameters:

- Population size: number of trees in a population.
- Maximum number of generation: number of population created during evolution process.
- Probability of crossover: number of trees in population taking part in crossover operation.
- Probability of reproduction: number of trees in population taking part in reproduction operation.

6.2.2 Genetic Algorithms

The difference between genetic algorithms and genetic programming is the way they represent candidate solution. While genetic programming uses trees; strings (also known as chromosomes) are used by genetic algorithms. Each chromosome of the population is a candidate solution to the problem. So a chromosome should in some way containing information about the solution that it represents. Encoding of chromosome depends on the problem heavily. There are various types of encoding: binary encoding where each chromosome is a string of bits 0 and 1 (Fig. 6.2), value encoding where each chromosome is a sequence of some values being anything connected to the problem, such as numbers, chars, or objects (Fig. 6.3).

Chromosome 1:	1	1	0	1	0	0	1
Chromosome 2:	0	0	1	1	1	0	1

Fig. 6.2 Two chromosomes are encoded as binary strings

Chromosome 1:	0.3266	-1.5692	563.36	-236.36
Chromosome 2:	A	C	T	G
Chromosome 3:	Red	Green	Blue	White

Fig. 6.3 Three chromosomes are encoded as value strings

6.2.3 *Parallel Evolutionary Computation*

One of the disadvantages of evolutionary computation methods is that it takes a large amount of computing time. Majority of computing time is used for evaluating fitness function. Conversely, genetic operators are not time consuming. So if we wish to increase computational speed, parallel computing is necessary. This procedure is feasible because multicore processors have become more or more popular and built in most computers. There are two basic approaches to parallelization as follows.

6.2.3.1 Parallelism at Fitness Level

The aim of these models is to increase the performances. In these models, there is only one population in each run. There are three levels at which parallelism of fitness evaluation is possible. In parallelism at the fitness-case level, a set of all fitness cases of the problem is partitioned into subsets of fitness cases, and each subset is assigned to one central processing processor (CPU) (Fig. 6.4). In parallelism at the individual level, when computing the fitness value of individuals, a population is divided into subpopulations and each subpopulation is set to one processor (CPU). The scheme is shown in Fig. 6.5. In parallelism at the independent-runs level, each processor is assigned one or more full runs for the maximum number of generations to be run. The final result is the best of all the runs from all processors. Figure 6.6 is a flowchart of this model.

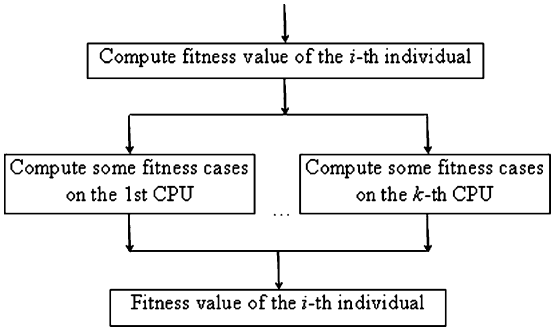

Fig. 6.4 Parallel computing at fitness cases level

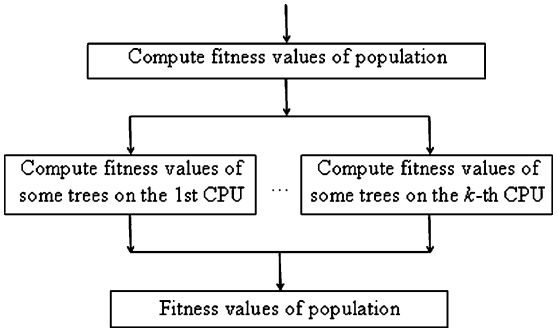

Fig. 6.5 Parallel computing at individuals level (M is number of individuals)

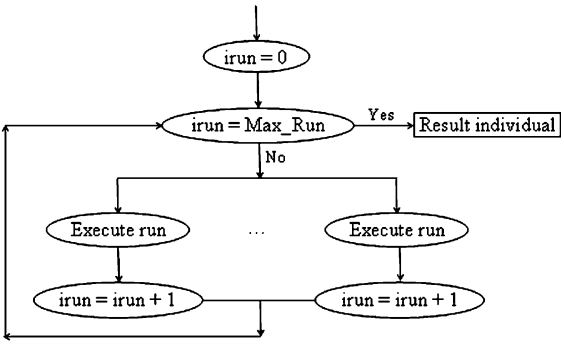

Fig. 6.6 Parallel computing at independent runs level

6.2.3.2 Parallelism at Population Level (Island Model or Cellular Model)

Island model is rather complicated and is the most popular type (Erick 2001). In this model, population is partitioned into subpopulations. Each subpopulation called island is assigned to one processor and runs independently. After a predefined number of generations, islands exchange individuals with each other called migration. This model has been applied to many problems (Alba et al. 2005; Calegari et al. 1997; Fernandez de Vega 2005) and shown that it not only increases the performance of algorithm but also gives better results than the sequential algorithm. This chapter introduces how to use the island model for improving the results.

6.2.3.3 Parameters of Island Model

Topology

Andre and Koza (1996) introduced grid topology which connected each subpopulation with four neighbors in the North, East, West, and South directions (Fig. 6.8). Punch (1998) used a typical island model with a ring topology (Fig. 6.7). Fernandez de Vega (2005) introduced a random topology (Fig. 6.9) and compared it with the grid and ring topology; then the author concluded that if all other parameters are fixed then there are no significant differences when changing the topology.

Migration Rate

Migration rate is defined as how many individuals should migrate at each migration step. The following are results of some previous work:

- Juille and Pollack (1995): One migrating tree per subpopulation.
- Andre and Koza (1996): 08% migrating trees per subpopulation.
- Punch (1998): Two migrating trees per subpopulation.

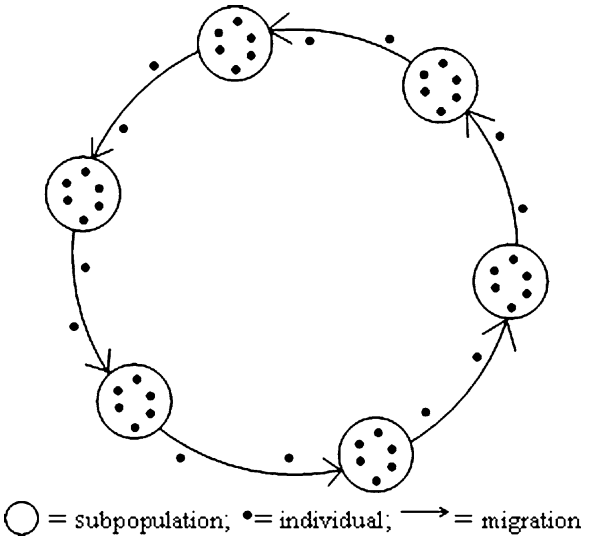

Fig. 6.7 Island model □ = subpopulations; • = individual; → = migration

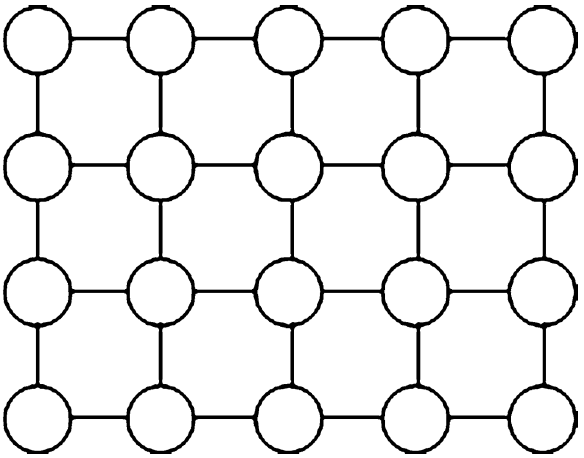

Fig. 6.8 Grid topology

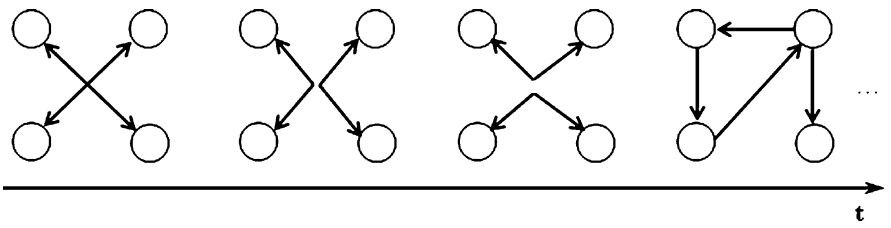

Fig. 6.9 Random topology

- Fernandez de Vega (2005): Best migration rate is between 5% and 10% in four test problems (two classic and two real-life problems: even parity 5, ant problem, routing and placing circuits on FPGAs, and medical diagnosing).

Migration Frequency

After a fixed number of generations, *mf*, islands exchange their individuals. Number of generations, *mf*, is called migration frequency. According to Juille and Pollack (1995) and Andre and Koza (1996), islands exchange individuals at every generation. Migration is executed after ten generations in results of Punch (1998). Fernandez de Vega (2005) had a wider study on comparing different frequencies, and gave conclusion that the best convergence results appear when about 10% of individuals from each subpopulation are sent every five to ten generations.

Subpopulation Size

This size concerns with suitable numbers of subpopulations and individuals per subpopulation. Andre and Koza (1996) used 64 islands each which had 500 individuals. Punch (1998) used 5 subpopulations and 200 individuals per subpopulation; and also 7 subpopulations, 700 individuals for each subpopulation. Fernandez de Vega (2005) had a set of trials and concluded that there are a number of trees with which the best results are obtained (regardless of the number of subpopulations). According to Calegari et al. (1997), the solutions which obtained with 4 islands of 40 individuals, each was better than that found with a single population of 160 individuals. The solution obtained with 40 islands of 4 individuals each was even better.

6.2.3.4 Application Program Interface Tools

We know that each island runs on one processor and after a fixed number of generations, islands exchange individuals. In other words, island model is a parallel computing model. Therefore, we need an application program interface (API) tool which can distribute islands on available processors and communicate between them. The following API tools are popular for parallel computing:

- Parallel virtual machine (PVM) is a portable message-passing programming system, designed to link separate host machines to form a "virtual machine" which is a single, manageable computing resource. PVM supports C, C++, and Fortran languages and runs on Unix platform.
- Message-Passing Interface (MPI) is a library of functions and macros that can be used in C, C++, and FORTRAN programs. MPI is intended for use in programs that exploit the existence of multiple processors by message passing. MPI supports Windows and Linux.

- OpenMP may be used to explicitly direct multithreaded, shared memory parallel programming in C/C++ and Fortran on all architectures, including Unix and Windows NT platforms.
- The p4 system is a library of macros and subroutines developed at Argonne National Laboratory for programming a variety of parallel machines. The p4 system supports both the shared-memory model (based on monitors) and the distributed-memory model (using message-passing). For the shared-memory model of parallel computation, p4 provides a set of useful monitors as well as a set of primitives, from which monitors can be constructed. For the distributed-memory model, p4 provides typed send and receive operations and creation of processes according to a text file describing group and process structure.
- Linda is a concurrent programming model that has evolved from a Yale University research project. The primary concept in Linda is that of a "tuple-space," an abstraction via which cooperating processes communicate. This central theme of Linda has been proposed as an alternative paradigm to the two traditional methods of parallel processing: that based on shared memory and that based on message passing. The tuple-space concept is essentially an abstraction of distributed shared memory, with one important difference (tuple-spaces are associative), and several minor distinctions (destructive and nondestructive reads and different coherency semantics are possible).
- There are some languages for parallel programming such as Orca, Ada, Cilk, NESL, and mpC.
- The very low level of parallel programming is that we use socket programming. We need to build up everything if we select this alternative.

6.3 Analysis of Cancer Data

6.3.1 Genetic Programming for Binary Classification

In supervised pattern classification, algorithms use training data that contain labeled (preclassified) patterns to train a classifier. The classifier is then used to predict the class of unknown patterns. It is important to notice that pattern classification is an ill-defined, nondeterministic task, in the sense that, using only training data, one cannot be sure that a discovered classification rule will have a high predictive accuracy on test set (database), which contains patterns unseen during training (Freitas 2002). There are normally three types of classification: single-class classification, binary classification, and multiclass classification. Multiclass classification is usually converted into multiple binary classifications. Single-class classification is a novel trend of pattern classification.

In binary classification, algorithms are given a training set. The training set consists of two subsets, namely positive $(+1)$ and negative (-1). The positive set contains similar patterns that algorithms will search in a database. The negative

set contains any other patterns of the rest of the database (not similar to patterns of positive set). Each pattern in the training set is an n dimension vector.

Let a training set be a set of pairs: $\mathrm{TS} = \{(\mathbf{x}_i, y_i) \subset \mathrm{R}^n \times \{-1, +1\}, i = 1..m\}$, where y_i is label (class) of a pattern (point), $\mathbf{x}_i$; $i = 1, \ldots, m$ are the number of patterns in a training set. Algorithms find classifiers are given as

$$\begin{aligned} &f : \mathrm{R}^n \to \mathrm{R} \\ &f(\mathbf{x}) \mapsto \{-1, +1\} \end{aligned} \tag{6.2}$$

which are supposed to discover relationships among patterns of the training set. It is probable that there are many kinds of relationship such as decision trees, mathematical functions, etc. In this part, we present an algorithm that uses genetic programming to create a decision tree for binary classification.

6.3.1.1 Method

Let $\mathbf{x} = [x_1, x_2, \ldots, x_n]^{\mathrm{T}}$ be a pattern in the training set. The aim of this algorithm is to create a decision tree that consists of terminal nodes and internal nodes. Terminal nodes are class labels (-1 or $+1$). Internal nodes are either $x_k \geq v_k$ or $x_k \leq v_k$ with $1 \leq k \leq n$; where x_k is one of variables of pattern $\mathbf{x}$, v_k is a random number within the range of x_k of pattern $\mathbf{x}$ in the training set. The fitness function is given as

$$\text{Fitness} = \frac{\text{number of correct classified}}{\text{number of patterns in the training set}} \tag{6.3}$$

The control parameters of the genetic programming are listed in Table 6.1.

The performance of the algorithm can be evaluated using two indicators, namely sensitivity (Se) and specificity (Sp) given by the following equations:

$$\mathrm{Se} = \frac{\mathrm{TP}}{|\mathrm{C}|} \tag{6.4}$$

$$\mathrm{Sp} = \frac{|\mathrm{R}| - \mathrm{FP}}{|\mathrm{R}|} \tag{6.5}$$

where TP is true positive, FP is false positive, $|\mathrm{C}|$ is the total number of patterns in C, and $|\mathrm{R}|$ is the total number of other patterns in R. Sensitivity (Se) indicates the number of patterns being correctly classified, where the value 1 means all similar

Table 6.1 Control parameters of genetic programming

Number of generations	500
Population size	1000
Probability of crossover	0.9
Probability of reproduction	0.1

patterns are found. Specificity (Sp) is number of misclassified patterns, where the value 1 means no misclassified patterns found. The values of both Se and Sp should approach 1 for high performance.

6.3.1.2 Experiments

Ovarian Cancer Data

Ovarian cancer data were produced using the WCX2 protein chip (Petricoin et al. 2002). The authors employed an upgraded PBSII SELDI-TOF mass spectrometer to generate the spectra. Different sets of ovarian serum samples were used compared to previous studies. The sample set included 91 controls and 162 ovarian cancers, which were not randomized so that the authors could evaluate the effect of robotic automation on the spectral variance within each phenotypic group. This database has 253 patterns each of which belongs to ovarian cancer class or control class. Each pattern is a time series whose length is 15,154. The rate of the training set and test set is 50–50%. The results obtained from GP and SVM are listed in Table 6.2, which shows the better performance of the GP. Se and Sp values are obtained using Eqs. (6.4) and (6.5).

Wisconsin Diagnostic Breast Cancer

This database was first used by Street et al. (1993). There are 569 instances each of which belongs to benign class or malignant class (357 benign, 212 malignant). Each instance is described by 30 real-valued attributes. Attributes are computed from a digitized image of a FNA of a breast mass. They describe characteristics of the cell nuclei present in the image. The diagnosis of breast tumors has traditionally been performed by a full biopsy, an invasive surgical procedure. FNAs provide a way to examine a small amount of tissue from the tumor; however, diagnosis with this procedure has met with mixed success. By carefully examining both the characteristics of individual cells and important contextual features such as the size of cell clumps, physicians at some specialized institution have been able to diagnose successfully using FNAs. However, many different features are though to be correlated with malignancy, and the process remains highly subjective, depending on the skill and experience of the physician. In order to increase the speed, correctness, and objective of the diagnosis process, we have used data mining methods. The rate of training set and test set is 50–50%. To compare with other methods, we used some well-known classifiers such as SVM, LB, logistic regression (LR), linear

Table 6.2 Ovarian cancer results

Methods	Se	Sp
SVM	0.870	0.848
GP	0.935	0.978

Table 6.3 Wisconsin diagnostic breast cancer results

Methods	Se	Sp
SVM	0.918	0.811
LS	0.959	0.893
LDA	0.959	0.893
LR	0.936	0.926
LB	0.940	0.910
GP	0.966	0.975

discriminant analysis (LDA), linear regression, and LS. The results obtained from various classification methods are given in Table 6.3.

6.3.2 Genetic Algorithms for Binary Classification

6.3.2.1 Concepts from Geometry

Hyperplane

Let $u_1, u_2, \ldots, u_n, v \in \mathrm{R}$, where at least one of the u_i is nonzero. The set of all points $\mathbf{x} = (x_1, x_2, \ldots, x_n)^{\mathrm{T}}$ that satisfy the linear equation

$$\sum_{i=1}^{n} u_i x_i = v \tag{6.6}$$

is called a hyperplane of the space R^n, which is defined by

$$\{\mathbf{x} \in \mathrm{R}^n : \mathbf{u}^{\mathrm{T}}\mathbf{x} = v\} \tag{6.7}$$

The hyperplane $H = \{\mathbf{x} \in \mathrm{R}^n : \mathbf{u}^{\mathrm{T}}\mathbf{x} = v\}$ divides R^n into two half-spaces. One of these half-spaces consists of the points satisfying the inequality $\sum_{i=1}^{n} u_i x_i \geq v$, denoted as

$$H_+ = \{\mathbf{x} \in \mathrm{R}^n : \mathbf{u}^{\mathrm{T}}\mathbf{x} \geq v\} \tag{6.8}$$

The other half-space consists of the points satisfying the inequality $\sum_{i=1}^{n} u_i x_i \leq v$, denoted

$$H_- = \{\mathbf{x} \in \mathrm{R}^n : \mathbf{u}^{\mathrm{T}}\mathbf{x} \leq v\} \tag{6.9}$$

The half-space H_+ is called the positive half-space, and the half-space H_- is called the negative half-space.

Distance from Point to Hyperplane

Given a point $\mathbf{a} = (a_1, a_2, \ldots, a_n)^{\mathrm{T}}$ and a hyperplane $H = \{\mathbf{x} \in \mathrm{R}^n : \mathbf{u}^{\mathrm{T}}\mathbf{x} = v\}$, distance from point to hyperplane in geometry is defined as

$$d(\mathbf{a}, H) = \frac{|\mathbf{u}^{\mathrm{T}}\mathbf{a} - v|}{\sqrt{\sum_{i=1}^{n} u_i^2}} \tag{6.10}$$

6.3.2.2 Nonlinear Programming Problem

Let a training set be a set of pairs, denoted $\mathrm{TS} = \{(\mathbf{x}_i, y_i) \subset \mathrm{R}^n \times \{-1, +1\}, i = 1..m\}$. The main idea of binary genetic algorithm is to find a hyperplane H that maximizes total distance from all points in training set to hyperplane H. That means we have a nonlinear programming problem as:

Find $\mathbf{z} = (u_1, u_2, \ldots, u_n, v)^{\mathrm{T}}$ which maximizes

$$f(\mathbf{z}) = \sum_{k=1}^{m} d(\mathbf{x}_k, H) = \sum_{k=1}^{m} \frac{|\mathbf{u}^{\mathrm{T}}\mathbf{x}_k - v|}{\sqrt{\sum_{i=1}^{n} u_i^2}} \tag{6.11}$$

Subject to:

$$\begin{cases} -1 \le u_i \le 1, i = 1..n, \text{ and } \exists u_i \neq 0 \\ -1 \le v \le 1 \\ (\mathbf{u}^{\mathrm{T}}\mathbf{x}_k - v) y_k \ge 0, \ k = 1..m \end{cases} \tag{6.12}$$

6.3.2.3 Prediction

After we have the resultant hyperplane H, if tested point $\mathbf{a}$ satisfying the inequality $\mathbf{u}^T\mathbf{a} - v \ge 0$, then tested point $\mathbf{a}$ belongs to positive set; otherwise tested point $\mathbf{a}$ belongs to negative set.

6.3.2.4 Solving Nonlinear Programming Problem by Genetic Algorithms

Genetic algorithm approach is one of the evolutionary computation methods and has been used in a wide variety of optimization tasks, including numerical optimization and such combinatorial optimization problems as circuit layout and job-shop scheduling (Mitchell 2001; Chong and Zak 2001). And many previous researches show that GA is a powerful method for optimization problems. Therefore, we used genetic algorithm to solve Eq. (6.11).

Chromosome

Each chromosome represents a hyperplane. So each chromosome is encoded as a fixed-length string of $(n + 1)$ real numbers (value encoding). The first n real values represent $u_i (i = 1..n)$; the last real value represents v.

Fitness Function

Because each chromosome of initial population is randomly created as a set of $(n + 1)$ real number within the range $[-1, 1]$, and mutation operation is not used, so the first and second inequalities of (6.12) are satisfied during search process.

The third inequality of Eq. (6.12) and Eq. (6.11) are used to calculated fitness value of each chromosome. First, each chromosome must satisfy the third inequality of Eq. (6.12). Second, the total distance from all points in training set to hyperplane (chromosome) is computed.

The best chromosome (hyperplane) is the one that satisfies the third inequality of Eq. (6.12) and has the maximum total distance.

6.3.2.5 Experiments

We applied genetic algorithms for binary classification using the Wisconsin diagnostic breast cancer data. All control parameters of genetic algorithm are listed in Table 6.1. The Se and Sp values are obtained using Eqs. (6.4) and (6.5). Likewise, we used the SVM, LB, LR, LDA, linear regression, and LS for comparison the results with the GA approach. All results are listed in Table 6.4, which again show the best performance of the GA.

Proteomic Database

Although this dataset is not cancer data but we present the analysis here to further illustrate the robust performance of the evolutionary computation approach and the efficiency of its parallel computation. The database has 145 patterns whose dimension size was 5 (Grunenfelder et al. 2001; Vohradsky et al. 2003). The database is

Table 6.4 Wisconsin diagnostic breast cancer results

Methods	Se	Sp
SVM	0.918	0.811
LS	0.959	0.893
LDA	0.959	0.893
LR	0.936	0.926
LB	0.940	0.910
GA	0.983	0.830

Table 6.5 Sp and Se of proteomic database of six algorithms

	GA		SVM		LB		LR		LDA		LS	
Cluster	Se	Sp	Se	Sp	Se	Sp	Se	Sp	Se	Sp	Se	Sp
1–3	1	0.7438	1	0.7190	1	0.7438	1	0.7025	1	0.6529	1	0.6364
4–7	1	0.8783	1	0.8609	1	0.9304	1	0.8870	1	0.8783	1	0.8435
10–14	1	0.7870	0.9730	0.9167	1	0.7407	1	0.75	1	0.7130	1	0.7130
17–19	1	0.8296	1	0.6370	1	0.7481	1	0.7481	0.9	0.7407	0.7	0.7556
20–23	1	0.9478	1	0.9478	1	0.6348	1	0.6348	1	0.6261	1	0.6261
1–7	1	0.9780	0.9630	1	1	1	1	1	1	0.9670	1	0.9670
17–23	1	0.9524	0.9750	0.9238	0.975	0.8476	0.975	0.8381	1	0.9238	1	0.9238

Table 6.6 Sp and Se using sequential and parallel computing

	Sequent GA		Parallel GA (two islands)		Parallel GA (four islands)	
Cluster	Se	Sp	Se	Sp	Se	Sp
1–3	1	0.7438	1	0.7521	-	-
4–7	1	0.8783	1	0.9565	-	-
10–14	1	0.7870	1	0.8333	1	0.8889
20–23	1	0.9478	1	0.9739	-	-
1–7	1	0.9780	1	1	-	-

Null value means parallel computing does not give better result

available at http://proteom.biomed.cas.cz. The data were first analyzed using a clustering method. The average pattern of each cluster was then calculated and used in the initial training set for the algorithm to find other patterns of clusters. For example, the initial training set of cluster 1–7 problem contained average patterns from 1 to 7. The algorithm was tested with seven clusters and the results are summarized in Table 6.5.

On the basis of the previous researches of parallel evolutionary computing, we applied the island model to this algorithm with the following parameters: the topology was ring, migration rate was from 5% to 10%, migration was executed after ten generations, and subpopulation sizes were 500 and 260 for two and four islands, respectively. The results are listed in Table 6.6.

6.3.3 Genetic Algorithm for Single-Class Classification

In binary classification, we use the training set that consists of two subsets, namely positive set and negative set. The positive set contains similar patterns which we want to search in a database. The negative set contains some arbitrary patterns which are not similar to the patterns of the positive set. In fact, when users use the program to search for a pattern they only know the pattern that they want to search in a database. So the selection of the negative set makes it difficulties to the users. In order to tackle some disadvantages of binary classification algorithm, we present

an algorithm that uses only one set in the training process. This set contains only similar patterns that the user wants to search in a database. This problem is called the single-class classification (Scholkopf and Smola 2002).

6.3.3.1 Nonlinear Programming Problem

Let the training set be a set of patterns, $\text{TS} = \{\mathbf{x}_i \in \text{R}^n, i = 1..m\}$, with m is number of patterns in the training set. The main idea of the algorithm is to find a hyperplane H that can contains all points of the training set. In other words, a hyperplane H that minimizes the total distance from all points of the training set to the hyperplane H is found. Therefore, we have the following nonlinear programming problem:

Find $\mathbf{z} = [u_1, u_2, \ldots, u_n, v]^{\text{T}}$ which minimizes

$$f(\mathbf{z}) = \sum_{k=1}^{m} d(\mathbf{x}_k, H) = \sum_{k=1}^{m} \frac{|\mathbf{u}^{\text{T}}\mathbf{x}_k - v|}{\sqrt{\sum_{i=1}^{n} u_i^2}} \tag{6.13}$$

Subject to:

$$\begin{cases} -1 \le u_i \le 1, i = 1..n, \text{ and } \exists u_i \ne 0 \\ -1 \le v \le 1 \end{cases} \tag{6.14}$$

In order to solve the above nonlinear programming problem, we use genetic algorithms.

6.3.3.2 Prediction

Let H be the best hyperplane; $Min\_Dis = \min\{d(\mathbf{x}_i, H), \forall \mathbf{x}_i \in \text{training set}\}$; $Max\_Dis = \max\{d(\mathbf{x}_i, H), \forall \mathbf{x}_i \in \text{training set}\}$. If the distance $d(\mathbf{a}, H)$ from tested point $\mathbf{a}$ to hyperplane H is within the range $[Min\_Dis, Max\_Dis]$, then tested point $\mathbf{a}$ is similar to training set *TS*.

6.3.3.3 Using GA to Solve Nonlinear Programming Problem

Chromosome

Each chromosome represents a hyperplane. So each chromosome is encoded as a fixed-length string of $(n + 1)$ real numbers (value encoding). The first n real values represent u_i $(i = 1..n)$; the last real value represents v.

Fitness Function

Because each chromosome of initial population is randomly created as a set of $(n + 1)$ real numbers within the range $[-1, 1]$, and mutation operation is not used, so inequalities in Eq. (6.14) are satisfied during the search process.

Equation (6.13) is used to calculated fitness value of each chromosome. In other word, the total distance from all points in training set to hyperplane (chromosome) is computed. The best chromosome (hyperplane) is the one that has the minimum total distance.

6.3.3.4 Experiments

Wisconsin Diagnostic Breast Cancer

All the control parameters of the genetic algorithm approach are listed in Table 6.1. Se and Sp values were obtained using Eqs. (6.4) and (6.5). Results obtained from the GA are listed in Table 6.7. It can be seen that the GA for single-class classification performs better than the GA for binary classification being shown in Table 6.4.

Proteomic Database

On the basis of the results in Table 6.8, it can be seen that the GA-based method is superior to the other classifiers.

Table 6.7 Wisconsin diagnostic breast cancer results

Methods	Se	Sp
SVM	0.918	0.811
LS	0.959	0.893
LDA	0.959	0.893
LR	0.936	0.926
LB	0.940	0.910
GA	1.0	0.830

Table 6.8 Se and Sp results obtained from various algorithms where single SVM stand for single class using SVM

	GA		Binary SVM		Single SVM		LB		LR		LDA		LS	
Cluster	Se	Sp	Se	Sp	Se	Sp	Se	Sp	Se	Sp	Se	Sp	Se	Sp
1–3	1	0.992	1	0.719	0.542	0.959	1	0.744	1	0.702	1	0.653	1	0.636
4–7	1	1	1	0.861	0.80	0.809	1	0.930	1	0.887	1	0.878	1	0.843
10–14	1	0.972	0.9730	0.917	0.946	0.870	1	0.741	1	0.75	1	0.713	1	0.713
17–19	1	0.918	1	0.637	0.80	0.889	1	0.748	1	0.748	0.9	0.741	0.7	0.756
20–23	1	0.991	1	0.948	0.767	0.844	1	0.635	1	0.635	1	0.626	1	0.626
1–7	1	1	0.9630	1	0.889	0.989	1	1	1	1	1	0.967	1	0.967
17–23	1	0.971	0.9750	0.924	0.80	0.857	0.975	0.848	0.975	0.838	1	0.924	1	0.924

6.4 Conclusion

On the basis of several experimental results previously presented, we can see that both SVM and GP gave high performance. Other traditional methods such as LB, LR, LDA, and LS do not achieve superior performance because of the high dimensions of the data. This is the reason why feature selection methods are used to reduce dimension when these methods are applied to high dimension patterns. SVM depends on kernel; for example, (1) Gaussian kernel tends to give good results whereas linear kernel gives poor results with some training, (2) different selections of the parameters of kernel may yield different results, and (3) parameter $\nu \in (0, 1]$ can affect the classification. Although SVM gives a fixed result on fixed parameters, there are many parameters, resulting in expensive computation.

Binary classification has been known as the most popular method in supervised learning. In this chapter, we proposed two methods for binary classification: the first method is based on genetic programming to search decision trees; in the second method, genetic algorithms were used to search hyperplanes that are the solutions of the nonlinear programming.

Although binary classification is a very popular, it has several disadvantages. We know that the negative set is very important because it helps classifier to recognize patterns which differ from the positive set. But there are some features of the negative set that can affect the searching for the right pattern. If the number of patterns in the negative set is high, then number of misclassified patterns is usually low and the number of true patterns is low as well. On the contrary, if the number of patterns in the negative set is low, then number of misclassified patterns is usually high as well as the number of true patterns. There may be no general rule to determine how many patterns in the negative set is the best for every database; and it is still a trial and error task.

Method that does not use the negative set in the training set can remove disadvantages of the negative set. This method is called single-class classification. According to our best knowledge, only one single-class classification method and single-class SVMs have been reported in literature. This chapter proposes a GA-based method for single-class classification, where genetic algorithms were used to solve nonlinear programming. Results of this work are able to illustrate that genetic algorithms and genetic programming are powerful methods for optimization and regression, respectively. The biggest disadvantage of evolutionary computation is that they require considerable time for computing, but this problem can be overcome by parallel computing which computing has become a feasible task for large computations. There are several parallel models for evolutionary computing. In this work, we used the island model because it not only gives high performance but also improves the overall results. In binary classification, we usually find a classifier that maps patterns into $\{-1, +1\}$ set. In our opinion, it is not natural; therefore, we will investigate to develop an algorithm that uses only one-class information [do not map pattern into $\{-1, +1\}$ set] to train the classifier: if being successful, there is no need to convert multiclass classification into multiple-binary classifications.

References

Alba E., Laguna M., Luque G. (2005). "Workforce planning with a parallel genetic algorithm." Proceedings of CEDI-MAEB'05: 911–919.

Alon U., Barkai N., et al. (1999). "Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays." Proc Natl Acad Sci USA 96: 6745–6750.

Andre D., Koza J.R. (1996). "Parallel genetic programming: a scalable implementation using the transputer network architecture," in Advances in Genetic Programming 2, Cambridge, MA, MIT Press.

Bittner M., Meltzer P., Chen Y., et al. (2000). "Molecular classification of cutaneous malignant melanoma by gene expression profiling." Nature 406, 536–540.

Calegari P., Guidec F., Kuonen P., Kobler D. (1997). "Parallel island-based genetic algorithm for radio network design." Journal of Parallel and Distributed Computing 47(1): 86–90.

Chong K.P.E., Zak H.S. (2001). An Introduction to Optimization. New York, John Wiley & Sons.

Dettling M., Buhlmann P. (2003). "Boosting for tumor classification with gene expression data." Bioinformatics 19: 1061–1069.

Dudoit S., Fridlyand J., et al. (2002). "Comparison of discrimination methods for the classification of tumors using gene expression data." Journal of the American Statistical Association 97: 77–87.

Erick Cantu-Paz (2001). Efficient and Accurate Parallel Genetic Algorithms. Kluwer Academic Publishers.

Fernandez de Vega F. (2005). "Parallel genetic programming," Workshop of the 2005 IEEE Congress on Evolutionary Computation.

Freitas A.A. (2002). Data Mining and Knowledge Discovery with Evolutionary Algorithms. Berlin, Springer Verlag.

Furey T.S., Cristianini N., et al. (2000). "Support vector machine classification and validation of cancer tissue samples using microarray expression data." Bioinformatics 16: 906–914.

Golub T.R., Slonim D.K., et al. (1999). "Molecular classification of cancer: class discovery and class prediction by gene expression monitoring." Science 286: 531–537.

Gray H.F., Maxwell R.J., et al. (1996). "Genetic programming for classification of brain tumours from nuclear magnetic resonance biopsy spectra." Genetic Programming 1996: Proceedings of the First Annual Conference: 28–31.

Grunenfelder B., Rummel G., Vohradsky J., Roder D., Langen H., Jenal U. (2001). "Proteomic analysis of the bacterial cell cycle." Proc Natl Acad Sci USA, 98(8):4681–4686.

Guyon I., Weston J., et al. (2002). "Gene selection for cancer classification using support vector machines." Machine Learning 46: 389–422.

Juille H., Pollack J.B. (1995). "Parallel genetic programming and fine-grained SIMD architecture" in Working Notes for The AAAI Symp. 31–37.

Khan J., Wei J.S., et al. (2001). "Classification and diagnostic prediction of cancers using gene expression profiling and artificial neural networks." Nature Medicine 7: 673–679.

Koza J.R. (1992). Genetic Programming: on the Programming of Computers by Means of Natural Selection. London, MIT Press.

Lee Y., Lee C.-K. (2003). "Classification of multiple cancer types by multicategory support vector machines using gene expression data." Bioinformatics 19: 1132–1139.

Liu J.J., Cutler G., et al. (2005). "Multiclass cancer classification and biomarker discovery using GA-based algorithms." Bioinformatics 21: 2691–2697.

Mangasarian O.L., Street W.N., Wolberg W.H. (1995). "Breast cancer diagnosis and prognosis via linear programming." Operations Research, 43(4), pages 570–577.

Mitchell M. (2001). An Introduction to Genetic Algorithm. London, MIT Press.

Nguyen D.V., Rocke D.M. (2002). "Tumor classification by partial least squares using microarray gene expression data." Bioinformatics 18: 39–50.

Perou C.M., Jeffery S.S., et al. (1999). "Distinctive gene expression patterns in human mammary epithelial cells and breast cancers." Proc Natl Acad Sci USA 96: 9212–9217.

Petricoin E.F., Ardekani A.M., et al. (2002). "Use of proteomic patterns in serum to identify ovarian cancer." The Lancet 359: 9306, 572–577.

Pham T.D. (2008). "Computational prediction models for cancer classification using mass spectrometry data." Int. J. Data Mining and Bioinformatics, Vol. 2, No. 4: 405–422.

Punch W.F. (1998). "How effective are multiple populations in genetic programming." Proceedings of the Third Annual Genetic Programming Conference: 313–318.

Scholkopf B., Smola J.A. (2002). Learning with Kernels. MIT Press.

Street W.N., Wolberg W.H., Mangasarian O.L. (1993). "Nuclear feature extraction for breast tumor diagnosis." IS&T/SPIE 1993 International Symposium on Electronic Imaging: Science and Technology 1905: 861–870.

Su M., Basu M., et al. (2002). "Multi-domain gating network for classification of cancer cells using gene expression data." Proceedings of the International Joint Conference on Neural Networks 1: 286–289.

Toure A., Basu M. (2001). "Application of neural network to gene expression data for cancer classification." Proceedings of the International Joint Conference on Neural Networks 1: 583–587.

Vohradsky J., Janda I., Grunenfelder B., Berndt P., Roder D., Langen H., Weiser J., Jenal U. (2003). "Proteome of Caulobacter crescentus cell cycle publicly accessible on SWICZ server." Proteomics, 3(10):1874–1882.

Zhang H., Yu C.-Y., et al. (2001). "Recursive partitioning for tumor classification with gene expression microarray data." Proc Natl Acad Sci USA 98: 6730–6735.

Chapter 7
Analysis of Population-Based Genetic Association Studies Applied to Cancer Susceptibility and Prognosis

Xavier Solé, Juan Ramón González, and Víctor Moreno

Abstract Along hundreds of thousands of years, genetic variation has been the keystone for human evolution and adaptation to the surrounding environment. Although this fact has supposed a great progress for the species, mutations in our DNA sequence may also lead to an increased risk of developing some diseases with an underlying genetic basis, such as cancer. Among different genetic epidemiology branches, population-based association studies are one of the tools that can help us decipher which of these mutations are involved in the appearance or progression of the disease. This chapter aims to be a didactic but thorough review for those who are interested in genetic association studies and its analytical methodology. It will mainly focus on SNP-array analysis techniques, covering issues such as quality control, assessment of association with disease, gene–gene and gene–environment interactions, haplotype analysis, and genome-wide association studies. In the last part, some of the existing bioinformatics tools that perform the exposed analyses will be reviewed.

7.1 Genetic Variation and Its Implication in Cancer

The implication of genes in cancer has long been suspected because this disease shows familial aggregation, in some instances remarkably. The study of cancer cells shows extensive genomic alterations, ranging from mutations in target genes – known as oncogenes and tumor suppressor genes – to large chromosomal aberrations. These alterations are supposed to be triggered by initial events that accumulate and confer the cancer cells proliferation advantage and escape to control of DNA damage. Alterations are acquired during the carcinogenesis process and are called somatic alterations. However, individuals that carry alterations in germ line are

X. Solé (✉)
Biostatistics and Bioinformatics Unit, Catalan Institute of Oncology – IDIBELL, Av. Gran Via s/n Km 2.7, 08907 L'Hospitalet de Llobregat, Barcelona, Spain
e-mail: x.sole@iconcologia.net

T. Pham (ed.), *Computational Biology: Issues and Applications in Oncology*, Applied Bioinformatics and Biostatistics in Cancer Research,
DOI 10.1007/978-1-4419-0811-7_7,

known to have susceptibility to develop cancer. Mutations in a few genes have already been identified as responsible for cancer syndromes like Li–Fraumeni (p53), familial breast cancer (BRCA1, BRCA2), adenomatous polyposis coli (APC), and Lynch syndrome (MLH1, MSH2, MSH6, PMS2) (Foulkes 2008). These mutations show high penetrance, but are rare and do not explain more than 5% of all cancers, while other 15–25% are thought to have a relevant genetic contribution.

Though families share genes and environment, and part of the familial aggregation could be related to shared lifestyles, diet, and other exposures, twin studies allow estimation of the relative contribution of genes and environment. An important fraction of most frequent cancers is related to genetic factors: 42% of prostate, 35% of colorectal, and 27% of breast, and similar estimates were observed for other less frequent tumors (Lichtenstein et al. 2000).

The discrepancy between heritability estimates and the proportion of cases associated to known genes raised the hypothesis that other genes should be involved in cancer etiology, though with lower penetrance and probably high frequency. An intensive search for these susceptibility genes has been triggered when genotyping technologies have emerged that allow easy simultaneous analysis of thousands to millions of genetic markers. Also, the knowledge that recombination is not occurring at random throughout the chromosomes, but in specific regions that delimit blocks of nucleotides that are transmitted together (Hap 2003) (see Sect. 7.8.3.2), has helped designing strategies to extensively explore the genetic variation at a genome-wide scale in order to identify cancer susceptibility loci.

Heritable genomic variations are called polymorphisms, which occur by mutation of DNA in germinal cells and are transmitted to descendants. Most of these polymorphisms have no functional impact, either because they occur in noncoding regions or do not modify the protein product qualitatively or quantitatively. Some of these polymorphisms do have a functional impact and are the base of evolution. Usually when the effect provides an advantage to the individual the polymorphism increases in frequency in the population. Conversely, deleterious mutations tend to disappear, though they can reach relatively high frequency in the population if the heterozygous status provides some advantage like sickle cell anemia carriers, which are more resistant to malaria.

There are three types of polymorphisms at the genetic level: single nucleotide polymorphisms (SNP), variable number tandem repeats (VNTR) and copy number variations (CNV).

SNPs, the most frequent polymorphisms, are changes in one nucleotide at a given genomic position. Usually one nucleotide is substituted by other, but sometimes one or a few nucleotides are deleted or inserted (Ins/Del). The results of these minor changes are diverse. If the SNP is in an exon, it may confer a change in the aminoacid chain of the resulting protein, or a truncated protein if the SNP results in an stop codon. SNPs in introns and noncoding regions may also be functional by altering splicing sites or the binding of transcription factors. The ENCODE project (Birney et al. 2007) is revealing that DNA expression is frequent in noncoding regions. SNPs in resulting RNAs might also have relevant functions.

Nonfunctional SNPs are scattered throughout the genome with one average distance of one SNP every 1,000 bp. The average haplotype block has a size of 20,000 bp in non-African populations and 10,000 bp in African populations. Thus, there are about 20 SNPs per haplotype block on average, but only 5–6 different haplotypes per block, as there is high redundancy. Only few SNPs per haplotype block are needed to ascertain most of the variation and identify which haplotype is carrying a causal polymorphism, if it exists. These minimum number of selected SNPs are called haplotype-tagging SNPs (htSNPs).

VNTRs appear with less frequency and consist in serial repetitions of a short series of nucleotides with length variability among individuals. For example, ATATAT $= (AT)_3$, ATATATATAT $= (AT)_5$. The repeats may be mononucleotide (AAAA), dinucleotide (AT), or even larger repeats. These polymorphisms are also called microsatellites and most often are multiallelic, since the number or repeats may vary greatly. This condition increases the likelihood of heterozygosity and makes VNTRs very informative for some genetic analyses, particularly linkage. VNTRs may also have a functional effect if present in relation to coding regions. As a typical example, type 1 diabetes has been associated to a VNTR in the insulin gene. Subjects with a short number of repeats (less than 50) have double risk than subjects with more than 200 repeats (Bennett et al. 1995). More recent findings link VNTRs and predisposition to early-onset colorectal cancer (Yeh et al. 2009). Though VNTR are very informative, their genotyping usually requires more elaborated and expensive methods than SNPs (usually sequencing) and for this reason these polymorphisms are less often used for linkage and association studies nowadays.

CNVs have been identified more recently as an additional source of genomic variation. These are relatively large regions spanning kilobases, sometimes covering multiple genes, that appear in multiple copies with a variable number of repetitions, in the range of 0 (deletion) to tens (Redon et al. 2006). CNVs are a typical genomic somatic alteration in most cancers. Germ line CNVs are also being studied as a potential cancer susceptibility source (Shlien et al. 2008).

7.2 Evolution of Genetic Epidemiology: From Family-Based to Population-Based Association Studies

Finding cancer genes is a long task that needs to answer a series of questions (see Table 7.1). Each question usually requires a specific study design and measures genetic information with different levels of precision. Though the methods in this chapter will focus on association, it is important to know where this design is in relation to other alternatives.

The first and most important question is: Are genes involved in cancer? Case–control studies showing familial aggregation of cancer provide indirect information about the potential implication of genetic factors. Having a first degree relative with cancer is a risk factor for most frequent cancers. However, this is a very crude measure that might be confounded by shared environmental exposures. Studies in

Table 7.1 Relevant questions and study designs in genetic epidemiology

Question	Study design
Are genes involved in cancer?	Familial aggregation, twin studies
What is the inheritance model?	Segregation
Where are the genes?	Linkage
Which are the genes?	Association
What is the causal variant?	Fine-mapping
Which is the mechanism?	Functional studies
Interactions	G×G and G×E association

migrants may also be informative. Cancer rates in second generations of migrants that are more similar to their origin than the country of residence are indicative of a genetic component. Twin studies are the most powerful to estimate heritability (i.e., the proportion of cases attributable to genetic factors). The comparison of concordance rates between monozygotic and dizygotic twins, when combined with information about shared environment, provides most valuable information (Lichtenstein et al. 2000).

The occurrence of specific cancers sometimes is a recurrent event in some families. In such situations, when a major gene is suspected to be responsible for the disease, segregation analysis of the pedigrees can provide information about the inheritance model and estimates of penetrance (Bailey-Wilson et al. 1995). These studies use only phenotype information and family structure and do not require DNA markers.

When enough information is accumulated about genetic factors as a cause of a specific cancer, the next question is: which are the genes? When genotyping of genetic markers became feasible, before the genome was completely sequenced, it was easier to identify regions of the genome associated to cancer and, in a second step, try to identify which gene in that region was responsible. Linkage studies explore a series of polymorphic markers carefully selected across the genome in large pedigrees of affected families. When at least three generations are genotyped, polymorphic markers can identify which alleles cosegregate with the disease and identify the regions most likely to carry the causal genes. Linkage analysis can combine the information of multiple families and is very powerful to detect signal when the penetrance is high, but since only about 400 markers are used to cover the genome, the level of resolution is in the range of megabases. After a consistent linkage signal has been detected, sometimes hundreds of genes may be in the region and this technique is not always able to improve the resolution even when increasing the number of markers because the number of subjects from the affected families is relatively small.

In order to identify the specific genes related to cancer, association studies with unrelated individuals using SNPs as genetic markers are the most powerful approach. Unrelated individuals increase the likelihood of recombination events and increase the resolution of the signal. Careful selection of SNPs, nowadays using information about haplotype blocks, can identify which genes are involved in the

disease. Association studies compare the genotype frequencies of a series of SNPs between a sample of unrelated cases and a sample of controls from the same population. The possibility to include unrelated cases and controls allows the usage of large sample sizes to increase detection power. Association studies are usually focused on selected candidate genes belonging to regions that have shown linkage or because their known mechanism of action makes the gene possibly related to cancer. For example, typical genes studied in cancer are involved in cell cycle control, inflammation, metabolism, or DNA repair (Landi et al. 2005; Moreno et al. 2006).

Since current large-scale genotyping technology allows to simultaneously genotype millions of SNPs, currently Genome-Wide Association Studies (GWAS) are being conducted to identify susceptibility loci not necessarily related to coding regions. In fact, the first finding of these studies in prostate cancer has identified a region in 8q24 where no genes can be clearly imputed as responsible (Haiman et al. 2007b). POU5F1 is the nearest expressed region, but corresponds to a pseudogene. MYC, a known oncogene that lies downstream the region, is also suspect of being involved, but the evidence is indirect (Sole et al. 2008).

Even when a gene has been clearly associated to a disease, finding the causal variant usually requires resequencing and intensive genotyping to fine-map the region. Identifying the causal variant will also need functional studies to document the mechanism of action that determines the risk.

For some genes, the genetic variation is probably not sufficient to cause cancer unless an environmental exposure is acting simultaneously. For example, polymorphisms in NAT2 have been associated to an increased risk of bladder cancer among smokers; for nonsmokers the risk is not increased (Garcia-Closas et al. 2005). This is an example of gene–environment interaction that is probably relevant in many genetic determinants. Ignoring the environmental effect leads to an attenuated risk (average of smokers and nonsmokers) that is difficult to detect unless the study has a large sample size. Similar to gene–environment interactions, it is likely that gene–gene interactions may exist and only carriers of multiple variants are at increased risk of developing cancer. The difficulty in detecting such interactions is that, without prior hypothesis, the search domain is huge and very large sample sizes are needed.

7.3 Technical Issues and Data Quality Control for SNP-Array Association Studies

All biological experiments are subject to different sources of variability. Particularly, large-scale techniques, such as DNA microarrays, may be specially sensitive to specific experimental conditions that are not easy to keep under control (Spruill et al. 2002). Although there are some methods which try to minimize this variability, such as data normalization or experiment replication, it is not possible to remove it completely. Thus, besides being extremely careful about how all the experiments are performed and the data normalized, before going on with our analysis we will

also need to check the quality of the obtained data to increase the reliability of the study results. As we previously stated, this chapter will mainly focus on SNP-array analysis techniques. First, we will briefly review some of the different genotyping algorithms that have been used to infer the calls from raw data. Once the genotypes are obtained, SNP-array quality control can be performed at different levels: SNP and sample (array). In the following sections, we are going to briefly review some of the different calling algorithms, as well as explaining in detail this quality-control procedure and all the steps it comprises.

7.3.1 Introduction to Genotype Calling Algorithms

The *call* of a specific SNP for a single sample is essentially its genotype, that is, the combination of its two corresponding alleles. Since most usually we will be working with two-allele SNPs (also called biallelic), for a given SNP with alleles A and B there will be three possible calls: two homozygous (AA and BB) and one heterozygous (AB or equivalently BA). These calls are automatically obtained using algorithms that process raw intensities coming from the scanned image of the microarray. Usually, for a given SNP and sample we will have two intensity values, each one corresponding to one of the two alleles. Some array platforms, however, have also probes which are strand specific (sense and antisense), finally leading to four intensity values.

Over the last few years, genotyping algorithms have evolved in accordance with the size of the available arrays. The embryo technology of the SNP arrays, Affymetrix Variation Detection Arrays (VDAs), contained about 1,500 SNPs. An algorithm called Adaptive Background Genotype Calling Scheme (ABACUS) was then designed to extract the calls (Cutler et al. 2001). As well as showing a certain trend to drop heterozygous calls, this method was clearly unsuitable when the first SNP arrays appeared (e.g., Affymetrix 10K). Thus, ABACUS was soon replaced by newer algorithms such as Modified Partitioning Around Medoids (MPAM) (Liu et al. 2003). This algorithm was based on the robust classification method called Partitioning Around Medoids (PAM), but it was modified to penalize small between-group distances, since PAM tends to split large clusters into two different groups to minimize the total sum of distances of all the observations to their corresponding nearest medoid. MPAM worked well for SNPs that had enough data in each of the three genotypes, but not for SNPs with one missing or very small genotype or when the number of arrays to be analyzed was small.

With the advent of 100K arrays, a lot of SNPs with low minor allele frequency (see Sect. 7.3.2.3) were included in the new platform, making the performance of MPAM decrease remarkably. Thus, it was replaced by the newer DM (Dynamic Model) algorithm (Di et al. 2005), in which four Gaussian models were fitted for the probe intensities of each SNP (one for each genotype and one for the null values), and then a genotype call was assigned to each sample depending on its likelihood.

The DM algorithm had a main limitation: it was a single-array algorithm, that is, it could not take profit of aggregating data sets to better assess how each SNP behaved. Furthermore, it seemed to poorly classify heterozygous samples when compared to MPAM.

Arguing that neither MPAM nor the DM algorithm were using currently available genotypic information, and only about a year after the publication of the DM method, Rabbee and Speed (2006) proposed a new algorithm, called Robust Linear Model based on Mahalanobis distance classification (RLMM). This method had two main advantages over the formerly designed DM algorithm: first, it was a multichip algorithm, thus allowed to assess both probe effects and allele signals for each SNP. Second, genotypes were estimated by means of a multiple-sample classification, that is, using information of other SNPs to better define the properties of the three groups corresponding to the three possible genotypes. To combine intensities across probes and arrays and produce allele-based summaries it used the robust multichip average method (Irizarry et al. 2003). This method took advantage of the large amount of publicly available information on genotype calls (i.e., HapMap) to define regions for each genotype group, thus improving the accuracy of the classification. Nevertheless, although this remarkable increase in accuracy, RLMM still had some problems in dealing with the interstudy or interlaboratory variability, which may be caused by slight variations in the sample preparation procedure, among other reasons.

Affymetrix soon adopted RLMM as the standard methodology to analyze 100K and 500K SNP arrays. The method was slightly modified with the addition of a Bayesian approach which yielded differences in the clustering space transformation and in the estimation of both cluster centers and variances. Although it was its main aim, the resulting algorithm, known as BRLMM (Bayesian RLMM) (BRL 2006), still did not seem to handle accurately the interstudy variability.

To solve this issue, Carvalho et al. (2007) proposed another modified version of the RLMM, known as Corrected Robust Linear Model with Maximum Likelihood Classification (CRLMM). Essentially, it uses an adapted version of RMA preprocessing method for SNPs (called SNP-RMA), which is designed to remove most of the study/laboratory effect. In much the same way as BRLMM does, it also uses Bayesian approach to inform lowly populated clusters. Recently, CRLMM designers have added a new recalibration step to the algorithm which further increases its accuracy level (Lin et al. 2008). The new version also incorporates a new quality metric to assess call confidences at the SNP level, which may be very useful to filter out poor-quality SNPs. This method seems to perform better than all the previous algorithms explained, and even better than Birdseed (Korn et al. 2008), which is the recently designed algorithm by Affymetrix and the Broad Institute for the 6.0 generation of SNP arrays, so it may be a good choice if we need to decide which calling method we are going to use.

Finally, we must point out that although all the calling algorithms we have mentioned in this section have been basically designed to be applied to Affymetrix SNP arrays, the underlying basis of the analysis can also be suitable for arrays made by other manufacturers, such as Illumina's BeadChip technology.

7.3.2 SNP-Level Quality Control

This is the first level of quality control. Once we have the SNP calls, it is important to check their quality one-by-one in order to detect uninformative or poor-quality SNPs, so that they can be permanently removed from all the samples contained in our dataset.

SNP-level quality control can be divided into different parts, each one of them checking different quality issues. SNPs that meet *all* the requirements are the ones that will be kept for further analysis.

7.3.2.1 Percentage of Present Calls

As we stated in Sect. 7.3, defective hybridizations or incorrect analytical processes may result in poor-quality data and could hamper obtaining reliable genotype calls. In the case of SNP-level percentage of present calls, difficulties may arise mainly from improper functioning of genotype calling algorithms. Some of them, such as BRLMM, may introduce some systematic bias in the missing values they report (Hong et al. 2008). As a consequence, this bias may not randomly affect all the three different genotypes of a SNP, but only some of them. Having the calls for all SNPs and samples, we can then assess the missing rate for each SNP across all the hybridizations. Since missing call values will potentially be related to low levels of genotyping quality, we should discard from our study those SNPs with a poor call rate. Although rather subjective, an 80% of present calls is usually considered as the minimum threshold applied to filter out potential low-quality or highly biased SNPs.

7.3.2.2 Hardy–Weinberg Equilibrium

Given a SNP and its allele frequencies for a specific population, the Hardy–Weinberg principle determines what the expected genotype frequencies should be, assuming that the different alleles are transmitted independently from one generation to another and with no selective pressure over them. Therefore, if we have a SNP with two alleles, A and B, with population frequencies p and q (or equivalently $1 - p$) respectively, the expected genotype probabilities are:

$$f_{AA} = p^2$$
$$f_{AB} = 2 \cdot p \cdot q$$
$$f_{BB} = q^2$$

To assess if one SNP follows the Hardy–Weinberg law we can use the Pearson's goodness-of-fit Chi-square test statistic, χ^2, with 1 degree of freedom (in case we

have a biallelic polymorphism). The null hypothesis is that the SNP is indeed under HWE, so we will reject SNPs with p values smaller than a specific significance level. The Chi-square statistic, however, may have a poor performance when we have small genotype counts, so in that case it will be better to use a Fisher's Exact test instead (Guo and Thompson 1992; Wigginton et al. 2005).

The fact that a SNP does not follow the Hardy–Weinberg law may be due to different reasons:

- Small population size.
- The allele-calling algorithm is underperforming for one of the genotypes (i.e., it fails to correctly call heterozygotes).
- The SNP is mapping to multiple genomic locations.
- The genotyped individuals are not independent (i.e., because of inbreeding).
- There has been a positive selection of a certain allele (i.e., an allele associated to longevity).
- If we use a significance level of a 5%, we may find by chance different observed frequencies from the ones we expect. This would happen for a 5% of the SNPs for which we are evaluating HWE. Theoretically, we would need to perform p value adjustment (see Sect. 7.6.4) to solve this issue. Nevertheless, what is usually done in this context is to set a more restrictive threshold for significance for HWE tests, but not as restrictive as it would be using standard p value correction methods. Researchers have widely accepted 0.001 as a suitable boundary, being 0.0001 in the case of GWAS, where more tests are performed. Even if after setting this more astringent threshold we still find SNPs with genotype frequencies under no HWE then one of the other issues on this list may be the reason, so we will need to evaluate our data in detail to find what is causing this genotypic imbalance.

In the typical case–control study, HWE may be only evaluated in control populations, which is where it should hold true. If we do not identify clearly the reason of the disequilibrium, it may be necessary to remove the affected SNPs. In case we want to keep them, association results for those SNPs need to be checked carefully, as there may be some influence from one of the items mentioned above. As a guidance, we can also look at HWE among the cases, since a SNP with no equilibrium might be potentially related with the disease.

7.3.2.3 Minor Allele Frequency

The *minor allele frequency*, or MAF, of a SNP is its lowest allele frequency. There is a huge variation in the MAF among different SNPs, from a very low percentage (e.g., less than 1%) to almost a 50%, meaning that in fact there is no minor allele. Although the MAF is not a quality measure by itself, it might be useful to filter SNPs according to it for subsequent analysis. It must be taken into account that, if a certain SNP has a very low MAF, we will have very little statistical power to detect its potential association with the disease (see Sect. 7.6.3). Furthermore, these SNPs are more difficult to genotype reliably. Therefore, removing those SNPs, from which

a priori it will be hard to get any useful information, might increase the quality of our data and will slightly reduce the number of hypothesis tested, so we will be a bit less restrictive in the step of p value correction for multiple hypothesis testing.

7.3.2.4 Genotype Calling and Exploration of Signal Intensity Plots

As we have seen in Sect. 7.3.1, SNP allele intensities are the data we use to infer the genotypes. Since all preprocessing and calling procedures are mostly automatic, we do not usually work with these intensities directly. Nonetheless, we can still use them if we are specially interested in checking a few specific SNPs. To do so, signal intensity plots are mainly used. For a given SNP, these plots are useful to visually inspect the intensity values for both alleles across all samples. Under an ideal situation we will observe three clouds, one for each genotype (Fig. 7.1, left panel). However, some issues may influence this intensity values, thus distorting the plot and making it more difficult to visually define the three clusters (Fig. 7.1, right panel). This will happen mainly for bad-quality SNPs, SNPs with poor intensities, SNPs with homologous sequences in different parts of the genome, or SNPs involved in a Copy Number Variant (CNV), among other reasons.

Since checking this plots is mainly a visual inspection that needs to be done SNP-by-SNP, it will be virtually infeasible to have a look at the hundreds of thousands of SNPs contained in an array. Thus, more than a preanalysis quality-control step, this should be considered a postanalysis quality-control procedure. That is, when

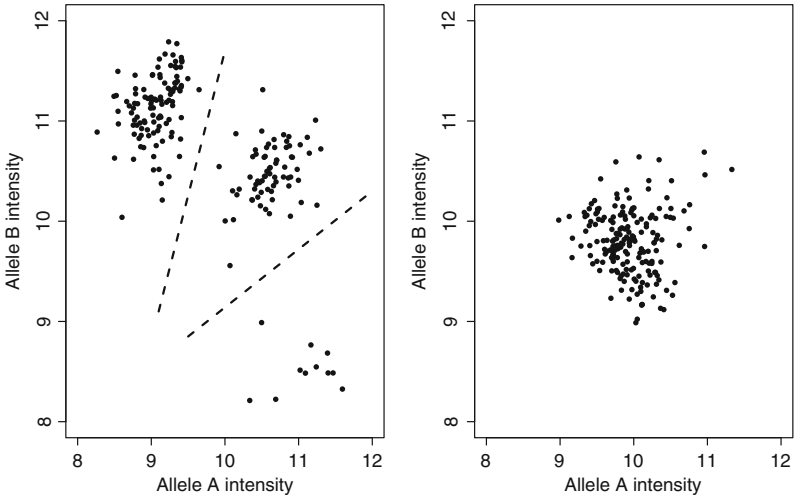

Fig. 7.1 Intensity plots of allele *A* vs. allele *B* for two SNPs. In the *left panel* we see a good-quality SNP, where the boundaries between the three genotype regions (*dashed lines*) can be clearly defined. The *right panel* belongs to a defective SNP, where no clear boundaries between genotypes regions can be defined, yielding to potentially incorrect calling results

we have a few candidate SNPs and we want to ensure the reliability of the obtained results, we can plot their intensities and see how the calling algorithm has created the different groups. Furthermore, if we find a strange negative result (i.e., lack of association with the disease when we already expected it) we can do the same to check if it has been caused by any of the technical reasons stated above.

7.3.3 Sample-Level Quality Control

As well as performing SNP-level quality control, it is important to check whether there are any poor-quality samples in our dataset or not. This fact could happen not only because of bad quality of the biological material to be hybridized (e.g., DNA), but also because there may be some problem with the hybridization. That is, since samples and arrays are confounded, sometimes it will be hard to tell where the problem comes from unless we perform replicates of our experiments, which is rather expensive. Independently of the underlying reasons though, all samples with a poor-quality level should be removed from our dataset for further analysis, as they could be a potential source of error in our study. In the following sections we will review which parameters can be used to detect these defective hybridizations.

7.3.3.1 Percentage of Present Calls

Much in the same way as we explained in Sect. 7.3.2.1, percentage of present calls in a sample can be a helpful quality index. In the case of sample percentage of present calls, this fact could be mainly related to the quality of the hybridized DNA (i.e., it may be degraded or the amount of DNA hybridized may be too small), as well as with some technical problem with the hybridization or with the microarray itself. Usually, those samples with less than 95 or 97% of present SNPs should be discarded, since missing genotypes tend to be nonrandomly distributed. This threshold may be increased or decreased depending on how strict we want to be with our data. If for any reason we decide to lower it significantly (e.g., less than 90%), we must always bear in mind that our final results may be influenced by this potential artifact.

7.3.3.2 Sample Heterozygosity

Total heterozygosity, understood as the number (or proportion) of heterozygous SNPs in one sample, can be a good-quality indicator at the sample level. As an example, individuals having a large proportion of heterozygous SNPs may be more likely to have their DNA contaminated. On the contrary, a too low level of heterozygosity could indicate that there may be some problem with the hybridization or even a sign of inbreeding for that individual. A rather simple but typically used

methodology to filter out those samples with an odd level of heterozygosity is to compute the mean and the standard deviation of this index across all samples and then filter out those individuals falling outside the mean ± 3SD.

Regarding heterozygosity analysis, it is interesting to remark that we should pay special attention to SNPs located in the X chromosome, since presence or absence of heterozygous SNPs in a specific sample will help us decipher the gender of that individual (i.e., only females can have heterozygous SNPs in the X chromosome). This will enable us to check for possible mistakes during the process of sample annotation.

7.3.3.3 Using Principal Components Analysis as a Method to Detect Outliers or Related Samples

Even after removing those defective SNPs and samples by methods such as the ones described in previous sections, to reach the maximum level of quality in our data we must still ensure that none of our individuals displays an irregular genotype pattern. As an example, this could happen if, by mistake, a Chinese or African individual falls into a study of Caucasians. By applying all the filters mentioned above we may not detect this fact, so we need to use techniques capable of discovering this kind of outliers. Additionally, another issue we must take into account is to search for any underlying relationships between individuals in our cohort. That is, if we have samples with a higher level of concordance between their genotypes than we would expect by chance. This relationship could be due to technical (e.g., date of hybridization, batch effect, etc.) or biological reasons (e.g., inbreeding).

A useful technique to perform all these quality controls is Principal Components Analysis (PCA). Basically, it is a dimensionality reduction technique which transforms an undefined number of correlated variables (which in this case would be the samples) into a smaller number of uncorrelated and ordered variables, called principal components. The order of the principal components is arranged according to the amount of variability explained by each one of them, being the first principal component the one that accounts for most of the variability.

Before doing the analysis, we will need to perform a simple transformation of the genotype matrix into a numerical one. That is, we will recode genotypes, such as AA–AB–BB, into numbers (e.g., 0–1–2). Although this may be enough to detect outliers in our dataset, a transformation of the matrix as the one suggested by Price et al. (2006), which takes into account the differences in the MAF of all the SNPs, may be appropriate. Performing a PCA is a rather straightforward task. Nonetheless, it may be memory consuming for very large datasets, so in this case we may need a computer with a fairly big amount of RAM memory for this purpose.

Once the analysis is done, a rather simple but useful way to check for outliers or unlikely relationships among samples is to plot the values of the first principal components for all the samples contained in our dataset. As an example, in Fig. 7.2 we can see a plot of the two first principal components for a dataset containing 94 HapMap samples and 6359 SNPs located in the genomic region 8q24. Two of the

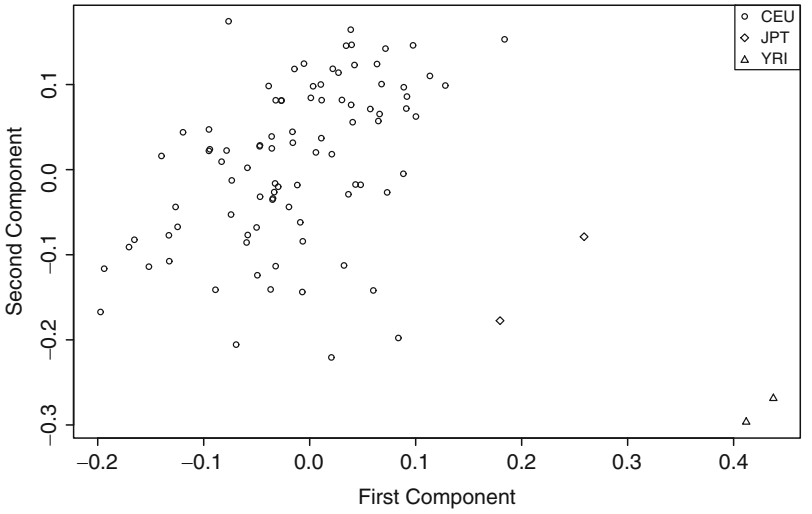

Fig. 7.2 Principal components analysis plot of 94 HapMap samples using 6359 SNPs located in the 8q24 region. We can clearly see how the two African and the two Asian individuals separate from the rest

samples have Asian origin (Japanese in Tokyo, Japan), and two more come from Africa (Yoruba in Ibadan, Nigeria), being the remaining 90 CEPH (Utah residents with ancestry from northern and western Europe). In the plot we can clearly see how JPT and YRI individuals can be clearly distinguished from CEU, which form a relatively homogeneous group. Therefore, this procedure has enabled us to uncover those samples that may be defective or have a different origin than we could have expected. Although the example is shown to reveal different ethnic origins, it can be also useful to detect technical biases or batch effects in our dataset. A more extensive application of PCA to genome-wide association studies is reviewed in Sect. 7.6.1.

7.4 Single-SNP Analysis: Association Between SNPs and a Trait

Single-SNP analysis is usually the first step of the analytical process after performing quality control in our dataset. Essentially, it consists on assessing the association between the different genotypes of a SNP and a response variable. This has usually been the most straightforward and computationally feasible type of analysis in association studies. Nonetheless, with the recent advent of more dense platforms such as Affymetrix's SNP array 6.0 or Illumina's Human1M–Duo BeadChip, single-SNP association analysis has also become a challenge both for statisticians and bioinformaticians, specially for those association studies with a large sample size. The vast amount of data generated by these arrays demands good computational and

statistical skills, as well as a computing infrastructure powerful enough to handle it properly. In the following sections, we are going to review which are the different available tests to evaluate association between a SNP and a specific trait. We will divide the different types of analysis in terms of the type of outcome we have: binary, quantitative, or prognosis.

7.4.1 Binary Outcome

Having a categorical outcome is one of the most, or maybe the most, usually found situations in population association studies. More specifically, what we typically have is a binary *case–control* phenotype, with unrelated affected and unaffected samples. Cases and controls are usually matched by third variables such as gender or age. In this case, the most straightforward manner to test the SNP-outcome association can be done based on the 3×2 contingency table (Table 7.2). To test the null hypothesis of no association between genotypes and the response variable we can perform a 2df χ^2 test. Nonetheless, when there are low genotype frequencies in one or more cells, a Fisher's exact test would be desirable.

The model which takes into account the full table with the three genotypes is usually called codominant. Basically, it assumes a phenotypic intermediate effect (but not necessarily half-way) for the heterozygotes compared with the two homozygotes. However, sometimes we will expect our SNPs to follow other inheritance patterns, such as dominant or recessive. To do so, we will rearrange the full contingency table as shown in Tables 7.3 and 7.4.

Analysis of Tables 7.3 and 7.4 can also be done both with a χ^2 test (with 1 df) or a Fisher's exact test, depending on the number of counts found in each cell.

One important fact to take into account is that, when dealing with complex traits, genetic effects of single SNPs will likely be additive instead of dominant, recessive,

Table 7.2 Contingency table with genotype counts for cases and controls

Genotype	Controls	Cases
AA	$n_{AA_{\text{co}}}$	$n_{AA_{\text{ca}}}$
AB	$n_{AB_{\text{co}}}$	$n_{AB_{\text{ca}}}$
BB	$n_{BB_{\text{co}}}$	$n_{BB_{\text{ca}}}$

Table 7.3 Contingency table for the dominant model, with allele B being the risk allele

Genotype	Controls	Cases
AA	$n_{AA_{\text{co}}}$	$n_{AA_{\text{ca}}}$
AB+BB	$n_{AB_{\text{co}}} + n_{BB_{\text{co}}}$	$n_{AB_{\text{ca}}} + n_{BB_{\text{ca}}}$

In this case heterozygous individuals are expected to have the same phenotype as BB homozygotes, so both categories are collapsed into a single one

Table 7.4 Contingency table for the recessive model, with allele B being the risk allele

Genotype	Controls	Cases
AA+AB	$n_{AA_{\text{co}}} + n_{AB_{\text{co}}}$	$n_{AA_{\text{ca}}} + n_{AB_{\text{ca}}}$
BB	$n_{BB_{\text{co}}}$	$n_{BB_{\text{ca}}}$

Heterozygous individuals are only *carriers* of the disease, but they display a nondisease phenotype. Thus, they are merged with the AA subjects, which have a wild-type genotype

or codominant. Additive models assume that heterozygotes risk will be half-way between the two homozygote risks. Since the statistical tests explained above (2df χ^2 and Fisher's exact test) have a reduced power to detect this kind of effects, we need to address this issue and find other ways to detect this additive association. The Cochran–Armitage test (Armitage 1955) is a good model to detect this trend in the proportion of cases for each one of the genotypes. It tests against the null hypothesis of a zero slope for the line that fits the three genotype risks best. Actually, this test corresponds to the *multiplicative* model for effects of alleles on odds scale. An important characteristic of this model is that it does not rely on the assumption of Hardy–Weinberg equilibrium (HWE), so it may be useful in case HWE does not hold for our complete population of individuals (cases and controls altogether).

Compared to the contingency table approach, logistic regression offers a more flexible environment to assess the association between a SNP and a binary outcome. For large sample sizes, the likelihood ratio test of the logistic model against the null hypothesis $\beta_{AA} = \beta_{AB} = \beta_{BB}$ is equivalent to the 2df χ^2 test. However, logistic regression can be extended to further SNPs (epistasis), environmental, or clinical variables, which usually need to be taken into account.

To specify inheritance models in a logistic regression, we just need to restrict the values of the β coefficients. Thus, forcing that $\beta_{AB} = \beta_{BB}$ or $\beta_{AA} = \beta_{AB}$ would test for a dominant and recessive effects, respectively. If we restrict β_{AB} to be half-way between β_{AA} and β_{BB}, then the logistic model will be equivalent to the Cochran–Armitage trend test.

7.4.2 Quantitative Outcome

A typical example of association study with a quantitative outcome is the one where we want to test if the expression value of a gene is affected by the genotype of a specific SNP (which may or may not be located in the same gene). This kind of association may be relevant for diseases with an important genetic basis, such as cancer.

A first and simple approach to assess the degree of association between a SNP and a trait would be to categorize the quantitative response into two classes (e.g., "low value" and "high value"), and then apply one of the approaches described in

Sect. 7.4.1. Nevertheless, this approach is suboptimal, since it would carry a loss of statistical power to detect significant changes between groups. Therefore, instead of that approach, a more natural and optimal model to test association with a quantitative response is to use statistical tests such as ANOVA and linear regression.

While ANOVA model is equivalent to the 2df χ^2 test, linear regression assumes linearity between genotypes and the response means, so the degrees of freedom are reduced to one. Furthermore, both tests require the trait to be normally distributed and with equal variance across all genotypes.

In a similar manner as what is explained in Sect. 7.4.1, inheritance models can also be specified in this case by merging the proper genotypes to generate a dominant, recessive, or additive genetic pattern.

7.4.3 Prognosis Outcome

In the last few years, a vast number of studies have investigated the association between polymorphisms and cancer survival. Some of the more recent findings include studies for breast cancer (Cox et al. 2007; Hunter et al. 2007; McKay et al. 2008; Wang et al. 2008), colorectal (Broderick et al. 2007; Haiman et al. 2007b; Jaeger et al. 2008; Tenesa et al. 2008) or prostate (Haiman et al. 2007a,b; Sun et al. 2008; Witte 2007; Yeager et al. 2007). From a statistical point of view, the Kaplan–Meier estimator is most widely used to estimate the survival function. As an example, we could model the time to develop metastasis after the resection of a primary tumor in terms of the genotype of a specific SNP. To assess the significance of survival differences in different groups, a log-rank test can be used. However, Cox proportional hazards model will allow us to quantify the increase or decrease of risk for each one of the genotypes. Analogously to what is explained in Sects. 7.4.1 and 7.4.2, in this case we can also force our SNPs to follow a specific inheritance pattern.

7.5 Multiple-SNP Analysis

Association studies may not restrict only to single genetic markers, specially when most recent large-scale techniques have broaden the experiments up to more than a million SNPs. Although useful as a first approach to detect potential association with a trait, single-SNP analyses have shown to be somehow inefficient, because they do not integrate information of nearby markers. Since it may be rather unlikely that we have the *real* causative marker genotyped, multiple-SNP associations can provide a great advantage over pointwise estimations.

There are two main approaches to assess multiple marker association: regression and haplotype-based methods. Regression methods are mainly based on logistic or linear models (depending on the type of response we have). Nonetheless, as genotyping densities have dramatically increased over the last few years, correlations

among neighboring SNPs can cause model instability. Backward or forward stepwise procedures may overcome this limitation, but they tend to overfit the observed genotype and phenotype data, making permutation testing procedures necessary to control the type I error rate. Another feasible approach is to select only tag-SNPs (i.e., loci that can serve as proxies for many other SNPs, see Sect. 7.5.2), but at the expense of losing potentially valuable information. This motivates us to focus on haplotype-based approaches, which constitute an attractive alternative. In the following sections, we are going to introduce the basic concepts of haplotype theory, and then we will review haplotype-based association methods.

7.5.1 Introduction to Haplotypes

Haplotypes are combinations of alleles at multiple polymorphic loci along a chromosome. Although an entire chromosome could be seen as a haplotype, usually only regions no longer than 100 kbp with highly linked polymorphisms are considered. Thus, for a given set of markers, each person has two haplotypes, each one inherited from one of the progenitors. As one can easily calculate, a set of n biallelic SNPs generate 2^n potential haplotypes in the population. However, recombination rates commonly make the actual occurring number of haplotypes to be much smaller than this theoretical upper bound.

The usefulness of haplotypes in association studies is justified by several reasons. First, since they are combinations of multiple SNPs, haplotypes have been demonstrated to be more informative than individual markers. Furthermore, haplotype association studies show greater statistical power than single-SNP association analyses (Akey et al. 2001). From a biological perspective, there are evidences that a set of pointwise *cis*-mutations (i.e., located in the same copy of the chromosome) within the same gene can interact to have a greater effect on a subject's phenotype. Despite this, the association of a haplotype with a phenotype does not necessarily mean that the haplotype itself is biologically related to the trait, since it may be possible that an unexplored locus located in the haplotype region was the marker biologically functionally related with the phenotype.

One serious drawback of any analysis involving haplotypic information is that genotyping studies usually generate unphased data. That is, for a given subject we do not really know which alleles come from each one of the progenitors. Laboratory techniques which allow to obtain phase information, such as allele-specific PCR or cloning, are rather expensive and time consuming. Thus, to overcome this lack of information we need a statistical approach that enables us to *infer* haplotypes for a given set of unrelated samples and genotypic markers.

7.5.2 Linkage Disequilibrium, Linkage Blocks, and Tag-SNPs

Linkage disequilibrium (LD) statistics describe the deviation of observed haplotype frequencies from what is expected. Let A and B be two SNPs with alleles A_1, A_2,

B_1, and B_2. Thus, the combination of these SNPs can generate four possible haplotypes: A_1B_1, A_1B_2, A_2B_1, and A_2B_2, with relative frequencies $f_{A_1B_1}$, $f_{A_1B_2}$, $f_{A_2B_1}$, and $f_{A_2B_2}$, respectively. The basic statistic to assess the LD between both markers, named D, is defined as follows:

$$D = f_{A_1B_1} - f_{A_1} \cdot f_{B_1} \tag{7.1}$$

D equals to 0 in the case of *complete equilibrium*. Positive D values indicate that A_1 and B_1 tend to appear together more than expected by chance, while negative values would indicate the opposite. A major inconvenient with the D statistic is that its range depends on the MAF of the two SNPs, making it desirable to find a measure with a standardized range. Thus, a normalized version of D, called D', is defined as:

$$D' = \frac{D}{D_{\max}} \tag{7.2}$$

where

$$D_{\max} = \begin{cases} \frac{D}{\min(f_{A_1} f_{B_1}, f_{A_2} f_{B_2})} & \text{if } D > 0 \\ \frac{D}{\min(f_{A_1} f_{B_2}, f_{A_2} f_{B_1})} & \text{if } D < 0, \end{cases} \tag{7.3}$$

D' ranges from -1 to 1, and usually takes extreme values when allele frequencies are small. If $D' = 1$ or $D' = -1$, it means there is no evidence for recombination between the two markers. Moreover, if allele frequencies are similar, high D means the SNPs are good surrogates for each other. Nonetheless, this statistic has an important drawback, which is that it is inflated for small sample sizes or when one allele is rare. Therefore, another measure based on the correlation between alleles, called r^2, can be defined as follows:

$$r^2 = \frac{D^2}{f_{A_1} \cdot f_{A_2} \cdot f_{B_1} \cdot f_{B_2}}, \tag{7.4}$$

r^2 ranges from 0 (i.e., perfect equilibrium) to 1 (i.e., both markers provide identical information), and its expected value is $1/2n$. It has become one of the most used statistics to assess LD between pairs of markers.

Comparison of haplotypes and the scope of LD across individuals allows us to identify segments or haplotype blocks that correspond to minimal units of recombination. Usually, one or few alleles within these haplotype blocks will be predictive of the other alleles. This predictive SNPs are called *tag-SNPs*. Therefore, GWAS can be accomplished by genotyping a collection of tag-SNPs which define the haplotype blocks along the complete genome. As an example, this is the approach followed by Illumina's BeadChip technology.

7.5.3 Haplotype Inference

In the last two decades, several methods of haplotypic reconstruction have been developed in order to solve the problem of haplotype inference. Since Clark, in 1990, developed a parsimony algorithm to estimate haplotype frequencies from a sample of genotypes, quite a large number of methods have been developed (Clark 1990). Most of them rely on the use of different techniques to calculate the Maximum Likelihood Estimator (MLE).

In 1995, Excoffier and Slatkin (1995) adapted the Expectation-Maximization algorithm, an iterative algorithm of maximization developed by Dempster in 1977 to maximize the likelihood function of the haplotypes given the genotypes at specific loci (Dempster et al. 1977). This method has some limitations and convergence to a local maximum may occur in some situations (Celeux and Diebolt 1985).

Some authors have attempted to minimize these limitations in their works, like Qin et al. (2002) using *Divide and conquer* strategies, or David Clayton, implementing an EM-algorithm which adds SNPs one by one and estimates haplotype frequencies, discarding haplotypes with low frequency as it progresses. In the context of Bayesian statistics, Stephens et al. in 2001 proposed an algorithm based on coalescent theory with a especial prior based on the general mutational model (Stephens et al. 2001). Niu et al. (2002) implemented another Bayesian approach using a Markov Chain Monte Carlo method. In general, algorithms dealing with Bayesian models are suitable to infer haplotypes from genotypes having a large number of polymorphisms. More recent methods work with clusters of haplotypes in order to avoid the major limitations of many current haplotype-based approaches (Waldron et al. 2006).

Once the frequencies have been estimated by any of the methods mentioned above, the next goal is to test the association between haplotypes and the disease. The most accurate strategy in order to take into account the uncertainty of the sample is to estimate simultaneously haplotype frequencies and haplotype effects. Some works are focusing on this approach (Iniesta and Moreno 2008; Tanck et al. 2003; Tregouet et al. 2004).

7.5.4 Haplotype Association with Disease

Since haplotypes capture variation in small genomic regions, the analysis of association between haplotypes and disease is a potentially more powerful way to identify cancer genes when the causal variant is unknown (Akey et al. 2001). Haplotypes inferred from a series of SNPs should also capture variation in VNTR and CNVs.

From an analytical point of view, the possible haplotypes at a given region conform a categorical variable that can be analyzed in regression models when appropriately coded with indicator variables. There are a few technical difficulties with this analysis. First, haplotypes are inferred from genotypes, as we have seen previously, and for subjects heterozygous at more than two SNPs there is uncertainty

about the pair of haplotypes. A similar problem arises when one or more genotypes are missing. For these cases, the inference algorithm used provides a posterior probability for each compatible combination. These probabilities can be used as weights in a regression model to transfer the uncertainty in the haplotype estimation to the estimates of association. This is the method used in haplo.stats software (see Sect. 7.8.2.3). The second problem is the inheritance model. Since each subject carries two haplotypes (though in the dataset is further expanded to account for uncertainty), the most frequent inheritance model used is the log-additive, where the risk for each haplotype is compared to the one selected as reference in logistic regression. Usually the most frequent haplotype is this reference. The odds ratio estimates obtained should be interpreted as per-haplotype relative risks, similar to the per-allele relative risk in a log-additive model for genotypes. There is a possibility to encode the haplotypes to model dominant or recessive effects (Iniesta and Moreno 2008), but the interpretation is not as simple and for the recessive effects the power is generally very limited. A final consideration in these analyses is the treatment of rare haplotypes (i.e., those with an observed frequency lower than a previously defined threshold – usually 0.01%). The inference process usually results in a series of haplotypes with very low inferred frequency in the studied population. If these rare haplotypes are considered in the logistic regression, the model becomes unstable because most probably these haplotypes have only been observed or inferred for cases or for controls, and the standard errors of the regression coefficients become very large. The typical solution is to pool these rare haplotypes into a common group that is not interpreted. If the cumulative frequency of these rare haplotypes is not high and the reference category is large enough, a better option might be to pool them into the reference category. With this method, 1 degree of freedom is gained for the test of global association between the haplotypes and the disease.

The analysis of haplotypes can also be useful to identify genes associated to prognosis of cancer. For this analysis, if a Cox proportional hazards model is desired, the combined estimation of haplotypes and regression parameters is more difficult computationally due to the semiparametric nature of the Cox model. This analysis, could be approached within a Bayesian framework.

7.6 Genome-Wide Association Studies

As already mentioned in Sect. 7.2, candidate gene association studies are designed to assess association between a moderate number of SNPs and disease. These kind of studies can be viewed as hypothesis-based studies in which the a priori knowledge of the disease plays an important role. Another reason for adopting candidate gene approach is the low costs of genotyping since only a moderate number of markers have to be genotyped. The price we have to pay, however, is that only those genes with known functional impact are included in the analysis. The continuous improvements in genotyping technologies and, above all, their decreasing cost has made it possible to perform GWAS where the entire human genome is interrogated. GWAS

use large-scale genotyping technologies to assay hundreds of thousands of SNPs and relate them to clinical conditions or measurable traits. One of the main strength of a GWAS is that they are unconstrained by prior hypotheses with regard to genetic associations with disease (Hirschhorn and Daly 2005). Recently, there has been a vast number of GWAS to determine new susceptibility locus contributing to complex diseases such as Alzheimer (Beecham et al. 2009), cancer (Easton and Eeles 2008), Schizophrenia (O'Donovan et al. 2009), or Parkinson (Pankratz et al. 2009) as well as quantitative traits such as metabolic traits (Sabatti et al. 2009) or serum IgE (Weidinger et al. 2008), among others. A remaining obstacle of GWAS is that a massive number of statistical tests is performed. This may lead to a huge number of false-positive results making necessary to adopt further statistical corrections in the assessment of association or replication.

In this section, we will give an overview about GWAS, including study designs and statistical tests. We will also illustrate how to determine the statistical power and suitable sample size of our study, as well as addressing the multiple comparisons problem.

7.6.1 Study Designs

GWAS can be defined as the study of common genetic variations across the entire human genome. They are designed to identify associations with observable or quantitative traits (such as blood pressure or weight), or the presence or absence of a disease or condition (i.e., discrete traits). Depending on the nature of the trait (e.g., quantitative or discrete) the study design may differ.

The mostly used design for GWAS has been, so far, the case–control design, which has been frequently used in traditional epidemiological studies. They are less expensive to conduct than cohort studies, and genetic studies can be easily incorporated. This is possible because many epidemiological studies collect blood samples at the beginning of the study that afterward can be used to perform genetic tests. The aim of genetic case–control studies is to compare allele (or genotype) frequencies in diseased individuals with frequencies in healthy controls. The only difference between GWAS and other genetic association studies is simply the number of SNPs analyzed. Their characteristic limitations are those of case–control studies such as recall, selection, and confounding bias. Recall bias arises when cases report their exposure history in a different manner than controls. This is not a problem when assessing genotype–phenotype associations, because genotypes (i.e., *exposure*) are measured from DNA samples. Nonetheless, this may be relevant when studying gene–environment (G×E) interactions. Furthermore, in some occasions DNA collections may differ with regard to storage, technicians, or genotyping methods that could induce to some systematic bias (Clayton et al. 2005). On the other hand, selection bias occurs when controls do not come from the same population as cases. In this case, genetic or environmental background may differ as a result of the study design and not due to genetic differences. This can be a concern in studies that use

controls who were selected and genotyped for a previous study (Consortium 2007). Finally, confounding bias occurs when a risk factor for disease is also associated with the marker. Some authors stated that genetic association studies are protected against this bias since genotypes at a given locus are independent of most environmental factors (Clayton and McKeigue 2001). In genetic association studies, there is a special case of confounding bias known as *population stratification*. This situation appears when both disease and allele frequencies are correlated across ethnicity. This difficulty may be overcome either in the study design or in the analysis process. When designing the study, one may select controls matched by ethnicity with cases or select controls from the same family as cases (paired design). However, if this matching cannot be performed, population stratification needs to be addressed at the analytical stage. There are several procedures for addressing population stratification from a statistical point of view. Some of them are based on correcting the test statistics used to assess association by computing an inflation parameter, while others try to find the underlying structure of the data and its variability and incorporate it into the analysis.

Population stratification inflates chi-square values when assessing association among markers and disease. After estimating the inflation parameter, λ, one can correct the chi-square test of association dividing it by λ. Different methods exist to estimate λ. One of them, known as *genomic control* (Devlin and Roeder 1999), uses the variance inflation factor to correct for the variance distortion estimates. In this case, the inflation parameter can be estimated as:

$$\hat{\lambda} = \frac{\text{median}\left(\chi_1^2, \chi_2^2, \ldots, \chi_M^2\right)}{0.466},$$

where $\chi_1^2, \chi_2^2, \ldots, \chi_M^2$ are the chi-square test statistics for the M markers analyzed. Another approach, known as *delta centralization* (Gorroochurn et al. 2006), centralizes the noncentral chi-square distribution of the test statistic. In the presence of population stratification, the test statistic used for assessing association follows a noncentral chi-square distribution with noncentrality parameter δ^2. In this case, δ^2 is used to correct the test statistics as λ (Gorroochurn et al. 2006). Finally, Clayton et al. (2005) uses the relationship between observed and expected test statistics for disease association. The tests are ranked from smallest to largest and plotted against their expected order statistics under the global hypothesis of no association. Under no population stratification the points should be in the diagonal line. The authors estimate λ, by calculating the ratio between the mean across the smallest 90% of observed test statistics and the mean of the corresponding expected values. Ninety percent of tests are considered because it is expected to have a little proportion of SNPs that are truly associated with the disease (i.e., they do not hold null hypothesis) for which the observed test statistic should be inflated. Nevertheless, methods based on adjusting association statistics at each marker by uniform overall inflation factor may be insufficient for markers having unusually strong differentiation across different populations, leading to a loss in power (Price et al. 2006).

Another approach is based on finding structured associations with the program STRUCTURE (Pritchard et al. 2000). This method assigns individuals to discrete subpopulation clusters and then aggregates evidence of association within each cluster. It produces highly accurate assignments using few loci. One limitation of STRUCTURE, though, is that it is computationally intensive and cannot be applied to the whole set of SNPs comprised in a GWAS. However, this method has been demonstrated to work reasonably well only with a subset of the SNPs. Furthermore, the assignment of individuals to clusters is very sensitive to the number of clusters, which is not well defined (Price et al. 2006). A better alternative is proposed in Price et al. (2006). The authors proposed a method called EIGENSTRAT consisting of three steps. First, PCA of genotype data is applied to infer continuous axes of genetic variation. They show that the top two axes of variation describe most of the observed variability and can be used to correct for population stratification. Second, they adjust genotypes and phenotypes by amounts attributable to unobserved population (e.g., ancestry) along each axis. Finally, in the third step, association statistic is computed using population-adjusted genotypes and phenotypes. This last step can be performed by fitting a logistic regression model adjusted by the first (generally the two first) principal components. One of the main advantages of using continuous measurements for adjusting population stratification is that it provides a better description of genetic variation among individuals. Another important point is that EIGENSTRAT is computationally tractable on a genome-wide scale.

To illustrate how EIGENSTRAT works, we use HapMap samples (see Sect. 7.8.3.2 for more details). We randomly selected a set of 9,307 SNPs from the entire genome for 270 individuals from different CEPH (Utah residents with ancestry from northern and western Europe – abbreviated as CEU), subjects with African origin (Yoruba in Ibadan – YRI) and Asian individuals with Japanese (JPT) or Chinese (CHB) origin. Figure 7.3 shows the two first components (axes) of variation and the position for each individual. We observe that the first component reflects genetic variation between YRI and CEU plus CHP+JPT populations, while the second component separates CEU and CHB+JPT populations. This example illustrates how well EIGENSTRAT is able to capture the genetic difference among individuals of different populations. If we are then interested in assessing association between two groups of individuals we will use a logistic regression model (see Sect. 7.4.1) adjusted by subject score components (loading values). This incorporates genetic differences among individuals due to ancestry correcting population stratification.

The often abbreviated description of participants and lack of comparison of key characteristics can make evaluation of potential biases and replication of findings quite difficult, as described in Chanock et al. (2007). To overcome the difficulties typical of case–control studies, other different designs based on trios or in cohort studies can be adopted. The trio design includes affected case individuals and both parents (Spielman et al. 1993). Classification of affected status is given only in the offspring and only affected offspring are included, but genotyping is performed in all three trio members. The frequency with which an allele is transmitted to an affected offspring from heterozygous parents is then estimated (Spielman et al. 1993).

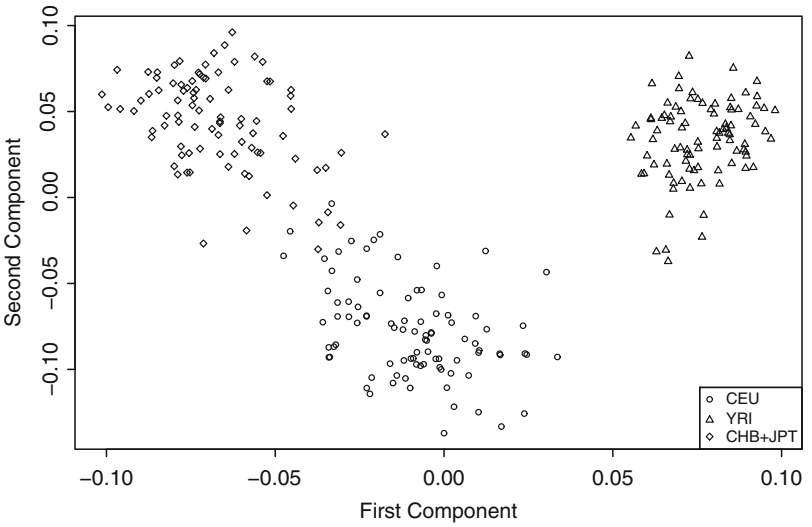

Fig. 7.3 The top two components (axes) of variation of HapMap samples obtained by using EIGENSTRAT approach. Data corresponds to 270 individuals from European (CEU), Yoruba (YRI), and Chinese plus Japanese (CHB+JPT) populations. This example analyze 9,307 randomly selected SNPs from the entire genome

Under the null hypothesis of no association with disease, the transmission will be 50%. Therefore alleles associated with the disease will be transmitted in excess to the affected case individuals. Cohort studies collect baseline information in a large number of individuals, who are then assessed for the appearance of the disease over time depending on different genetic variants. These studies are normally more expensive, but the results are often more representative than case–control studies. Consequently, GWAS have been recently performed using cohort studies such as the European Prospective Investigation on Cancer (EPIC) (McKay et al. 2008), Study French Prostate Cancer Study (CeRePP) and MultiEthnic Cohort Study, and Physicians' Health Study (Yeager et al. 2007). One of the main disadvantages of cohort studies is the large investment required both in time and money. A large cohort needs decades of follow-up time to achieve the number of cases required to detect moderate genetic effects. However, if the cohort is established, genetic association studies can be performed using a nested case–control strategy, where observed cases and a subset of appropriately matched controls are genotyped, rather than the entire cohort (Langholz et al. 1999).

Most of GWAS studies also adopt a multistage design, mainly aimed to reduce the number of false-positive results while minimizing the number of genotyped individuals and keeping statistical power (Hirschhorn and Daly 2005). In practice, analysis of GWAS is often performed in two (or sometimes even more) stages. First, at step 1, the full marker set is genotyped for an initial group having a moderate number of cases and controls. Then, at stage 2, only the most promising markers (e.g., SNPs that have been statistically significant associated with the trait in the

first step) are regenotyped for another group using smaller SNP arrays. The number of SNPs and individuals included in these consecutive steps may vary depending on budget. The question of what significance threshold is appropriate for GWAS studies is somewhat unresolved. In Sect. 7.6.4, we outline some approaches that are currently adopted.

Pahl et al. (2009) pointed out that two-stage designs can be seen as a special case of general group sequential designs used in clinical trials, consisting of a single interim analysis (stage 1) and a final analysis (stage 2). They extend the special case of two-stage designs to the general framework of multistage designs, deriving optimal multistage designs with respect to both minimal overall study costs and maximal study power. The authors concluded that economical benefits can be obtained by using more than two stages. In particular, they found that using no more than four stages is sufficient for practical purposes (Pahl et al. 2009).

7.6.2 Assessing Association in GWAS

After designing the study and obtaining genotype information, we have to assess association between the variants and the disease. The significance of association between a marker and disease is done by analyzing each SNP at time, determined by calculating any of the test statistics described in Sect. 7.4. For autosomal SNPs, the test statistic can be based on assuming dominant, recessive, and additive models. This means that we have to perform three times the number of SNPs tests. Considering that currently Affymetrix or Illumina platforms are able to genotype 1 million SNPs, 3 million of tests are needed to be computed. One may be tempted to avoid performing such a high number of tests by calculating the most powerful test (additive model) or Armitage's trend test (Freidlin et al. 2002; Slager and Schaid 2001). However, assuming a model different from the real one leads to loss of power (Freidlin et al. 2002; Schaid et al. 2005; Slager and Schaid 2001). Therefore, when the underlying genetic model is unknown, association may be assessed using the max-statistic, which selects the largest test statistic from the dominant, recessive, and additive models (Gonzalez et al. 2008). This statistic is written as:

$$\chi^2_{\mathrm{MAX}} = \max\{\chi^2_{\mathrm{DOM}}, \chi^2_{\mathrm{REC}}, \chi^2_{\mathrm{ADD}}\}. \tag{7.5}$$

A naive approach to determine whether a given marker is associated with the disease using max-statistic is to consider the smallest p value between dominant, recessive, and additive tests (Gonzalez et al. 2008). This approach does not maintain the overall type I error rate since it does not account for either multiple testing or correlation among the three tests as showed by several authors (Freidlin et al. 2002; Gonzalez et al. 2008; Schaid et al. 2005; Slager and Schaid 2001). Hence, the statistical significance of association using max-statistics has to be addressed with other methods. Sladek et al. (2007) consider χ^2_{MAX} test to identify novel risk variants for type 2 diabetes. The authors stated that as the distribution of max-statistic

is unknown, a permutation approach can be used to estimate statistical significance. This procedure is extremely expensive computationally for GWAS. For instance, Sladek et al. (2007) needed to calculate around 11,800 million tests (only in the first stage) for 392,935 markers and three inheritance models performing 10,000 permutations to compute p values for the max-statistic. Gonzalez et al. (2008) derived the asymptotic form for max-statistic that can be used to compute the correct p value when it is used. The authors also found through simulations studies that the effective number of tests when dominant, recessive, and additive models are fitted is 2.2. This number can be used to correct the significance level or p values as a rule of thumb. Since this value is based on simulations, large sample sizes might be required.

7.6.3 Statistical Power Calculations

As in many other situations, when a GWAS is performed, one can be interested in estimating the probability that the statistical test used to assess association yields significant results when the null hypothesis is false (e.g., power). In other words, we would like to know the chance that the study will be successful in detecting a true effect. Power calculations are normally performed during the planning stages of a study. However, in some occasions, they can also be used to interpret negative results. Herein, we are describing the fundamentals of statistical power for genetic case–control studies. We will also illustrate how to perform power calculations in the context of GWAS, where other issues such as multiple testing, coverage, and staged designs need to be considered.

Power in association studies depends on a number of factors, such as the total number of available samples, the size of the genetic effect, its mode of inheritance, the prevalence of the disease, the ratio of case to control individuals, allelic frequencies, and the extent of linkage disequilibrium between marker and trait loci (Cai and Zeng 2004; Purcell et al. 2003). It is possible to obtain closed-form expressions to compute statistical power for genetic associations (Schork 2002). These formulas can be used to calculate expected power under a range of scenarios. For instance, Table 7.5 displays the effect of varying sample size, linkage disequilibrium, or disease allele frequency on the power to detect association under a case–control setting. Power of case–control studies changes as a function of linkage disequilibrium (Sect. 7.5.2), as can be seen in Table 7.5. Values of $D' = 1$ indicate an absence of ancestral recombination between marker and disease loci and thus complete disequilibrium. In contrast, $D' = 0.8$ indicates independence between marker and trait loci. In this case, we can observe that power to detect association is greater when linkage disequilibrium is high, as well as when trait and marker loci have similar allele frequency. As a final conclusion of these examples, we can also observe how power to detect association is strongly related to allele frequency.

Power of two-stage GWAS depends on the same factors as case–control studies. Additionally, these studies also depend on how markers are selected for being analyzed in the second stage, how samples are divided between stages 1 and 2, and the

Table 7.5 Power calculation for a case–control study varying sample size, linkage disequilibrium (D'), and allele frequency

	Sample size					
	100	200	500	1,000	1,500	2,000
$D' = 1$						
0.1	0.24	0.42	0.79	0.98	1.00	1.00
0.2	0.19	0.33	0.67	0.92	0.99	1.00
0.3	0.15	0.25	0.53	0.82	0.94	0.98
0.4	0.12	0.19	0.40	0.67	0.84	0.92
0.5	0.09	0.14	0.28	0.49	0.66	0.79
$D' = 0.9$						
0.1	0.20	0.36	0.71	0.94	0.99	1.00
0.2	0.16	0.28	0.58	0.86	0.96	0.99
0.3	0.13	0.21	0.45	0.74	0.89	0.96
0.4	0.11	0.16	0.33	0.58	0.76	0.87
0.5	0.09	0.12	0.24	0.42	0.58	0.70
$D' = 0.8$						
0.1	0.17	0.30	0.61	0.89	0.97	0.99
0.2	0.14	0.23	0.49	0.78	0.92	0.97
0.3	0.11	0.18	0.38	0.64	0.81	0.91
0.4	0.09	0.14	0.28	0.49	0.66	0.78
0.5	0.08	0.11	0.20	0.35	0.48	0.60

proportion of markers tested in stage 2. Power also depends on the significance level we consider for the entire genome α_{genome}. Table 7.6 shows the obtained power for a hypothetical two-stage GWAS where a case–control study was used including 1,000 cases and 1,000 with a prevalence of the disease equal to 0.1, the allele frequency equal to 0.4, $\alpha_{\text{genome}} = 0.05$, and 300,000 markers ($M$). The table presents different scenarios by varying the proportion of individuals genotyped at first step, and the proportion of markers tested in second stage. Following the recommendations given by Skol et al. (2006), we also present the power for the joint analysis (e.g., joint analysis of data from both stages) that is expected to be more efficient. We observe that the power for two-stage design increases as the number of individuals genotyped in the first stage decreases. For example, two-stage design has 31% power to detect association in the case of analyzing 50% of cases in the first step and genotype 10% of SNPs in the second phase (30,000 markers), while the power is 61% when 20% of cases are analyzed in the first step. We also notice that joint analysis can achieve nearly the same power as the one-stage design in which all samples are genotyped on all markers.

7.6.4 Statistical Level Correction for Multiple Testing

It is well known that multiple testing problem arises when many hypotheses are tested simultaneously using the same data because some test statistics can be

Table 7.6 Power calculation for a two-stage GWAS varying the percentage of individuals ($\%n$) to be genotyped at stage 1, the percentage of markers ($\%M$) to be re-genotyped in the second stage

			Power		
$\%n$	$\%M$	OR	Step 1	Step 2	Joint
1.00	1.00	1.40	1.00	0.00	0.74
0.50	0.10	1.40	0.99	0.31	0.74
0.50	0.05	1.40	0.99	0.36	0.74
0.50	0.01	1.40	0.94	0.48	0.74
0.40	0.10	1.40	0.98	0.46	0.74
0.40	0.05	1.40	0.96	0.50	0.74
0.40	0.01	1.40	0.87	0.58	0.71
0.30	0.10	1.40	0.94	0.57	0.73
0.30	0.05	1.40	0.90	0.60	0.71
0.30	0.01	1.40	0.74	0.58	0.63
0.20	0.10	1.40	0.84	0.61	0.67
0.20	0.05	1.40	0.75	0.58	0.62
0.20	0.01	1.40	0.52	0.45	0.47

The results are computed assuming a case–control design with 1,000 cases and 1,000 controls, where the prevalence of the disease is 0.1, the allele frequency is equal to 0.4, the α_{genome} is 0.05, and 300,000 markers (M) are analyzed. Table also shows the power of a joint analysis test

extreme even if no association exists. Multiple correction procedures are designed to control the set of hypotheses and to prevent false-positive conclusions that could be attributed to chance. Correcting for multiple comparisons requires determining a threshold for which p values are considered as statistically significant. There are several approaches to establish such threshold.

The simplest one is based on controlling the family-wise error rate (FWER), defined as the probability of committing at least one type-I error. FWER can be controlled in a weak sense by using procedures such as Bonferroni or Sidak corrections or in a strong sense by considering that any subset of hypothesis is true (Hoh and Ott 2003). On one hand, Bonferroni correction for multiple testing simply requires a p value of α/M, α denoting the desired nominal level which normally is set equal to 0.05 and M is the number of genotyped SNPs. On the other hand, Sidak's correction needs a p value of $1-(1-\alpha)^{(1/M)}$, which is similar to Bonferroni's.

By using these corrections, a GWAS including 1,000,000 SNPs will lead to a Bonferroni significance threshold of 5.0×10^{-8} and 5.12×10^{-8} using Sidak's formula. These assumptions are too conservative, which may lead to a high false-negative rate (Dudbridge and Gustano 2008). The main problem of using a FWER correction in GWAS is that, by applying it, we are assuming that all markers are independent. This hypothesis makes sense in the context of targeted studies, where SNPs are selected to cover a given region. However, genome-wide scans includes

SNPs that are in a strong LD, making the independent assumption too restrictive. In these situations, permutation test can be used to estimate the "effective number of tests" (Churchill and Doerge 1994). FWER can then be applied to correct for multiple comparisons using the effective number of tests.

Due to linkage disequilibrium between SNPs, FWER control may be too conservative for GWAS, where the goal is to screen the genome for a further study of very few promising candidates (e.g., replication, fine mapping, functional studies, ...). As a consequence, several authors proposed other methods for false-positive rate control, such as false discovery rate (FDR) (Benjamini and Hochberg 1995), posterior error rates (Manly et al. 2004; Wacholder et al. 2004), or permutation testing. Dudbridge et al. (2006) pointed out that FDR is not appropriate for association studies and that other methods such as permutation approach based on minimum p value should be employed.

Permutation tests are based on computing significance levels by estimating the underlying null distribution when theoretical results do not hold. The unknown null distribution is estimated by simulating data in a given manner. Before addressing the problem of how to compute the corrected p value in a GWAS using permutation procedure, let us start by illustrating how to use it in a single analysis where association between trait and a given SNP is assessed. For the benefit of simplicity, let us assume that our trait is dichotomous (case–control), and that we are interested in comparing the proportion of carriers of a rare allele (e.g., assume a dominant model) between the two groups of individuals. As mentioned in Sect. 7.4.1, the null hypothesis of no association can be addressed by using the χ^2 test. In general, the test is based on comparing the proportion of cases who carry the susceptibility allele with the proportion of controls who do not. In the case of having association, we will observe a large value for χ^2 statistic. That is, far away from the null hypothesis of no association where χ^2 is equal to 0. Following theoretical results, the significance of this observed statistic is computed by using a χ^2 distribution with 1 degree of freedom. If this distribution cannot be assumed by any reason, the significance has to be computed by estimating the distribution of the test statistic under the null hypothesis. Under these circumstances, permutation can be used as following. Case–control labels are exchanged among individuals by keeping genotypes fixed and test statistic is then computed. It is expected that after random assignment of case and control status the test statistic would be close to the null hypothesis (e.g., near 0). This procedure is repeated B times and the significance level is the proportion of replicate statistics exceeding the observed statistic (Welch 1990). If there is no association, we expect to have the χ^2 value as one of those obtained by permutating data and a large p value, meaning that the null hypothesis of no association cannot be rejected. On the other hand, if the variant is related to the disease, the number of χ^2 values larger than that obtained by analyzing the observed data will be low. In this case the permuted p value will be lower that the nominal level, rejecting the null hypothesis and concluding that there is association between the marker and the disease.

This permutation procedure can be extended in GWAS to compute a corrected p value. By permuting data, we are able to capture the correlation among markers. Table 7.7 shows the main steps we have to perform for obtaining the corrected

Table 7.7 Steps to correct nominal level by using permutation approach in GWAS

Repeat B times:
1. Randomly assign traits among samples, while keeping the genotype fixed
2. Compute the p value for each SNP by using any selected test (e.g., log-additive, max-statistic, . . .)
3. Keep the minimum p value
The corrected significance level is estimated by selecting the 5% quantile of the replicate minimum p values
or
Estimate Beta$(1, n_E)$ by using the replicated minimum p values and computing its 5% quantile

level of significance by permutation testing. As in the single case, we first randomly assign case and control labels to individuals by keeping genotypes fixed. Then we compute p values of association for each SNP using a statistical test (i.e., dominant, log-additive, max-statistic, . . .). For each permutation, we retain the minimum p value obtained among all SNPs analyzed. The corrected significance level is computed as the 5% quantile point of the empirical distribution for the minimum p values. Alternatively, one can assume that the minimum p value follows a Beta distribution with parameters $(1, n_E)$, n_E being the number of effective tests (Dudbridge and Koeleman 2004). One can fit the Beta distribution to the minimum p value of the permutation replicates and then estimate the 5% quantile point from this theoretical distribution. Using the approximation to a Beta distribution, a moderate number of permutations (e.g., 10,000) can be enough to correct estimate the corrected nominal level. This approach has also another important advantage. Let us assume that we are in the context of performing a GWAS, where several diseases are analyzed by using a shared group of controls like in the Wellcome Trust Case Control Consortium study (Consortium 2007). In this case, for each analysis, we should have to perform a permutation analysis for each disease. However, Dudbridge and Koeleman (2004) pointed out that this approach needs to be done once only because the distribution of p values under the null hypothesis is the same in all studies. Using Beta approximation, we can also test whether the minimum p value is consistent with an effective number of independent tests, by testing whether the first parameter is 1(Dudbridge and Koeleman 2004).

To illustrate how the permutation approach works, we used 9,307 SNPs from the HapMap (Hap 2003) project randomly selected from the entire genome. We compared genotype frequencies between European (CEU) and Yoruba (YRI) populations. The corrected nominal level by using Bonferroni correction is equal to 5.37×10^{-6}. However using the distribution of the minimum p value we obtained a corrected p value of 2.32×10^{-5} (Fig. 7.4). The p value from the empirical distribution is 2.22×10^{-5}. This shows an excellent agreement between the empirical and theoretical distributions of minimum p values, as expected. Notice that, using this permutation approach, we obtained a not so stringent significance level leading to an increase in power. This procedure is implemented in SNPassoc software (see Sect. 7.8.1.2).

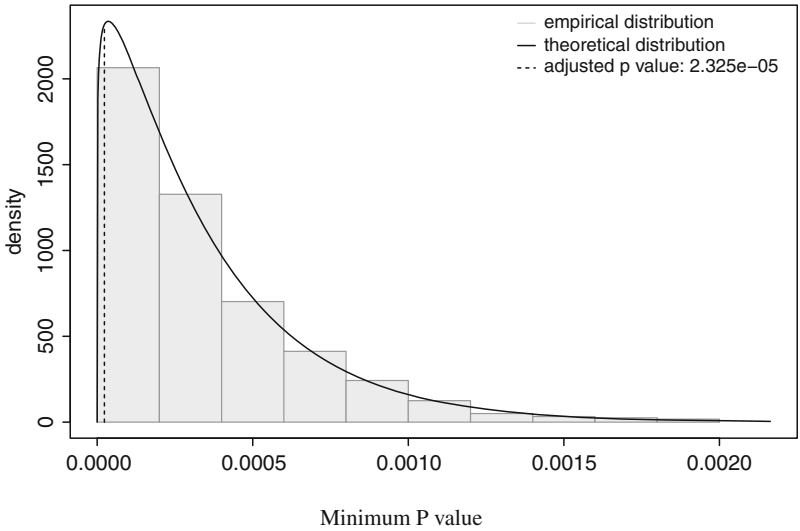

Fig. 7.4 Empirical (*histogram*) and beta distributions (*straight line*) of minimum p values obtained from the permutation procedure applied to 9,307 randomly selected SNPs from HapMap project. The p values are obtained by assessing differences between CEU and YRI populations assuming an additive model. The results are based on 10,000 permutations

7.7 Gene–Gene and Gene–Environment Interactions

The analysis of interactions involves the assessment of an effect modification: the risk associated with a genotype varies according to a third variable, which can be an environmental exposure or other genetic factors. The assessment is a comparison of the effects among subgroups of the data. Usually regression models are used to test for interactions. In these models, next to main effects (gene and environment), an additional term corresponding to the product of the main effects is added.

The coefficient for this product term, if zero, is interpreted as no interaction: gene and environment are independent and act additively in the model scale. If the coefficient is different from zero, it is interpreted as a difference in the effect for the combination of gene and environment with respect to the expected under independence. A positive coefficient means synergism: the combined effect is larger than expected. A negative coefficient means antagonism or that the combined effect is not larger than the sum of the main effects. Usually, when this analysis is performed for a case–control study, logistic regression is used and the interactions must be interpreted in a multiplicative scale. A negative interaction coefficient may mean that the combined effect is additive and not multiplicative, as the logistic regression imposes.

Power to detect interactions is smaller than power to detect main effects because it is effectively a comparison between two (or more) risk estimates. Power also depends on the degrees of freedom used to test the interaction. For this reason, it is

usually desirable to define a binary environmental factor and collapse the genotypes into an inheritance model with only 1 degree of freedom (dominant, recessive, or log-additive).

One strategy to increase the power to detect gene–environment or gene–gene interactions is to use a case-only analysis (Piegorsch et al. 1994). The association between gene and environment when the sample is restricted to cases only estimates the interaction under the condition that there is no association between the gene and the environment in controls and it is important to recognize this requirement (Albert et al. 2001). Recent developments in this methodology allow using the advantages of the case-only design when the independence assumption is met and use the complete dataset otherwise in a weighted empirical-Bayes approach (Mukherjee and Chatterjee 2008).

7.8 Bioinformatics Tools and Databases for Genetic Association Studies

In previous sections, we have described the whole process of a population-based genetic association analysis, from initial quality control to correction for multiple testing in GWAS. As it can be easily seen, the illustrated statistical analysis is complex and composed of many different parts. Thus, it is difficult and time consuming to perform a complete association analysis using only general-purpose statistical suites (e.g., R, SAS). Fortunately, there are plenty of bioinformatics tools that will allow researchers to successfully complete the whole analysis without the need of a deep knowledge of computing or bioinformatics. Before using these tools, however, we need to be careful about choosing the right type of analysis for our data, so that we avoid any potential errors and take the most advantage of our data.

In the next sections, we will briefly list some of the most used software tools. For a better clarity, they have been classified according to their main functionalities.

7.8.1 Genetic Association Suites

These pieces of software deal with most of the analytical steps explained. They are suitable tools for quality control, single-SNP association, or haplotype analysis, and in the case of PLINK it can handle GWAS datasets.

7.8.1.1 SNPStats

SNPStats (Sole et al. 2006) is an easy and ready-to-use Web tool for association analysis, specially designed for small to mid-size studies (i.e., up to a thousand

of SNPs approximately). It can perform quality-control procedures (genotyping rate, HWE, allele and genotype frequencies), can perform analysis of association with a response variable based on linear or logistic regression, accepts multiple inheritance models (e.g., codominant, dominant, recessive, overdominant and log-additive), and can perform analysis of gene–gene or gene–environment interactions. If multiple SNPs are selected, SNPStats offers the possibility to compute LD statistics between SNPs, haplotype frequency estimation, and association with the response and analysis of haplotype–environment interactions. Web site: http://bioinfo.iconcologia.net/snpstats

7.8.1.2 SNPassoc

The R package SNPassoc (Gonzalez et al. 2007) is useful to carry out most common parts of a GWAS analysis in an efficient manner. These analyses include descriptive statistics and exploratory analysis of missing values, calculation of Hardy–Weinberg equilibrium, analysis of association based on generalized linear models (either for quantitative or binary traits), and analysis of multiple SNPs (haplotype and epistasis analysis, p value correction for multiple testing). Permutation tests and other related tests (sum statistic and truncated product) are also implemented. Compared to other R packages with similar purposes, SNPassoc offers a greater usability. Web site: http://www.creal.cat/jrgonzalez/software.htm

7.8.1.3 PLINK

PLINK (Purcell et al. 2007) is a free, open-source GWAS toolset, designed to perform a complete range of basic, large-scale analyses in a computationally efficient manner. The focus of PLINK is purely on analysis of genotype/phenotype data, so there is no support for steps prior to this (e.g., study design and planning, generating genotype or CNV calls from raw data). PLINK is designed as a command-line tool, but through its recent integration with a JAVA graphical interface (called gPLINK) and Haploview, there is some support for the subsequent visualization, annotation, and storage of results. Web site: http://pngu.mgh.harvard.edu/∼purcell/plink/

7.8.1.4 GAP

Genetic Analysis Package (GAP) (Zhao 2007) is implemented as a package for R. It has functions for Hardy–Weinberg equilibrium tests, measures of linkage disequilibrium between SNPs or multiallelic markers and haplotype analysis. It is also useful for two-stage case–control power calculations. Web site: http://www.mrc-epid.cam.ac.uk/Personal/jinghua.zhao/r-progs.htm

7.8.2 Haplotype-Only Software

Haplotype-related analysis is an important part of association studies. Thus, there are some tools focused specifically on this area. Regarding their functionality, these tools can be mainly divided into those which only infer haplotype frequencies and those which perform both the inference and the association with a trait.

7.8.2.1 Haploview

Haploview (Barrett et al. 2005) is a graphical tool nicely designed to simplify and expedite the process of haplotype analysis by providing a common interface to several tasks relating to such analyses. Haploview currently supports a wide range of haplotype functionalities such as: LD and haplotype block analysis, haplotype population frequency estimation, single SNP and haplotype association tests, permutation testing for association significance, implementation of Paul de Bakker's Tagger tag SNP selection algorithm, automatic download of phased genotype data from HapMap, and visualization and plotting of PLINK genome-wide association results including advanced filtering options. Haploview is fully compatible with data dumps from the HapMap project and the Perlegen Genotype Browser. It can analyze thousands of SNPs (tens of thousands in command line mode) in thousands of individuals. Web site: http://www.broad.mit.edu/mpg/haploview/

7.8.2.2 PHASE/fastPHASE

PHASE (Stephens et al. 2001; Stephens and Donnelly 2003; Stephens and Scheet 2005) and fastPHASE (Scheet and Stephens 2006) are command-line pieces of software used for haplotype reconstruction, as well as estimation of missing genotypes from population data. They do not compute association with a phenotype. Although fastPHASE can handle larger datasets than its previous version PHASE (e.g., hundreds of thousands of markers in thousands of individuals), it does not provide estimates of recombination rates (while PHASE does). Experiments suggest that fastPHASE haplotype estimates are slightly less accurate than from PHASE, but missing genotype estimates appear to be similar or even slightly better than PHASE. Web site: http://stephenslab.uchicago.edu/software.html

7.8.2.3 Haplo.stats

Haplo.stats (Schaid 2004; Schaid et al. 2002, 2005) is a suite of R routines for the analysis of indirectly measured haplotypes. The statistical methods implemented assume that all subjects are unrelated and that haplotypes are ambiguous (due to unknown linkage phase of the genetic markers). The genetic markers are assumed to be codominant (i.e., one-to-one correspondence between their genotypes and

their phenotypes). Some tools, such as SNPStats (see Sect. 7.8.1.1) and SNPassoc (see Sect. 7.8.1.2), use Haplo.stats as the underlying software to compute all haplotype related computations. Web site: http://mayoresearch.mayo.edu/mayo/research/schaid_lab/software.cfm

7.8.2.4 THESIAS

The aim of THESIAS (Tregouet et al. 2004; Tregouet and Tiret 2004; Tregouet and Garelle 2007) is to perform haplotype-based association analysis in unrelated individuals. The program is based on the maximum likelihood model and is linked to the Stochastic EM algorithm. THESIAS allows the simultaneous estimation of haplotype frequencies and of their associated effects on the phenotype of interest. Quantitative, qualitative, categorical, and, more interestingly, survival outcomes can be studied. Covariate-adjusted haplotype effects as well as haplotype–environment interactions can be investigated. THESIAS began as a command-line tool which was not too user friendly, but in the latest version its creators have added a JAVA interface which has improved the usability of the program, although it is still a bit rigid at some points. Web site: http://ecgene.net/genecanvas/uploads/THESIAS3.1/Documentation3.1.htm

7.8.3 Web Databases

The dramatically increasing amount of genomic information generated in the last few years has made essential the development of information systems which provide easy procedures of storage and retrieval of data. Therefore, public repositories of genetic data have been created and are maintained on a daily basis due to the effort of large consortiums.

7.8.3.1 dbSNP

In collaboration with the National Human Genome Research Institute, The National Center for Biotechnology Information has established the dbSNP (Sherry et al. 2001) database to serve as a central repository for both single-base nucleotide substitutions and short deletion and insertion polymorphisms. Once discovered, these polymorphisms can be used by additional laboratories, using the sequence information around the polymorphism and the specific experimental conditions. Note that dbSNP takes the looser "variation" definition for SNPs, so there is no requirement or assumption about minimum allele frequency. Data in dbSNP can be integrated with other NCBI genomic data. As with all NCBI projects, data in dbSNP is freely available to the scientific community and made available in a variety of forms. Web site: http://www.ncbi.nlm.nih.gov/projects/SNP/

7.8.3.2 Hapmap

The International HapMap Project (Hap 2003; Thorisson et al. 2005) is a multi-country effort to identify and catalog genetic similarities and differences in human beings. It describes what these variants are, where they occur in our DNA, and how they are distributed among people within populations and among populations in different parts of the world. Using the information in the HapMap, researchers will be able to find genes that affect health, disease, and individual responses to medications and environmental factors. In the initial phase of the project, genetic data are being gathered from four populations with African, Asian, and European ancestry. Ongoing interactions with members of these populations are addressing potential ethical issues and providing valuable experience in conducting research with identified populations. Web site: http://www.hapmap.org/

7.8.3.3 Genome Variation Server

The Genome Variation Server (GVS) is a local database hosted by the SeattleSNPs Program for Genomic Applications. The objective of this database is to provide a simple tool for rapid access to human genotype data found in dbSNP. The database includes a suite of analysis tools such as linkage disequilibrium plots, tag SNPs, and more. In addition you can upload our own data and use the GVS analysis and visualization tools. Web site: http://gvs.gs.washington.edu/GVS/

7.8.4 Statistical Power Calculation

Statistical power calculation is an important issue in association studies, specially for GWAS. Before we proceed with the experiments, we need to ensure that with our sample size we will be able to detect changes of a specific size in terms of the minimum allele frequency we expect to have in our SNPs of interest. Ignoring this information may cause our failure in detecting existing genetic associations due to the inadequate sample size of our study.

7.8.4.1 QUANTO

QUANTO computes sample size or power for association studies of genes, gene–environment interaction, or gene–gene interaction. Available study designs include the matched case–control, case–sibling, case–parent, and case-only designs. It is a stand-alone 32-bit Windows application. Its graphical user interface allows the user to easily change the model and view the results without having to edit an input file and rerun the program for every model. Web site: http://hydra.usc.edu/gxe/

7.8.4.2 Genetic Power Calculator

Designed by the same authors of PLINK, GPC is a Web site tool that performs power calculations for the design of linkage and association genetic mapping studies of complex traits (Purcell et al. 2003). Web site: http://pngu.mgh.harvard.edu/~purcell/gpc/

7.8.4.3 CaTS

CaTS (Skol et al. 2006) is a simple and useful multiplatform interface for carrying out power calculations for large genetic association studies, including two-stage genome-wide association studies. Web site: http://www.sph.umich.edu/csg/abecasis/cats/index.html

References

Affymetrix (2006) Brlmm: An improved genotype calling method for the genechip human mapping 500k array set. Technical Report, Affymetrix, Inc., Santa Clara, CA

Akey J, Jin L, Xiong M (2001) Haplotypes vs single marker linkage disequilibrium tests: What do we gain? Eur J Hum Genet 9(4):291–300

Albert PS, Ratnasinghe D, Tangrea J, Wacholder S (2001) Limitations of the case-only design for identifying gene-environment interactions. Am J Epidemiol 154(8):687–693

Armitage P (1955) Tests for linear trends in proportions and frequencies. Biometrics 11(3): 375–386

Bailey-Wilson JE, Sorant B, Sorant AJ, Paul CM, Elston RC (1995) Model-free association analysis of a rare disease. Genet Epidemiol 12(6):571–575

Barrett JC, Fry B, Maller J, Daly MJ (2005) Haploview: Analysis and visualization of ld and haplotype maps. Bioinformatics 21(2):263–265

Beecham GW, Martin ER, Li YJ, Slifer MA, Gilbert JR, Haines JL, Pericak-Vance MA (2009) Genome-wide association study implicates a chromosome 12 risk locus for late-onset alzheimer disease. Am J Hum Genet 84(1):35–43

Benjamini Y, Hochberg Y (1995) Controlling the false discovery rate – a practical and powerful approach to multiple testing. J R Stat Soc B 57:289–300

Bennett ST, Lucassen AM, Gough SC, Powell EE, Undlien DE, Pritchard LE, Merriman ME, Kawaguchi Y, Dronsfield MJ, Pociot F, et al (1995) Susceptibility to human type 1 diabetes at iddm2 is determined by tandem repeat variation at the insulin gene minisatellite locus. Nat Genet 9(3):284–292

Birney E, Stamatoyannopoulos JA, Dutta A, Guigo R, Gingeras TR, Margulies EH, Weng Z, Snyder M, Dermitzakis ET, Thurman RE, Kuehn MS, Taylor CM, Neph S, Koch CM, Asthana S, Malhotra A, Adzhubei I, Greenbaum JA, Andrews RM, Flicek P, Boyle PJ, Cao H, Carter NP, Clelland GK, Davis S, Day N, Dhami P, Dillon SC, Dorschner MO, Fiegler H, Giresi PG, Goldy J, Hawrylycz M, Haydock A, Humbert R, James KD, Johnson BE, Johnson EM, Frum TT, Rosenzweig ER, Karnani N, Lee K, Lefebvre GC, Navas PA, Neri F, Parker SC, Sabo PJ, Sandstrom R, Shafer A, Vetrie D, Weaver M, Wilcox S, Yu M, Collins FS, Dekker J, Lieb JD, Tullius TD, Crawford GE, Sunyaev S, Nobl WS (2007) Identification and analysis of functional elements in 1% of the human genome by the encode pilot project. Nature 447(7146):799–816

Broderick P, Carvajal-Carmona L, Pittman AM, Webb E, Howarth K, Rowan A, Lubbe S, Spain S, Sullivan K, Fielding S, Jaeger E, Vijayakrishnan J, Kemp Z, Gorman M, Chandler I, Papaemmanuil E, Penegar S, Wood W, Sellick G, Qureshi M, Teixeira A, Domingo E, Barclay E, Martin L, Sieber O, Kerr D, Gray R, Peto J, Cazier JB, Tomlinson I, Houlston RS (2007) A genome-wide association study shows that common alleles of smad7 influence colorectal cancer risk. Nat Genet 39(11):1315–1317

Cai J, Zeng D (2004) Sample size/power calculation for case-cohort studies. Biometrics 60(4):1015–1024

Carvalho B, Bengtsson H, Speed TP, Irizarry RA (2007) Exploration, normalization, and genotype calls of high-density oligonucleotide snp array data. Biostatistics 8(2):485–499

Celeux G, Diebolt J (1985) The sem algorithm: A probabilistic teacher derived from the em algorithm for the mixture problem. Comput Stat Q 2:73–82

Chanock SJ, Manolio T, Boehnke M, Boerwinkle E, Hunter DJ, Thomas G, Hirschhorn JN, Abecasis G, Altshuler D, Bailey-Wilson JE, Brooks LD, Cardon LR, Daly M, Donnelly P, Fraumeni J J F, Freimer NB, Gerhard DS, Gunter C, Guttmacher AE, Guyer MS, Harris EL, Hoh J, Hoover R, Kong CA, Merikangas KR, Morton CC, Palmer LJ, Phimister EG, Rice JP, Roberts J, Rotimi C, Tucker MA, Vogan KJ, Wacholder S, Wijsman EM, Winn DM, Collins FS (2007) Replicating genotype–phenotype associations. Nature 447(7145):655–660

Churchill GA, Doerge RW (1994) Empirical threshold values for quantitative trait mapping. Genetics 138(3):963–971

Clark AG (1990) Inference of haplotypes from pcr-amplified samples of diploid populations. Mol Biol Evol 7(2):111–122

Clayton D, McKeigue PM (2001) Epidemiological methods for studying genes and environmental factors in complex diseases. Lancet 358:1356–1360

Clayton DG, Walker NM, Smyth DJ, Pask R, Cooper JD, Maier LM, Smink LJ, Lam AC, Ovington NR, Stevens HE, Nutland S, Howson JM, Faham M, Moorhead M, Jones HB, Falkowski M, Hardenbol P, Willis TD, Todd JA (2005) Population structure, differential bias and genomic control in a large-scale, case–control association study. Nat Genet 37(11):1243–1246

Consortium WTCC (2007) Genome-wide association study of 14,000 cases of seven common diseases and 3,000 controls. Nature 447:661–678

Cox A, Dunning AM, Garcia-Closas M, Balasubramanian S, Reed MW, Pooley KA, Scollen S, Baynes C, Ponder BA, Chanock S, Lissowska J, Brinton L, Peplonska B, Southey MC, Hopper JL, McCredie MR, Giles GG, Fletcher O, Johnson N, dos Santos Silva I, Gibson L, Bojesen SE, Nordestgaard BG, Axelsson CK, Torres D, Hamann U, Justenhoven C, Brauch H, Chang-Claude J, Kropp S, Risch A, Wang-Gohrke S, Schurmann P, Bogdanova N, Dork T, Fagerholm R, Aaltonen K, Blomqvist C, Nevanlinna H, Seal S, Renwick A, Stratton MR, Rahman N, Sangrajrang S, Hughes D, Odefrey F, Brennan P, Spurdle AB, Chenevix-Trench G, Beesley J, Mannermaa A, Hartikainen J, Kataja V, Kosma VM, Couch FJ, Olson JE, Goode EL (2007) A common coding variant in casp8 is associated with breast cancer risk. Nat Genet 39(3):352–358

Cutler DJ, Zwick ME, Carrasquillo MM, Yohn CT, Tobin KP, Kashuk C, Mathews DJ, Shah NA, Eichler EE, Warrington JA, Chakravarti A (2001) High-throughput variation detection and genotyping using microarrays. Genome Res 11(11):1913–1925

Dempster A, Laird N, Rubin D (1977) Maximum likelihood from incomplete data via the EM-algorithm. J R Stat Soc 39:1–38

Devlin B, Roeder K (1999) Genomic control for association studies. Biometrics 55(4):997–1004

Di X, Matsuzaki H, Webster TA, Hubbell E, Liu G, Dong S, Bartell D, Huang J, Chiles R, Yang G, Shen MM, Kulp D, Kennedy GC, Mei R, Jones KW, Cawley S (2005) Dynamic model based algorithms for screening and genotyping over 100 k snps on oligonucleotide microarrays. Bioinformatics 21(9):1958–1963

Dudbridge F, Gustano A (2008) Estimation of significance thresholds for genomewide association scans. Genet Epidemiol 32:227–234

Dudbridge F, Koeleman BP (2004) Efficient computation of significance levels for multiple associations in large studies of correlated data, including genomewide association studies. Am J Hum Genet 75(3):424–435

Dudbridge F, Gusnanto A, Koeleman BP (2006) Detecting multiple associations in genome-wide studies. Hum Genomics 2(5):310–317

Easton DF, Eeles RA (2008) Genome-wide association studies in cancer. Hum Mol Genet 17(R2):R109–R115

Excoffier L, Slatkin M (1995) Maximum-likelihood estimation of molecular haplotype frequencies in a diploid population. Mol Biol Evol 12(5):921–927

Foulkes WD (2008) Inherited susceptibility to common cancers. N Engl J Med 359(20):2143–2153

Freidlin B, Zheng G, Li Z, Gastwirth JL (2002) Trend tests for case-control studies of genetic markers: power, sample size and robustness. Hum Hered 53(3):146–152

Garcia-Closas M, Malats N, Silverman D, Dosemeci M, Kogevinas M, Hein DW, Tardon A, Serra C, Carrato A, Garcia-Closas R, Lloreta J, Castano-Vinyals G, Yeager M, Welch R, Chanock S, Chatterjee N, Wacholder S, Samanic C, Tora M, Fernandez F, Real FX, Rothman N (2005) Nat2 slow acetylation, gstm1 null genotype, and risk of bladder cancer: Results from the spanish bladder cancer study and meta-analyses. Lancet 366(9486):649–659

Gonzalez JR, Armengol L, Sole X, Guino E, Mercader JM, Estivill X, Moreno V (2007) Snpassoc: An r package to perform whole genome association studies. Bioinformatics 23(5):644–645

Gonzalez JR, Carrasco JL, Dudbridge F, Armengol L, Estivill X, Moreno V (2008) Maximizing association statistics over genetic models. Genet Epidemiol 32(3):246–254

Gorroochurn P, Heiman GA, Hodge SE, Greenberg DA (2006) Centralizing the non-central chi-square: A new method to correct for population stratification in genetic case–control association studies. Genet Epidemiol 30(4):277–289

Guo SW, Thompson EA (1992) Performing the exact test of Hardy–Weinberg proportion for multiple alleles. Biometrics 48(2):361–372

Haiman CA, Le Marchand L, Yamamato J, Stram DO, Sheng X, Kolonel LN, Wu AH, Reich D, Henderson BE (2007a) A common genetic risk factor for colorectal and prostate cancer. Nat Genet 39(8):954–956

Haiman CA, Patterson N, Freedman ML, Myers SR, Pike MC, Waliszewska A, Neubauer J, Tandon A, Schirmer C, McDonald GJ, Greenway SC, Stram DO, Le Marchand L, Kolonel LN, Frasco M, Wong D, Pooler LC, Ardlie K, Oakley-Girvan I, Whittemore AS, Cooney KA, John EM, Ingles SA, Altshuler D, Henderson BE, Reich D (2007b) Multiple regions within 8q24 independently affect risk for prostate cancer. Nat Genet 39(5):638–644

Hirschhorn JN, Daly MJ (2005) Genome-wide association studies for common diseases and complex traits. Nat Rev Genet 6(2):95–108

Hoh J, Ott J (2003) Mathematical multi-locus approaches to localizing complex human trait genes. Nat Rev Genet 4:701–709

Hong H, Su Z, Ge W, Shi L, Perkins R, Fang H, Xu J, Chen JJ, Han T, Kaput J, Fuscoe JC, Tong W (2008) Assessing batch effects of genotype calling algorithm brlmm for the affymetrix genechip human mapping 500 k array set using 270 hapmap samples. BMC Bioinformatics 9(Suppl 9):S17

Hunter DJ, Kraft P, Jacobs KB, Cox DG, Yeager M, Hankinson SE, Wacholder S, Wang Z, Welch R, Hutchinson A, Wang J, Yu K, Chatterjee N, Orr N, Willett WC, Colditz GA, Ziegler RG, Berg CD, Buys SS, McCarty CA, Feigelson HS, Calle EE, Thun MJ, Hayes RB, Tucker M, Gerhard DS, Fraumeni J J F, Hoover RN, Thomas G, Chanock SJ (2007) A genome-wide association study identifies alleles in fgfr2 associated with risk of sporadic postmenopausal breast cancer. Nat Genet 39(7):870–874

Iniesta R, Moreno V (2008) Assessment of genetic association using haplotypes inferred with uncertainty via markov chain monte carlo. In: Keller A, Heinrich S, Niederreiter H (eds) Monte Carlo and Quasi-Monte Carlo Methods 2006. Springer, New York, pp 529–535

International HapMap Consortium (2003) The international hapmap project. Nature 426(6968): 789–796

Irizarry RA, Hobbs B, Collin F, Beazer-Barclay YD, Antonellis KJ, Scherf U, Speed TP (2003) Exploration, normalization, and summaries of high density oligonucleotide array probe level data. Biostatistics 4(2):249–264

Jaeger E, Webb E, Howarth K, Carvajal-Carmona L, Rowan A, Broderick P, Walther A, Spain S, Pittman A, Kemp Z, Sullivan K, Heinimann K, Lubbe S, Domingo E, Barclay E, Martin L, Gorman M, Chandler I, Vijayakrishnan J, Wood W, Papaemmanuil E, Penegar S, Qureshi M, Farrington S, Tenesa A, Cazier JB, Kerr D, Gray R, Peto J, Dunlop M, Campbell H, Thomas H, Houlston R, Tomlinson I (2008) Common genetic variants at the crac1 (hmps) locus on chromosome 15q13.3 influence colorectal cancer risk. Nat Genet 40(1):26–28

Korn JM, Kuruvilla FG, McCarroll SA, Wysoker A, Nemesh J, Cawley S, Hubbell E, Veitch J, Collins PJ, Darvishi K, Lee C, Nizzari MM, Gabriel SB, Purcell S, Daly MJ, Altshuler D (2008) Integrated genotype calling and association analysis of snps, common copy number polymorphisms and rare cnvs. Nat Genet 40(10):1253–1260

Landi S, Gemignani F, Moreno V, Gioia-Patricola L, Chabrier A, Guino E, Navarro M, de Oca J, Capella G, Canzian F (2005) A comprehensive analysis of phase i and phase ii metabolism gene polymorphisms and risk of colorectal cancer. Pharmacogenet Genomics 15(8):535–546

Langholz B, Rothman N, Wacholder S, Thomas D (1999) Cohort studies for characterizing measured genes. Monogr Natl Cancer Inst 26:39–42

Lichtenstein P, Holm NV, Verkasalo PK, Iliadou A, Kaprio J, Koskenvuo M, Pukkala E, Skytthe A, Hemminki K (2000) Environmental and heritable factors in the causation of cancer – analyses of cohorts of twins from Sweden, Denmark, and Finland. N Engl J Med 343(2):78–85

Lin S, Carvalho B, Cutler DJ, Arking DE, Chakravarti A, Irizarry RA (2008) Validation and extension of an empirical bayes method for snp calling on affymetrix microarrays. Genome Biol 9(4):R63

Liu WM, Di X, Yang G, Matsuzaki H, Huang J, Mei R, Ryder TB, Webster TA, Dong S, Liu G, Jones KW, Kennedy GC, Kulp D (2003) Algorithms for large-scale genotyping microarrays. Bioinformatics 19(18):2397–2403

Manly KF, Nettleton D, Hwang JT (2004) Genomics, prior probability, and statistical tests of multiple hypotheses. Genome Res 14(6):997–1001

McKay JD, Hung RJ, Gaborieau V, Boffetta P, Chabrier A, Byrnes G, Zaridze D, Mukeria A, Szeszenia-Dabrowska N, Lissowska J, Rudnai P, Fabianova E, Mates D, Bencko V, Foretova L, Janout V, McLaughlin J, Shepherd F, Montpetit A, Narod S, Krokan HE, Skorpen F, Elvestad MB, Vatten L, Njolstad I, Axelsson T, Chen C, Goodman G, Barnett M, Loomis MM, Lubinski J, Matyjasik J, Lener M, Oszutowska D, Field J, Liloglou T, Xinarianos G, Cassidy A, Vineis P, Clavel-Chapelon F, Palli D, Tumino R, Krogh V, Panico S, Gonzalez CA, Ramon Quiros J, Martinez C, Navarro C, Ardanaz E, Larranaga N, Kham KT, Key T, Bueno-de Mesquita HB, Peeters PH, Trichopoulou A, Linseisen J, Boeing H, Hallmans G, Overvad K, Tjonneland A, Kumle M, Riboli E, Zelenika D, Boland A, Delepine M, Foglio M, Lechner D, Matsuda F, Blanche H, Gut I, Heath S, Lathrop M, Brennan P (2008) Lung cancer susceptibility locus at 5p15.33. Nat Genet 40(12):1404–1406

Moreno V, Gemignani F, Landi S, Gioia-Patricola L, Chabrier A, Blanco I, Gonzalez S, Guino E, Capella G, Canzian F (2006) Polymorphisms in genes of nucleotide and base excision repair: Risk and prognosis of colorectal cancer. Clin Cancer Res 12(7 Pt 1):2101–2108

Mukherjee B, Chatterjee N (2008) Exploiting gene–environment independence for analysis of case-control studies: An empirical bayes-type shrinkage estimator to trade-off between bias and efficiency. Biometrics 64(3):685–694

Niu T, Qin ZS, Xu X, Liu JS (2002) Bayesian haplotype inference for multiple linked single-nucleotide polymorphisms. Am J Hum Genet 70(1):157–169

O'Donovan MC, Norton N, Williams H, Peirce T, Moskvina V, Nikolov I, Hamshere M, Carroll L, Georgieva L, Dwyer S, Holmans P, Marchini JL, Spencer CC, Howie B, Leung HT, Giegling I, Hartmann AM, Moller HJ, Morris DW, Shi Y, Feng G, Hoffmann P, Propping P, Vasilescu C, Maier W, Rietschel M, Zammit S, Schumacher J, Quinn EM, Schulze TG, Iwata N, Ikeda M, Darvasi A, Shifman S, He L, Duan J, Sanders AR, Levinson DF, Adolfsson R, Osby U, Terenius L, Jonsson EG, Cichon S, Nothen MM, Gill M, Corvin AP, Rujescu D, Gejman PV,

Kirov G, Craddock N, Williams NM, Owen MJ (2009) Analysis of 10 independent samples provides evidence for association between schizophrenia and a snp flanking fibroblast growth factor receptor 2. Mol Psychiatry 14(1):30–36

Pahl R, Schafer H, Muller HH (2009) Optimal multistage designs – A general framework for efficient genome-wide association studies. Biostatistics 10(2):297–309

Pankratz N, Wilk JB, Latourelle JC, DeStefano AL, Halter C, Pugh EW, Doheny KF, Gusella JF, Nichols WC, Foroud T, Myers RH (2009) Genomewide association study for susceptibility genes contributing to familial parkinson disease. Hum Genet 124(6):593–605

Piegorsch WW, Weinberg CR, Taylor JA (1994) Non-hierarchical logistic models and case-only designs for assessing susceptibility in population-based case-control studies. Stat Med 13(2):153–162

Price AL, Patterson NJ, Plenge RM, Weinblatt ME, Shadick NA, Reich D (2006) Principal components analysis corrects for stratification in genome-wide association studies. Nat Genet 38(8):904–909

Pritchard JK, Stephens M, Donnelly P (2000) Inference of population structure using multilocus genotype data. Genetics 155(2):945–959

Purcell S, Cherny SS, Sham PC (2003) Genetic power calculator: Design of linkage and association genetic mapping studies of complex traits. Bioinformatics 19(1):149–150

Purcell S, Neale B, Todd-Brown K, Thomas L, Ferreira MA, Bender D, Maller J, Sklar P, de Bakker PI, Daly MJ, Sham PC (2007) Plink: A tool set for whole-genome association and population-based linkage analyses. Am J Hum Genet 81(3):559–575

Qin ZS, Niu T, Liu JS (2002) Partition-ligation-expectation-maximization algorithm for haplotype inference with single-nucleotide polymorphisms. Am J Hum Genet 71(5):1242–1247

Rabbee N, Speed TP (2006) A genotype calling algorithm for affymetrix snp arrays. Bioinformatics 22(1):7–12

Redon R, Ishikawa S, Fitch KR, Feuk L, Perry GH, Andrews TD, Fiegler H, Shapero MH, Carson AR, Chen W, Cho EK, Dallaire S, Freeman JL, Gonzalez JR, Gratacos M, Huang J, Kalaitzopoulos D, Komura D, MacDonald JR, Marshall CR, Mei R, Montgomery L, Nishimura K, Okamura K, Shen F, Somerville MJ, Tchinda J, Valsesia A, Woodwark C, Yang F, Zhang J, Zerjal T, Armengol L, Conrad DF, Estivill X, Tyler-Smith C, Carter NP, Aburatani H, Lee C, Jones KW, Scherer SW, Hurles ME (2006) Global variation in copy number in the human genome. Nature 444(7118):444–454

Sabatti C, Service SK, Hartikainen AL, Pouta A, Ripatti S, Brodsky J, Jones CG, Zaitlen NA, Varilo T, Kaakinen M, Sovio U, Ruokonen A, Laitinen J, Jakkula E, Coin L, Hoggart C, Collins A, Turunen H, Gabriel S, Elliot P, McCarthy MI, Daly MJ, Jarvelin MR, Freimer NB, Peltonen L (2009) Genome-wide association analysis of metabolic traits in a birth cohort from a founder population. Nat Genet 41(1):35–46

Schaid DJ (2004) Evaluating associations of haplotypes with traits. Genet Epidemiol 27(4): 348–364

Schaid DJ, Rowland CM, Tines DE, Jacobson RM, Poland GA (2002) Score tests for association between traits and haplotypes when linkage phase is ambiguous. Am J Hum Genet 70(2): 425–434

Schaid DJ, McDonnell SK, Hebbring SJ, Cunningham JM, Thibodeau SN (2005) Nonparametric tests of association of multiple genes with human disease. Am J Hum Genet 76(5):780–793

Scheet P, Stephens M (2006) A fast and flexible statistical model for large-scale population genotype data: Applications to inferring missing genotypes and haplotypic phase. Am J Hum Genet 78(4):629–644

Schork NJ (2002) Power calculations for genetic association studies using estimated probability distributions. Am J Hum Genet 70(6):1480–1489

Sherry ST, Ward MH, Kholodov M, Baker J, Phan L, Smigielski EM, Sirotkin K (2001) dbSNP: The NCBI database of genetic variation. Nucleic Acids Res 29(1):308–311

Shlien A, Tabori U, Marshall CR, Pienkowska M, Feuk L, Novokmet A, Nanda S, Druker H, Scherer SW, Malkin D (2008) Excessive genomic DNA copy number variation in the Li–Fraumeni cancer predisposition syndrome. Proc Natl Acad Sci USA 105(32):11264–11269

Skol AD, Scott LJ, Abecasis GR, Boehnke M (2006) Joint analysis is more efficient than replication-based analysis for two-stage genome-wide association studies. Nat Genet 38(2):209–213

Sladek R, Rocheleau G, Rung J, Dina C, Shen L, Serre D, Boutin P, Vincent D, Belisle A, Hadjadj S, Balkau B, Heude B, Charpentier G, Hudson TJ, Montpetit A, Pshezhetsky AV, Prentki M, Posner BI, Balding DJ, Meyre D, Polychronakos C, Froguel P (2007) A genome-wide association study identifies novel risk loci for type 2 diabetes. Nature 445(7130):881–885

Slager SL, Schaid DJ (2001) Case–control studies of genetic markers: Power and sample size approximations for armitage's test for trend. Hum Hered 52(3):149–153

Sole X, Guino E, Valls J, Iniesta R, Moreno V (2006) SNPStats: A web tool for the analysis of association studies. Bioinformatics 22(15):1928–1929

Sole X, Hernandez P, de Heredia ML, Armengol L, Rodriguez-Santiago B, Gomez L, Maxwell CA, Aguilo F, Condom E, Abril J, Perez-Jurado L, Estivill X, Nunes V, Capella G, Gruber SB, Moreno V, Pujana MA (2008) Genetic and genomic analysis modeling of germline c-myc overexpression and cancer susceptibility. BMC Genomics 9:12

Spielman RS, McGinnis RE, Ewens WJ (1993) Transmission test for linkage disequilibrium: the insulin gene region and insulin-dependent diabetes mellitus (iddm). Am J Hum Genet 52(3):506–516

Spruill SE, Lu J, Hardy S, Weir B (2002) Assessing sources of variability in microarray gene expression data. Biotechniques 33(4):916–920, 922–923

Stephens M, Donnelly P (2003) A comparison of bayesian methods for haplotype reconstruction from population genotype data. Am J Hum Genet 73(5):1162–1169

Stephens M, Scheet P (2005) Accounting for decay of linkage disequilibrium in haplotype inference and missing-data imputation. Am J Hum Genet 76(3):449–462

Stephens M, Smith NJ, Donnelly P (2001) A new statistical method for haplotype reconstruction from population data. Am J Hum Genet 68(4):978–989

Sun J, Zheng SL, Wiklund F, Isaacs SD, Purcell LD, Gao Z, Hsu FC, Kim ST, Liu W, Zhu Y, Stattin P, Adami HO, Wiley KE, Dimitrov L, Li T, Turner AR, Adams TS, Adolfsson J, Johansson JE, Lowey J, Trock BJ, Partin AW, Walsh PC, Trent JM, Duggan D, Carpten J, Chang BL, Gronberg H, Isaacs WB, Xu J (2008) Evidence for two independent prostate cancer risk-associated loci in the hnf1b gene at 17q12. Nat Genet 40(10):1153–1155

Tanck MW, Klerkx AH, Jukema JW, De Knijff P, Kastelein JJ, Zwinderman AH (2003) Estimation of multilocus haplotype effects using weighted penalised log-likelihood: Analysis of five sequence variations at the cholesteryl ester transfer protein gene locus. Ann Hum Genet 67(Pt 2):175–184

Tenesa A, Farrington SM, Prendergast JG, Porteous ME, Walker M, Haq N, Barnetson RA, Theodoratou E, Cetnarskyj R, Cartwright N, Semple C, Clark AJ, Reid FJ, Smith LA, Kavoussanakis K, Koessler T, Pharoah PD, Buch S, Schafmayer C, Tepel J, Schreiber S, Volzke H, Schmidt CO, Hampe J, Chang-Claude J, Hoffmeister M, Brenner H, Wilkening S, Canzian F, Capella G, Moreno V, Deary IJ, Starr JM, Tomlinson IP, Kemp Z, Howarth K, Carvajal-Carmona L, Webb E, Broderick P, Vijayakrishnan J, Houlston RS, Rennert G, Ballinger D, Rozek L, Gruber SB, Matsuda K, Kidokoro T, Nakamura Y, Zanke BW, Greenwood CM, Rangrej J, Kustra R, Montpetit A, Hudson TJ, Gallinger S, Campbell H, Dunlop MG (2008) Genome-wide association scan identifies a colorectal cancer susceptibility locus on 11q23 and replicates risk loci at 8q24 and 18q21. Nat Genet 40(5):631–637

Thorisson GA, Smith AV, Krishnan L, Stein LD (2005) The international hapmap project web site. Genome Res 15(11):1592–1593

Tregouet DA, Garelle V (2007) A new java interface implementation of thesias: testing haplotype effects in association studies. Bioinformatics 23(8):1038–1039

Tregouet DA, Tiret L (2004) Cox proportional hazards survival regression in haplotype-based association analysis using the stochastic-em algorithm. Eur J Hum Genet 12(11):971–974

Tregouet DA, Escolano S, Tiret L, Mallet A, Golmard JL (2004) A new algorithm for haplotype-based association analysis: The stochastic-EM algorithm. Ann Hum Genet 68(Pt 2):165–177

Wacholder S, Chanock S, Garcia-Closas M, El Ghormli L, Rothman N (2004) Assessing the probability that a positive report is false: An approach for molecular epidemiology studies. J Natl Cancer Inst 96(6):434–442

Waldron ER, Whittaker JC, Balding DJ (2006) Fine mapping of disease genes via haplotype clustering. Genet Epidemiol 30(2):170–179

Wang Y, Broderick P, Webb E, Wu X, Vijayakrishnan J, Matakidou A, Qureshi M, Dong Q, Gu X, Chen WV, Spitz MR, Eisen T, Amos CI, Houlston RS (2008) Common 5p15.33 and 6p21.33 variants influence lung cancer risk. Nat Genet 40(12):1407–1409

Weidinger S, Gieger C, Rodriguez E, Baurecht H, Mempel M, Klopp N, Gohlke H, Wagenpfeil S, Ollert M, Ring J, Behrendt H, Heinrich J, Novak N, Bieber T, Kramer U, Berdel D, von Berg A, Bauer CP, Herbarth O, Koletzko S, Prokisch H, Mehta D, Meitinger T, Depner M, von Mutius E, Liang L, Moffatt M, Cookson W, Kabesch M, Wichmann HE, Illig T (2008) Genome-wide scan on total serum IgE levels identifies fcer1a as novel susceptibility locus. PLoS Genet 4(8):e1000166

Welch WJ (1990) Construction of permutation tests. J Am Stat Assoc 85:693–698

Wigginton JE, Cutler DJ, Abecasis GR (2005) A note on exact tests of Hardy–Weinberg equilibrium. Am J Hum Genet 76(5):887–893

Witte JS (2007) Multiple prostate cancer risk variants on 8q24. Nat Genet 39(5):579–580

Yeager M, Orr N, Hayes RB, Jacobs KB, Kraft P, Wacholder S, Minichiello MJ, Fearnhead P, Yu K, Chatterjee N, Wang Z, Welch R, Staats BJ, Calle EE, Feigelson HS, Thun MJ, Rodriguez C, Albanes D, Virtamo J, Weinstein S, Schumacher FR, Giovannucci E, Willett WC, Cancel-Tassin G, Cussenot O, Valeri A, Andriole GL, Gelmann EP, Tucker M, Gerhard DS, Fraumeni J J F, Hoover R, Hunter DJ, Chanock SJ, Thomas G (2007) Genome-wide association study of prostate cancer identifies a second risk locus at 8q24. Nat Genet 39(5):645–649

Yeh CC, Santella RM, Hsieh LL, Sung FC, Tang R (2009) An intron 4 VNTR polymorphism of the endothelial nitric oxide synthase gene is associated with early-onset colorectal cancer. Int J Cancer 124(7):1565–1571

Zhao J (2007) Gap: Genetic analysis package. J Stat Soft 23(8)

Chapter 8
Selected Applications of Graph-Based Tracking Methods for Cancer Research

Pascal Vallotton and Lilian Soon

Abstract Cell appearance is highly variable, particularly for cancer cells. Consequently, massive amounts of image data need to be screened rapidly and reproducibly to ascertain phenotypic information of interest. This demands a high level of automation and chosen image analysis techniques.

A very relevant phenotype for cancer is motile behaviour as it is a fundamental ingredient of its development. We aim to measure normal and abnormal motile processes, identify small molecules and genotypes that modulate these processes, and develop mathematical models of the disease towards identifying specific targets or target sets. Here, we describe the use and principles underlying representative software tools to read motile phenotypes from time-lapse data, with emphasis on improved graph-based tracking methods.

8.1 Introduction

Cancer is a disease of cell behaviour whereby a few cells evade the regulatory mechanisms obeyed by all others, multiplying without control and forming a tumour (Alberts 2002). Cancers have a capacity to metastasize, that is to initiate new tumours distinct from the initial lesion. This capacity depends on motile properties of cancer cells to penetrate the bloodstream and leave it later to initiate new tumours. When tumours metastasize, the disease is usually lethal. Therefore, considerable efforts have been devoted to understand the mechanisms underlying cell motility, both intra- and extra-cellular, and in both normal and abnormal cells.

The role of imaging in cancer research is central not only for motility studies but also for understanding angiogenesis and the genetic basis of the disease, for example, in the context of karyotyping, gel-based, or bead-based methods. Our scope in this chapter is to introduce a set of image analysis tools and techniques

P. Vallotton (✉)
CSIRO Mathematical and Information Sciences, Locked Bag 17, North Ryde, NSW 1670, Australia
e-mail: Pascal.Vallotton@csiro.au

T. Pham (ed.), *Computational Biology: Issues and Applications in Oncology*, Applied Bioinformatics and Biostatistics in Cancer Research,
DOI 10.1007/978-1-4419-0811-7_8,

that permit to identify objects automatically in image sequences and characterize their dynamics quantitatively. The importance of such a quantitative approach for biology and cell motility in particular is well appreciated; it represents the only viable route to meaningfully compare subtle differences observed across multiple biological experiments. Typically, these experiments aim to understand the role of particular proteins in various cellular processes, for example, by observing the consequences of a loss or inactivation of that protein (Petri Seiler et al. 2008).

The techniques that we will describe have been applied primarily to advance our understanding of the lamellipodium – the organelle of cell motility, and of the mitotic spindle – the organelle of cell division. Both the lamellipodium and the mitotic spindle are foci of considerable attention in their own right: small molecules that interfere with mitosis and motility are sought actively to inhibit cancer progression (Petri Seiler et al. 2008; Duffy et al. 2008). Many other areas have benefited from such tools including cell tracking (*see* Sect. 8.9), organelle tracking, virology, and super-resolution tracking of single molecules (Grunwald 2008). More exotic applications include the description of granular flow, MRI cardiac imaging, flow measurements in channels, or sport biomechanics (Kaitna 2007). This chapter describes the mathematical underpinnings of our methods – particularly, graph theory, and we present three applications illustrating their value for cancer research.

8.2 Object Detection

The brain has a natural tendency to divide the image field into discrete objects that evolve with some degree of independence over time and display finite lifetimes. This strategy provides a symbolic representation that enables prediction of the future and understanding of the present. Given the obvious success of the approach, one endeavours in computer vision to mimic it; an activity termed object segmentation or object detection (without worrying here about whether one knows a priori that an object is present or not). The simplest methodology relies on local intensity maximum detection and image filtering as described in more detail below. Depending on the application, more complex tools are required. For example, if objects extend spatially over relatively large area and feature sharp boundaries, it is best to perform edge detection first. Typically, the centre of mass of the regions identified in this manner is then taken to represent the objects.

In the situation, where objects of interest are circularly symmetric but do not present a clear local intensity maximum or minimum, it is sometimes useful to define a small window around one representative object and use this template to identify all other objects by correlation. For example, this technique is useful to identify virus particles in electron microscopy images (Sintorn et al. 2004).

One may choose to perform object detection operations either on the original images or on the pre-processed images. For example, Hadjidemetriou et al. describe how they applied the Hough transform to process phase contrast images of dividing cells appearing as bright rings to obtain well-defined intensity maxima at

the cell centre. This considerably simplified the cell tracking process in this case (Hadjidemetriou et al. 2008). More elementary pre-processing operations may include background subtraction to remove coarse scale illumination inhomogeneities. Many methodologies have been reported in that context including the use of morphological top hats, or the subtraction of a massively smoothed image from the original image (Soille 2004).

8.3 Local Maximum and Blurring

An elementary but robust strategy to identify objects, particularly if they stretch only over a few pixels on the image sensor, is to identify all local intensity maxima in the image. Each such maximum then "represents" the object – with ideally one maximum for each object. The reason why this situation often occurs in biological imaging is that the image of a diffraction limited point source (i.e., an object smaller than $\sim$0.3 μm, such as a single molecule, a small vesicle, or a small organelle) is a Gaussian spot in good approximation (Siebrasse et al. 2007). Larger objects often owe their visibility to some dye that diffuses, or is distributed homogeneously within them. This tends to give rise to the same ideal situation of a single intensity maximum per object.

If several local intensity maxima occur within objects, it is often possible to remove "secondary" maxima by blurring the image using Gaussian filtering. However, if genuine objects lie in proximity, excessive filtering should be avoided because this leads to merging objects that one wishes to treat independently. Many attempts have been reported in the literature to automate the filtering using ideas from scale space theory, for example (Lindeberg 1994). We find that optimal filtering is as much a matter of expert domain knowledge as a property of the image itself. Thus, we prefer to leave it to users to make the informed decision of the extent of image filtering optimal in each particular situation. Tuning such image analysis steps is very fast provided the image analysis functionalities are organized in an accessible manner.

8.4 Object Segmentation

In many cases, it is desired not only to represent objects by placing a marker in their centre but also to determine the boundaries of these objects. The watershed transform has proven a very robust and generic method to achieve this goal (Soille 2004). Briefly, as the name suggests, the watershed transform can be simulated by filling of the image, viewed as a surface in 3D, starting from seed positions placed in an independent manner (e.g., using local intensity maxima detection). This process creates watershed regions that grow concurrently as the algorithm progresses until they meet a neighbouring region; a constructive process that defines boundaries between the watersheds. A major difficulty in watershed segmentation is to position the seeds – both foreground seeds (those that will give rise to objects of interest) and

background seeds (those that will give rise to the image background). Another difficulty is over-segmentation, typically occurring when several seeds are placed within a single object. We avoid over-segmentation by linking every foreground seed with its nearest neighbours and testing whether the intensity along the corresponding segment presents a drop less than 10%, compared with the interpolated values along the segment. If it does, we instate the whole line as a foreground seed for the watershed transform, ensuring that a single object only be created. This method is also effective for 3D segmentation.

8.5 Object Tracking

Time-lapse image sequences contain considerable information on dynamic biological processes such as cell migration, intracellular polymer flow, protrusive activity, protein trafficking, or cell division. In order to analyse this data, it is desirable to identify discrete objects in every frame of the sequence and attempt to establish a correspondence between them in different frames that posits their physical identity. This process when performed across all frames of the sequence delivers a set of trajectories, one for each physical object. Object tracking is not an absolute necessity. For example, kymograph analysis has been in use long before object tracking and is still widespread nowadays. To conduct a kymograph analysis, the user defines a line segment in the first frame of a sequence. This line is then cut out digitally from each image and mounted side by side to form a montage. Oblique streaks indicate the presence of features moving along the axis, with a slope that reveals the velocity of these features. When feasible, object tracking provides considerably more information and does not depend on the arbitrary choice of an axis.

In algorithmic terms, object tracking may pose significant difficulties. Objects may appear or disappear either genuinely, by division, fusion, diffusion, or because of out of focus motion. Object dynamics may be very different for different objects or may be changing over time; finally, the identification of the objects themselves may be problematic, either because of low signal-to-noise ratio in the images or because of the imprecise nature of the object boundaries. Our contributions have mostly focused on establishing optimal correspondences (matches) between object from different frames. We find that mathematical graphs are the natural structures to deal with such problems (Vallotton et al. 2003). These methods have considerably evolved since their inception.

8.6 Algorithms on Graphs

Graphs are mathematical objects composed of edges linking nodes (Fig. 8.1). Edges can be endowed with integer descriptors such as the maximum flow that they can sustain – the analogy with pipes through which water can flow is useful, although

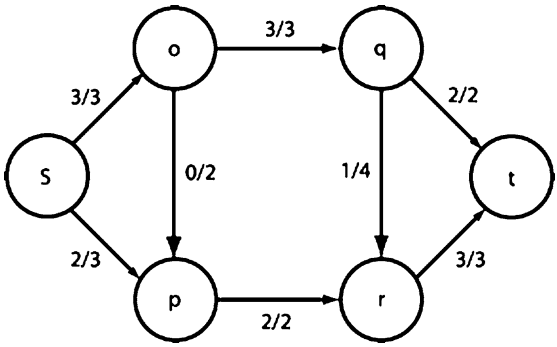

Fig. 8.1 A maximum amount of flow circulates from the source **s** to the sink **t** in this directed graph. The notation 1/4 for the edge linking node q to node r means that a flow of one unit traverses the edge of maximum capacity equal to four units. In this example, the solution is unique. Reversing the direction of edge o-p creates a graph where there are two solutions to the Max-Flow problem. Reproduced from Nils Grimsmo

not compelling, because the flow in a real pipe is not limited to integer values (Sedgewick 2002).

A central algorithm in graph theory is the Max-flow algorithm that determines the maximum amount of flow that can possibly go through the graph from one node designated as the source to another node designated as the sink. The solution of the algorithm is a set of integer flow values, one for each edge, such that the capacity of each edge is not exceeded, and such that no other assignment of flow values to edges results in a greater flow. Clearly, there may be several solutions to the Max-flow problem delivering the same total flow (*see* Fig. 8.1). It can be shown that the Max-flow problem is equivalent to the Min-cut problem; a formulation useful for the segmentation of digital images (Li et al. 2006).

The Max-flow algorithm can be used to solve bipartite matching problems where one aims to form as many couples as possible, respecting strictly the preferences of both males and females as represented by edges of capacity equal to one. Bipartite matching in this original form is not ideally suited for object tracking as we explain below. This led us to consider the Max-flow Min-cost graph algorithm, where edges are additionally endowed with a cost to pay per unit of flow traversing them. The Max-flow Min-cost solution selects the particular Max-flow solution for which the total cost is minimal. A Max-flow Min-cost solution is a Max-flow solution but the converse is not necessarily true.

In an object tracking context, the Max-flow Min-cost algorithm is particularly appealing because it is expected that small object displacements are more likely than larger ones under usual dynamical conditions. The edge cost in the tracking graph can be set in relation to the amplitude of the object displacement to reflect this preference. The strength of the Max-flow Min-cost algorithm is then that it is able to retrieve an optimal solution considering all possible displacements simultaneously.

This should be contrasted with a sequential approach, where matches are assigned in a particular order. In this type of so-called "greedy" approaches, tracking mistakes typically propagate to trigger other tracking mistakes later during the assignment process (Vallotton et al. 2003).

When applicable, the knowledge that one is tracking a constant number of objects is very powerful. It conveniently translates into a maximum matching (flow) requirement in the tracking graph. There is little doubt that as human beings, we take this cue into account when attempting to resolve dynamics in cluttered scenes.

However, if objects genuinely appear or disappear, strictly imposing maximum matching tends to create matching mistakes. Therefore, the constraint has to be relaxed while maintaining a global view on solving the matching problem. It is possible to achieve this by modifying the graph structure itself, allowing for some flow to leak out from the solution. In this form, the algorithm has proved extremely versatile, and performs remarkably in both 2D and 3D. Variations of this algorithm for particular applications such as coherent or anti-parallel flows have also been implemented as described in the application sections below.

8.7 Application to Lamellipodium Dynamics

In cells migrating over flat substrates, the lamellipodium is a thin sheet of cytoskeleton, consisting mostly of F-actin (filamentous actin) protruding at the cell front. The lamellipodium exerts tractile forces on the substrate, driving the cell forward (*see* Fig. 8.2). A major research endeavour is to understand the detailed mechanisms underlying this process, as described in more detail, for example by Small and Resch (2005).

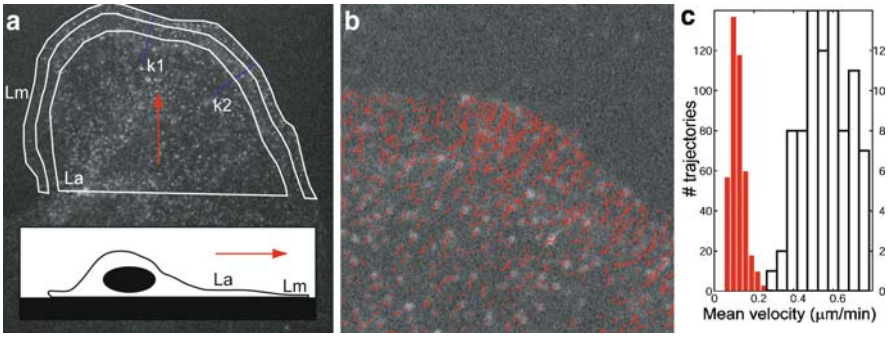

Fig. 8.2 (**a**) Motile cell imaged under FSM. Lamellipodium (*Lp*) and lamella (*La*) are outlined with closed contours. (**b**) Fluorescent speckles, tracked using improved graph-based methods. Results are compatible with manual tracking of the same data (not shown). (**c**) Distribution of speed along trajectories for tracks having their origin within the lamellipodium (resp. lamella) shown in white (resp. red). The two distributions show very limited overlap, reflecting the functional and structural difference between these two organelles

By introducing extremely low concentrations of fluorescently tagged actin monomers in live cells, it is possible via endogenous polymerization to generate single molecule fluorescent markers in the actin cytoskeleton and track them (Watanabe and Mitchison 2002). Even at slightly higher concentrations, the statistical variations in the local density of the polymerized fluorescent F-actin can give rise to distinct features, termed speckles, which can in principle be tracked (*see* Fig. 8.2). This very time-consuming task can easily be mastered by a human operator. Understandably, there is considerable interest in automating this process. Yet, it is important to remember that automated tracking aims at mimicking the results of manual tracking; not the opposite! Manual tracking and experiments on simulated data should systematically be performed to ascertain that the automated tracking results are sound.

A striking features of cytoskeleton motion as revealed by fluorescence speckle microscopy (FSM, cf previous paragraph) is retrograde motion, directed from the cell front towards the cell nucleus. Retrograde flow is powered by actin polymerization at the leading edge. It is fast in the lamellipodium and slower in the lamella, with a well-defined interface between these two regions where, by virtue of conservation of matter, considerable depolymerization takes place (Vallotton et al. 2004). By viewing movies of retrograde flow carefully, it becomes apparent that speckle motion occurs along straight paths directed perpendicular to the cell leading edge. Yet, early tracking results showed considerable jaggedness in the speckle trajectories. Also, early flow map results tended to considerably underestimate the flow velocity in the lamellipodium. The reason is that under usual imaging conditions, speckles move so fast in the lamellipodium that it is difficult to track them. This difficulty is compounded by the fact that considerable speckle appearance activity occurs at the cell front and this confused most trackers.

By introducing a preference for coherent motion among neighbouring speckles directly in our graph algorithm, it is possible to overcome these difficulties and obtain correct trajectories (i.e., trajectories that would have been obtained by manual tracking). We show results for speckles trajectories at the cell front in Fig. 8.2. These results correspond to the expectations of a naive observer: Tracks are mostly linear uniform, they are perpendicular to the leading edge, and jumps of up to 10 pixels can be observed. Incidentally, the average distance between speckles is also of this order, which characterizes difficult tracking problems.

Early on, we threw the concept of "secondary speckle" to describe the idea of a speckle that, it was felt, would trigger a local intensity maximum if the optical resolution was better. It was speculated that tracking results could potentially be improved by taking these secondary speckles into account. However, increasing the density of speckles in the lamellipodium in this manner makes matters actually worse for the reason mentioned earlier. If at all, secondary speckles should be taken into account only after a good matching has been achieved for "primary" speckles.

Speckle behaviour in the lamella and the lamellipodium are markedly different. For example, one can measure the average speed along tracks that have their origin in the lamellipodium and in the lamella, respectively (Fig. 8.2). The lack of overlap between these two distributions stresses the structural and functional differences between the two cell compartments.

8.8 Application to Mitotic Dynamics

The motion of Alexa-tubulin speckles in reconstituted mitotic spindles at metaphase was described (Vallotton et al. 2003). G-rhodamine-tubulin polymerizes into the microtubule system of the spindle and gives rise to discrete fiduciary markers: the speckles. At metaphase, microtubules undergo treadmilling, with the speckles converging to their associated spindle pole along the tracks defined by the microtubules. Using a so-called "three frame tracker," we were able to recover the anti-parallel nature of the speckle motion. However, careful review of the results (Fig. 8.6 in Vallotton et al. 2003) reveal that a significant number of tracks are interrupted or jitter more than would be expected for motion along a microtubule. Since then, our algorithms have improved by encoding within the graph structure the dual, anti-parallel nature of the motion in the spindle. In this manner, the algorithm specifically targets locally anti-parallel jumps, rather than a collection of triplets of points that are approximately aligned. As can be seen in Fig. 8.3, this leads to very significant improvements in the solution. Using this method, there is no constraint that would force speckles to move systematically in one direction or the other, as one can expect based on physical grounds. Therefore, the relatively low number of direction reversals in our results is encouraging (Fig. 8.3).

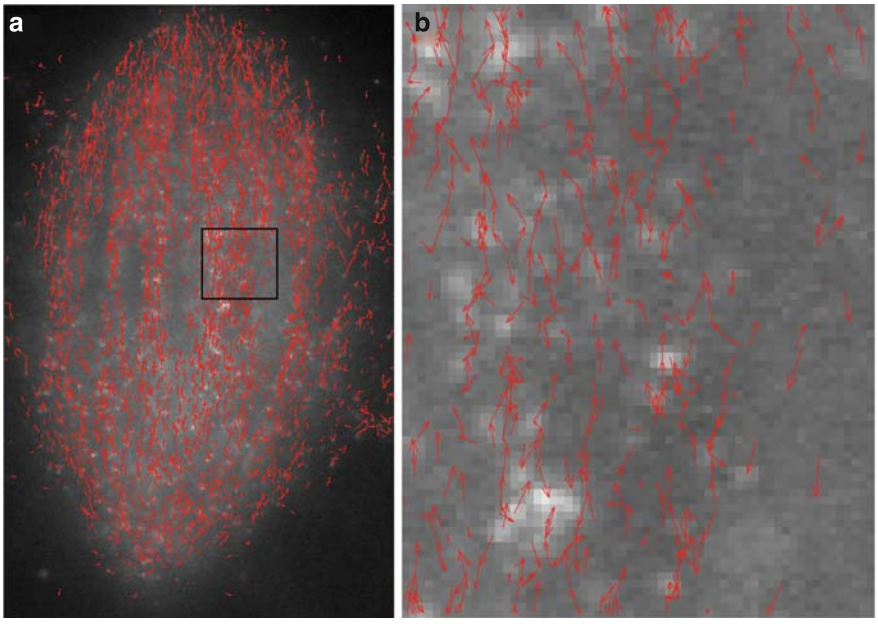

Fig. 8.3 (**a**) Reconstituted mitotic spindle seen under FSM. Individual speckles move towards either spindle pole at the top and bottom part of the figure, creating an anti-parallel motion field. (**b**) Zoom on the region shown in (**a**) showing anti-parallel tracks predominantly moving towards the upper and lower spindle poles. Tracks are mostly linear, as expected for motion along microtubules and only few direction reversals can be seen

The conclusions that can be drawn from these new results vary little from the conclusions that were drawn using our earlier three frame tracker. These conclusions were based on a large number of displacement vectors; the great majority of which had been correctly tracked. Generally, it is important not to attempt to draw conclusions requiring tools of a greater sophistication than those available. For example, because our three frame tracker delivered only partially formed tracks, it would have been unreasonable to use it to conduct an analysis of speckle lifetime. This analysis would still be unreasonable today because significant overlap between speckles moving in opposite directions prevents such an analysis.

8.9 Application to Cell Tracking

In the simplest embodiment of an assay to investigate migration, live cells are left for a specified amount of time over a substrate, from which they displace fluorescent beads. This creates traces whose length or area can be calculated easily on an endpoint image. This type of measurements conveniently summarizes the overall motile behaviour of a cell population. However, it omits interesting concepts such as velocity, acceleration, or trajectory entropy. Additionally, there is no handle for assessing the evolution of quantities of interest over time. By using the methodology described in this chapter, it is possible to segment every cell in every frame of a time-lapse sequence, track them, and investigate their dynamics systematically. We illustrate this point on cells that migrate in a chemo-attractant gradient created and maintained using a method developed by one of the authors (Fok 2008; Soon et al. 2005) and described elsewhere (Li Jeon 2002). A representative frame of the segmented cells is shown in Fig. 8.4a. The centre of mass of the fluorescently labelled cells is indicated by a red dot and the cell boundaries as determined by the watershed transform are shown as black contours. The trajectories of the cells as they unfold over 50 frames are shown in Fig. 8.4c, with time represented as the third dimension. Figure 8.4b is a plot of the evolution of the image intensity measured along the trajectories. Many other plots showing the evolution of quantities such as the area, the perimeter, and the eccentricity can be produced in a similar manner. The results can easily be aggregated to produce, for example, distributions of average velocities along tracks, or lifetime distributions, particularly useful to investigate cell division.

Tumour cells are known to be heterogeneous in nature, both within the lesion and following isolation as cell lines. This is reflected by the different types of movement that cancer cells exhibit in the absence of gradient. For example, amoeboid-like cancer cells tend to have low internal polarity, have a rounded appearance, and change directions frequently. Mesenchymal-like cells, on the other hand, are more polarized with distinctive front and tail regions. These cells tend to have high motion persistence when stimulated globally (Tchou-Wong 2006).

When the migratory behaviour of breast cancer cells under gradient conditions is analysed by cell tracking and analysis, two types of chemotactic behaviours are seen; linear and oscillatory. The former involves high directionality where cells tend to face the gradient throughout the chemotaxis run. The latter display a 10-min cycle

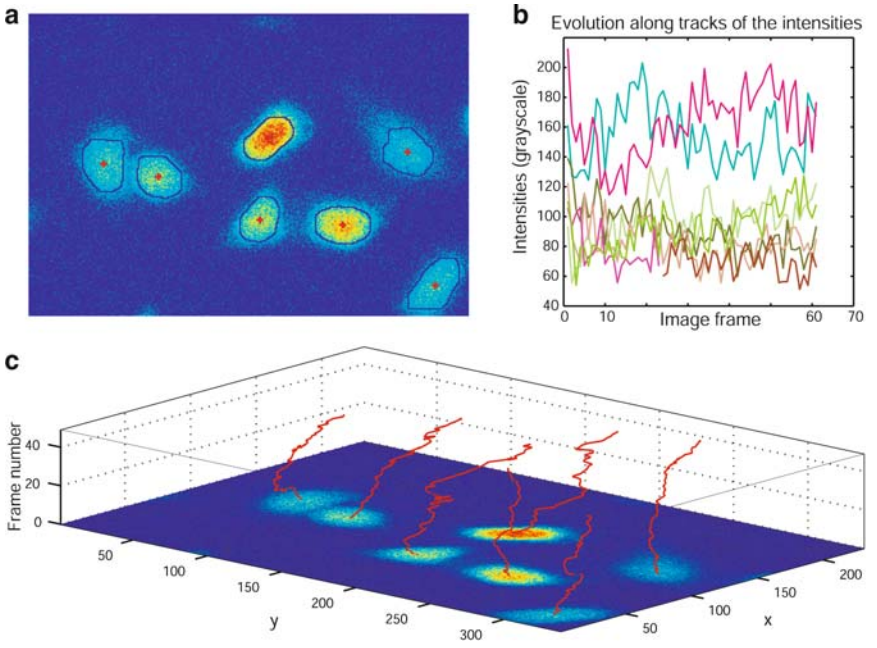

Fig. 8.4 Tracking cancer cells. (**a**) Segmentation of the cells. Local intensity maxima detection was used to seed automatically the watershed transform on the gradient image. The centre of gravity of cells is shown with red dots. (**b**) Evolution of the image intensity measured along trajectories. (**c**) Trajectories represented in three dimensions, with the z axis as the time. A drift towards a chemotaxis gradient is clearly apparent

when the cells wavered or loose directionality before recovering (Fok 2008). The linear movement resembles mesenchymal motility, whereas the oscillatory movement is similar to amoeboid cells motion, in this case under gradient conditions. The utility of quantitative tools is evident in the demonstration of variability in migration styles of cancer cells (Fok 2008; Soon 2007).

8.10 Conclusions and Perspectives

Object tracking methods are becoming increasingly useful in quantitative biology, particularly in cancer research where they contribute to our ability to understand the dynamics of cell populations as well as the root causes of their invasiveness. It is important for the development of the field that automated methods become as reliable as the manual methods that they are progressively replacing. In this chapter, we have argued for and demonstrated the value of graph-based algorithms for object tracking, with particular emphasis on challenging examples from fluorescent speckle microscopy and cell migration.

References

Alberts, B., et al. *Molecular Biology of the Cell* (Garland, 2002).

Duffy, M.J., McGowan, P.M. & Gallagher, W.M. Cancer invasion and metastasis: changing views. *J Pathol.* **214**, 283–293. (2008).

Fok, S., et al. Planar microfluidic chamber for generation of stable and steep chemoattractant gradients. *Biophysical Journal* **95**, 1523–1530. (2008).

Grunwald, D., et al. Probing intranuclear environments at the single-molecule level. *Biophys J.* **94**, 2847–2858. Epub 2007 Dec 2847. (2008).

Hadjidemetriou, H., Gabrielli, B., Mele, K. & Vallotton, P. Detection and tracking of cell divisions in phase contrast video microscopy, in *MICCAI* (New York, 2008).

Kaitna, R. Experimental study on rheologic behaviour of debris flow material. *Acta Geotechnica* **2**, 71–85. (2007).

Li, K., Wu, X., Chen, D.Z. & Sonka, M. Optimal surface segmentation in volumetric images–a graph-theoretic approach. *IEEE Trans Pattern Anal Mach Intell.* **28**, 119–134. (2006).

Li Jeon, N., et al. Neutrophil chemotaxis in linear and complex gradients of interleukin-8 formed in a microfabricated device. *Nat Biotechnol.* **20**, 826–830. Epub 2002 Jul 2001. (2002).

Lindeberg, T. *Scale-Space Theory in Computer Vision* (Kluwer Academic Publishers, 1994).

Petri Seiler, K., Kuehn, H., Pat Happ, M., Decaprio, D. & Clemons, P.A. Using ChemBank to probe chemical biology. *Curr Protoc Bioinformatics.* **Chapter**, Unit 14–15. (2008).

Sedgewick, R. *Algorithms in C++ Part 5: Graph Algorithms* (Addison-Wesley, Boston, 2002).

Siebrasse, J.P., Grunwald, D. & Kubitscheck, U. Single-molecule tracking in eukaryotic cell nuclei. *Anal Bioanal Chem.* **387**, 41–44. Epub 2006 Oct 2011. (2007).

Sintorn, I.M., Homman-Loudiyi, M., Soderberg-Naucler, C. & Borgefors, G. A refined circular template matching method for classification of human cytomegalovirus capsids in TEM images. *Comput Methods Programs Biomed.* **76**, 95–102. (2004).

Small, J.V. & Resch, G.P. The comings and goings of actin: coupling protrusion and retraction in cell motility. *Curr Opin Cell Biol.* **17**, 517–523. (2005).

Soille, P. *Morphological Image Analysis* (Springer, 2004).

Soon, L.L. A discourse on cancer cell chemotaxis: where to from here? *Iubmb Life* **59**, 60–67. (2007).

Soon, L., Mouneimne, G., Segall, J., Wyckoff, J. & Condeelis, J. Description and characterization of a chamber for viewing and quantifying cancer cell chemotaxis. *Cell Motil Cytoskeleton.* **62**, 27–34. (2005).

Tchou-Wong, K.M., et al. Rapid chemokinetic movement and the invasive potential of lung cancer cells; a functional molecular study. *BMC Cancer.* **6**, 151. (2006).

Vallotton, P., Ponti, A., Waterman-Storer, C.M., Salmon, E.D. & Danuser, G. Recovery, visualization, and analysis of actin and tubulin polymer flow in live cells: a fluorescent speckle microscopy study. *Biophys J.* **85**, 1289–1306. (2003).

Vallotton, P., Gupton, S.L., Waterman-Storer, C.M. & Danuser, G. Simultaneous mapping of filamentous actin flow and turnover in migrating cells by quantitative fluorescent speckle microscopy. *Proc Natl Acad Sci U S A.* **101**, 9660–9665. Epub 2004 Jun 9621. (2004).

Watanabe, N. & Mitchison, T.J. Single-molecule speckle analysis of actin filament turnover in lamellipodia. *Science.* **295**, 1083–1086. (2002).

Chapter 9
Recent Advances in Cell Classification for Cancer Research and Drug Discovery

Dat T. Tran and Tuan Pham

Abstract Drug effects on cancer cells are investigated through measuring cell cycle progression in individual cells as a function of time. This investigation requires the processing and analysis of huge amounts of image data obtained in time-lapse microscopy. Manual image analysis is very time consuming thus costly, potentially inaccurate, and poorly reproducible. Stages of an automated cellular imaging analysis consist of segmentation, feature extraction, classification, and tracking of individual cells in a dynamic cellular population. The feature extraction and classification of cell phases are considered the most difficult tasks of such analysis. We review several techniques for feature extraction and classification. We then present our work on an automated feature weighting technique for feature selection and combine this technique with cellular phase modeling techniques for classification. These combined techniques perform the two most difficult tasks at the same time and enhance the classification performance. Experimental results have shown that the combined techniques are effective and have potential for higher performance.

9.1 Introduction

High-content screening is an integrated solution that uses images of living cells as the basic unit to produce information on drug responses for accelerated molecule drug discovery (Giuliano et al. 2003) such as functions of genes, proteins, and other molecules in normal and abnormal cellular functions (Abraham et al. 2004). Fluorescence microscopy for cell biological studies in small-scale cell biology is used in research imaging-microscopy systems to collect image data from a small number of experimental samples (Abraham et al. 2004).

High-content screening by automated fluorescence microscopy is becoming one of the most widely used research tools to assist scientists in understanding the

D.T. Tran (✉)
Faculty of Information Sciences and Engineering, University of Canberra, ACT 2601, Australia
e-mail: dat.tran@canberra.edu.au

T. Pham (ed.), *Computational Biology: Issues and Applications in Oncology*,
Applied Bioinformatics and Biostatistics in Cancer Research,
DOI 10.1007/978-1-4419-0811-7_9,

complex process of cell division or mitosis (Dunkle 2002; Fox 2003). Its power comes from the sensitivity and resolution of automated light microscopy with multiwell plates, combined with the availability of fluorescent probes that are attached to specific subcellular components, such as chromosomes and microtubules, for visualization of cell division or mitosis using standard epifluorescence microscopy techniques (Yarrow 2003).

By employing a carefully selected reporter probes and filters, fluorescence microscopy allows specific imaging of phenotypes of essentially any cell component (Murphy 2001). With these probes, we can determine both the amount of a cell component and, most critically, its distribution within the cell relative to other components. Typically, three to four different components are localized in the same cell using probes that excite at different wavelengths. Any change in cell physiology would cause a redistribution of one or more cellular components, and this redistribution provides a certain cytological marker that allows for scoring of the physiological change.

An essential task for high content screening is to measure cell cycle progression (interphase, prophase, metaphase, and telophase) in individual cells as a function of time. Cell cycle progress can be identified by measuring nuclear changes. Automated time-lapse fluorescence microscopy imaging provides an important method for the observation and study of cellular nuclei in a dynamic fashion (Hiraoka and Haraguchi 1996; Kanda et al. 1998). Stages of an automated cellular imaging analysis consist of segmentation, feature extraction, classification, and tracking of individual cells in a dynamic cellular population (Chen et al. 2006). Automatic classification of cell nuclei into interphase, prophase, metaphase, or anaphase is still a challenge in cell biology studies using fluorescence microscopy.

In time-lapse microcopy, images are usually captured in a time interval of more than 10 min. During this period, dividing nuclei may move far away from each other and daughter cell nuclei may not overlap with their parents. Given the advanced fluorescent imaging technology, there still remain technical challenges in processing and analyzing large volumes of images generated by time-lapse microscopy. The increasing quantity and complexity of image data from dynamic microscopy renders manual analysis unreasonably time consuming (Wang et al. 2007). Therefore, automatic techniques for analyzing cell-cycle progress are of considerable interest in the drug discovery process.

Being motivated by the desire to study drug effects on HeLa cells, an ovarian cancer cell line, we applied several computational techniques for identifying individual cell phase changes during a period of time. To extract useful features for the cell-phase identification task, the image segmentation of large image sequences acquired by time-lapse microscopy is necessary. The extracted data can then be used to analyze cell phase changes under drug influence. Segmenting nuclei in time-lapse microscope can be performed by various methods such as thresholding, region growing, or edge detection (MacAulay and Palcic 1988). Most of these algorithms take into account either the morphological information or the intensity information of the image. Problems may arise when trying to segment touching nuclei because it is very difficult to define the boundary of each individual nuclear. Watershed

techniques can be used to segment touching objects (Bleau and Leon 2000). To deal with the oversegmentation problem, a post process is needed to merge the fragments. A connectivity-based merging method is used to merge a tiny cell fragment with a nearby cell if it shares the maximum boundary with that cell (Umesh and Chaudhuri 2001). These authors applied their method on a set of 327 cells and a 98% correct segmentation result was reported. This method can only merge small fragments and fails if the size of cell fragments is above a preset value. The bigger fragments are considered as cells by this method. Bleau and Leon (2000) used an iterative trial and test approach to merge small regions with their nearby larger regions based on a set of volume, depth, and surface criteria. These authors applied their method to segment the vesicles in live cells; however, no experimental results were reported.

To automate the process of identifying cellular phases using time-lapse fluorescence microscopic image sequences, we first apply a shape- and size-based method which merges the oversegmented nuclear fragments. Second, we extract useful features to discriminate the shapes and intensities of different image cell phases. We then use these image features to build cell phase models using several computational algorithms. To classify an unknown cell, we extract its features, then compare those with the phase models. Figure 9.1 presents a block diagram of a typical cell phase classification system.

This chapter presents a variety of modeling techniques for cell phase classification including vector quantization (VQ), Gaussian mixture model (GMM), and hidden Markov model (HMM). In these techniques, the extracted cell features are treated equally, but they may not have the same importance as the data distribution for each feature may be different. To overcome this, we present a new approach to feature selection using an automated feature weighting technique. This technique is combined with those modeling techniques to provide an efficient way for modeling and to reduce cell phase classification error rates. We also present further investigation on fuzzy fusion. Fuzzy fusion is a formal framework that provides computational tools for the integration of data originating from different sources of information, so that the information or result of quality being better than that obtained from any single source can be obtained.

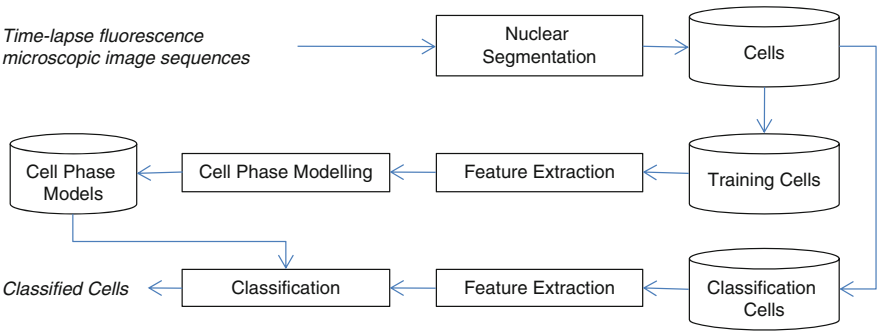

Fig. 9.1 Block diagram of a typical cell phase classification system

The chapter is organized as follows. Section 1 introduces cell classification for cancer research and drug discovery. Section 2 reviews nuclear segmentation techniques. Section 3 presents techniques for feature selection. Cell phase modeling techniques are presented in Section 4. Algorithms for modeling and classifying cell phases are presented in Section 5. Section 6 introduces fuzzy fusion. Section 7 presents experimental results. Finally, the chapter is concluded in Section 8.

9.2 Nuclear Segmentation

Nuclear segmentation is the first part of the cell phase classification system. This is a critical part since the segmentation results directly affect the classification accuracy. Manual segmentation is very time consuming and the results may not be reproducible (Ma et al. 2008). Automated segmentation techniques have been proposed to overcome this problem. The most popular techniques are based on thresholds in images and are presented here.

9.2.1 Threshold-Based Segmentation

In the threshold-based segmentation techniques, thresholds are selected manually according to a priori knowledge or automatically through image information. Some rules are then defined and segmentation is a procedure of searching for pixels that satisfy these rules. Edge-based algorithms use the edge information and find edge pixels while eliminating the noise influence (Canny 1986). Since these algorithms are based on pixels, the detected edges can contain discrete pixels, and therefore may be incomplete or discontinuous. It is necessary to apply a post-processing method such as morphological operation to make them continuous. Region-based algorithms assume that pixels inside a structure would have similar intensities (Adams and Bischof 1994). Initial seeds are first selected, then the thresholds are used to define intervals. These algorithms search for the neighbored pixels whose intensities are inside the intervals, and then merge them to expand the regions. Statistical information and a priori knowledge can be incorporated to the algorithms to eliminate the dependence on initial seeds and make the algorithm automatically (Pohle and Toennies 2001). As these algorithms mainly rely on the image intensity information, they are hard to handle the partial volume effects and control the leakage. A new two-step approach has been proposed to achieve the nuclear segmentation (Bell et al. 2008). First, a mean shift segmentation which is an over segmentation of the image is applied. These segments are then grouped together into the desired nuclear segmentation in the second step. This is achieved by a model-guided region grouping.

9.2.2 Image Thresholding

Time-lapse fluorescent microscopy images of nuclei are bright objects protruding out from a relatively uniform dark background. Thus, they can be segmented by histogram thresholding. Nuclear images are first segmented and then extracted from their background by applying a global threshold technique. The iterative self-organizing data analysis technique algorithm (ISODATA) was used to perform image thresholding (Norberto et al. 1997; Otsu 1978). By applying the ISODATA technique, an image is initially segmented into two parts using an initial threshold value. The sample mean of the gray values associated with the nuclear pixels and the sample mean of the gray values associated with the background pixels are computed. A new threshold value is then computed as the average of the two sample means. The process is repeated until the change of threshold values reaches convergence. This algorithm correctly segments most isolated nuclei, but it is unable to segment touching nuclei. The algorithm fails because it assigns the pixels to only two different groups (nuclear and background). If two nuclei are so close and there are no background pixels between them, the algorithm will not be able to separate them.

To overcome this problem, a watershed algorithm was used (Norberto et al. 1997; Otsu 1978; Bleau and Leon 2000). The watershed algorithm first calculates the Euclidean distance map (EDM) of the binary image obtained from the ISODATA algorithm. It then finds the ultimate eroded points (UEP), which are the local maxima of the EDM. The watershed algorithm then dilates each of the UEP as far as possible – either until the edge of the nuclear or the edge of the region of another UEP is reached. However, the watershed algorithm fails when there is more than one ultimate eroded point within the same nucleus. In such cases, the nuclear will be incorrectly divided into several fragments. A fragment merging algorithm below is therefore needed to correct such segmentation errors.

9.2.3 Fragment Merging Algorithm

Nuclei are usually elliptic objects with various shape parameters. In such cases, the compactness can be used to describe the shape of the nuclei. Compactness is defined as the ratio of the square of the perimeter of the nuclear to the area of the nuclear. The value of 1 indicates a circular nuclear. Compactness increases as the contour of the nuclear deviates from the circular shape. If a round nuclear is divided into several fragments, the compactness of each fragment will be larger than the compactness of the entire nuclear. On the basis of the observation of nuclear shapes and sizes, we have developed a fragment merging technique. This technique can identify oversegmented nuclear fragments and then merges them into single nuclear units.

The procedure can be described as follows. Let N be the total number of segmented objects found by the watershed segmentation algorithm. Let T be the minimum size of a nuclear in the image. In this work, a threshold value of 100

pixels is chosen as no single nuclear size is smaller than 100, and larger threshold value will cause small nuclei be identified as fragments and merged with nearby touching nuclei.

All touching objects (nuclei) are evaluated and checked. Two objects are considered touching if they belong to the same object in the binary image before the watershed algorithm is applied. This iterative merging process finds the smallest touching objects in each iteration and then uses the checking process to update the segmentation until no more touching objects can be merged. The checking process can be described as follows:

- If the size of a touching object is less than T, it is merged with its smallest touching neighbor.
- If the size of a touching object is greater than T, three compactness values are calculated: the object, touching neighbor of the object, and the two objects as a whole. If the calculated compactness decreases after this merging, these two objects are merged.

9.3 Feature Extraction

After the nuclear segmentation has been performed, it is necessary to perform a morphological closing process on the resulting binary images to smooth the nuclear boundaries and fill holes insides the nuclei. These binary images are then used as a mask to define relevant features in the original image. From this resulting image, features can be extracted.

The ultimate goal for feature selection is to assign correct phase to cells via the training of some identification technique. A large set of cell-nuclear features are extracted based on the experience of biologists. An optimal feature subset is then determined to minimize the classification error rate. It is impossible to use an exhaustive search to determine the subset due to the large amount testing that would be involved. Sequential forward selection and automated feature weighting methods are used for this purpose.

9.3.1 Sequential Forward Selection

Sequential forward selection is a bottom-up search method where features are aded to the optimal feature subset one by one. Initially the subset is empty. In each stage, only one feature is selected from the feature set to add to the subset. If the subset does not yield a lower classification error rate, then the added feature will be replaced by a new feature selected from the feature set. The procedure is terminated when no feature that can reduce the classification error rate is found.

This method was applied to the data set provided by the Department of Cell Biology at the Harvard Medical School. The feature set based on the experience of

biologists contained 13 features including maximum intensity, mean, standard deviation, major axis, minor axis, perimeter, compactness, minimum intensity, sum, area, CP, roughness, and ratio. After applying the sequential forward selection method, the optimal subset of seven features was selected. These features include maximum intensity, mean, stand deviation, major axis, minor axis, perimeter, and compactness (Chen et al. 2006).

9.3.2 Automated Feature Weighting

Each feature in the feature set based on the experience of biologists is associated by a weight whose value represents the degree of selection. The values of these weights are determined when the cell phase models are built. Modeling algorithms need to be extended accordingly to integrate the feature weighting algorithm. When the cell phase models are built, the weight values are also determined and the features whose weight values are smaller than a preset threshold will be discarded.

The advantage of this method is that it is faster than the sequential forward selection method and that it can be integrated to most of current modeling techniques such as Markov modeling, GMM, and VQ.

9.3.3 Feature Scaling

Because the feature values have different ranges, the scaling of features is therefore calculated as follows

$$x'_{tm} = \frac{x_{tm} - \mu_m}{\sigma_m}, \quad \sigma_m = \frac{1}{T}\sum_{t=1}^{T} |x_{tm} - \mu_m| \tag{9.1}$$

where x_{tm} is the mth feature of the tth nucleus, μ_m the mean value of all T cells, and σ_m the mean absolute deviation.

9.4 Cell Phase Modeling

Several statistical techniques have been applied to cell phase modeling. HMM is the most important technique since it employs temporal information in cell phase sequences. The underlying assumption of the HMM is that the considering cell phases can be well characterized as a parametric random process, and that the parameters of the stochastic process can be estimated in a precise, well-defined manner. The HMM technique also provides a reliable way of recognizing speech for a wide range of applications (Furui 1997; Juang 1998; Kulkarni 1995; Rabiner and Juang 1993).

There are two assumptions in the first-order HMM. The first one is the Markov assumption, that is a new state is entered at each time t based on the transition probability, which only depends on the previous state. It is used to characterize the sequence of the time frames of a pattern. The second is the output-independence assumption, that is the output probability depends only on the state at that time regardless of when and how the state is entered. A process satisfying the Markov assumption is called a Markov model. An observable Markov model (OMM) is a process where the output is a set of states at each instant of time and each state corresponds to an observable event. The HMM is a doubly stochastic process with an underlying Markov process that is not directly observable (hidden) but can be observed through another set of stochastic processes that produce observable events in each of the states.

If the temporal information is not taken into account, GMM is used. The GMM technique uses a mixture of Gaussian densities to model the distribution of cell feature vectors extracted from the training data. The GMM technique is also regarded as the one-state continuous HMM technique. When little training data are available, VQ technique is also effective. VQ modeling partition the cell feature space to convert the cell feature set into a small set of distinct feature vectors using a clustering technique. Advantages of this reduction are reduced storage and computation. The distinct vectors are called code vectors and the set of code vectors that best represents the training set is called the codebook regarded as a cell phase model. Currently, k-means clustering is used in VQ modeling.

Fuzzy techniques have also been applied to cell phase modeling. Fuzzy clustering techniques such as fuzzy c-means and fuzzy entropy have been used to design reestimation algorithms for fuzzy HMM, fuzzy GMM, and fuzzy VQ (Tran and Wagner 2002; Tran and Pham 2007; Tran et al. 2008).

A recent technique that can integrate to both statistical modeling and fuzzy modeling is automated feature weighting, which has been mentioned in Sect. 9.3. The current modeling techniques cannot select features automatically and they also treat all features equally. The combined techniques between the automated feature weighting and the current modeling techniques provide an efficient way to select appropriate features for cell phase models when they are being built to reduce the cell phase classification error rate. There have been some algorithms that were proposed to calculate weights for VQ and applied to cell phase modeling (Tran and Pham 2007; Tran et al. 2008). However, a generic framework that can apply to HMM, GMM, and VQ modeling does not exist.

This chapter proposes a novel combined feature weighting-modeling framework for HMM using maximum likelihood (ML) estimation. A generic objective function is proposed and maximizing this function will result in an algorithm for calculating weights as well as HMM parameters. Algorithms for the combined feature weighting-GMM and feature weighting-VQ techniques will also be determined from the algorithm for the combined feature weighting-HMM.

9.4.1 Feature Weighting-HMM

Let $S = \{s_1, s_2, \ldots, s_T\}$ and $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_T\}$ be a sequence of states and a sequence of continuous cell feature vectors, respectively. The compact notation $\Lambda = \{\pi, A, B\}$ indicates the complete parameter set of the HMM where

- $\pi = \{\pi_i\}$, $\pi_i = P(s_1 = i|\Lambda)$: the initial state distribution
- $A = \{a_{ij}\}$, $a_{ij} = P(s_t = j|s_{t-1} = i, \Lambda)$: the state transition probability distribution, and
- $B = \{b_j(\mathbf{x}_t)\}$, $b_j(\mathbf{x}_t) = P(\mathbf{x}_t|s_t = j, \Lambda)$: the output probability distribution of feature vector $\mathbf{x}_t$ in state j.

The following constraints are applied:

$$\sum_{i=1}^{N} \pi_i = 1, \quad \sum_{j=1}^{N} a_{ij} = 1, \text{ and } \int b(\mathbf{x}_t)\mathrm{d}\mathbf{x}_t = 1 \tag{9.2}$$

The HMM parameters are estimated such that in some sense they best match the distribution of the feature vectors in $\mathbf{x}$. The most widely used training method is the ML estimation. For a sequence of feature vectors $\mathbf{x}$, the likelihood of the HMM is

$$P(\mathbf{X}|\Lambda) = \prod_{t=1}^{T} P(\mathbf{x}_t|\Lambda) \tag{9.3}$$

The aim of ML estimation is to find a new parameter model $\bar{\Lambda}$ so that $P(\mathbf{X}|\bar{\Lambda}) \geq P(\mathbf{X}|\Lambda)$. Since the expression in (9.3) is a nonlinear function of parameters in Λ its direct maximization is not possible. However, parameters can be obtained iteratively using the expectation-maximization (EM) algorithm (Dempster et al. 1997). An auxiliary function Q is used

$$Q(\Lambda, \bar{\Lambda}) = \sum_{t=1}^{T-1} \sum_{i=1}^{N} \sum_{j=1}^{N} P\left(s_t = i, s_{t+1} = j|\mathbf{X}, \Lambda\right) \log\left[\bar{a}_{ij}\bar{b}_{ij}\left(\mathbf{x}_{t+1}\right)\right] \tag{9.4}$$

where $\bar{\pi}_{s_1=j}$ is denoted by $\bar{a}_{s_0=i\, s_1=j}$ for simplicity. The most general representation of the output probability distribution is a mixture of Gaussians

$$b_j(\mathbf{x}_t) = P(\mathbf{x}_t|s_t = j, \Lambda) = \sum_{k=1}^{K} P(k|s_t = j, \Lambda) P(\mathbf{x}_t|k, s_t = j, \Lambda) \tag{9.5}$$

This can be re-written as

$$b_j(\mathbf{x}_t) = \sum_{k=1}^{K} c_{jk} N\left(\mathbf{x}_t, \boldsymbol{\mu}_{jk}, \Sigma_{jk}\right) \tag{9.6}$$

where $c_{jk} = P(k|s_t = j, \Lambda)$, $j = 1, \ldots, N$, $k = 1, \ldots, K$ are mixture coefficients, and $N(\mathbf{x}_t, \boldsymbol{\mu}_{jk}, \Sigma_{jk})$ is a Gaussian with mean vector $\boldsymbol{\mu}_{jk}$ and covariance matrix Σ_{jk} for the kth mixture component in state j. The mixture coefficients satisfy the following conditions for all $j = 1, \ldots, N$

$$c_{jk} > 0 \quad \text{and} \quad \sum_{k=1}^{K} c_{jk} = 1 \tag{9.7}$$

In order to combine with the automated feature weighting technique, a weight w_{jkm}^{α} is associated with the mth feature as follows

$$\log N(\mathrm{x}_t, \bar{\mu}_{jk}, \bar{\Sigma}_{jk}) = \sum_{m=1}^{M} w_{jkm}^{\alpha} \log P(x_{tm}|k, s_t = j, \bar{\Lambda}) \tag{9.8}$$

where

$$P(x_{tm}|k, s_t = j, \Lambda) = \frac{1}{\sqrt{2\pi\sigma_{jkm}^2}} e^{-(x_{tm}-\mu_{jkm})^2/2\sigma_{jkm}^2} \tag{9.9}$$

σ_{jkm} is the mth variance component in Gaussian k and state j, w_{jkm}^{α}, $m = 1, 2, \ldots, M$ are components of an M-dimensional weight vector $\mathbf{w}_{jm}^{\alpha}$, and α is a fuzzy parameter weight for w_{jkm}^{α}. The weight values satisfy the following conditions:

$$0 \leq w_{jkm} \leq 1 \quad \forall m, \sum_{m=1}^{M} w_{jkm} = 1 \tag{9.10}$$

It can be seen that if all of the weight values are equal, the proposed expression of Gaussian distribution $N(\mathbf{x}_t, \bar{\boldsymbol{\mu}}_{jk}, \bar{\Sigma}_{jk})$ in Eq. (9.8) becomes the normal expression for Gaussian distribution as seen in statistics and probability theory (Tran and Wagner 1998).

Maximizing the likelihood function in (9.2) can be obtained by maximizing the objective function in (9.3) over $\bar{\Lambda}$ and the weight vector $\mathbf{w}_{jm}^{\alpha}$. The basic idea of this combined technique is that the function $Q_j(\Lambda, \bar{\Lambda})$ is maximized over the variable w_{jkm} on the assumption that the weight vector $\mathbf{w}_{jm}$ identifies a good contribution of the features. Using the well-known Lagrange multiplier method, maximizing the function $Q_j(\Lambda, \bar{\Lambda})$ in (9.4) using (9.7) and (9.10) gives

$$w_{jkm} = \frac{1}{\sum_{n=1}^{M} \left(D_{jkm}/D_{jkn}\right)^{1/(\alpha-1)}} \tag{9.11}$$

where $\alpha \neq 1$ and

$$D_{jkm} = -\sum_{t=1}^{T} P(k|\mathrm{x}_t, s_t = j, \Lambda) \log P(x_{tm}|k, s_t = j, \bar{\Lambda}) \tag{9.12}$$

The mixture coefficients, mean vectors, and covariance matrices are calculated by maximizing the function in (9.4) over $\bar{\Lambda}$ using (9.2) and (9.7). We obtain

$$\bar{c}_{jk} = \frac{1}{T}\sum_{t=1}^{T} P(k|\mathbf{x}_t, s_t = j, \Lambda) \tag{9.13}$$

$$\bar{\mu}_{jk} = \frac{\sum_{t=1}^{T} P(k|\mathrm{x}_t, s_t = j, \Lambda)\mathrm{x}_t}{\sum_{t=1}^{T} P\left(k|\mathrm{x}_t, s_t = j, \Lambda\right)} \tag{9.14}$$

$$\bar{\Sigma}_{jk} = \frac{\sum_{t=1}^{T} P(k|\mathrm{x}_t, s_t = j, \Lambda)\left(\mathrm{x}_t - \mu_{jk}\right)\left(\mathrm{x}_t - \mu_{jk}\right)'}{\sum_{t=1}^{T} P(k|\mathrm{x}_t, s_t = j, \Lambda)} \tag{9.15}$$

where the prime denotes vector transposition, and

$$P\left(k|\mathrm{x}_t, s_t = j, \Lambda\right) = \frac{c_{jk} N\left(\mathrm{x}_t, \mu_{jk}, \Sigma_{jk}\right)}{\sum_{n=1}^{K} c_{jn} N\left(\mathrm{x}_t, \mu_{jn}, \Sigma_{jn}\right)} \tag{9.16}$$

The initial state distribution and state transition distribution are also determined:

$$\bar{\pi}_i = \gamma_1(i) \quad \bar{a}_{ij} = \frac{\sum_{t=1}^{T-1} \xi_t\left(i, j\right)}{\sum_{t=1}^{T-1} \gamma_t\left(i\right)} \tag{9.17}$$

where

$$\gamma_t\left(i\right) = \sum_{j=1}^{N} \xi_t\left(i, j\right),\ \xi_t\left(i, j\right) = P\left(s_t = i, s_{t+1} = j|\mathrm{X}, \Lambda\right) \tag{9.18}$$

The advantage of this approach is that when the weighting values w_{jkm}^{α} have the same value, the combined feature weighting-HMM becomes the standard HMM in the ML estimation. Therefore, the proposed approach can be considered as a generic framework and can extend to other models that relate to the HMM such as GMM and VQ and other estimation methods such as minimum classification error and maximum a posteriori.

9.4.2 Feature Weighting-OMM

If the sequence of continuous cell feature vectors $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_T\}$ is replaced by the sequence of cell phases and is also used as the sequence of states, the feature weighting-HMM becomes the feature weighting-OMM. The sequence of cell phases is observable and hence the sequence of states is not hidden.

The initial state distribution and state transition distribution in (9.17) and (9.18) are recalculated as follows:

$$\bar{\pi}_i = \frac{n_i}{\sum_{s=1}^{M} n_s}, \quad \bar{a}_{ij} = \frac{n_{ij}}{\sum_{s=1}^{M} n_{is}} \tag{9.19}$$

where n_i and n_{ij} are the number of occurrences of π_i and a_{ij}, respectively, in the sequence of cell phases. Other equations remain unchanged.

9.4.3 Feature Weighting-GMM

The feature weighting-GMM can be obtained by setting the number of states in feature weighting-HMM to one. The feature weighting-GMM parameters consist of the mixture weight c_{jk}, mean vector $\boldsymbol{\mu}_{jk}$, covariance matrix Σ_{jk}, and feature weight $\mathbf{w}_{jm}$. The estimation equations in (9.11), (9.13), (9.14), (9.15), and (9.16) are used to calculate the GMM parameters.

9.4.4 Feature Weighting-Fuzzy GMM

The feature weighting-fuzzy GMM is a fuzzy version of the feature weighting-GMM. Parameters in (9.13), (9.14), (9.15), and (9.16) are recalculated as follows

$$\bar{c}_{jk} = \frac{\sum_{t=1}^{T} u_{jkt}^{\alpha}}{\sum_{k=1}^{K}\sum_{t=1}^{T} u_{jkt}^{\alpha}} \tag{9.20}$$

$$\bar{\mu}_{jk} = \frac{\sum_{t=1}^{T} u_{jkt}^{\alpha} \mathrm{x}_t}{\sum_{t=1}^{T} u_{jkt}^{\alpha}} \tag{9.21}$$

$$\bar{\Sigma}_{jk} = \frac{\sum_{t=1}^{T} u_{jkt}^{\alpha} \left(\mathrm{x}_t - \mu_{jk}\right)\left(\mathrm{x}_t - \mu_{jk}\right)'}{\sum_{t=1}^{T} u_{jkt}^{\alpha}} \tag{9.22}$$

where

$$u_{jkt} = \left[\sum_{m=1}^{K} \left(\frac{d_{jkt}}{d_{jmt}}\right)^{2/(m-1)}\right]^{-1} \tag{9.23}$$

$$d_{jkt}^2 = -\log P\left(\mathbf{x}_t | k, s_t = j, \bar{\Lambda}\right) \tag{9.24}$$

9.4.5 Feature Weighting-VQ

In feature weighting-VQ modeling (Tran and Pham 2008), the model Λ is a set of cluster centers $\Lambda = \{\boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \ldots, \boldsymbol{\mu}_K\}$ where $\boldsymbol{\mu}_k = (\mu_{k1}, \mu_{k2}, \ldots, \mu_{kM})$, $k = 1, 2, \ldots, K$ are code vectors (also mean vectors). Each code vector $\boldsymbol{\mu}_k$ is assigned to an encoding region R_k in the partition $\Omega = \{R_1, R_2, \ldots, R_K\}$. Then the source vector $\mathbf{x}_t$ can be represented by the encoding region R_k, and is expressed by

$$V(\mathrm{x}_t) = \mu_k \quad \text{if} \quad \mathrm{x}_t \in R_k \tag{9.25}$$

Let $U = [u_{kt}]$ be a matrix whose elements are memberships of x_t in the nth cluster, $k = 1, 2, \ldots, K, t = 1, 2, \ldots, T$. A K-partition space for $\mathbf{X}$ is the set of matrices U such that

$$u_{kt} \in \{0, 1\} \; \forall \, k, t, \quad \sum_{k=1}^{K} u_{kt} = 1 \; \forall \, t, \quad 0 < \sum_{t=1}^{T} u_{kt} < T \; \forall k \tag{9.26}$$

where $u_{kt} = u_k(\mathbf{x}_t)$ is 1 or 0, according to whether $\mathbf{x}_t$ is or not in the kth cluster, $\sum_{k=1}^{K} u_{kt} = 1 \; \forall \, t$ means each $\mathbf{x}_t$ is in exactly one of the K clusters, and $0 < \sum_{t=1}^{T} u_{kt} < T \; \forall \, k$ means that no cluster is empty and no cluster is all of $\mathbf{X}$ because of $1 < K < T$.

The feature weighting-VQ technique is based on minimization of the $J(U, W, \Lambda)$ function obtained from the $Q\left(\Lambda, \bar{\Lambda}\right)$ function in (9.2) by removing the expressions that contains state parameters in the HMM and Gaussian parameters in the GMM. The $J(U, W, \Lambda)$ function is also considered as the sum-of-squared-errors function (the index j for state is omitted) as follows

$$J(U, W, \Lambda) = \sum_{k=1}^{K} \sum_{t=1}^{T} u_{kt} \sum_{m=1}^{M} w_{km}^{\alpha} d_{ktm} \tag{9.27}$$

where $\bar{\Lambda}$ is included in d_{ktm}, which is the Euclidean norm of $(\mathbf{x}_t - \boldsymbol{\mu}_k)$. Similarly, the well-known Lagrange multiplier method is used to obtain the following equations for feature weighting-VQ

$$\mu_k = \frac{\sum_{t=1}^{T} u_{kt} \mathrm{x}_t}{\sum_{t=1}^{T} u_{kt}}, \quad 1 \le k \le K \tag{9.28}$$

$$u_{kt} = \begin{cases} 1: & d_{kt} < d_{jt}, \quad j = 1, ..., K, j \neq k \\ 0: & \text{otherwise} \end{cases} \tag{9.29}$$

$$w_{km} = \frac{1}{\sum_{n=1}^{M} (D_{km}/D_{kn})^{1/(\alpha-1)}}, \quad D_{km} = \sum_{t=1}^{T} u_{kt} d_{ktm} \tag{9.30}$$

where

$$d_{ktm} = (c_{km} - x_{tm})^2, \quad d_{kt} = \sum_{m=1}^{M} w_{km}^{\alpha} d_{ktm} \tag{9.31}$$

9.4.6 Feature Weighting-Fuzzy VQ

In feature weighting-fuzzy VQ modeling, the matrix $U = [u_{kt}]$ is redefined as follows (Tran and Pham 2008)

$$u_{kt} \in [0, 1] \; \forall k, t, \quad \sum_{k=1}^{K} u_{kt} = 1 \; \forall t, \quad 0 < \sum_{t=1}^{T} u_{kt} < T \; \forall k \tag{9.32}$$

where $0 \leq u_{kt} \leq 1$ denoting the degree of fuzziness.

The feature weighting-fuzzy VQ technique is based on minimization of the following function

$$J(U, W, \Lambda) = \sum_{k=1}^{K} \sum_{t=1}^{T} u_{kt}^{\beta} \sum_{m=1}^{M} w_{km}^{\alpha} d_{ktm} \tag{9.33}$$

where $\beta > 1$ and others are the same as those in feature weighting-VQ. The Eqs. (9.28), (9.29), and (9.30) are replaced by (9.34), (9.35), and (9.36), respectively, as follows

$$\mu_k = \frac{\sum_{t=1}^{T} u_{kt}^{\beta} x_t}{\sum_{t=1}^{T} u_{kt}^{\beta}}, \quad 1 \leq k \leq K \tag{9.34}$$

$$u_{kt} = \left[\sum_{j=1}^{K} \left(d_{kt} / d_{jt} \right)^{2/m-1} \right]^{-1} \tag{9.35}$$

$$w_{km} = \frac{1}{\sum_{n=1}^{M} \left(D_{km} / D_{kn} \right)^{1/(\alpha-1)}}, \quad D_{km} = \sum_{t=1}^{T} u_{kt}^{\beta} d_{ktm} \tag{9.36}$$

9.5 Algorithms for Modeling and Classifying Cell Phases

The modeling and classifying algorithms for the combined techniques are summarized in following sections.

9.5.1 Modeling Algorithm

1. Give a training data set $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_T\}$, where $\mathbf{x}_t = (x_{t1}, x_{t2}, \ldots, x_{tM})$, $t = 1, 2, \ldots, T$.
2. Initialize parameters at random satisfying (9.2), (9.7), and (9.10).
3. Give $\alpha \neq 1$ and $\varepsilon > 0$ (small real number).
4. Set $i = 0$ and $Q^{(i)}\left(\Lambda, \bar{\Lambda}\right)$ in (9.4) or $J^{(i)}(U, W, \Lambda)$ in (9.27) and (9.33) to a small real number. Iteration:
   a. Compute weight values:
      - For HMM, OMM, and GMM: using (9.11) and (9.12)
      - For VQ: using (9.30)
      - For fuzzy VQ: using (9.36)
   b. Compute initial state and state transition parameters:
      - For HMM: using (9.17) and (9.18)
      - For OMM: using (9.19)
   c. Compute model parameters:
      - For GMM: using (9.13)–(9.16)
      - For VQ: using (9.28) and (9.29)
      - For fuzzy VQ: using (9.34) and (9.35)
   d. Compute $Q^{(i+1)}\left(\Lambda, \bar{\Lambda}\right)$ using (9.4) or $J^{(i+1)}(U, W, \Lambda)$ using (9.27) and (9.33). If the difference between $Q^{(i+1)}\left(\Lambda, \bar{\Lambda}\right)$ and $Q^{(i)}\left(\Lambda, \bar{\Lambda}\right)$ or the difference between $J^{(i+1)}(U, W, \Lambda)$ and $J^{(i)}(U, W, \Lambda)$ is less than e, then set $Q^{(i)}\left(\Lambda, \bar{\Lambda}\right) = Q^{(i+1)}\left(\Lambda, \bar{\Lambda}\right)$ or $J^{(i)}(U, W, \Lambda) = J^{(i+1)}(U, W, \Lambda)$, and $i = i + 1$ and go to step (a).

9.5.2 Classification Algorithm

Assuming $\left\{\Lambda^{(1)}, \Lambda^{(2)}, \ldots, \Lambda^{(p)}\right\}$ are p cell phase models that are built using the modeling algorithm. Given an unknown cell feature vector $\mathbf{x}$, the task is to classify $\mathbf{x}$ into one of the p cell phase models. The following algorithm is proposed:

1. Given an unknown feature vector $\mathbf{x}$ and the set of models $\left\{\Lambda^{(1)}, \Lambda^{(2)}, \ldots, \Lambda^{(p)}\right\}$
2. Calculate the distance $d^{(i)} = \sum_{k=1}^{K} \sum_{m=1}^{M} w_{km}^{\alpha} d_{km}^{(i)}$, where $d_{km}^{(i)} = \left(c_{km}^{(i)} - x_m\right)^2$, $i = 1, \ldots, p$, or the probabilities $P\left(\mathbf{x}|\Lambda^{(i)}\right)$, $i = 1, \ldots, p$ as follows

$$P\left(x|\Lambda^{(i)}\right) = \sum_{k=1}^{K} c_k N\left(x, \mu_k^{(i)}, \Sigma_k^{(i)}\right) \tag{9.37}$$

3. The recognized model i^* is determined by one of the following rules:

$$i^* = \underset{i \in \{1,2,\ldots,p\}}{\arg\min} \; d^{(i)} \tag{9.38}$$

or

$$i^* = \underset{i \in \{1,2,\ldots,p\}}{\arg\max} \; P\left(\mathrm{x}|\Lambda^{(i)}\right) \tag{9.39}$$

9.6 Fuzzy Fusion of Classifiers

We have presented in the previous sections the modeling methods including HMM, GMM, VQ, and their combinations with the automated feature weighting technique as well as their fuzzy versions. With a number of classifiers obtained from those modeling techniques, we describe a novel way to combine their classification results using fuzzy fusion based on the mathematical concepts of fuzzy measures and fuzzy integrals (Pham et al. 2007). The concept of information fusion has become an active area of systems research and has many applications in pattern recognition, image analysis, robotics, and management science.

In general, information fusion is a formal framework that provides computational tools for the integration of data originating from different sources of information so that the information or result of quality being better than that obtained from any single source can be obtained. The motivation is based on the fact that it is theoretically difficult to design a single classifier that can detect well all the features with a similar level of accuracy. This difficulty is due to the complexity of implementing several learning algorithms in a single classifier because each independent algorithm may work better with some features than the other algorithms and vice versa.

In this study, we were motivated to use fuzzy integrals as the operators for information fusion because fuzzy integrals constitute a generalization of aggregation operators including many widely used operators such as minimum, maximum, order statistic, weighted sum, and ordered weighted sum. We describe the properties of a fuzzy measure and the operations of the fuzzy integrals on the subsets of the fuzzy measure as follows.

Let $Y = \{g_1, \ldots, g_n\}$ be a set of the degrees of importance of n classifiers. These elements can be considered as the classification rates of classifiers for the same cell phase. In this study, we model the fuzzy densities of a classifier on an adaptive way such that it is variable depending on its overall classification rate for a particular cell phase. In other words, the fuzzy density g_i for each cell phase of each classifier can be obtained as the percentage of the correct classification for the corresponding cell phase using the training data.

A fuzzy measure g over a finite set Y has the following properties (Sugeno 1977):

1. $g(\varnothing) = 0$; and $g(Y) = 1$
2. If $A \subset B$ then $g(A) \leq g(B)$

The Sugeno measure or the g_λ fuzzy measure satisfies the following additional condition for some $\lambda > -1$:

$$g_\lambda (A \cup B) = g_\lambda(A) + g_\lambda(B) + \lambda g_\lambda(A) g_\lambda(B) \tag{9.40}$$

The value of λ can be calculated regarding to the condition $g(Y) = 1$:

$$\lambda + 1 = \prod_{i=1}^{n} \left(1 + \lambda g^i\right) \tag{9.41}$$

Given the values of the g_λ fuzzy measure, which can be determined by using (9.40), a fuzzy integral, which is an aggregation operator, can be computed. The relationship is that fuzzy integrals are integrals of a real function with respect to a fuzzy measure. There are several definitions of fuzzy integrals but the most popular two are the Sugeno (1977) and the Choquet (1953) integrals. These two fuzzy integrals are defined as follows.

Let C be a set of classifiers, $h : C \rightarrow [0$ (Giuliano et al. 2003), and let $h(c_i)$ denote the classification score of classifier c_i. The Sugeno integral of h over $A \subset C$ with respect to the fuzzy measure g can be calculated as follows:

$$\int_A h(c_i) \circ g = \sup_{\alpha \in [0,1]} [\alpha \wedge g (A \cap H_\alpha)] \tag{9.42}$$

where $H_\alpha = \{c_i | h(c_i) \geq \alpha\}$.

For a finite set of elements $C = \{c_1, \ldots, c_n\}$ where the elements are sorted so that $h(c_i)$ is a descending function, that is $h(c_1) \geq h(c_2) \geq \ldots \geq h(c_n)$, the discrete Sugeno integral, which represents the fused result, can be calculated as follows:

$$S_g(h) = \vee_{i=1}^{n} [h(c_i) \wedge g (H_i)] \tag{9.43}$$

where $H_i = \{c_1, \ldots, c_i\}$.

The discrete Choquet integral is defined as

$$C_g(h) = \Sigma_{i=1}^{n} [h(c_i) - h (c_{i-1})] g (A_i)] \tag{9.44}$$

where $h(c_1) \leq h(c_2) \leq \ldots \leq h(c_n)$, $h(c_0) = 0$ and $A_i = \{c_1, \ldots, c_n\}$.

Let $f(C, p)$ be the output of either the Sugeno or the Choquet integral as the score for a particular cell phase p from the fusion of a set of classifiers C. On the basis of the fusion rule, the phase p^* is selected as the corrected phase if it has the maximum integrated score:

$$p^* = \arg\max_p f(C | p) \tag{9.45}$$

9.7 Experimental Results

9.7.1 Data Set

The data set described by Pham et al. (2006, 2007) contains 375,841 cells in 892 nuclear sequences. The average number of cells per sequence is 421. Imaging was performed by time-lapse fluorescence microscopy with a time interval of 15 min. Two types of sequences were used denoting drug treated and untreated. Cell cycle progress was affected by drug and some or all of the cells in the treated sequences were arrested in metaphase. Cell cycle progress in the untreated sequences was not affected. Cells without drug treatment will usually undergo one division during this period.

9.7.2 Feature Extraction

Extracting cell features has been presented in Sect. 9.3. The subset of seven features were used in the HMM, GMM, and VQ modeling techniques. The complete set of 13 cell features were used in the combined techniques including FW-HMM, FW-GMM, and FW-VQ, where FW stands for feature weighting. Because the feature values have different ranges, the scaling of features presented in Sect. 9.3.3 was applied.

9.7.3 Initialization and Constraints on Parameters During Training

It was shown in the literature that no significant difference in pattern recognition was found by using different initialization methods (Rabiner and Juang 1993). Therefore, modeling techniques based on GMM and VQ were initialized as follows:

- Feature Weighting (FW): The parameter α was set to 5.0 (Tran and Pham 2008).
- VQ: Fuzzy membership functions in fuzzy VQ models were randomly initialized. The degree of fuzziness β was set to 1.1. Choosing the appropriate values was based on our previous work for speech and speaker recognition (Tran and Wagner 1998).
- GMM: Mixture weights, mean vectors, covariance matrices, and fuzzy membership functions were initialized with essentially random choices. Covariance matrices are diagonal, that is $[\Sigma_k]_{ii} = \sigma_k^2$ and $[\Sigma_k]_{ij} = 0$ if $i \neq j$, where σ_k^2, $1 \leq k \leq K$ are variances. A variance limiting constraint was applied to all GMMs using diagonal covariance matrices. This constraint places a minimum variance value σ_k^2 on elements of all variance vectors in the GMM, that is, $\sigma_k^2 = \sigma_{\min}^2$ if $\sigma_k^2 \leq \sigma_{\min}^2$. In our experiments, $\sigma_{\min}^2 = 0.01$.

9.7.4 Experimental Results

There are five phases to be identified: interphase, prophase, metaphase, anaphase, and arrested metaphase. We divide the data set into five subsets for training five cell phase models and a subset for classification. Each of the 5 training subsets for 5 phases contains 5,000 cell feature vectors, which are extracted from the cell sequences labeled from 590 to 892. The classification subset contains sequences labeled from 1 to 589. There are 249,547 cell feature vectors in this classification subset.

The modeling methods HMM, GMM, and VQ used 7 features and the combined modeling methods FW-HMM, FW-GMM, and FW-VQ used 12 features to train cell phase models.

The cell phase classification results are presented in Fig. 9.2 for seven model sizes 2, 4, 8, 16, 32, 64, and 128. The VQ and FW-VQ model parameters are cluster centers only. The GMM and FW-GMM model parameters include mixture weights, mean vectors, and covariance matrices. The OMM and FW-OMM model parameters include the GMM parameters and Markov state parameters. It can be seen that the more model parameters were considered, the better cell phase classification rate is.

The results also show that the combined techniques always improve the cell phase classification rates. The feature weighting technique associates a weight to each

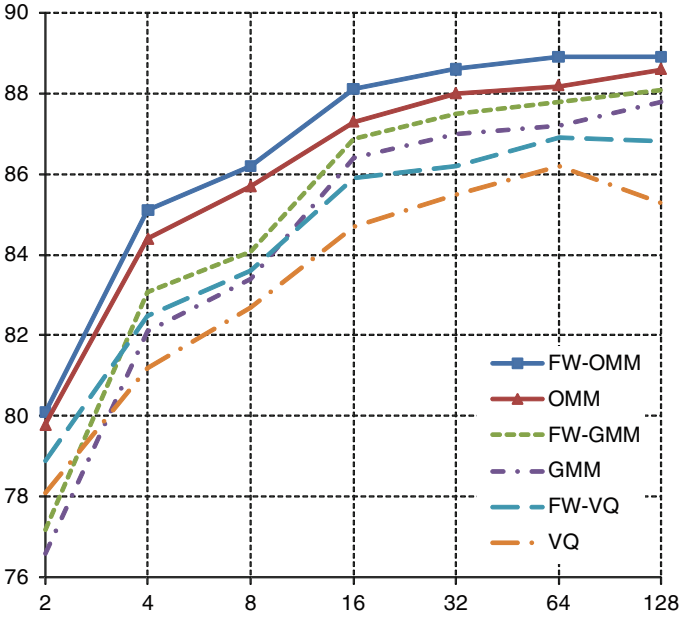

Fig. 9.2 Cell phase classification rates (in %) for modeling methods where FW, OMM, GMM, and VQ stand for feature weighting, observable Markov model, Gaussian mixture model, and vector quantization, respectively. Model sizes 2, 4, 8, 16, 32, 64, and 128 were considered

feature and these weight values were adjusted properly when the cell phase models are being trained; therefore, we obtained the optimal cell phase models when the combined techniques were used.

9.8 Conclusion

We have applied several pattern recognition methods for the classification of cell phases using time-lapse fluorescence microscopic image sequences. The selection of useful features is a very important task for any classifier. In this chapter, we have applied the automated feature weighting technique to select features when training cell phase models. The combined feature weighting and modeling techniques could certainly enhance the performance of these classifiers, particularly the combined feature weighting and OMM technique has achieved the highest classification rate.

Molecular imaging is an exciting area of research in life sciences by providing an outstanding tool for the study of diseases at the molecular or cellular levels. Some molecular imaging techniques have been implemented for clinical applications. To contribute to this emerging imaging technology, we have presented and discussed several computational models for the classification of cellular phases based on fluorescent imaging data. This task is an important component for any computerized imaging system that automates the screening of high-content, high-throughput fluorescent images of mitotic cells to aid biomedical or biological researchers to study the mitotic data at dynamic ranges for various applications including the study of the complexity of cell processes, and the screening of novel anti-mitotic drugs as potential cancer therapeutic agents.

Acknowledgement This work was supported by the ARC under project DP0665598 to T. Pham.

References

Giuliano, K.A., Haskins, J.R., and Taylor, D.L.: Advances in high content screening for drug discovery. In: ASSAY and Drug Development Technologies, vol. 1, no. 4, pp. 565–577 (2003)

Abraham, V.C., Taylor, D.L., and Haskins, J.R.: High content screening applied to large-scale cell biology. In: Trends in Biotechnology, Elsevier, vol. 22, no. 1, pp. 15–23 (2004)

Dunkle, R.: Role of image informatics in accelerating drug discovery and development. In: Drug Discovery World, vol. 7, pp. 7–11 (2002)

Fox, S.: Accommodating cells in HTS. In: Drug Discovery World, vol. 5, pp. 21–30 (2003)

Feng, Y.: Practicing cell morphology based screen. In: European Pharmaceutical Review, vol. 7, pp. 75–82 (2002)

Yarrow, J.C., et al.: Phenotypic screening of small molecule libraries by high throughput cell imaging. In: Comb Chem High Throughput Screen, vol. 6, pp. 279–286 (2003)

Murphy, D.B.: Fundamentals of light Microscopy and Electronic Imaging, Wiley-Liss (2001)

Hiraoka, Y. and Haraguchi, T.: Fluoresence imaging of mammalian living cells. In: Chromosome Res, vol. 4, pp. 173–176 (1996)

Kanda, T., Sullivan, K.F., and Wahl, G.M.: Histone-GFP fusion protein enables sensitive analysis of chromosome dynamics in living mammalian cells. In: Current Biology, vol. 8, pp. 377–385 (1998)

Chen, X., Zhou, X., and Wong, S.T.C.: Automated segmentation, classification, and tracking cancer cell nuclei in time-lapse microscopy. In: IEEE Trans. on Biomedical Engineering, vol. 53, no. 4, pp. 762–766 (2006)

Wang, M., Zhou, X., King, R.W., and Wong, S.T.: Context based mixture model for cell phase identification in automated fluorescence microscopy. In: BMC Bioinformatics vol. 8, no. 32 (2007)

MacAulay, C. and Palcic, B.A.: Comparison of some quick and simple threshold selection methods for stained cells. In: Anal. Quant. Cytol. Histol., vol. 10, pp. 134–138 (1988)

Bleau A. and Leon J.L.: Watershed-based segmentation and region merging. In: Computer Vision and Image Understanding, vol. 77, pp. 317–370 (2000)

Umesh, A.P.S. and Chaudhuri B.B.: An efficient method based on watershed amd rule-based merging for segmentation of 3-D histopathological images. In: Pattern Recognition, vol. 34, pp. 1449–1458 (2001)

Ma, Z., Tavares, J.M.R.S., and Jorge, R.N.: Segmentation of structures in medical images: review and a new computational framework. In: the Eigth International Symposium on Computer Methods in Biomechanics and Biomedical Engineering, Portugal (2008)

Canny, J.: A computational approach to edge detection. In: IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 8, pp. 679–714 (1986)

Adams, R. and Bischof L., Seeded region growing. In: IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 16, pp. 641–47 (1994)

Pohle, R. and Toennies, K.D.: Segmentation of medical images using adaptive region growing. In: *SPIE*, pp. 1337–1346 (2001)

Bell, A., Herberich, G., Dietrich, M.-E., Bocking, A., and Aach, T.: Analysis of silver stained cell specimens: nuclear segmentation and validation. In: International Conference on Medical Imaging (2008)

Norberto M., Andres S., Carlos Ortiz S. Juan Jose V., Francisco P., and Jose Miguel G.: Applying watershed algorithms to the segmentation of clustered nuclei. In: Cytometry, vol. 28, pp. 289–297 (1997)

Otsu N.: A threshold selection method from gray level histogram. In: IEEE Trans. System, Man, and Cybernetics, vol. 8, pp. 62–66 (1978)

Bleau A. and Leon, J.L.: Watershed-based segmentation and region merging. In: Computer Vision and Image Understanding, pp. 317–370 (2000)

Furui, S.: Recent advances in speaker recognition. In: Patter Recognition Letter, vol. 18, pp. 859–872 (1997)

Juang, B.-H.: The past, present, and future of speech processing. In: IEEE Signal Processing Magazine, vol. 15, no. 3, pp. 24–48 (1998)

Kulkarni, V.G.: Modeling and Analysis of Stochastic Systems. Chapman & Hall, UK (1995)

Rabiner, L.R. and Juang, B.H.: Fundamentals of Speech Recognition. Prentice Hall PTR, USA (1993)

Tran, D. and Wagner, M.: Generalised Fuzzy hidden Markov models for speech recognition. In: Lecture Notes in Computer Science: Advances in Soft Computing - AFSS 2002, N.R. Pal and M. Sugeno (Eds.), pp. 345–351, Springer-Verlag (2002)

Tran, D.T. and Pham, T: Modeling methods for cell phase classification, Book chapter in the book Advanced Computational Methods for Biocomputing and Bioimaging, Editors: Tuan Pham, Hong Yan, and Denis I. Crane, Nova Science Publishers, New York, USA, ISBN: 1–60021–278–6, chapter 7, pp. 143–166 (2007)

Tran, D., Pham, T., and Zhou, X.: Subspace vector quantization and Markov modeling for cell phase classification. In: Proceedings of the International Conference on Image Analysis and Recognition (ICIAR), in Image Analysis and Recognition of Lecture Notes in Computer Science, Portugal, vol. 5112, pp. 844–853 (2008)

Dempster, A.P., Laird, N.M., and Rubin, D.B.: Maximum likelihood from incomplete data via the EM algorithm. In: Journal of the Royal Statistical Society, Ser. B, 39: pp. 1–38 (1997)

Tran, D. and Wagner, M.: Fuzzy Gaussian mixture models for speaker recognition. In: special issue of the Australian Journal of Intelligent Information Processing Systems (AJIIPS), vol. 5, no. 4, pp. 293–300 (1998)

Tran, D. and Pham, T.: Automated feature weighting-based cell phase classification. In: Proc. IASTED International Symposium on Computational Biology and Bioinformatics, USA, pp. 274–277 (2008)

Pham, T., Tran, D., and Zhou, X.: Fuzzy information fusion of classification models for high-throughput image screening of cancer cells in time-lapse microscopy. In: KES Journal, vol. 11, no. 4, pp. 237–246, IOS Press (2007)

Sugeno, M.: Fuzzy measures and fuzzy integrals: a survey. In: M.M. Gupta, G.N. Saridis, and B.R. Gaines, eds, Fuzzy Automata and Decision Processes, North-Holland, New York, pp. 89–102 (1977)

Choquet, G.: Theory of capacities. In: Annales de l'Institut Fourier, vol. 5, pp. 131–295 (1953)

Pham, T., Tran, D.T., Zhou, X., and Wong, S.T.C.: Integrated algorithms for image analysis and identification of nuclear division for high-content cell-cycle screening. In: Internation Journal of Computational Intelligence and Applications, vol. 6, pp. 21–43 (2006)

Pham, T., Tran, D.T., Zhou, X., and Wong, S.T.C.: A microscopic image classification system for high-throughput cell-cycle screening. In: Proceeding of International Journal Intelligent Computing in Medical Sciences and Image Processing, vol. 1, no. 1, pp. 67–77 (2007)

Chapter 10
Computational Tools and Resources for Systems Biology Approaches in Cancer

Andriani Daskalaki, Christoph Wierling, and Ralf Herwig

Abstract Systems biology focuses on the study of interacting components of biological systems rather than on the analysis of single genes or proteins and offers a new approach to understand complex disease mechanisms by the use of computational models. The analysis of such models has become crucial to understand biological processes and their dysfunctions with respect to human diseases. A systems biology approach would be a key step in improving diagnosis and therapy of complex diseases such as cancer. It offers new perspectives for drug development, for example, in detecting drug side effects and alternative response mechanisms through the analysis of large cellular networks *in silico*.

In this chapter we review important cellular processes for cancer onset, progression, and response to anticancer drugs, provide a summary of existing pathway databases and tools for the construction and analysis of computational models, and discuss existing kinetic models for cancer-related signaling pathways.

10.1 Introduction

Dysfunctions in the molecular interaction network of the cell can cause severe diseases such as cancer. Computational methods and tools help significantly to understand the effects of such dysfunctions on the network. This requires the implementation and analysis of *in silico* models of the investigated biological systems. In this chapter we give a brief review of cancer-related signaling networks and describe databases and computational tools for their annotation and analysis.

A. Daskalaki (✉)
Max Planck Institute for Molecular Genetics, Ihnestr. 63–73, 14195 Berlin, Germany
e-mail: daskalak@molgen.mpg.de

T. Pham (ed.), *Computational Biology: Issues and Applications in Oncology*,
Applied Bioinformatics and Biostatistics in Cancer Research,
DOI 10.1007/978-1-4419-0811-7_10,

10.2 Molecular Networks Involved in Cancer

10.2.1 Pathways Affected by Cancer Onset and Progression

Cancer is a complex disease involving multiple genes and pathways and is considered to be a manifestation of severe functional changes (Schubbert et al. 2007) in cell physiology related to apoptosis and cell proliferation (Hanahan and Weinberg 2000; Bild et al. 2006; Weinberg 2007). These changes in biological pathways are due to mutations of oncogenes and tumor suppressor genes eventually causing cancer initiation and progression (Kinzler and Vogelstein 1996). Mutations in more than 1% of the human genes are known to contribute to the onset of cancer (Futreal et al. 2004). For instance, loss in the activity of the phosphatase and tensin homolog (PTEN) or hyperactivity of phosphatidylinositol-3-kinase (PI3K) due to a mutation have been found to cause increasing phosphatidylinositol 3,4,5-trisphosphate (PIP3) levels that subsequently initiate activation of the kinase AKT1, which is involved in cellular survival pathways, by inhibiting apoptotic processes and enhancing cell survival and proliferation (Vivanco and Sawyers 2002). Gene amplification or mutations of PIK3CA (the catalytic subunit of PI3K) have been reported in >40% of patients with specific types of cancer (Levine et al. 2005).

Crucial for the regulation of cell proliferation and apoptosis (Cummings et al. 2005) are the recognition and integration of growth and death signals by the cellular signal transduction network, a complex network exhibiting extensive crosstalk. Positive feedback loops between pathways can induce transitions from inactive to permanently activated states leading to continuous cell proliferation and, hence, contribute to the pathogenesis of cancer (Kim et al. 2007).

Although it is well known that these pathways have extensive crosstalk with other pathways involved in tumor progression, computational modeling of cancer processes has been focused so far mainly on individual pathways such as the MAP kinase pathway (Kim et al. 2007), the AKT pathway (Araujo et al. 2007), or WNT signaling (Kim et al. 2007) and apoptosis (Legewie et al. 2006).

10.2.2 Target Pathways of Cancer Treatment

Tumorigenesis in humans is a multistep process and reflects genetic alterations that drive the progressive transformation of normal human cells into highly malignant derivatives. Furthermore, mutations in certain oncogenes and tumor suppressor genes can occur early in some tumor progression pathways and late in others. As a consequence, resistance to apoptosis, sustained angiogenesis, and unlimited replication potential can appear at different times during tumor progression (Hanahan and Weinberg 2000).

Important signaling pathways (Fig. 10.3) crucial for cell growth and survival are frequently activated in human cancer due to genomic aberrations including

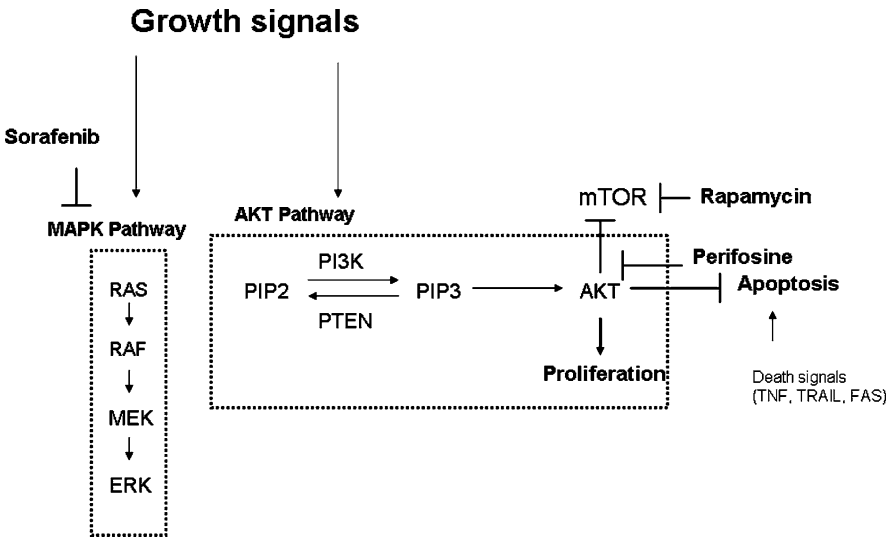

Fig. 10.1 Specific inhibition experiments targeting AKT and MAPK signaling: A schema presenting the concept related to the inhibition of a network regulated by AKT and MAPK with different drugs (Perifosine, Sorafenib Rapamycin). Changes in the state of AKT and MAPK influence proliferation, growth, and apoptosis. Inhibition is indicated by a *blunted line*. AKT protein kinase B, *PI3K* phosphatidylinositol 3-kinase, *mTOR* mammalian target of rapamycin, *MAPK* mitogen-activated protein kinase, *ERK* extracellular-signal-regulated kinase

mutations, amplifications, and rearrangements. An increasing number of rationally designed small molecule inhibitors directed against growth and survival pathways such as the mitogen-activated protein (MAP) kinase pathway, the PI3K-AKT-mTOR pathway, or the JAK-STAT signaling pathways are entering clinical testing for the treatment of cancer (Van Ummersen et al. 2004; Hennessy et al. 2005; McCubrey et al. 2008).

The PI3K/AKT signaling pathway (Fig. 10.1) has become a primary target for cancer treatment (Vivanco and Sawyers 2002). For instance, the AKT inhibitor perifosine (Fig. 10.1) was used in preclinical and clinical trials for several cancer entities, for example, prostate cancer (Van Ummersen et al. 2004). This drug interrupts the interaction between PIP3 and the pleckstrin homology (PH) domain of AKT and thus prevents membrane localization of AKT that is essential for its activation. Further drugs that present antitumor activity are Wortmannin (Rahn et al. 1994) and LY294002 (Hennessy et al. 2005). Another drug that targets the mammalian target of rapamycin (mTOR) pathway is rapamycin (Rapamune, Wyeth Ayerst). Rapamycin is a specific inhibitor of mTOR and functions downstream of AKT (Hay and Sonenberg 2004). mTOR inhibitors are being tested in clinical trials for patients with breast cancer and other solid tumors (Chan et al. 2005; Hidalgo and Rowinsky 2000; Nagata et al. 2004). Besides these inhibitors, many more drugs (Cho et al. 2006) are described in the literature that target components of the major cancer-related cellular pathways. Table 10.1 gives an overview of these 81.

Table 10.1 List of well-known cancer-related pathways

Pathways
EGF signaling
Cytokin signaling
E-cadherin signaling
FAS signaling
G-protein signaling
Hedgehog
IGF-1 signaling
Intrinsic apoptosis
Nerve growth factor
Notch signaling
Phospholipase C signaling
RB signaling in response to DNA damage
Toll-like receptor 3
Toll-like receptor 10
TGF beta signaling
BMP signaling
TNFR signaling
TRAIL signaling
WNT signaling

Literature review identifies at least 19 different molecular pathways that might be targeted by cancer therapies and are involved in tumor progression (Table 10.1). Components of these pathways and their molecular interactions are the basis of computational modeling. A cancer-related model should consider the extensive pathway crosstalk in cancer by the integration of different cellular pathways and processes that constitute signal transduction cascades activated by stimuli such as growth factors (EGF, NGF, IGF-1, TGF-beta), cell proliferation (Wnt, Rb, Notch, Hedgehog), cytokines (Interleukin 2, STAT-JAK), inflammation (Toll-like receptors), apoptosis (TNF-alpha, FAS, TRAIL), and metabolic regulation (G-protein-coupled receptors).

Based on this complex molecular background, modeling approaches can be used to address and analyze clinical problems regarding different response mechanisms in patients, which could be due to mutations of oncogenes or tumor suppressor genes leading to gain or loss of function of the related proteins.

10.3 Molecular Interaction Databases

The development of a computational model involves multiple steps. Figure 10.2 illustrates the general modeling workflow.

The first step in this workflow includes the annotation of cellular processes and pathways by the use of appropriate annotation tools in order to build the relevant computational network. Much of the existing knowledge on cancer-relevant reaction networks is agglomerated in pathway databases. These databases typically describe the signaling flow in the normal state and can be used as a first approximation for

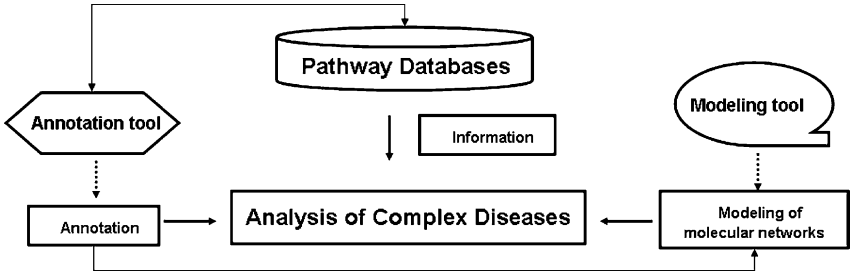

Fig. 10.2 Annotation workflow: The annotation workflow includes the following steps: (**a**) Based on pathway databases search for interaction between proteins involved in biological pathways related to cell proliferation and apoptosis. (**b**) Annotation step: The construction of the molecular network is based on information from the Pathway Databases Reactome and ConsensusPathDB. The Reactome Curator Tool is used for this annotation. (**c**) As a final step, the model topology can be further adapted and analyzed with a modeling tool such as PyBioS

cancer-related signaling networks. In this section pathway databases, annotation and simulation tools will be discussed that support the scientist in the development of those molecular models and their analysis.

10.3.1 BioCyc

The BioCyc databases (Karp et al. 2005) have been assembling a unique collection of chemical data for compounds involved in metabolic pathways (Darzentas et al. 2005; http:// biocyc.org). The BioCyc Open Chemical Database (BOCD) is a collection of chemical compound data from the BioCyc databases. The compounds act as substrates in enzyme-catalyzed metabolic reactions or serve as enzyme activators, inhibitors, and cofactors. Chemical structures are provided for the majority of compounds.

10.3.2 KEGG

KEGG (http://www.genome.jp/kegg/) is a database of biological systems that integrates genomic, chemical, and systemic functional information (Kanehisa et al. 2006). KEGG PATHWAY contains 26 maps for human diseases (Araki and Hirakawa 2006). The disease pathway maps are classed in four subcategories: 6 as neurodegenerative disorders, 3 as each of infectious diseases and metabolic disorders, and 14 as cancers.

10.3.3 Reactome

The Reactome project developed a curated resource of core pathways and reactions in human biology (Vastrik et al. 2007; http://reactome.org). The information in

this database is cross-referenced with multiple other databases, such as sequence databases at NCBI, Ensembl, UniProt, the UCSC Genome Browser, HapMap (http://www.hapmap.org), KEGG, ChEBI, PubMed, and GO. In addition to curated human events, inferred orthologous events in 22 nonhuman species including mouse, rat, chicken, puffer fish, worm, fly, yeast, two plants, and *Escherichia coli* are also available. Reactome is a free online resource, and Reactome software is open-source. The Reactome database already covers multiple cellular pathways, (Joshi-Tope et al. 2005) including different signal transduction pathways that are also related to cancer (de Bernard 2008).

10.3.4 ConsensusPathDB

To achieve a more comprehensive integration of interaction data, Kamburov et al. (2009) have developed ConsensusPathDB, a database integrating human molecular interaction networks of several public pathway resources (http://cpdb.molgen.mpg.de). The integrated content comprises different types of functional interactions that interconnect diverse types of cellular entities. To gain an immediate critical number of interactions, the authors have focused primarily on the integration of existing database resources. Currently, the database contains human functional interactions, including gene regulations, physical (protein–protein and protein-compound) interactions, and biochemical (signaling and metabolic) reactions, obtained by integrating such data from 12 publicly accessible databases.

10.3.5 TRANSPATH®

TRANSPATH is a database on regulatory network information, mostly in human, mouse, and rat cells. This database originally focuses on signal transduction pathways that aim at gene regulatory molecules (transcription factors), but also on metabolic enzymes and structural proteins as targets (Krull et al. 2006). Data about intracellular regulatory processes are collected, annotated, and stored at different levels of abstraction making use of both a molecular and a reaction hierarchy (Wingender et al. 2007).

10.3.6 Annotation Tools

A pathway model provides a starting point for searching the complex molecular background of cancer. The ability to incorporate experimental data in a curated model could be considered as a key step to gain insights through the correlation of data with network information (de Bono 2008).

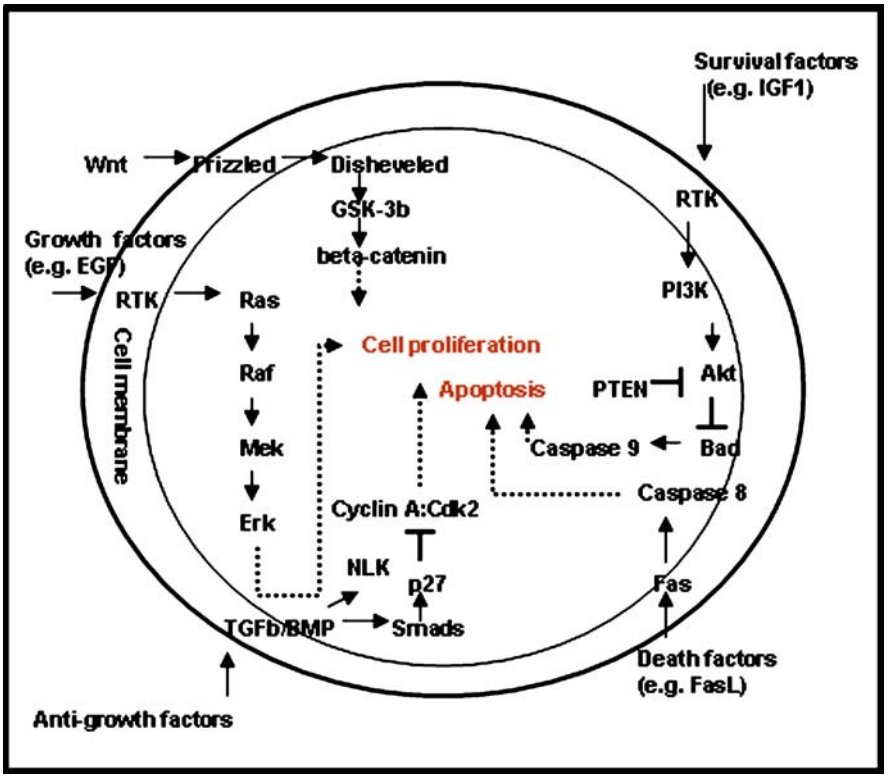

Fig. 10.3 Cancer model relevant pathways: Signaling pathways that comprise the minimal model of cancer-related signaling pathways

Annotation of pathways typically involves the translation of the human-readable maps (Bouwmeester et al. 2004) (Fig. 10.3) to computer-readable reaction systems (Kerrien et al. 2007). Several tools are available that offer this service. The Reactome Curator tool is a software application implemented in Java. It is designed for the annotation of data related to biological pathways. This tool enables the researcher to annotate pathways based on the existing data of the Reactome database but also to enter new data from other public resources. A set of reactions can be grouped to form a pathway (de Bono 2008). GO terms from "Molecular Function" are linked directly to the action of catalysts.

10.3.7 Modeling Tools

Modeling and simulation techniques are valuable tools for the understanding of complex biological systems involved in tumor formation and progression. Computational modeling requires the translation of the pathway schemas (Fig. 10.2) into computer models that can carry information on the concentrations of the model

components and on the kinetics of the reactions these components are involved in. This process contains the design of suitable computer objects, the implementation of the reactions, the assignment of kinetic laws to these reactions, and the model analysis (Klipp et al. 2005).

A modeling tool enables the study of the kinetic behavior of the system under analysis, for example, changes of the model components in time upon perturbations (e.g., due to mutations or drug effects) or external stimuli (e.g., growth factors). Once established, such a computational model can be used to accomplish specific *in silico* case studies. Perturbations can be integrated in strategic points of the model by modifying the parameters of the model. A minimal value rate constant of a reaction leading to the activation or deactivation of a model component by its phosphorylation can present significant changes. Furthermore, a higher value for the degradation rate constant could lead to a sustained signal. Therefore, the structure of the model should be based on different parameter values. The effect of these changes should be considered and validated with experimental results.

Computational tools that support model population, simulation, and analysis build the basis of systems biology (Wierling et al. 2007). Modeling tools that have been applied for analyzing molecular networks, are, for instance, CellDesigner (Funahashi et al. 2003), E-Cell (Tomita et al. 1999), Gepasi (Mendes 1993, 1997) and its successor Copasi (Hoops et al. 2006), ProMoT/Diva (Ginkel et al. 2003), JDesigner (Novikov and Barillot 2008), Virtual Cell (Loew and Schaff 2001; Slepchenko et al. 2003), and tools integrated in the Systems Biology Workbench (Hucka et al. 2002). These tools can be used for the development and analysis of small models. However, a model related to cancer should include a high number of components, reactions, and kinetic parameters. Thus, the analysis of such a model requires automation of the annotation of the reaction network, its generation, numerical integration of differential equations, simulation of the model, and further the visualization of the simulation results (Klamt et al. 2006).

10.3.8 Systems Biology Workbench

The Systems Biology Workbench (SBW) is a software framework that allows heterogeneous application components, which are written in diverse programming languages and running on different platforms, to communicate with each other. SBW enables modeling, analysis, visualization, and general data manipulation to communicate via a simple network protocol.

10.3.9 JDesigner

JDesigner is both a network design tool and simulator applied to draw a biochemical network and export it in SBML format (Novikov and Barillot 2008). The

software automatically derives the differential equations and solves them to generate a solution. JDesigner was developed at the Keck Graduate Institute, California, in collaboration with the California Institute of Technology.

10.3.10 CellDesigner

CellDesigner is a structured diagram editor for drawing gene-regulatory and biochemical networks. Networks are drawn based on the process diagram, with a graphical notation system proposed by Kitano et al. (2005). The network models are stored using the Systems Biology Markup Language (SBML), a standard for representing models of biochemical and gene-regulatory networks. Networks are able to link with simulation and other analysis packages through the Systems Biology Workbench (SBW).

10.3.11 PyBioS

PyBioS is an object-oriented tool for modeling and simulation of cellular processes. This tool has been established for the modeling of biological processes using biochemical pathways from databases like KEGG and Reactome. PyBioS (Wierling et al. 2007, http://pybios.molgen.mpg.de) acts as a model repository and supports the automatic generation of large models through interfaces to publicly available pathway databases, such as Reactome and KEGG. This allows a rapid and automated access to reaction systems. An ODE system of a model may be generated automatically based on pre- or user-defined kinetic laws and used for subsequent simulation of time course series and further analyses of the dynamic behavior of the system.

10.4 Computational Models for Cancer-Related Processes

10.4.1 BioModels Database

The BioModels Database (Le Novère et al. 2006) allows researchers to exchange and share their computational models. This database provides a free, centralized, publicly-accessible repository of annotated, computational models in SBML and other structured formats, which are linked to relevant data resources, publications, as well as databases of compounds and pathways.

10.4.2 Specific Kinetic Models Relevant for Cancer

Protein circuits in living human cells are characterized by variability in their behavior, both from cell to cell and in the same cell over time. To describe such variability, one can use kinetic models. Kinetic models of biochemical systems are often described with a set of first-order nonlinear ordinary differential equations. These systems have large numbers of unknown parameters, simplified dynamics, and uncertain connectivity. These key features are shared with many high-dimensional multiparameter nonlinear models. To investigate the dynamical characteristics of those models, a set of different parameter values should be tested. Brown and Sethna (2003) use a statistical method to study the behavior of these models to extract as much useful predictive information as possible from a model, given the available data used to constrain it. The authors present a unified methodology for the construction, evaluation, and use of models with many unknown parameters.

Experimental measurements in living cells as well as theoretical models enable to understand the dynamics and variability of protein circuitry. For instance, Geva-Zatorsky et al. (2006) measured the dynamics of fluorescently tagged p53 and MDM2 in living cells and worked out a corresponding model that takes into account the negative feedback loop between the tumor suppressor p53 and the oncogene MDM2. The model is characterized by variability: low-frequency noise in protein production rates, rather than noise in parameters regarding degradation rates.

Another pathway that is known to play a key role in the progression of multiple human cancers is EGF signaling Jones et al. (2006). The model of Birtwistle et al. (2007) provides a quantitative description of the activation of critical downstream proteins in the EGF pathway, the extracellular-signal-regulated kinase (ERK) and AKT after stimulation of the EGF pathway by binding of the EGF receptor with the different ligands [epidermal growth factor (EGF) or heregulin (HRG)]. Based on model analysis and experimental validation, activation of the EGF pathway with different ligands leads to a different signaling behavior. Thus, after the application of an ERK cascade inhibitor U0126, HRG-induced ERK activity will be less influenced in comparison to the EGF-induced ERK activity. Variation of EGF-induced ERK activity was due to the regulation by PI3K

Schoeberl et al. (2002) present a computational model, based on components of epidermal growth factor (EGF) receptor signal pathways. The model provides insights into signal-response relationships between the binding of EGF to its receptor at the cell surface and the activation of downstream proteins in the signaling cascade. Based on this model, the initial velocity parameter of the receptor activation was critical for the signal. The predictions of the model agree well with experimental analysis. Sasagawa et al. (2005) developed a model of ERK signaling networks by constraining *in silico* dynamics based on in vivo dynamics in PC12 cells. The authors predicted and validated that transient ERK activation depends on rapid increases of epidermal growth factor and nerve growth factor (NGF) but not on their final concentrations, whereas sustained ERK activation depends on the final concentration of NGF but not on the temporal rate of increase.

The Wnt and the ERK pathways are both involved in the pathogenesis of various kinds of cancers. Kim et al. (2007) showed that because of a positive feedback loop embedded in a crosstalk between the Wnt and the ERK pathways, changes in proteins based on gene mutations result in changes in pathways beyond the pathway in which they directly act. Thus, crosstalk between signaling pathways can affect properties of the system at a larger scale. Based on experimental reports and established basic mathematical models of each pathway, the authors studied the role of this hidden positive feedback loop between the Wnt and the ERK pathways and showed that the positive feedback loop can generate bistability in both the Wnt and the ERK signaling pathways. In particular, enhanced production of beta-catenin and reduction of the velocity of MAP kinase phosphatase(s) followed by mutations could evoke an irreversible response leading to a sustained activation of both pathways. This enables that high activities of the Wnt and the ERK pathways are maintained even without a persistent extracellular signal.

Investigation of dynamics and regulation of the TGF-beta signaling pathway are central to the understanding of complex cellular processes such as growth, apoptosis, and differentiation. Zi and Klipp (2007) proposed a constraint-based modeling method to construct a mathematical model for the SMAD-dependent TGF-beta signaling pathway by fitting the experimental data and incorporating qualitative constraints from experimental analysis. This constraint-based modeling method can be applied to quantitative modeling of other signaling pathways. The model agrees well with the experimental analysis of the TGF-beta pathway, such as the time course of nuclear phosphorylated SMAD, the subcellular location of SMAD, and the signal response of SMAD phosphorylation to different doses of TGF-beta.

Yamada et al. (2003) have developed a computational model related to the JAK/STAT signaling network. The authors investigated the role of the suppressor of cytokine signaling-1 (SOCS1), which is considered as the negative regulator of the Janus kinase (JAK) and signal transducer and activator of transcription (STAT) signal transduction pathway. The model was simulated based on various values of its parameters. Furthermore, the authors compared various initial concentrations and parameter values and investigated the peak and steady state concentration of activated transcription factors (STAT1).

Another pathway that is assumed to be dysregulated in cancer is the programmed cell death by apoptosis. Crucial for apoptosis is the activation of caspases. Caspases (cysteine-aspartic acid proteases) are a family of cysteine proteases, which play essential roles in apoptosis. The inhibitors of apoptosis (IAP) are a family of proteins, which can inhibit caspases. Based on the model of Legewie et al. (2006), inhibition of caspase-3 (Casp3) and caspase-9 (Casp9) by inhibitors of apoptosis (IAPs) results in an implicit positive feedback that leads to bistability, as well as irreversibility in caspase activation through Casp3-mediated feedback cleavage of Casp9. The feedback mechanism described by Legewie et al. provides insights on how cells achieve ultrasensitivity, bistability, and irreversibility.

10.5 Discussion

In this chapter we reviewed data resources and computational tools essential for the modeling of cancer pathways, such as currently available pathway databases and annotation and modeling tools. Furthermore, some existing kinetic models for important pathways involved in tumor formation and progression have been highlighted. These pathways are often targeted by currently available cancer treatments so that analysis of the dynamic features of these signaling pathways is crucial for understanding the response to treatment-induced network perturbations. However, these models are not able to explain these response mechanisms in a sufficient way. The focus of a model study related to cancer development should be to investigate the role of any factors and any parameters in the system by the simulation with various values. The modeling approach is important to focus on predicting differences in the model, for example, due to inhibition or activation of key components in the model. The study of gene regulation will have to be taken into account. In particular, network models should be adapted to specific tumor types and states. An enormous effect on the use of predictive modeling for cancer patients based on their molecular signatures can be expected due to the rapid development of next-generation sequencing techniques that parallelize the sequencing process and produce millions of sequences at once. The modified model parameters, which represent the behavior of mutated pathway components, should be compared with the control model and their results can be verified in the lab.

Furthermore, fitting of several experimental datasets simultaneously is a powerful approach to estimate parameter values (Cho et al. 2003), to check the validity of a given model, and to discriminate competing model hypotheses. It requires high-performance integration of ordinary differential equations and robust optimization.

The integration of several isolated pathways into a larger framework, which also captures crosstalk between pathways, might however be crucial for the prediction of drug action. Having agglomerated information about drugs, their molecular targets or set of targets, and the cellular interaction network (Schulze et al. 2005) they function in, the next step is to translate the effects of the drug in the computer (Cho et al. 2006; Jones et al. 2006).

Current anticancer drugs are designed to target specific pathway components (Cummings et al. 2005; Holcomb et al. 2008). Nevertheless, side effects occur, because of the drug effects in other pathways. In clinical testing, many inhibitors fail due to unexpected toxicities caused by previously unknown targets, or because the drug target itself is involved in multiple functional interactions that can be sensitive to deregulation. In addition, clinical failure of targeted drugs is also caused by the existence of unexpected feedback loops, compensatory upregulation of alternative signaling pathways, or drug resistance mutations, all of which avoid the effects of target inhibition and allow tumor cell survival and proliferation (Burchert et al. 2005).

In particular, mutations in signaling proteins can contribute to cancerogenesis, because of a sustained activation of pathways. Therefore, predictive models should include relevant protein interactions in order to cope with the complexity of multiple

targets and crosstalk between pathways. Such models could provide significant support for the development of novel targeted drugs (Strumberg 2005).

To identify models according to existing biological knowledge and experimental measurements (Luo et al. 2005), the dynamic properties of the model have to be investigated (Maraziotis et al. 2007). To generate new hypotheses about the reaction networks or to postulate new system variables, it is important to analyze the model, in close relation to the laboratory data (Jones et al. 2006). The necessary functionalities range from real-time changing of parameter values and characteristics of driving input functions to efficient refinement of the model structure itself. Powerful fitting procedures are required to calibrate model parameters in the context of several experimental datasets, under different experimental settings and with different sets of measured species. Model-data-compliance and model discrimination should be quantified by statistical tests (Maiwald and Timmer 2008).

The robustness of sensitivity analysis to parameter perturbation at different ligand doses should also be taken into account. A modeling approach can lead to a novel approach of personalized medicine by generating detailed predictions for the therapy of complex diseases like cancer and could be integrated in routine diagnostics in oncology.

Acknowledgments This work was supported by the EU FP6 grant SysCo (LSHG-CT-2006–37231), the Mutanom project (01GS08105) supported by the German Federal Ministry of Education and Research (BMBF) and the Max Planck Society.

References

Araujo RP, Liotta LA, Petricoin EF (2007) Proteins, drug targets and the mechanisms they control: the simple truth about complex networks. Nat Rev Drug Discov 6:871–880

Bild AH, Potti A, Nevins JR (2006) Linking oncogenic pathways with therapeutic opportunities. Nat Rev Cancer 6:735–741

Birtwistle MR, Hatakeyama M, Yumoto N, Ogunnaike BA, Hoek JB, Kholodenko BN (2007) Ligand-dependent responses of the ErbB signaling network: experimental and modeling analyses. Mol Syst Biol 3:144

Bouwmeester T, Bauch A, Ruffner H, Angrand PO, Bergamini G, Croughton K, Cruciat C, Eberhard V, Gagneur J, Ghidelli S et al (2004) A physical and functional map of the human TNF-a/NF-kB signal transduction pathway. Nat Cell Biol 6:97–105

Brown KS, Sethna JP (2003) Statistical mechanical approaches to models with many poorly known parameters. Phys Rev E Stat Nonlin Soft Matter Phys 68(2, Part 1):021904

Burchert A, Wang Y, Cai D, von Bubnoff N, Paschka P, Muller-Brusselbach S, Ottmann OG, Duyster J, Hochhaus A, Neubauer A (2005) Compensatory PI3-kinase/Akt/mTor activation regulates imatinib resistance development. Leukemia 19:1774–1782

Chan S, Scheulen ME, Johnston S, Mross K, Cardoso F, Dittrich C, Eiermann W, Hess D, Morant R, Semiglazov V, Borner M, Salzberg M, Ostapenko V, Illiger HJ, Behringer D, Bardy-Bouxin N, Boni J, Kong S, Cincotta M, Moore L (2005) Phase II study of temsirolimus (CCI-779), a novel inhibitor of mTOR, in heavily pretreated patients with locally advanced or metastatic breast cancer. J Clin Oncol 23:5314–5322

Cho CR, Labow M, Reinhardt M, van Oostrum J, Peitsch MC (2006) The application of systems biology to drug discovery. Curr Opin Chem Biol 10(4):294–302

Cho K-H, Shin S-Y, Kolch W, Wolkenhauer O (2003) Experimental design in systems biology, based on parameter sensitivity analysis using a Monte Carlo method: a case study for the TNFalpha-mediated NF-kappa B signal transduction pathway. SIMULATION 79:726–739

Cline MS, Smoot M, Cerami E, Kuchinsky A, Landys N, Workman C, Christmas R, Avila-Campilo I, Creech M, Gross B et al (2007) Integration of biological networks and gene expression data using Cytoscape. Nat Protoc 2:2366–2382

Cummings J et al (2005) Validation of pharmacodynamic assays to evaluate the clinical efficacy of an antisense compound (AEG 35156) targeted to the X-linked inhibitor of apoptosis protein XIAP. Br J Cancer 92:532–538

de Bernard B (2008) The breadth and depth of biomedical molecular networks: the Reactome perspective. In: Daskalaki A (ed) Handbook of research on systems biology applications in medicine, 1st edn. Medical Information Science Reference, Hershey, PA

Faivre S, Kroemer G, Raymond E (2006) Current development of mTOR inhibitors as anticancer agents. Nat Rev Drug Discov 5:671–688

Funahashi A, Tanimura N, Morohashi M, Kitano H (2003) CellDesigner: a process diagram editor for gene-regulatory and biochemical networks. Biosilico 1:159–162

Futreal PA, Coin L, Marshall M, Down T, Hubbard T, Wooster R, Rahman N, Stratton MR (2004) A census of human cancer genes. Nat Rev Cancer 4:177–183

Geva-Zatorsky N et al (2006) Oscillations and variability in the p53 system. Mol Syst Biol 2:2006.0033. doi: 10.1038/msb4100068

Gills JJ, Holbeck S, Hollingshead M, Hewitt SM, Kozikowski AP, Dennis PA (2006) Spectrum of activity and molecular correlates of response to phosphatidylinositol ether lipid analogues, novel lipid-based inhibitors of Akt. Mol Cancer Ther 5:713–722

Ginkel M, Kremling A, Nutsch T, Rehner R, Gilles ED (2003) Modular modeling of cellular systems with ProMoT/Diva. Bioinformatics 19:1169–1176

Hanahan D, Weinberg RA (2000) The hallmarks of cancer. Cell 100:57–70

Hay N, Sonenberg N (2004) Upstream and downstream of mTOR. Genes Dev 18:1926–1945

Hennessy BT, Smith DL, Ram PT, Lu Y, Mills GB (2005) Exploiting the PI3K/AKT pathway for cancer drug discovery. Nat Rev Drug Discov 4:988–1004

Hidalgo M, Rowinsky EK (2000) The rapamycin-sensitive signal transduction pathway as a target for cancer therapy. Oncogene 19:6680–6686

Holcomb B et al (2008) Pancreatic cancer cell genetics and signaling response to treatment correlate with efficacy of gemcitabine-based molecular targeting strategies. J Gastrointest Surg 12:288–296

Hoops S, Sahle S, Gauges R, Lee C, Pahle J, Simus N, Singhal M, Xu L, Mendes P, Kummer U (2006) COPASI – a COmplex PAthway Simulator. Bioinformatics 22:3067–3074

Hucka M, Finney A, Sauro HM, Bolouri H, Doyle J, Kitano H (2002) The ERATO Systems Biology Workbench: enabling interaction and exchange between software tools for computational biology. Pac Symp Biocomput 450–461

JDesigner: a biochemical network layout tool. http://sbw.kgi.edu/software/jdesigner.htm. Accessed 4 Dec 2009

Jiang N, Cox RD, Hancock JM (2007) A kinetic core model of the glucose-stimulated insulin secretion network of pancreatic beta cells. Mamm Genome 18:508–520

Jones RB, Gordus A, Krall JA, MacBeath G (2006) A quantitative protein interaction network for the ErbB receptors using protein microarrays. Nature 439:168–174

Joshi-Tope G, Gillespie M, Vastrik I, D'Eustachio P, Schmidt E, de Bono B, Jassal B, Gopinath GR, Wu GR, Matthews L, Lewis S, Birney E, Stein L (2005) Reactome: a knowledgebase of biological pathways. Nucleic Acids Res 33:D428–D432

Kamburov A, Wierling C, Lehrach H, Herwig R (2009) ConsensusPathDB—a database for integrating human functional interaction networks. Nucleic Acids Res 37:D623–D628

Kanehisa M, Goto S, Hattori M, Aoki-Kinoshita KF, Itoh M, Kawashima S, Katayama T, Araki M, Hirakawa M (2006) From genomics to chemical genomics: new developments in KEGG. Nucleic Acids Res 34:D354–D357

Karp PD, Ouzounis CA, Moore-Kochlacs C, Goldovsky L, Kaipa P, Ahren D, Tsoka S, Darzentas N, Kunin V, Lopez-Bigas N (2005) Expansion of the BioCyc collection of pathway/genome databases to 160 genomes. Nucleic Acids Res 33:6083–6089

Kerrien S, Alam-Faruque Y, Aranda B, Bancarz I, Bridge A, Derow C, Dimmer E, Feuermann M, Friedrichsen A, Huntley R et al (2007) IntAct – open source resource for molecular interaction data. Nucleic Acids Res 35:D561–D565

Kim D, Rath O, Kolch W, Cho KH (2007) A hidden oncogenic positive feedback loop caused by crosstalk between Wnt and ERK pathways. Oncogene 26:4571–4579

Kinzler KW, Vogelstein B (1996) Breast cancer. What's mice got to do with it? Nature 382:672

Kitano H, Funahashi A, Matsuoka Y, Kanae OD (2005) Using process diagrams for the graphical representation of biological networks. Nat Biotechnol 23:961–966

Klamt S, Saez-Rodriguez J et al (2006) A methodology for the structural and functional analysis of signaling and regulatory networks. BMC Bioinformatics 7:56

Klipp E, Herwig R, Kowald A, Wierling C, Lehrach H (2005) Systems biology in practice: concepts, implementation and application. Wiley-VCH, Weinheim

Krull M, Pistor S, Voss N, Kel A, Reuter I, Kronenberg D, Michael H, Schwarzer K, Potapov A, Choi C, Kel-Margoulis O, Wingender E (2006) TRANSPATH: an information resource for storing and visualizing signaling pathways and their pathological aberrations. Nucleic Acids Res 34:D546–D551

Legewie S, Blüthgen N, Herzel H (2006) Mathematical modeling identifies inhibitors of apoptosis as mediators of positive feedback and bistability. PLoS Comput Biol 2(9):e120

Le Novère N, Bornstein B, Broicher A, Courtot M, Donizelli M, Dharuri H, Li L, Sauro H, Schilstra M, Shapiro B, Snoep JL, Hucka M (2006) BioModels database: a free, centralized database of curated, published, quantitative kinetic models of biochemical and cellular systems. Nucleic Acids Res 34:D689–D691

Levine DA, Bogomolniy F, Yee CJ, Lash A, Barakat RR, Borgen PI, Boyd J (2005) Frequent mutation of the *PIK3CA* gene in ovarian and breast cancers. Clin Cancer Res 11:2875–2878

Loew LM, Schaff JC (2001) The virtual cell: a software environment for computational cell biology. Trends Biotechnol 19:401–406

Luo M, Reyna S, Wang L, Yi Z, Carroll C, Dong LQ, Langlais P, Weintraub ST, Mandarino LJ (2005) Identification of insulin receptor substrate 1 serine/threonine phosphorylation sites using mass spectrometry analysis: regulatory role of serine 1223. Endocrinology 146:4410–4416

Maiwald T, Timmer J (2008) Dynamical modeling and multi-experiment fitting with PottersWheel. Bioinformatics 24(18):2037–2043

Maraziotis IA, Dimitrakopoulou K, Bezerianos A (2007) Growing functional modules from a seed protein via integration of protein interaction and gene expression data. BMC Bioinformatics 8:408

McCubrey JA, Steelman LS, Abrams SL, Bertrand FE, Ludwig DE, Basecke J, Libra M, Stivala F, Milella M, Tafuri A, Lunghi P, Bonati A, Martelli AM (2008) Targeting survival cascades induced by activation of Ras/Raf/MEK/ERK, PI3K/PTEN/Akt/mTOR and Jak/STAT pathways for effective leukemia therapy. Leukemia 22:708–722

Mendes P (1993) GEPASI: a software package for modelling the dynamics, steady states and control of biochemical and other systems. Comput Appl Biosci 9:563–571

Mendes P (1997) Biochemistry by numbers: simulation of biochemical pathways with Gepasi 3. Trends Biochem Sci 22:361–363

Nagata Y, Lan KH, Zhou X, Tan M, Esteva FJ, Sahin AA, Klos KS, Li P, Monia BP, Nguyen, NT, Hortobagyi GN, Hung MC, Yu D (2004) PTEN activation contributes to tumor inhibition by trastuzumab, and loss of PTEN predicts trastuzumab resistance in patients. Cancer Cell 6:117–127

Novikov E, Barillot E (2008) Regulatory network reconstruction using an integral additive model with flexible kernel functions. BMC Syst Biol 2:8

Rahn T, Ridderstrale M, Tornqvist H, Manganiello V, Fredrikson G, Belfrage P, Degerman E (1994) Essential role of phosphatidylinositol 3-kinase in insulin-induced activation and phosphorylation of the cGMP-inhibited cAMP phosphodiesterase in rat adipocytes. Studies using the selective inhibitor wortmannin. FEBS Lett 350:314–318

Sasagawa S, Ozaki Y, Fujita K, Kuroda S (2005) Prediction and validation of the distinct dynamics of transient and sustained ERK activation. Nat Cell Biol 7(4):365–373

Schoeberl B, Eichler-Jonsson C, Gilles ED, Müller G. (2002) Computational modeling of the dynamics of the MAP kinase cascade activated by surface and internalized EGF receptors. Nat Biotechnol 20(4):370–375

Schubbert S, Shannon K, Bollag G (2007) Hyperactive Ras in developmental disorders and cancer. Nat Rev Cancer 7:295–308

Schulze WX, Deng L, Mann M (2005) Phosphotyrosine interactome of the ErbB-receptor kinase family. Mol Syst Biol 1:42–54

Slepchenko BM, Schaff JC, Macara I, Loew LM (2003) Quantitative cell biology with the Virtual Cell. Trends Cell Biol 13:570–576

Strumberg D (2005) Preclinical and clinical development of the oral multikinase inhibitor sorafenib in cancer treatment. Drugs Today (Barc) 41:773–784

Tomita M, Hashimoto K, Takahashi K, Shimizu TS, Matsuzaki Y, Miyoshi F, Saito K, Tanida S, Yugi K, Venter JC, Hutchison CA III (1999) E-CELL: software environment for whole-cell simulation. Bioinformatics 15:72–84

Van Ummersen L, Binger K, Volkman J, Marnocha R, Tutsch K, Kolesar J, Arzoomanian R, Alberti D, Wilding G (2004) A phase I trial of perifosine (NSC 639966) on a loading dose/maintenance dose schedule in patients with advanced cancer. Clin Cancer Res 10:7450–7456

Vastrik I, D'Eustachio P, Schmidt E, Joshi-Tope G, Gopinath G, Croft D, de Bono B, Gillespie M, Jassal B, Lewis S, Matthews L, Wu G, Birney E, Stein L (2007) Reactome: a knowledge base of biologic pathways and processes. Genome Biol 8:R39

Vivanco I, Sawyers CL (2002) The phosphatidylinositol 3-Kinase AKT pathway in human cancer. Nat Rev Cancer 2:489–501

Weinberg RA (2007) The biology of cancer. Garland Science, New York

Wierling C, Herwig R, Lehrach H (2007) Resources, standards and tools for systems biology. Brief Funct Genomic Proteomic 6:240–251

Wingender E, Crass T, Hogan JD, Kel AE, Kel-Margoulis OV, Potapov AP (2007) Integrative content-driven concepts for bioinformatics "beyond the cell." J Biosci 32:169–180

Yamada S, Shiono S, Joo A, Yoshimura A (2003) Control mechanism of JAK/STAT signal transduction pathway. FEBS Lett 534(1–3):190–196

Zi Z, Klipp E (2007) Constraint-based modeling and kinetic analysis of the Smad dependent TGF-beta signaling pathway. PLoS ONE 2(9):e936

Chapter 11
Laser Speckle Imaging for Blood Flow Analysis

Thinh M. Le, J.S. Paul, and S.H. Ong

Abstract Laser speckle imaging (LSI) has increasingly become a viable technique for real-time medical imaging. However, the computational intricacies and the viewing experience involved limit its usefulness for real-time monitors such as those intended for neurosurgical applications. In this paper, we report a proposed technique, tLASCA, which processes statistics primarily in the temporal direction using the laser speckle contrast analysis (LASCA) equation, proposed by Briers and Webster. This technique is thoroughly compared with the existing techniques for signal processing of laser speckle images, including the spatial-based sLASCA and the temporal-based mLSI techniques. sLASCA is an improvement of the basic LASCA technique in which the derived contrasts are further averaged over a predetermined number of raw speckle images. mLSI, on the other hand, is the modified laser speckle imaging (mLSI) technique in which temporal statistics are processed using the technique developed by Ohtsubo and Asakura. tLASCA preserves the original image resolution similar to mLSI. tLASCA performs better than sLASCA (window size $M = 5$) with faster convergence of K values (5.32 vs. 20.56 s), shorter per-frame processing time (0.34 vs. 2.51 s), and better subjective and objective quality evaluations of contrast images. tLASCA also performs better than mLSI with faster convergence of K values (5.32 s) than N values (10.44 s), shorter per-frame processing time (0.34 vs. 0.91 s), smaller intensity fluctuations among frames (8–10% vs 15–35%), and better subjective and objective quality evaluations of contrast images. The computation of speckle contrast and flow rate has been updated with both Lorentzian and Gaussian models. Using tLASCA, the minimally invasive and optically derived flow rates (370–490 μL/min using Lorentzian and 464–614 μL/min using Gaussian model) are found to be in good agreement with the invasively measured flow rate (218–770 μL/min) at similar-sized arteriole (270 μm in diameter). The LSI technique for real-time monitoring of blood flows and vascular perfusion, with proper experimental setups and quantitative analyses, may lay new bricks for research in diagnostic radiology and oncology.

T.M. Le (✉)
Department of Electrical and Computer Engineering, Faculty of Engineering, National University of Singapore, 4 Engineering Drive 3, Singapore 117576
e-mail: elelmt@nus.edu.sg;thinh.le@ieee.org

T. Pham (ed.), *Computational Biology: Issues and Applications in Oncology*,
Applied Bioinformatics and Biostatistics in Cancer Research,
DOI 10.1007/978-1-4419-0811-7_11,

11.1 Introduction

Laser speckle imaging has increasingly become a viable technique for real-time medical imaging (Briers 2001). Laser speckle is formed when laser light shines on an object, and the speckle contrast values are estimated from the time-varying statistics. By using speckle statistics, blood flow and perfusion can be reliably estimated by processing only windows of pixels from successive raw speckle images (Briers and Webster 1996).

Current conventional methods include laser-Doppler flowmetry (LDF), which provides information about cerebral blood flow (CBF) but is limited by the number of isolated points in the brain (approximately 1 mm^3) (Dirnagl et al. 1989) and does not show spatial evolution of CBF changes. Other methods, based on magnetic resonance imaging (MRI) (Calamante et al. 1999) and emission tomography (ET) (Heiss et al. 1994), provide spatial maps of CBF, but once again are limited in their spatiotemporal resolution. The drawback of the aforementioned methods is the need to mechanically scan the probe or beam over the test area. A simple method that is able to provide real-time spatially resolved CBF images would thus aid in the experimental studies of functional cerebral activation and cerebral pathology.

Single-exposure speckle photography was first mentioned in Fercher and Briers (1981) where the laser-illuminated area in study was photographed, and the exposure time was set long enough to allow the faster fluctuating speckles (generated by moving particles) to be averaged out. The technique was then proposed for retinal blood flow in Briers and Fercher (1982). The technique was based on a two-stage process where film had to be developed and the image analyzed subsequently. Fully digital techniques were reported in Webster and Briers (1994), Webster (1995), and Briers and Webster (1995). A change in technique to reflect the nonanalog nature was called laser speckle contrast analysis (LASCA). LASCA is a means of providing full-field and real-time measurement of blood flow using first-order statistics of the time-integrated speckle as suggested by Briers and Webster (1996). A 2-D array CCD camera with focusing optics was used to detect the speckle pattern formed by light reflected from tissue illuminated by a divergent laser beam. Analyses of the speckle pattern contrast may provide information about the average velocity of red blood cells. The speckle imaging technique has been used as a minimally invasive method of imaging transport in biological tissues such as the retina (Aizu et al. 1992), skin (Ruth 1994), capillary blood flow in a human hand (Briers and Webster 1995, 1996), as well as CBF in rats (Dunn et al. 2001).

In LASCA (Briers and Webster 1996), an MxM pixel window is used. The smaller the value of M, the lower the statistical validity, while a larger the value M lowers the resulting effective (or perceived) resolutions. The main disadvantage of LASCA is the loss of effective resolution caused by the downsampling of a window of pixels to obtain the spatial statistics required in the analysis. For applications in which analysis of smaller areas such as blood vessels is required, spatial resolution should not be compromised.

In this chapter, we report a new technique, tLASCA that processes statistics primarily in the temporal direction using the LASCA equation, proposed by Briers

and Webster (1996). This technique is thoroughly compared with existing techniques for signal processing of laser speckle images, including the spatial-based sLASCA and the temporal-based mLSI techniques. sLASCA (Dunn et al. 2001) is an improvement of the basic LASCA technique. In sLASCA, the derived contrasts are further averaged over a predetermined number of raw speckle images. mLSI (Cheng et al. 2003), on the other hand, is the modified laser speckle imaging (mLSI) technique in which temporal statistics are processed using technique developed by Ohtsubo and Asakura (1976).

The rest of this chapter is organized as follows. The experimental techniques and setup are discussed in Sect. 11.2. In Sect. 11.3, the results are presented, followed by a discussion of the results obtained. The chapter ends with conclusions and observations in Sect. 11.4.

11.2 Experimental Techniques and Setup

11.2.1 Experimental Setup

Male Sprague–Dawley rats (250–300 g) were used for the imaging. Animal care and experimental procedures were carried out in accordance with University guidelines laid out adhering to the *Basic Principles of the International Guiding Principles for Biomedical Research Involving Animals* (1985). The animals were anaesthetized with urethane (1.25 mg/kg) and mounted onto a stereotaxic frame (Stoelting). A burr hole of diameter 6 mm was drilled into the skull and thinned to the dura mater. The site of the imaging hole was made 2 mm anterior to Bregma and 3 mm lateral to the midline. Saline was used to lower the temperature during surgery and to keep the exposed surfaces moist until imaging was commenced.

Figure 11.1 shows the experimental setup of the speckle imaging components. A laser diode (Sanyo, 782.6 nm) was used to irradiate the imaging site. The laser operation was controlled with a shutter (UniBlitz) of frequency 10 Hz. A monochrome 12-bit charge-coupled device (CCD) Coolsnap camera (Roper Scientific), with 1,040 lines by 1,392 columns corresponding to 1,040 × 1,392 pixels resolution, was positioned over the imaging site.

Each pixel is of size 4.65 μm × 4.65 μm. 2 × 2 hardware binning was performed by the camera, and a frame of 520 × 696 CCD readouts was output to the PC for software processing. Therefore, the true image is of size 4.8 mm × 6.5 mm. We note here that the values commonly used for laser speckle statistical analysis are in fact the CCD readouts, but most literature refer to them as pixels, possibly under the assumption that the camera binning factor is set at 1 × 1. The distinction between pixels and readouts only matters when the real size of the vessel is to be considered. We use the term pixels in all our formulae to be consistent with the literature, and use readouts when calculating the real size of the vessels. Speckle image processing was then performed on an Intel 1.7-GHz processor with 1-GB RAM using

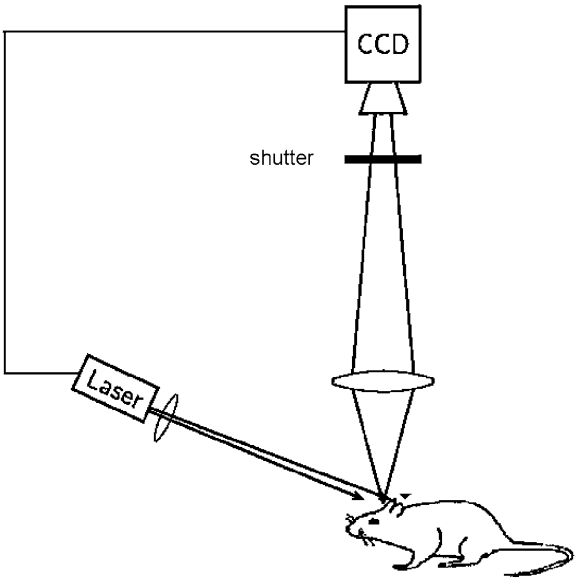

Fig. 11.1 Experimental setup of speckle imaging of cerebral blood flow under no external stimulation

MATLAB software. The exposure time of the CCD was adjusted to be 10 ms, and the images were acquired over a period of 6 s under baseline conditions (with no external stimulation). A total of 60 images were acquired in the process.

11.2.2 Laser Speckle Imaging

Laser speckle is a random interference pattern produced by the coherent addition of scattered laser light with slightly different path lengths. This random interference pattern can be captured on camera when an area is illuminated by a laser beam. The resultant image of 520×696 CCD readouts is a grainy speckle pattern. No apparent useful information can be obtained, and hence statistical image processing must be performed to obtain the speckle contrast.

When an object moves, the speckle pattern it produces changes. For short movements of a solid object, the speckles move with the object and they remain correlated. For longer movements, they decorrelate and the speckle pattern changes completely (Briers and Webster 1996). Decorrelation also occurs when the light is scattered from a large number of individual moving scatterers, such as particles in a fluid.

If the light scattering particles are in motion, a time-varying laser speckle is produced at each pixel on the captured image. Assuming ideal conditions for producing the speckle pattern – a single-frequency laser and a perfectly diffusing surface with a Gaussian distribution of surface height – it can be shown that the standard deviation

of the intensity variations is equal to the mean intensity (Briers and Webster 1996). Hence quantitative blood flow information can be obtained by the spatial intensity variations of the speckle contrast varying from 0 to 1. A speckle contrast of 1 indicates no blurring and thus no motion, whereas a speckle contrast of 0 means that the light scattering particles are moving fast enough to average out all the speckles.

In the following sections, background mathematical derivations are reviewed for completeness, and the applications of them in various image processing techniques are discussed in Sects. 11.2.3–11.2.6.

11.2.2.1 Effect of 2 × 2 or *n* × *n* Hardware Binning on the Camera

As suggested by Dunn et al. (2001), the speckle size should be set equal to the pixel size. If the resolution of the camera is 1,040 lines by 1,392 columns and there is no binning (binning = 1), the capture image may experience aliasing. To avoid, while not being able to eliminate completely, aliasing, 2 × 2 (or higher) hardware binning can be applied, so that each CCD readout is a spatial average of the values of its constituting pixel sensors, as indicated in here (Jain 1989):

$$v\,(m,n) = \frac{1}{N_{\mathrm{W}}} \sum_{k \in W} \sum_{l \in W} y\,(m-k, n-l)\,, \tag{11.1}$$

where $v(m,n)$ and $y(m,n)$ are pixels of the output and input images, respectively, $a\,(k,l)$ is $1/N_{\mathrm{W}}$, and N_{W} is the number of pixels in the window W.

11.2.2.2 Speckle Contrast, *K*

The speckle contrast for a center pixel is computed using the speckle intensity distribution obtained from an M x M window of surrounding pixels (Briers and Webster 1996):

$$K = \delta_{\mathrm{s}} / \langle I \rangle\,, \tag{11.2}$$

where K, δ_s, and $\langle I \rangle$, are the speckle contrast, the spatial standard deviation, and the spatial mean intensity of a window of pixels, respectively. As a result, the effective resolution of the contrast image will be reduced by a factor of M in each dimension of the original image. It has been reported that a 5 × 5 or 7 × 7 pixel window is used to generate the spatial statistics. In this chapter, we examine the use of 3 × 3, 5 × 5, 7 × 7, and 9 × 9 pixels window to see how valid their statistics are, and how well their resulting speckle contrast is viewed.

11.2.2.3 Decorrelation Time, τ_{c}

Assuming that the scattering particles (in this case red blood cells) are of uniform size and have Newtonian flow, the speckle contrast K of a time-integrated speckle

over the CCD *exposure time* T is examined using the model reported by Fercher and Briers (1981):

$$K = \{(\tau_c/2T)\,[1 - \exp(-2T/\tau_c)]\}^{1/2} \tag{11.3a}$$

The homogeneous Lorentzian (Weisstein 2008) profile is given by:

$$K = \left\{\frac{\tau_c}{2T}\left[2 - \frac{\tau_c}{T}\left(1 - e^{-2T/\tau_c}\right)\right]\right\}^{1/2} \tag{11.3b}$$

and the inhomogeneous Gaussian (Jakeman and Ridley 2006) profile by:

$$K = \left\{\frac{\tau_c}{2T}\left[\sqrt{2\pi}erf\left(\frac{\sqrt{2}T}{\tau_c}\right) - \frac{\tau_c}{T}\left(1 - e^{-2(T/\tau_c)^2}\right)\right]\right\}^{1/2}, \tag{11.3c}$$

where τ_c is the decorrelation time of the intensity fluctuations. Although pointed out in Duncan and Kirkpatrick (2008) that (11.3a) is an incorrect formula for K, we keep it so that quantitative comparison with other models, especially when used with in vivo data, can be made. The exposure time T is an important factor in image capturing because if it is set too long, the speckles would average out, yielding a low contrast image. Conversely, if the exposure time is set to be sufficiently short, a high contrast image can be produced. In this experiment, the CCD exposure time T was set to 10 ms.

11.2.2.4 Mean Flow Velocity, υ_c

The relationship between τ_c and *mean flow velocity*, υ_c given by Briers and Webster (1996) is as follows:

$$\upsilon_c = \lambda/2\pi\tau_c, \tag{11.4}$$

where λ is the wavelength of the laser light used. We note that this relationship is seen as speculative and gives no first principles argument as to its veracity (Duncan and Kirkpatrick 2008). However, like many other researchers (Dunn et al. 2001), we use it to gauge the possible velocity range measured using laser speckle imaging, and compare it against the values invasively measured using the technique reported in Mesenteric Arterial Branches Measurement in the R (http://www.transonic.com).

As the $\lambda/2\pi$ term is constant, the mean velocity is directly proportional to $1/\tau_c$. Since the wavelength used is 782.6 nm and π is known, we now have $\upsilon_c = 0.12/\tau_c\,\mu$m/s. A more complicated approximation given by Bonner and Nossal (1981) that takes particle size into consideration calculates velocity as $\upsilon_c = 3.5/\tau_c\,\mu$m/s. From the statistically obtained value K in (11.2) and the relationship in (11.3a–11.3c), $1/\tau_c$ can be derived and used to estimate the scatterers' velocity υ_c in the applications of LASCA.

We note that, K is proportional to $\log(\tau_c/T)$, while velocity υ_c is proportional to $1/\tau_c$. Therefore, K, via τ_c, possesses a "log-inverse" relationship to scatterers' velocity υ_c.

11.2.2.5 Parameter, *N*

Cheng et al. (2003) proposed a technique in which the first-order temporal statistics of a time-integrated speckle pattern can be used to obtain velocity information as a 2-D derivation from the technique developed by Ohtsubo and Asakura (1976). The mathematical equation for obtaining parameter N uses the mean intensity and mean-square intensity of a group of pixels over n temporal frames. In particular, each pixel (i,j) of a particular frame can be computed by:

$$N_{\mathrm{mLSI}(i,j)} = \frac{\left[\left\langle I^2_{i,j,t}\right\rangle_n - \left\langle I_{i,j,t}\right\rangle^2_n\right]}{\left\langle I_{i,j,t}\right\rangle^2_n}, \tag{11.5}$$

where $I_{i,j,t}$ and $I^2_{i,j,t}$ are the instantaneous intensity and instantaneous square intensity of the (ith, jth) pixel at the tth frame of the raw speckle image, respectively. $\left\langle I_{i,j,t}\right\rangle_n$ and $\left\langle I^2_{i,j,t}\right\rangle_n$ are the mean intensity and mean-square intensity of the (ith, jth) pixel over n consecutive frames, respectively. $N_{\mathrm{mLSI}(i,j)}$ is said to be inversely proportional to the velocity of the scattering particles. A porcelain plane connected to a stepper motor moving at a controlled speed of 0.018–2.3 mm/s was used to model the moving particles and used as a reference for velocity derivation based on the $1/N_{\mathrm{mLSI}}$ value.

In the following paragraphs, two reported K-value speckle image processing techniques are described in Sects. 11.2.3 and 11.2.4. A temporal-based K-value technique is reported in Sect. 11.2.5. The N-value image processing technique is discussed in Sect. 11.2.6. The performances and processing times associated with the techniques are studied and analyzed in Sect. 11.3.

11.2.3 Laser Speckle Contrast Analysis

In the LASCA technique (Briers and Webster 1996), the speckle contrast K for a center pixel is computed using the intensity distribution obtained from an $M \times M$ window of surrounding pixel values using (11.1). The *speckle image* of size 520 rows by 696 columns is used to generate the *contrast image* of size (520) × (696).

The algorithm works as follows. Each pixel in the contrast image is obtained by replacing the center pixel in the surrounding $M \times M$ window of the speckle image with its K_{LASCA} contrast value computed by (11.2). The block is then moved by one pixel and the process is repeated. The resulting contrast image has values ranging

from 0.0 to 1.0. It is then contrast-stretched and converted to a color-mapped image for display. Note here that the computed K values are used for statistical and velocity analyses, whereas the contrast-stretched and color-mapped values are for viewing. Even though the display resolution of the contrast image maintains at 520×696, the effective (or perceived) resolution has been reduced to $(520/M) \times (696/M)$.

11.2.4 Spatially Derived Contrast Using Temporal Frame Averaging

Spatially derived contrast using temporal frame averaging (sLASCA) (Dunn et al. 2001) is an improvement of the basic LASCA technique in which the derived contrasts are further averaged over a predetermined number of raw speckle images. Using sLASCA, the display resolution remains at 520×696, while the effective resolution is $(520/M) \times (696/M)$. The averaging operation should make viewing the resulting image more pleasant.

The algorithm works as follows. For each frame, a sliding $M \times M$ window is used to compute and generate a temporary frame of contrast values using (11.1). After the K values of all the frames are computed, they are averaged according to a preset number of frames n. If $K_{i,j,1}$, $K_{i,j,2}$, . . ., and $K_{i,j,n}$ denote the respective consecutive contrast values at pixel (i, j) in frames 1, 2, . . ., n, the contrast K_{sLASCA} is given by:

$$K_{\text{sLASCA}}(i, j) = \frac{K_{i,j,1} + K_{i,j,2} + \ldots + K_{i,j,n}}{n}. \tag{11.6}$$

After the computation of K_{sLASCA} values, the resulting image is contrast-stretched and converted to a color-mapped image for display. It is noted that the number of pixels involved using an $M \times M$ sliding window and averaging over n frames is $(M)(M)(n/2)$, where n is the number of temporal frames.

11.2.5 Temporally Derived Contrast

In this section, we discuss the first-order temporal statistics of time-integrated speckle pattern called temporally derived contrast (tLASCA). tLASCA works on the statistics along n frames in the temporal dimension. Therefore, it is able to maintain both display and effective resolutions of an image. Also, as long as the number of temporal statistics is adequate, the spatial window size M does not affect the validity of the contrast values.

The algorithm works as follows. For each frame, the contrast value K_{sLASCA} of pixel (i, j) of a particular frame is computed by:

$$K_{\text{tLASCA}(i,j)} = \frac{1}{9} \sum_{r=i-1}^{r=i+1} \sum_{c=j-1}^{c=j+1} \frac{\delta_{i,j,t}}{\langle I_{i,j,t} \rangle}, \tag{11.7}$$

where $\delta_{i,j,t}$ is the standard deviation of all pixels at (i,j) in n frames along the temporal dimension, and $\langle I_{i,j,t} \rangle$ is the mean intensity of all pixels at (i,j) in n frames along the temporal dimension, and K_{sLASCA} is calculated as an average over a 3×3 spatial observation window. It is noted that the number of pixels involved in a 3×3 pixel observation window using tLASCA is (3)(3)(n/2), where n is the number of temporal frames. This small window of observation ensures good statistics while not including statistics of slower moving or nonmoving particles. After the computation of K_{sLASCA} values, the resulting image is contrast-stretched and converted to color-mapped image for display.

11.2.6 Modified Laser Speckle Imaging

Cheng et al. (2003) proposed a technique in which N values are calculated by sampling one point in each frame and collecting the points along the temporal dimension. Recall from Sect. 11.2.2.4 that $N_{\text{mLSI}(i,j)}$ is inversely proportional to the velocity of the scattering particles.

The algorithm works as follows. For each frame, the contrast value of each pixel $N_{\text{mLSI}(i,j)}$ of a particular frame is computed by (11.5). It is not mentioned in Cheng et al. (2003) whether an observation window was used to collect the value N_{mLSI}. However, for completeness, an observation window of 3×3 pixels is assumed in this chapter. After the computation of N_{mLSI} values, the resulting image is contrast-stretched and converted to color-mapped image for display. Again, the N_{mLSI} values are used for statistical and velocity analyses, whereas the contrast-stretched and color-mapped values are for viewing.

11.3 Results and Discussion

In this section, we examine the performances of four image processing techniques – LASCA, sLASCA, tLASCA, and mLSI – in terms of K or N, subjective and objective visual qualities, and processing times. The flow rate at specific sites are also estimated according to the Lorentzian and Gaussian models. The images were acquired under baseline conditions with no external stimulations applied to the test animal. Under such conditions, we to observe a *normal* velocity of blood flow resulting in reasonably fluctuating values of K or N. Blood flow and pressure are unsteady. The cyclic nature of the heart pump creates pulsatile conditions in all arteries. The heart ejects and fills with blood in alternating cycles called *systole* and *diastole*. Blood is pumped out of the heart during *systole*. The heart rests during *diastole*, and no blood is ejected. In this case, our intention was not to estimate the instantaneous flow, but the average flow rate over a group of frames. The rat cortex image taken under white light illumination is shown in Fig. 11.2. Two windows of observation, L and S, were selected.

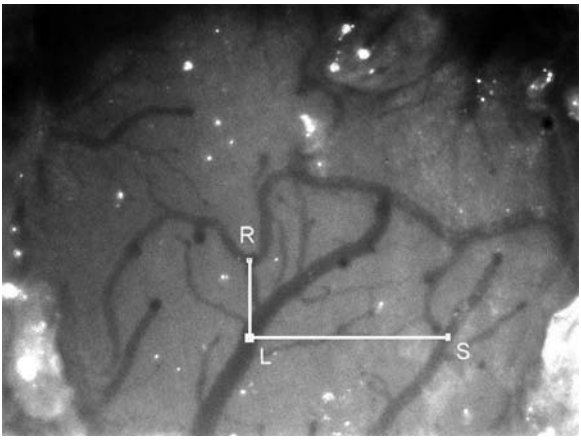

Fig. 11.2 Image of rat cortex illuminated with white light. Two regions of interest overlying a large arteriole (L) and a smaller arteriole (S) are selected for computing speckle contrasts and blood flow rates if possible. A point of reference (R) is also shown

A small window L was placed at the center of the large arteriole. L was selected directly below reference R (center of Y-shaped vessel over the largest vessel). Point R was selected so that a common reference can be identified among many white light and speckle images. A small window S was placed at the center of the smaller arteriole, to the right of L. Under white light illumination, although it was possible to view the blood vessels clearly, no quantitative data regarding flows could be obtained. The diameters of the cross sections of the large and small arterioles were approximately 279 μm (30 CCD readouts × 2 × 4.65) and 186 μm (20 CCD readouts × 2 × 4.65), respectively. The factor 2 comes from the fact that 2 × 2 hardware binning was performed at the CCD image sensor.

Since the sizes of the large and small blood vessels are different, data collection windows L and S should be of variable sizes to ensure correct statistics for the derivations of K or N values. Note that there are two different window concepts used in this paper. An *averaging window* of $M \times M$ pixels is used for downsampling image using LASCA and sLASCA techniques. On the other hand, an *observation window* of $O \times O$ is used for statistics collection at L and S using tLASCA and mLSI techniques. In the following sections, the effects of window size M on K_{LASCA} and the number of frames n on K_{sLASCA}, K_{tLASCA}, and N_{mLSI} are studied.

11.3.1 Effects of Window Size M on K_{LASCA}

We first studied the effect of window size M on the contrast value K using LASCA, denoted as K_{LASCA}. To obtain the contrast image using LASCA, a window of size

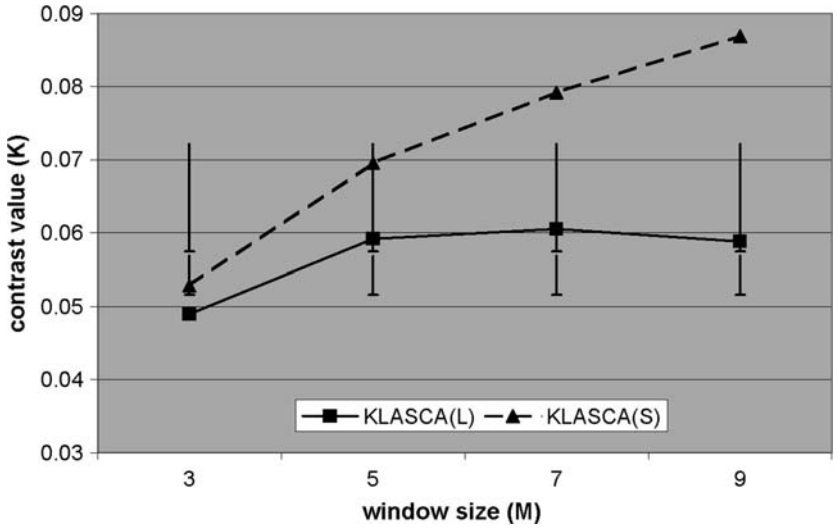

Fig. 11.3 Contrast values (K) at L and S calculated using LASCA with windows $M = 3$, 5, 7, and 9. The K_{LASCA}(L,S) at $M = 3$ exceed the 1 standard deviation (SD) error range as shown by the lower-half error bars. The K_{LASCA}(L,S) at $M = 5$, 7, and 9 lie within the 1-SD error range

M was used to generate K_{LASCA}. The corresponding K_{LASCA} values were obtained at both L and S. Figure 11.3 plots the K_{LASCA} values at L and S using window sizes $M = 3$, 5, 7, and 9.

In Fig. 11.3, the solid line connecting the squares represents K_{LASCA} obtained at L, denoted by K_{LASCA}(L), while the dotted line connecting the triangles represent K_{LASCA} obtained at S, denoted by K_{LASCA}(S). K_{LASCA}(L or S) is the average K_{LASCA} value carefully obtained by placing over the center of the vessel in the study an observation window whose size depends on the size of this vessel. For small vessels careful selection of observation window size must be observed.

As window size M increases, it is likely that more pixel values corresponding to the slower or non-moving scatterers are included in the computation of K_{LASCA}. Therefore, K_{LASCA} tends to increase as M increases. K_{LASCA}(L) is within 25% of each other, while K_{LASCA}(S) is within 65% of each other (bars not shown). As selection or rejection criterion, we use one standard deviation error (SD) as the threshold to select the K values. When a K value exceeds 1-SD, we reject this value.

From Fig. 11.3, K_{LASCA}(L,S) at $M = 3$ exceeds one SD error range as shown by the lower error bars. K_{LASCA}(L,S) at $M = 5$, 7, and 9 lie within one SD error range (not shown), and therefore are selected. Also, K_{LASCA}(L) values at $M = 5$, 7, and 9 are closer together and compare to those at K_{LASCA}(S). Note, however, that when $M = 9$, there is greater loss of effective resolution than at $M = 3$, 5, and 7. As a result, the smaller vessels disappear from the image. We will demonstrate this in Sect. D.1. $M = 9$ will not be discussed further. In the next section, we investigate the effects of n, the number of frames for temporal averaging, on K_{sLASCA}.

11.3.2 Effects of n on K_{sLASCA}

From Sect. 11.3.1, we reject the window size $M = 3$ due to its larger error in K value. We also reject window size $M = 9$ due to the loss in effective resolution. In the sLASCA technique, due to the nature of temporal frame averaging, more temporal statistics are involved in the computation of K_{sLASCA}; we keep $M = 3$, and use its contrast values to compare with other techniques.

Fig. 11.4a, b shows the plots of contrast value K at L and S computed using sLASCA technique, denoted by K_{sLASCA}, with window size $M = 3$, 5, and 7, and averaged over $n = 2$–30 frames. The value K(LM) represents the K_{sLASCA}(L) using window size M, while K(SM) represents K_{sLASCA}(S) using window size M.

In Fig. 11.4a, the values of K(L) using $M = 3$, 5, and 7 are within 8% of their respective values after $n = 4$, and their average values are K_{sLASCA} (L3) = 0.0434, K_{sLASCA} (L5) = 0.0530, and K_{sLASCA} (L7) = 0.0579. If 1-SD error bars are placed along the three average K values (plot not shown), the value at $M = 3$ slightly falls out of range, while those at $M = 5$, 7 are within. The correlation of [K(L3), K(L5)] is 0.99, of [K(L5), K(L7)] is 0.95, and of [K(L3), K(L7)] is 0.89.

In Fig. 11.4b, the values of K(S) using $M = 3$, 5, and 7 are within 8% of their respective values after $n = 10$, and their average values are K_{sLASCA} (L3) = 0.0590, K_{sLASCA} (L5) = 0.0734, and K_{sLASCA} (L7) = 0.0808. If 1-SD error bars are placed along the three average K values (plot not shown), the value at $M = 3$ slightly falls out of range, while those at $M = 5$, 7 are within. The correlation of [K(S3), K(S5)] and [K(S5), K(S7)] is 0.99, while the correlation of [K(S3), K(S7)] is 0.96.

By using the 1-SD error range, it can be shown that $M = 3$ could be used for sLASCA, but its validity is not guaranteed. On the other hand, the correlation values show that window size $M = 5$ is the best selection for computing K at both L and S.

11.3.3 Effects of n on K_{tLASCA} and N_{mLSI}

Figure 11.5 shows plots of K_{tLASCA} and N_{mLSI} with the number of frames $n = 2$–30.

K_{tLASCA} at L, denoted by K_{tLASCA}(L), is within 10% of each other after $n = 10$, while K_{tLASCA} at S, denoted by K_{sLASCA}(S), is within 8% after $n = 16$. K_{tLASCA}(L) is averaged (from frames 10–30) at 0.0564, while K_{tLASCA}(S) is averaged (from frames 16–30) at 0.0766.

Also in Fig. 11.5, N_{mLSI} at L, denoted by N_{mLSI}(L), is within 25% of each other after $n = 10$, while N_{mLSI} at S, denoted by N_{mLSI}(S), is within 15% after $n = 16$. N_{mLSI}(L) is averaged (from frames 10–30) at 0.00307, while N_{mLSI}(S) is averaged (from frames 16–30) at 0.00573. We note that the range of N_{mLSI} values is from 3.0×10^{-3} to 6.0×10^{-3}, which is reasonable and in agreement with the numbers reported by Cheng et al. (2003).

The correlation of K_{tLASCA}(L) and N_{mLSI}(L) is very high at 0.96, while the correlation of K_{tLASCA}(S) and N_{mLSI}(S) is moderately high at 0.89. The range of N

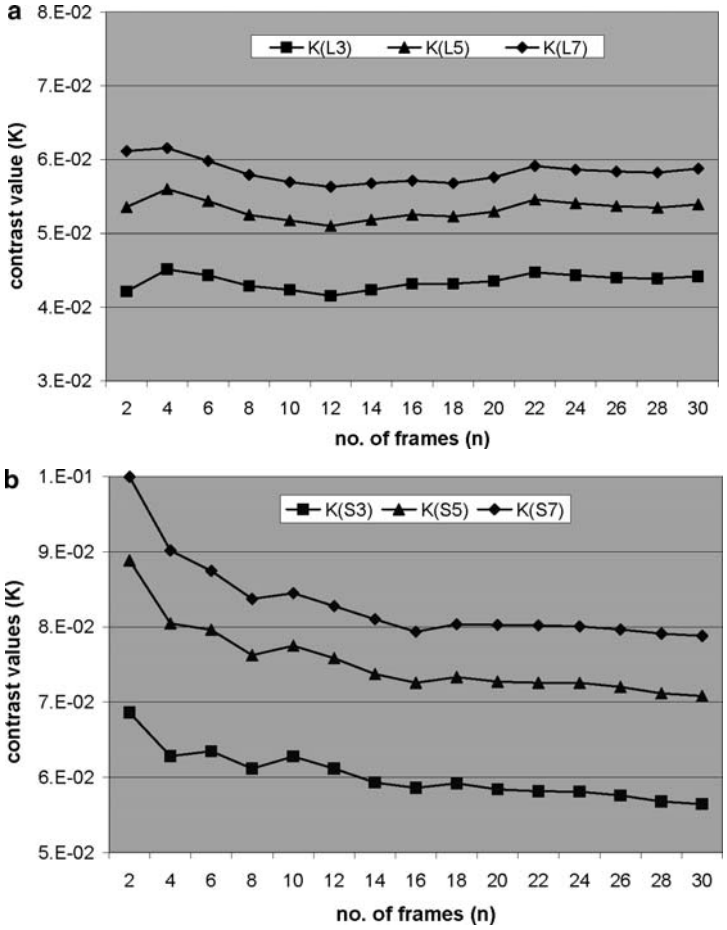

Fig. 11.4 (**a**) Contrast values (K) at L calculated using sliding windows and temporal frame averaging (sLASCA) over n frames. Window size (M) = 3, 5, and 7. K_{sLASCA} (L3) ~ 0.0434, K_{sLASCA} (L5) ~ 0.0530, and K_{sLASCA} (L7) ~ 0.0579. K_{sLASCA} values are within 8% after $n = 4$. Correlation [K_{sLASCA} (L3), K_{sLASCA} (L5)]= 0.99; correlation [K_{sLASCA} (L5), K_{sLASCA} (L7)]= 0.95; correlation [K_{sLASCA} (L3), K_{sLASCA} (L7)]= 0.89. (**b**) Contrast values (K) at S calculated using sliding windows and temporal frame averaging (sLASCA) over n frames. Window size (M) = 3, 5, and 7. K_{sLASCA} (S3) ~ 0.0590, K_{sLASCA} (S5) ~ 0.0734, and K_{sLASCA} (S7) ~ 0.0808. K_{sLASCA} values are within 8% after $n = 10$. Correlation [K_{sLASCA} (S3), K_{sLASCA} (S5)]= 0.99; correlation [K_{sLASCA} (S5), K_{sLASCA} (S7)]= 0.99; correlation [K_{sLASCA} (S3), K_{sLASCA} (S7)]= 0.96

values is small, and therefore the N values are subject to larger differences (25%). On the other hand, the range of K values is large, and therefore the K values are subject to smaller differences (10%).

Based on the knowledge that (a) $1/N_{\text{mLSI}}$ is proportional to the velocity of blood cells (Ohtsubo and Asakura 1976), (b) K_{tLASCA} possesses a "log-inverse" relationship to the estimated velocity of blood cells (11.2)–(11.4), and (c) the correlations

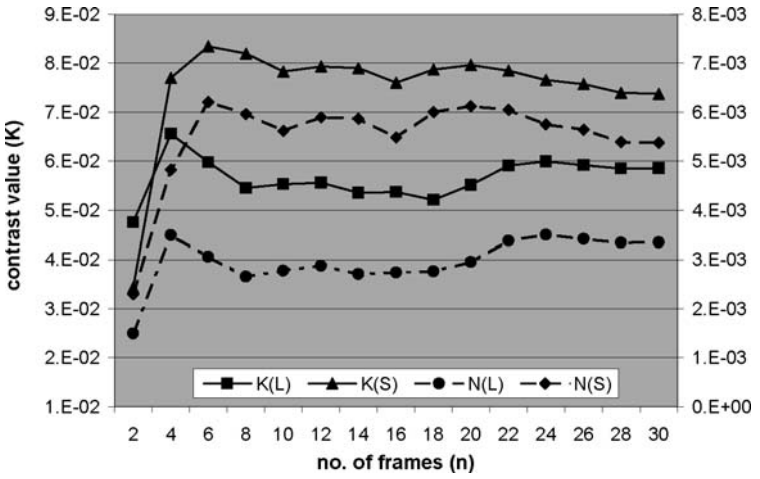

Fig. 11.5 Contrast values (K) – plotted on the left axis – calculated using temporally derived contrast (tLASCA) over n frames. K_{tLASCA} (L) ~ 0.0564 (within 10% of each other after $n = 10$), K_{tLASCA} (S) ~ 0.0766 (within 8% of each other after $n = 16$). Values (N) – plotted on the right axis – calculated using mLSI over n frames. N_{mLSI} (L) ~ 0.00307 (within 25% of each other after $n = 10$), N_{mLSI} (S) ~ 0.00573 (within 15% of each other after $n = 16$). Correlation [K_{tLASCA} (L), N_{mLSI} (L)]$= 0.96$, Correlation [K_{tLASCA} (S), N_{mLSI} (S)]$= 0.89$

of K_{tLASCA} and N_{mLSI} are reasonably high at L and S, we conclude that the tLASCA technique can be used to derive contrast values and velocity of blood cells.

We have used LASCA, sLASCA, tLASCA, and mLSI techniques to derive K and N values, which can further be used to estimate velocity in the blood vessel. The K(L) values range from 0.0432 to 0.0579 for $M = 3$, 5, and 7, while K(S) values from 0.0590 to 0.0808 also for $M = 3$, 5, and 7. The N(L) values are averaged at 0.00307, while N(S) values are averaged at 0.00573. LASCA reduces effective resolution and achieves stable statistics when $M = 5$ or 7. sLASCA improves effective resolution by frame averaging and achieves stable statistics when $M = 5$ or 7.

11.3.4 Comparisons of K_{LASCA}, K_{sLASCA}, and K_{tLASCA}

In Fig. 11.6a, we plot all the contrast values K corresponding to LASCA using $M = 3$, 5, and 7 (set of points 1–2–3); sLASCA using $M = 3$, 5, and 7 (set of points 4–5–6); and tLASCA (set of points 7).

When using 1-SD as selection criterion, it is found that the K values are most probable when calculated using LASCA ($M = 5$, 7), sLASCA ($M = 5$, 7), and tLASCA. Therefore, the following conclusions can be made. When using LASCA, the window size $M = 5$ or at most 7 can be used, and this confirms the conclusion

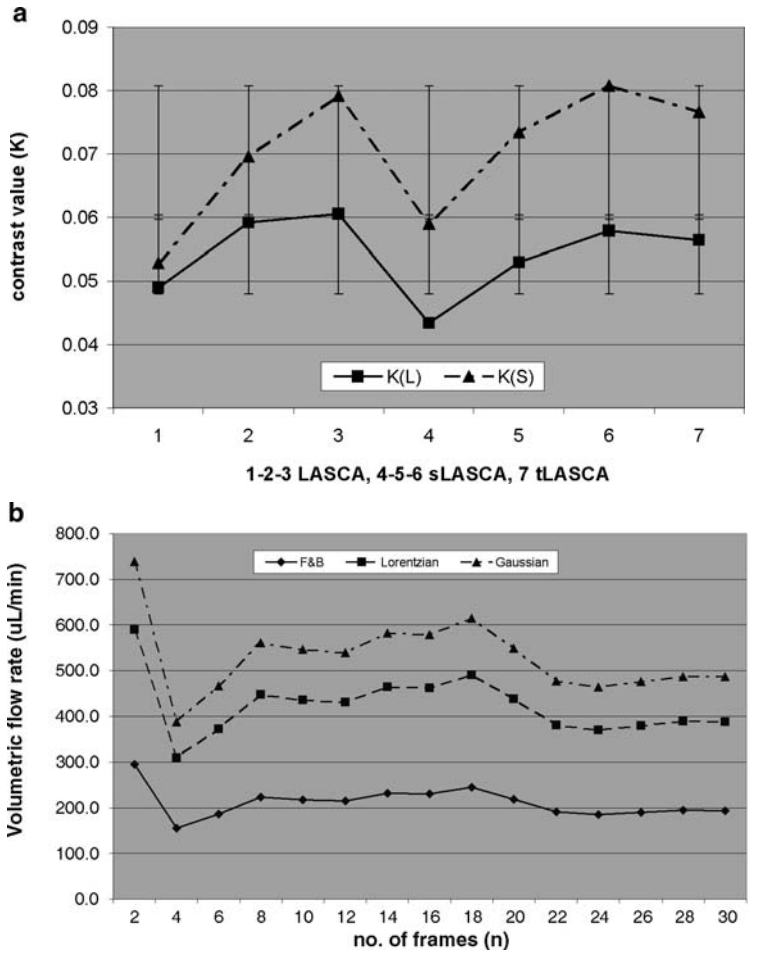

Fig. 11.6 (**a**) Contrast values (K) calculated using three techniques: LASCA (set of points 1–2–3), sLASCA (set of points 4–5–6), and tLASCA (set of points 7). When using 1-SD as selection criterion, it is found that the K values are most probable when computed using LASCA ($M = 5, 7$), sLASCA ($M = 5, 7$), and tLASCA. (**b**) The respective flow rates using Fercher and Briers', Lorentzian, and Gaussian models. Because of unstable data acquisition, the first two values are excluded in the computation of flow rates

made in Briers and Webster (1996). However, the results using $M = 5$ are more probable than $M = 7$. When using sLASCA, the window size $M = 5$ or 7 can be used, and this confirms the conclusion made in Dunn et al.(2001). Under both LASCA and sLASCA, K is not sufficiently probable when $M = 3$. It takes fewer frames ($n = 4$) to obtain stable statistics for large vessels than for smaller vessels ($n = 10$).

When using tLASCA, the values always fall within 1-SD, and thus tLASCA guarantees the value K regardless of the size of the averaging window M as long as more than 10 and 16 frames are used for computations at L and S, respectively. It is

noted that the number of pixels involved in a $O \times O$ pixel observation window using tLASCA is $(O) \times (O) \times (n/2)$ where n is the number of temporal frames involved. This small window of observation ensures good statistics while not including statistics of slower moving or non-moving scatterers.

Using the 1-SD error as the selection criterion, we have reduced the range of K(L) values to 0.0521–0.0600 and K(S) values from 0.0738 to 0.0835. Recall that the diameters of the cross sections of the large and small arterioles are approximately 279 and 186 μm, respectively. The respective flow rates at point L using Fercher and Briers', Lorentzian, and Gaussian models are shown in Fig. 11.6b:

- The flow rates at L are 185–245 μL/min, with a median of 215 μL/min using Fercher and Briers' derivation; 370–490 μL/min, with a median of 430 μL/min using Lorentzian model; and 464–614 μL/min, with a median of 539 μL/min using Gaussian model.
- The flow rates at S are 43–54 μL/min, with a median of 48 μL/min using Fercher and Briers' derivation; 85–109 μL/min, with a median of 96 μL/min using Lorentzian model; and 106–136 μL/min, with a median of 120 μL/min using Gaussian model.

The flow rate at L, using both Lorentzian and Gaussian models, is in good agreement with the flow rate of 218–770 μL/ min invasively measured at a 270-μm-diameter mesenteric arterial branch (Mesenteric Arterial Branches Measurement in the Rat). Using tLASCA accurate flow rates can also be obtained with a minimally invasive technique.

Until this point, the statistically derived contrast value K has been kept unchanged so that the velocities of blood cells can be derived. For better viewing, the image needs to be contrast-stretched and possibly color-mapped.

11.3.5 Evaluations on Visual Qualities

11.3.5.1 Subjective Quality

In this section, the contrast images obtained using the techniques LASCA, sLASCA, mLSI, and tLASCA will be viewed and compared. The purpose is to subjectively evaluate the quality of the resulting contrast images. In Fig. 11.7a–d gray images are *contrast-stretched* to [0, 1] to enhance viewing. The gray and color bars in Fig. 11.7a–h are scaled from 0.0 to 1.0 in increments of 0.1.

In Fig. 11.7, the contrast images resulting from LASCA are small and very grainy. The blood vessels are not connected as lines, but rather as sprays of dots. Among the images, it can be seen that the blood vessels are more visible using LASCA with window sizes $M = 3$ or 5. The vessels in the gray images 11.7a, b could be better perceived by the eyes than their color-mapped counterparts 11.7e, f. However, the color-mapped images 11.7e, f provide more information about the regions of relatively faster or slower moving or nonmoving blood cells. It is noted, however, that the point of reference R, L, and S cannot be clearly recognized.

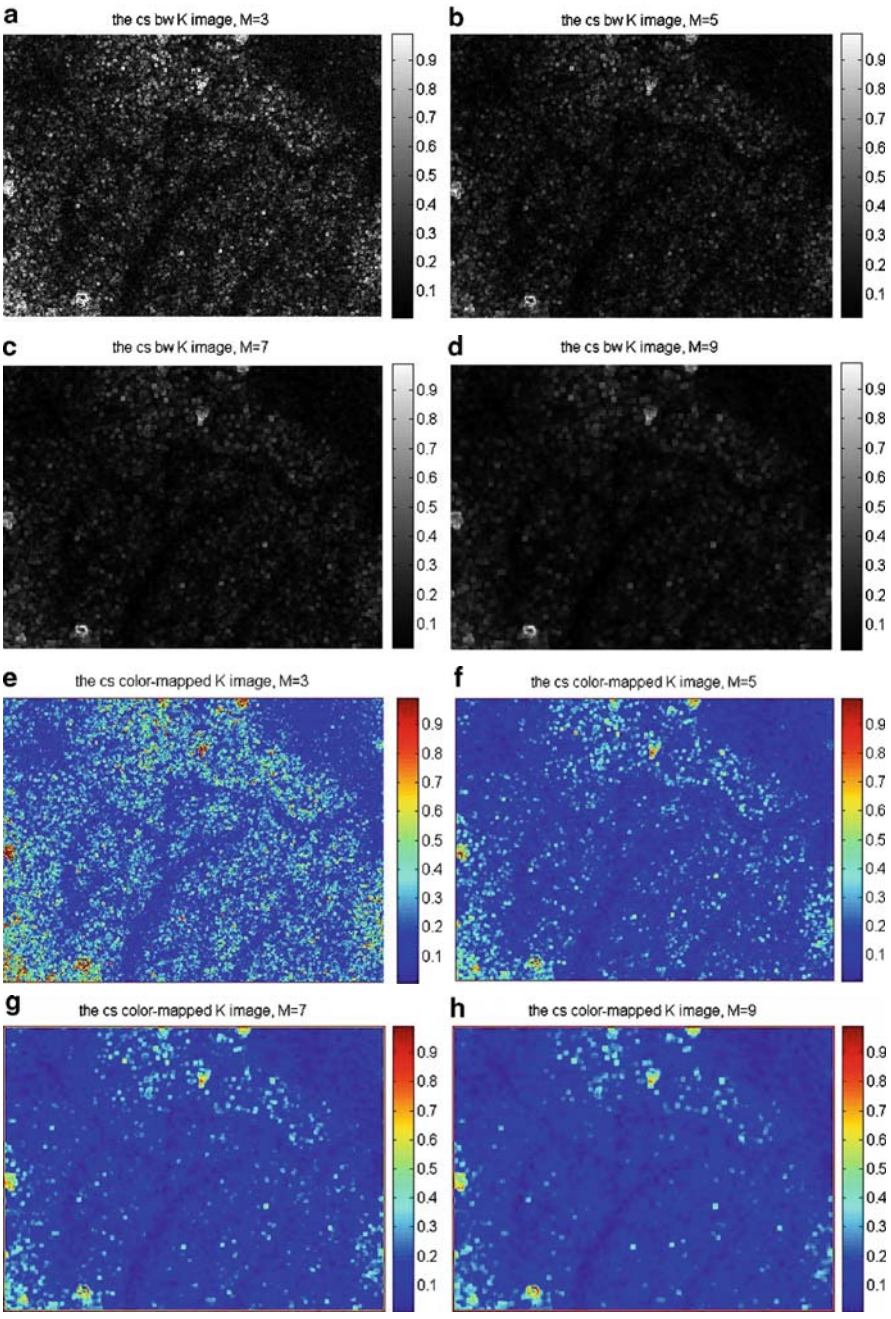

Fig. 11.7 (**a–d**) Gray-scale LASCA images using window sizes $M = 3, 5, 7$, and 9; (**e–h**) the corresponding color-mapped LASCA images. The larger the window size, the blockier the resulting contrast image. Using LASCA, the whole perfusion area (upper left corner) and the larger vessels are seen, but small vessels, especially at S, are too blurred and disconnected to be seen properly

Table 11.1 Subjective quality evaluations of Fig. 7a–h obtained using LASCA by 30 volunteers

Subjective quality	a (bw, $M = 3$)	b (bw, $M = 5$)	c (bw, $M = 7$)	d (bw, $M = 9$)	e (cl, $M = 3$)	f (cl, $M = 5$)	g (cl, $M = 7$)	h (cl, $M = 9$)
5	6.67%	3.33%	0.00%	0.00%	3.33%	3.33%	6.67%	3.33%
4	36.67%	16.67%	3.33%	3.33%	33.33%	30.00%	16.67%	16.67%
3	46.67%	36.67%	16.67%	3.33%	36.67%	36.67%	23.33%	13.33%
2	6.67%	43.33%	53.33%	6.67%	16.67%	20.00%	40.00%	26.67%
1	3.33%	0.00%	26.67%	86.67%	10.00%	10.00%	13.33%	40.00%
Weighted	3.37	2.80	1.97	1.23	3.03	2.97	2.63	2.17

Thirty volunteers were asked to subjectively evaluate the qualities of Fig. 11.7a–h and the results are listed in Table 11.1. All numbers are converted into percentages. The black and white figure is denoted by "bw," while color is denoted by "cl." The quality level-5 corresponds to the sharpest and most informative image, while quality level-1 the dullest and least informative one. The numbers in the bottom row are the weighted averages of the subjective qualities of the respective images.

The data in Table 11.1 show that color images are generally perceived as sharper and more informative. Specifically, Fig. 11.7a, b, e–f were graded above 2.8. The reason is that averaging window size $M = 3$ or 5 retains more information and provides better effective resolution compared to $M = 7$ or 9. Figure 11.7c, d were graded at level 1, while Fig. 11.7g, h were graded between levels 2 and 1. Using LASCA with window $M = 7$ or 9, the effective resolution of images and the associated traces of the vessels have been reduced or lost as expected.

In Fig. 11.8, all gray images are contrast-stretched by [0, 1]. The contrast images are obtained using sLASCA with $M = 3$, 5, and 7, and mLSI and tLASCA. The images shown are obtained by averaging over 4 or 10 frames (left column), and 10 or 16 frames (right column). In general, using fewer frames (n) results in more focused overall images, while using more frames results in more blurred and connected vessels.

Thirty volunteers were asked to evaluate the subjective qualities of Fig. 11.8a–j and the results are listed in Table 11.2. All numbers are converted to percentages. The sizes of the windows and the number of frames averaged are also listed. Quality level 5 corresponds to the sharpest image, and quality level 1 to the dullest one. The numbers in the bottom row are the weighted averages of the subjective qualities of the respective images.

In Table 11.2, Fig. 11.8a, b, i, j was graded above level 3, while Fig. 11.8c, d, g, h was graded above level 2. Figure 11.8e, f was graded below level 2, which is the least sharp. It can be concluded that, for gray-scale images, sLASCA ($M = 3$) and tLASCA ($n = 10$ or 16), followed by mLSI ($n = 10$, or 16) and sLASCA ($M = 5$) techniques, produce the best viewing experiences.

In Fig. 11.9, all color-mapped images are contrast-stretched by [0, 1] prior to color mapping. The color maps of the contrast images provide more details about the relative velocities of different regions. sLASCA ($M = 3$, $n = 4$ or 10), mLSI ($n = 10$ or 16), and tLASCA ($n = 10$ or 16) provide clearer definitions of the blood flows.

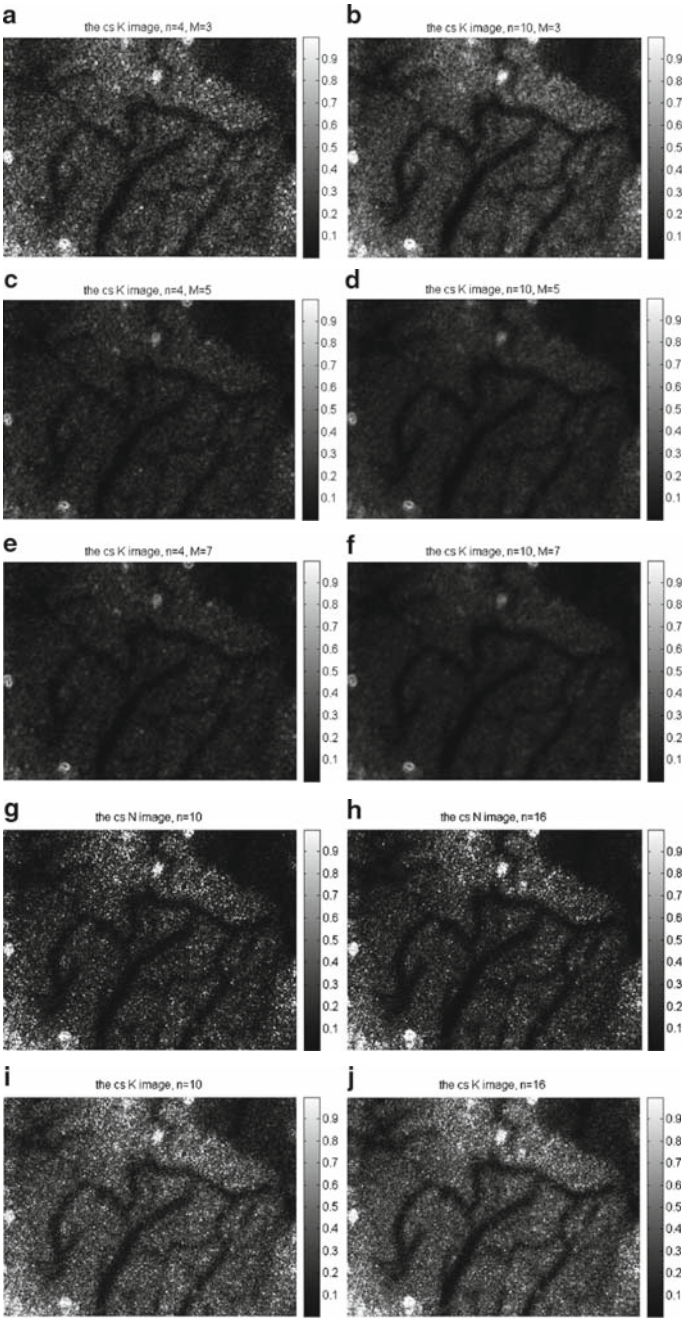

Fig. 11.8 (**a**, **b**) sLASCA ($M = 3$, $n = 4$, 10); (**c**, **d**) sLASCA ($M = 5$, $n = 4$, 10); (**e**, **f**) sLASCA ($M = 7$, $n = 4$, 10); (**g**, **h**) mLSI ($n = 10$, 16); (**i**, **j**) tLASCA ($n = 10$, 16). All gray-scale images have been contrast-stretched by [0, 1]. In general, using smaller number of frames results in more focused overall images, and using larger number of frames results in more blurred and connected vessels

Table 11.2 Subjective quality evaluations of Fig. 8a–j by 30 volunteers

Subjective quality	a (sLASCA, $M = 3$, $n = 4$)	b (sLASCA, $M = 3$, $n = 10$)	c (sLASCA, $M = 5$, $n = 4$)	d (sLASCA, $M = 5$, $n = 10$)	e (sLASCA, $M = 7$, $n = 4$)	f (sLASCA, $M = 7$, $n = 10$)	g (mLSI, $n = 10$)	h (mLSI, $n = 16$)	i (tLASCA, $n = 10$)	j (tLASCA, $n = 16$)
5	10.00%	20.00%	0.00%	3.33%	3.33%	3.33%	0.00%	3.33%	13.33%	20.00%
4	53.33%	30.00%	10.00%	6.67%	3.33%	3.33%	13.33%	13.33%	33.33%	36.67%
3	20.00%	36.67%	20.00%	20.00%	16.67%	10.00%	56.67	56.67%	40.00%	26.67%
2	16.67%	13.33%	60.00%	46.67%	26.67%	23.33%	26.67%	20.00%	10.00%	13.33%
1	0.00%	0.00%	10.00%	23.33%	50.00%	60.00%	10.00%	6.67%	3.33%	3.33%
Weighted	3.57	3.57	2.30	2.20	1.83	1.67	2.67	2.87	3.43	3.57

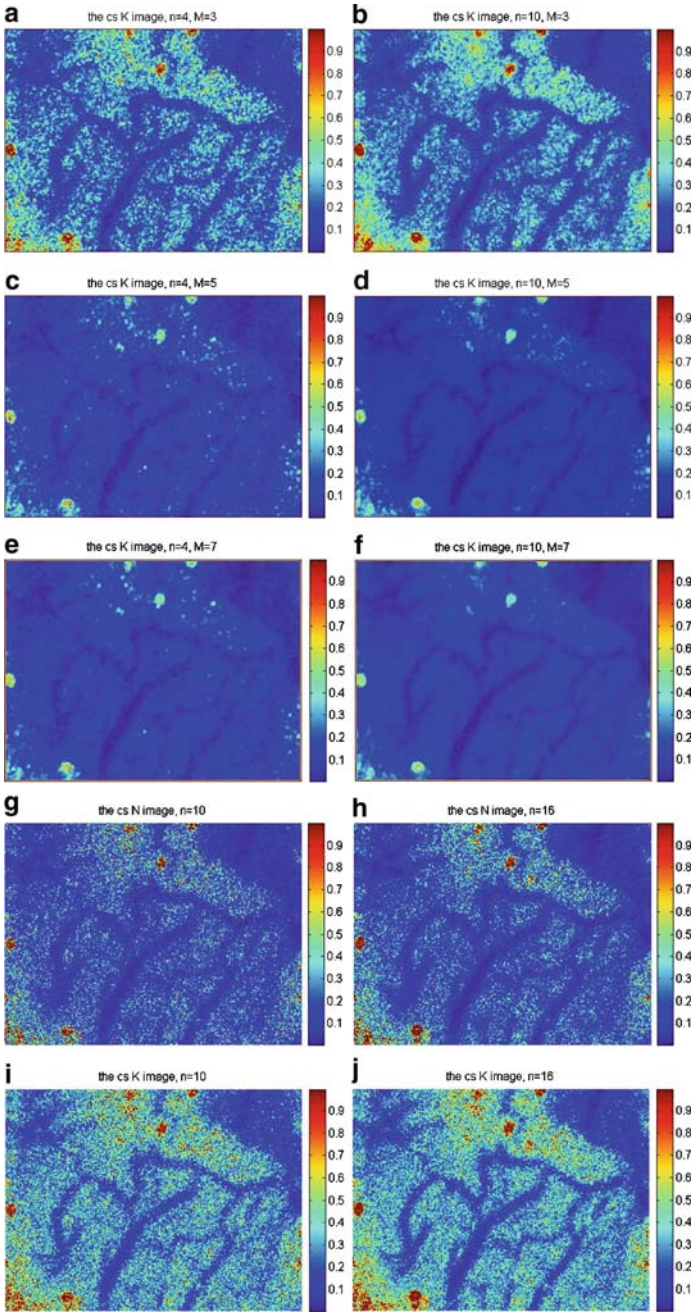

Fig. 11.9 (**a**, **b**) sLASCA ($M = 3$, $n = 4$, 10); (**c**, **d**) sLASCA ($M = 5$, $n = 4$, 10); (**e**, **f**) sLASCA ($M = 7$, $n = 4$, 10); (**g**, **h**) mLSI ($n = 10$, 16); (**i**, **j**) tLASCA ($n = 10$, 16). All color-mapped images had been contrast-stretched by [0, 1] prior to color mapping. Color-mapped images provide relative velocity of blood flows: dark blue representing fast movement of blood cells, whereas dark yellow or red no movement

Thirty volunteers were asked to evaluate the subjective qualities of Fig. 11.9a–j and the results are listed in Table 11.3. The sizes of the windows and the number of frames averaged are also listed. Quality level 5 corresponds to the sharpest image. The numbers in the bottom row are the weighted averages of the subjective qualities of the respective images.

Data from Table 11.3 show that color-mapped images are generally perceived as sharper and more informative. Fig. 11.9a, b, i, j was mostly graded at above level 3, while Fig. 11.9c, d, g, h was mostly graded above 2.5. Figure 11.9e, f was graded around 2.5. It can be concluded that, for color-mapped images, tLASCA ($n = 10$ or 16) and sLASCA ($M = 3$), followed by mLSI ($n = 10$ or 16) produce the best viewing experiences.

tLASCA and sLASCA have resulted in images that are visually superior to those produced by LASCA both in effective resolutions and clarity. The evaluations of image quality have been based on viewers' preferences. In the following section, objective quality will be analyzed and performances of sLASCA and tLASCA will be studied.

11.3.5.2 Objective Quality

We developed an objective evaluation technique for sharpness determination. An image is sharp if it has many well-contrasted lines. In other words, thin and connected lines should be visible to the eyes, and intensity changes drastically from one side of a line to the other. Each image (I) has undergone a blurring (Ib) and a separate sharpening (Is) operation. The image Ib or Is is further discrete-cosine transformed (DCT), and high-frequency coefficients are removed before reconstruction is made. The PSNRs of the respective images are calculated before and after DCT. The % change in PSNR between the blurred and sharpened versions of the same image indicates the level of sharpness of that image. This parameter can be used as the objective measure of speckle images.

For two images I_1 and I_2, if $\%\Delta$ PSNR (I_1) is greater than $\%\Delta$ PSNR (I_2), then I_1 is sharper than I_2. This behavior is interpreted as follows. If an image I_1 is very sharp, its sharpened version, I_1s, will not be any sharper, while its blurred version, I_1b, can be very blurred and significantly different from the original image I_1. Therefore, the difference between PSNR (I_1s) and PSNR (I_1b), denoted by $\%\Delta$ PSNR (I_1), will be large. On the other hand, if an image I_2 is less sharp, its sharpened version, I_2s, is slightly sharper, while its blurred version, I_2b, can be slightly blurred. Therefore, the difference between PSNR (I_2s) and $PSNR$ (I_2b), denoted by $\%\Delta$PSNR (I_2) will be small.

This technique is applied to the contrast gray images generated using sLASCA, mLSI, and tLASCA (after they are contrast-stretched for better viewing), and the results are tabulated in Table 11.4.

According to Table 11.4, the contrast images using tLASCA, ($n = 10$ or 16) produced the largest differences in $\%\Delta$ PSNR or highest objective measure as a result of having lost many sharpness features. sLASCA ($M = 3$) and mLSI come next in sharpness level.

Table 11.3 Subjective quality evaluations of Fig. 9a–j by 30 volunteers

Subjective quality	a (sLASCA, $M = 3$, $n = 4$)	b (sLASCA, $M = 3$, $n = 10$)	c (sLASCA, $M = 5$, $n = 4$)	d (sLASCA, $M = 5$, $n = 10$)	e (sLASCA, $M = 7$, $n = 4$)	f (sLASCA, $M = 7$, $n = 10$)	g (mLSI, $n = 10$)	h (mLSI, $n = 16$)	i (tLASCA, $n = 10$)	j (tLASCA, $n = 16$)
5	10.00%	10.00%	13.33%	10.00%	3.33%	10.00%	3.33%	3.33%	20.00%	26.67%
4	50.00%	46.67%	13.33%	13.33%	20.00%	16.67%	20.00%	23.33%	26.67%	36.67%
3	26.67%	33.33%	36.67%	40.00%	23.33%	20.00%	40.00%	43.33%	23.33%	16.67%
2	13.33%	10.00%	30.00%	26.67%	26.67%	23.33%	26.67%	23.33%	16.67%	16.67%
1	0.00%	0.00%	6.67%	10.00%	26.67%	30.00%	10.00%	6.67%	3.33%	3.33%
Weighted	3.57	3.57	2.97	2.87	2.47	2.53	2.80	2.93	2.53	3.67

Table 11.4 Objective and subjective quality evaluations of the discussed techniques using the proposed measure and their correlations

Techniques	Obj	Subj (BW)	Subj (CL)
sLASCA, $M = 3, n = 4$	39.22	3.57	3.57
sLASCA, $M = 3, n = 10$	37.49	3.57	3.57
sLASCA, $M = 5, n = 4$	30.35	2.30	2.97
sLASCA, $M = 5, n = 10$	30.01	2.20	2.87
sLASCA, $M = 7, n = 4$	25.59	1.83	2.47
sLASCA, $M = 7, n = 10$	25.28	1.67	2.53
mLSI, $n = 10$	36.97	2.67	2.80
mLSI, $n = 16$	36.12	2.87	2.93
tLASCA, $n = 10$	40.58	3.43	3.53
tLASCA, $n = 16$	39.30	3.57	3.67
Correlation with obj. evaluation		0.95	0.86

The correlation between the objective evaluation and the subjective evaluation of gray-scale speckle images is very high at 0.95. The correlation between the objective evaluation and the subjective evaluation of color-mapped speckle images is moderate at 0.86. One possible reason is that the objective evaluation technique is developed based on the gray-scale images, and therefore, its correlation with the subjective evaluation of gray-scale images is better.

tLASCA has been proven to provide more consistent contrast value K (after $n = 10$ or 16, $\pm 8\%$ error) and to yield superior viewing experience in both subjective and objective evaluations. Now that a framework on requirements of visual monitoring contrast images has been established, we will investigate their processing times in the next section.

11.3.6 Processing Time

11.3.6.1 LASCA

The frame processing times of LASCA using $M = 3$, 5, 7, and 9 are 0.4, 1.2, 2.2, and 3.7 s, respectively. Note that it takes from 0.4 to 1.2 s to obtain a viewable image using LASCA with $M = 3$ or 5. LASCA using $M = 7$ or 9 cannot be used because the small vessels cannot be seen (Fig. 11.7c, d, g, h). Compared to the processing times reported by Briers (2001), both groups Briers et al. at Kingston University and Boas et al. at Harvard University can (capture and) process a 640×480 pixel frame using a 5×5 window in 1 s. Had the same software and machine been used in the simulations, the processing times for LASCA would have been comparable (of the order of 1 s).

11.3.6.2 sLASCA, tLASCA, and mLSI with Varying *M* and *n*

Figure 11.10 shows the processing times in seconds of sLASCA using $M = 3$, 5, and 7, mLSI, and tLASCA with the number of frames from $n = 2$ to $n = 30$.

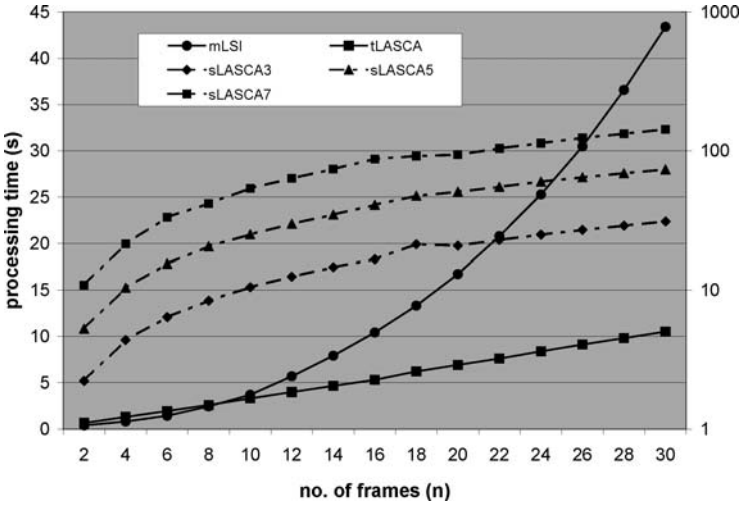

Fig. 11.10 Processing times of mLSI and tLASCA are plotted on a linear scale on the left, whereas sLASCA ($M = 3, 5, 7$) is plotted on a logarithmic scale on the right. tLASCA converges ($n = 10$, $t = 3.31$ s) faster than mLSI ($n = 10$, $t = 3.70$ s) and sLASCA ($M = 5$, $n = 4$, $t = 10.40$ s). Average frame processing time of tLASCA ($t = 0.34$ s) is also faster than that of mLSI ($t = 0.91$ s) and sLASCA ($M = 5$, $t = 2.51$ s)

sLASCA using $M = 3, 5$, and 7 are plotted on a logarithmic scale (on the right), while mLSI and tLASCA are plotted on a linear scale (on the left).

In Fig. 11.10, the processing times of sLASCA using $M = 3, 5$, and 7 increase relatively linearly with the number of frames required for averaging. To implement sLASCA using optimized Matlab *colfilt* operations, the average frame processing times are 1.1, 2.5, and 5.0 s for $M = 3, 5$, and 7, respectively. For sLASCA to achieve stable statistics, eight frames are generally required, which correspond to 8.4, 20.6, and 41.7 s using $M = 3, 5$, and 7, respectively. As discussed in Sects. 11.3.3 and 11.3.5.1, it is best to use sLASCA with $M = 5$. In particular, for $M = 5$, it takes 20.6 s to get the first statically stabled contrast values for the first frame and 2.5 s for subsequent frames.

The average processing times for mLSI and tLASCA are 0.9 and 0.3 s, respectively. For mLSI to achieve stable statistics, up to 16 frames or 10.4 s are required to generate N_{mLSI}. On the other hand, for tLASCA to achieve stable K values, up to 16 frames are also required to compute K_{tLASCA}, but the processing time is 5.3 s. In particular, if tLASCA technique is used, it takes 5.3 s to get the first statically stabled contrast values for the first frame and 0.3 s for subsequent frames. Once stability is achieved, tLASCA is 3.7–16.7 times faster than sLASCA and 3 times faster than mLSI.

11.4 Conclusions

We have reported a temporal-based technique, tLASCA, which processes statistics primarily in the temporal direction using the LASCA equation, proposed by Briers and Webster. We also thoroughly compared the proposed technique with the existing ones including the spatial-based sLASCA and the temporal-based mLSI techniques. Rat cortex has been used under baseline conditions and illuminated using white light. A window over a large arteriole at (L) and a smaller arteriole at (S) with respect to a reference point (R) is used to monitor K or N values and the derived velocities as a result. The contrast values K_{LASCA} are consistent when $M = 5$ or 7. K_{sLASCA} at L and S are within 8% difference when $n = 4$ and 10, respectively. Under both LASCA and sLASCA, K is not sufficiently probable when $M = 3$. On the other hand, K_{tLASCA} at L and S are within reasonable range compared with their K_{LASCA} and K_{sLASCA} when $M = 5$ or 7.

When the observation window size $O = 3$, K_{tLASCA} and N_{mLSI} also correlate to each other very well frame-by-frame even though they are computed differently using temporal statistics. K_{tLASCA} has been proven to provide similar flow rate, using both Lorentzian and Gaussian models, at the same size arteriole as that obtained by an invasive surgically method (Mesenteric Arterial Branches Measurement in the Rat).

From the subjective and objective image evaluation viewpoints, tLASCA also performs better than sLASCA and mLSI. In fact, for gray-scale and color-mapped images, tLASCA ($n = 10$ or 16), followed by mLSI ($n = 10$ or 16) and sLASCA ($M = 5$) techniques produce the best viewing experiences. tLASCA has proven to provide better views in both grayscale and color-mapped contrast images.

In terms of processing times, it took from 0.4 to 1.2 s to obtain a viewable image using LASCA. It took about 8.4, 20.6, and 41.7 s to generate stable contrast images using sLASCA with $M = 3$, 5, and 7, respectively. For mLSI, it took 10.4 s to obtain stable statistics for the first image and 0.9 s for subsequent images. On the other hand, using tLASCA, contrast images can be generated after 5.3 s for the first image and 0.3 s for subsequent images.

The computation of speckle contrast and flow rate has also been updated with both Lorentzian and Gaussian models. Using tLASCA, the minimally invasive and optically derived flow rates (370–490 μL/ min using Lorentzian and 464–614 μL/ min using Gaussian model) are found to be in good agreement with the invasively measured flow rate (218–770 μL/ min) at similar-sized arteriole (270 μm in diameter).

In summary, the tLASCA technique provides more consistent contrast values, more accurate flow rates, better contrast images in both subjective and objective senses, and is the fastest technique in its class. LSI technique for real-time monitoring of blood flows and vascular perfusion, with proper experimental setups and quantitative analyses, may lay new bricks for research in diagnostic radiology and oncology.

Acknowledgments This work is supported by the Faculty Research Committee grant (R-263-000-405-112 and R-263-000-405-133), Faculty of Engineering, National University of Singapore.

References

Aizu Y, Ogino K, Sugita T, Yamamoto T, Takai N, Asakura T (1992) Evaluation of blood flow at ocular fundus by using laser. Appl Opt 31(16):3020–3029

Briers JD (2001) Laser Doppler, speckle, and related techniques for blood perfusion mapping and imaging. Physiol Meas 22:35–66

Briers, JD Fercher AF (1982) Laser speckle technique for the visualization of retinal blood flow. Proc SPIE 369:22–28

Briers, JD Webster S (1995) Quasi-real time digital version of single-exposure speckle photography for full field monitoring of velocity or flow fields. Opt Commun 116:36–42

Briers, JD Webster S (1996) Laser speckle contrast analysis (LASCA): a non-scanning, full-field technique for monitoring capillary blood flow. J Biomed Opt 1(2):174–179

Bonner, R Nossal R (1981) Model for laser Doppler measurements of blood flow in tissue. Appl Opt 20(12):2097–2107

Calamante F, Thomas DL, Pell GS, Wiersma J, Turner R (1999) Measuring cerebral blood flow using magnetic resonance imaging techniques. J Cereb Blood Flow Metab 19:701–735

Cheng H, Luo Q, Zeng S, Chen S, Cen J, Gong H (2003) Modified laser speckle imaging method with improved spatial resolution. J Biomed Optics 8(3):559–564

Dirnagl U, Kaplan B, Jacewicz M, Pulsinelli W (1989) Continuous measurement of cerebral cortical blood flow by laser-Doppler flowmetry in a rat stroke model. J Cereb Blood Flow Metab 9:589–596

Duncan, D Kirkpatrick S (2008) Can laser speckle flowmetry be made a quantitative tool? J Opt Soc Am A 25(8):2088–2094

Dunn AK, Bolay H, Moskowitz MA, Boas DA (2001) Dynamic imaging of cerebral blood flow using laser speckle. J Cereb Blood Flow Metab 21:195–201. http://www.nmr.mgh.harvard.edu/∼adunn/speckle/software/speckle_software.html

Fercher, AF Briers JD (1981) Flow visualization by means of single exposure speckle photography. Opt Commun 37:326–329

Heiss WD, Graf R, Weinhard K, Lottgen J, Saito R, Fujita T, Rosner G, Wagner R (1994) Dynamic penumbra demonstrated by sequential multitracer PET after middle cerebral artery occlusion in cats. J Cereb Blood Flow Metab 14:892–902

Jain AK (1989) Fundamentals of digital image processing. Prentice Hall, Engelwood Cliffs, NJ

Jakeman, E Ridley KD (2006) Modeling fluctuations in scattered waves. Taylor & Francis, Bora Raton, FL

Ohtsubo, J Asakura T (1976) Velocity measurement of a diffuse object by using time-varying speckles. Opt Quant Electron 8:523–529

Ruth B (1994) Measuring the steady-state value and the dynamics of the skin blood flow using the non-contact laser speckle method. Med Eng Phys 16:105–111

Webster S (1995) Time-integrated speckle for the full-field visualization of motion, with particular reference to capillary blood flow. PhD Thesis, Kingston University, Kingston upon Thames, UK

Webster, S Briers JD (1994) Time-integrated speckle for the examination of movement in biological systems. Proc SPIE 2132:444–452

Weisstein EW (2008) Lorentzian function. From MathWorld – a Wolfram Web resource. http://mathworld.wolfram.com/LorentzianFunction.html

Thinh M. Le (S'98–M'00–SM'07) received the B.A.Sc degree in Computer Engineering in 1993, and M.A.Sc., and Ph.D. degrees in Electrical Engineering in 1995 and 2000, respectively, all from the University of Ottawa, Ontario, Canada. Prior to successfully completing his Ph.D. research in video compression and the associated high performance architectures, he joined Lumic Electronics as Technology Founder and Senior Systems Architect. This start-up company designed and marketed a successful commercial multimedia processor, rated as one of the world's top 100 products in 2002 by EDN magazine.

He is an Assistant Professor at the Department of Electrical and Computer Engineering, National University of Singapore. He is currently an Associate Editor to the IEEE Transactions on Biomedical Circuits and Systems, member of Technical Program Committee (TPC) of the IEEE RSP'2008, 2009; IEEE ICCS'08; member of the Technical Program Committee, International Workshop on System-on-Chip for Real-Time Applications, Banff, Canada, 2002; member of the Canadian Advisory Committee, Standards Council of Canada (CAC/JTC1/ SC29 – Coding of audio, picture, multimedia, and hypermedia information), from 2000 to 2006; senior member of the IEEE since 2007; and member of IEEE Engineering in Medicine and Biology Society and Circuits and Systems Society.

Dr. Le's research areas range from: Biomedical imaging including algorithm development for blood flow monitoring and estimation, and on-chip medical visual signal processing image sensor; SoC-based computing/communication algorithms and VLSI architectures including performance-complexity analyses, and power-aware IP block designs; and Human-centric embedded systems.

Assoc. Prof. Sim-Heng Ong is the Head of the Biomedical Engineering Group in the Department of Electrical and Computer Engineering, National University of Singapore. He also holds a joint appointment in the Division of Bioengineering at the same institution. He received his B.E. (Hons.) from the University of Western Australia and his Ph.D. from the University of Sydney. His primary research areas are computer vision and medical image processing. He collaborates extensively with clinicians from the National University Hospital, Singapore, in the analysis and visualization of a variety of medical images with the aim of developing software tools that will assist medical doctors in diagnosis, treatment planning and patient monitoring. He has published extensively in archival journals and international conference.

Joseph S. Paul received his Bachelors degree in Electrical engineering from the University of Calicut in 1986, following which he worked for the R&D Labs of the Indian Telephone Industries, Bangalore, India. He received his Masters and PhD degrees in Electrical Engineering from the Indian Institute of Technology in 1999. Subsequently, he was a postdoctoral fellow at Purdue school of Engineering and the Johns Hopkins School of medicine, Baltimore.

At Johns Hopkins, he underwent training in the area of neural signal processing and deep brain recordings under the supervision of Professor Nitish Thakor. From 2002 to 2004, he worked as assistant professor jointly in the departments of ECE and

Bioengineering at the National University of Singapore, where he set up an in vivo optical imaging laboratory and was a principal investigator in the NUS Life sciences research program. His current interests are in the area of signal processing of neural data and functional brain mapping. From 2004 to 2005, he received a fellowship from the National society for epilepsy for research in the area of functional MRI at the department of clinical epilepsy, University College London. Currently, he is an associate scientist with the department of Biomedical Engineering at UC Irvine, where he is developing multidimensional texture-based algorithms for the analysis of high-resolution MRI images.

Chapter 12
The Challenges in Blood Proteomic Biomarker Discovery

Guangxu Jin, Xiaobo Zhou, Honghui Wang, and Stephen T.C. Wong

Abstract Although discovering proteomic biomarker by using mass spectrometry technique is promising, its rate of introducing proteomic biomarker approved by the US Food and Drug Administration is falling every year and nearly 1 per year on an average since 1998. Apparently, there is a big gap between biomarker discovery and biomarker validation. Here, we reviewed the challenges appearing in the three key stages for the pipeline of proteomic biomarker, that is, blood sample preparation, bioinformatics algorithms for biomarker candidate discovery, and validation and clinical application of proteomic biomarkers. To analyze and explain the reasons for the gap between biomarker discovery and validation, we covered areas ranging from the techniques/methods used in biomarker discovery and their related biological backgrounds to the existing problems in these techniques/methods.

12.1 Introduction

With proteomic technologies, it is possible these days to realize a systematic interrogation of complex proteomics and the identification of differentially expressed proteins (proteomics biomarkers) in blood. The components in blood provide the indication of disease status, including various cellular elements such as tumor cells, cell-free DNA and RNA, proteins, peptides, and metabolites. Mass spectrometry was first used as a tool to identify and characterize isolated proteins and to profile the mass of components in clinical samples, such as surface enhanced laser desorption/ionization time-of-flight mass spectrometry (SELDI-TOF-MS) and matrix-assisted laser desorption/ionization time-of-flight mass spectrometry (MALDI-TOF-MS) (Anderson and Anderson 2002; Anderson et al. 2004).

A biomarker is a measurable indicator of a specific biological state, particularly of the one relevant to the risk of contraction, presence or the stage of disease

X. Zhou (✉)
6565 Fannin Street, B5–014, Houston, TX 77030–2707, USA
e-mail: XZhou@tmhs.org

T. Pham (ed.), *Computational Biology: Issues and Applications in Oncology*,
Applied Bioinformatics and Biostatistics in Cancer Research,
DOI 10.1007/978-1-4419-0811-7_12,

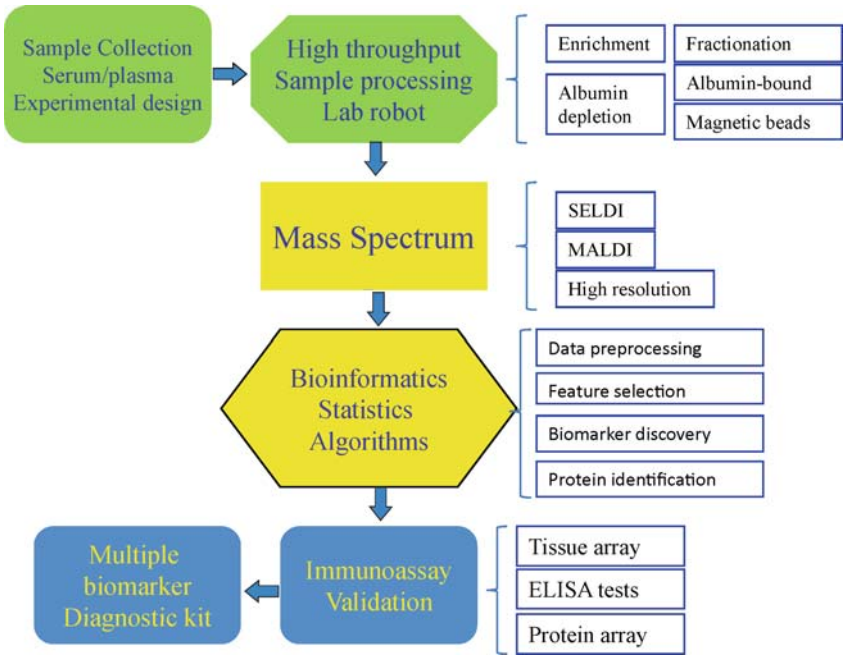

Fig. 12.1 The PepLine for proteomics biomarker discovery from blood

(Powell 2003; Joos and Bachmann 2005; Rifai et al. 2006; Hanash et al. 2008; Sawyers 2008). Proteomic biomarker field aims at developing simple noninvasive tests that indicate disease risk, allows early detection, regression, and recurrence, to monitor disease progression and to classify patients so that they can receive the most appropriate therapy (Rifai et al. 2006; Cox and Mann 2007; McGuire et al. 2008; Simpson et al. 2008). Discovering proteomics biomarkers from blood samples can be roughly divided into three stages: sample preparations for mass spectrometry, bioinformatics algorithms for biomarker discovery, and validation and clinical applications (Fig. 12.1). Despite a large interest and investment in this area, only a few new proteomics biomarkers are successfully used in clinical application. According to the report of the US Food and Drug Administration (FDA),the proteomics biomarker rate of introduction is falling every year and the rate of introduction of new protein analytes approved by FDA has fallen to 1 per year on an average since 1998 (Rifai et al. 2006). The reasons for this disjunction are due, in part, to the lack of a coherent pipeline connecting biomarker discovery process with well-established methods for clinical validation (Anderson and Anderson 2002; Rifai et al. 2006).

We review here some of the challenges that occur in the key stages in biomarker discovery from proteomics:

- Blood sample preparation
- Bioinformatics algorithms for biomarker candidate discovery
- Validation and clinical application

We will cover areas ranging from the techniques/methods used in biomarker discovery to the existing problems in these techniques/methods and aim to analyze and explain the reasons for the gap between biomarker discovery and validation.

12.2 Blood Samples Preparation for Biomarker Discovery

Discovering proteomics biomarker from blood samples by using SELDI-TOF-MS or MALDI-TOF-MS technologies starts with sample preparation. Proteomic studies to find new biomarker protein candidates that may be useful for further studies, for the development of diagnostic tests or even therapeutic targets, have to be planned very critically from the beginning to get accurate results [10]. The strategies used for blood samples preparation usually include deletion of high-abundance proteins from samples and fractionation of plasma proteins, digestion of protein into peptides, and overcoming other biological factors.

12.2.1 Dynamical Range of Proteins

The dynamic range of proteins in blood sample limits our capacity to directly interrogate the blood proteome for the purpose of biomarker discovery. In a typical blood sample, the protein abundance ranges from 40 mg/mL (albumin) to 5 pg/mL (cytokines), and the proteins, such as albumin, haptoglobin, IgA, IgM, α-1-antitrypsin, fibrinogen, α-2-macroglobulin, C3 Complement, transferrin, and IgG, account for about 90% of blood protein content [1, 10]. Thus, methods used for depletion of high-abundance proteins and fractionation of proteins from plasma have been proposed (Hoffmann et al. 2001; Miklos and Maleszka 2001; Thomas et al. 2002; Ahmed et al. 2003; Tam et al. 2004; Whelan et al. 2004; Cho et al. 2005; Tang et al. 2005; Wang et al. 2005; Lee et al. 2006; Shin et al. 2006). Two such methods: resin-based and antibody-based depletion, that is, multiple affinity removal system (MARS) [12, 13], and IgY-microbeads Kit [14] are mainly used these days. MARS provides high binding efficiency and specific, reproducible removal of the six most abundant plasma proteins (albumin, transferrin, IgG, IgA, haptoglobin, and antitrypsin) and their subtypes. IgY microbeads contain IgY antibodies against 12 most abundant plasma proteins [albumin, IgG, transferrin, a1antitrypsin, IgA, IgM, a2-macroglobulin, haptoglobin, HDL components (ApoAI and AII), a1-acid glycoprotein, and fibrinogen]. Four major types of fractionation methods have also been proposed for separate diverse abundant proteins from plasma. One such method, Gradiflow, is a type of 2D liquid enrichment system that uses membrane-based electrophoresis to fractionate protein samples through an uncharged membrane [15, 16]. The proteins are separated into 4–5 fractions based on pI ($<$5.25 vs.$>$5.25) and molecular size (e.g., $>$125 kDa, albumin-enriched region, $>$45 kDa and $<$ 45 kDa).

Other methods for protein fractionation, such as ProteomeLab[TM] PF2D [17], multichannel electrolyte (MCE) [18], microscale solution IEF (ZOOM) [19], and free flow electrophoresis (FFE) [20], are also in use.

While some methods can delete the high abundant proteins from plasma or separate diverse abundant proteins from each other, there are still challenges involved in the discovery of biomarkers from such samples derived by these methods. Although some depletion methods can reduce high abundant proteins ranging from 96% to 99%, the presence of albumin can still be $\sim$50 μg/mL, about 10^4-fold higher than blood CEA levels ($\sim$5 ng/mL) and 5×10^6-fold higher than blood IL-6 levels ($\sim$10 pg/mL) [10]. Another problem in depletion of high abundant proteins is that it is risky, as these proteins may be the carriers for low-abundance molecules. If highly abundant proteins interact with low-abundance proteins, such a depletion method is not efficient in detection of low-abundance proteins from plasma [10]. For example, albumin removal has been suggested to lead to the decrease of physiologically important proteins such as cytokines (Geho et al. 2006).

12.2.2 The Blood "Peptidome"

Unlike oligonucleotides in message RNAs, proteins cannot be amplified and therefore sensitivity is a major concern. To make proteins suitable to MS analysis, they first need to be converted into peptides. MALDI-TOF-MS and SELDI-TOF-MS can provide the surface upon which ionization can take place to provide a degree of fractionation due to variable absorbance of peptides. Such techniques are now being widely applied to analyze the peptides and proteins in relatively complex samples (Diamandis 2003; Yewdell 2003; Diamandis 2004; Geho et al. 2006; Hortin 2006; Petricoin et al. 2006; Davis and Patterson 2007). Based on the peptidome in plasma, it is now recognized that this region of the proteome can be expected to contain shed proteins and protein fragments emanating from physiologic and pathologic events taking place in all perfused tissues. Large proteins, unable to enter the circulation passively, on account of their size, could be represented as fragments in the low molecular weight region (LMW) of the blood proteome. For this reason, recent works have used mass spectrometry to interrogate this LMW region for disease-related information [22]. This method was first applied to serum from patients with ovarian cancer (Petricoin et al. 2002b) and then later to other cancers (Adam et al. 2002; Li et al. 2002; Petricoin et al. 2002b; Qu et al. 2002; Rosty et al. 2002; Hingorani et al. 2003; Poon et al. 2003; Vlahou et al. 2003; Won et al. 2003; Zhukov et al. 2003; Ebert et al. 2004; Villanueva et al. 2004; Brouwers et al. 2005).

But opposite opinions were also proposed. The SELDI-TOF technology that is currently used for serum analysis is not capable of detecting any serum component at concentrations of less than μg/mL. This range of concentrations is approximately 1,000-fold higher than the concentrations of known tumor markers in the circulation [23, 24]. The serum discriminatory peaks identified by mass spectrometry very

likely represent high-abundance molecules that were unlikely to have been released into the circulation by very small tumors or their microenvironments. Biomarker discovery studies of this nature have drawn cautionary notes owing to the problems in experimental design and data analysis or biases related to blood collection, processing, and/or storage protocols (Diamandis 2004; Ransohoff 2005).

12.2.3 Other Biological Factors

The quality and accuracy of discovery of biomarker candidates from blood sample is not only determined by technical variance but also affected by biological variances [10], such as [3] lack of standardized sample collection and storage, variably affecting comparison groups, differences between cases and controls in terms of sex, age, and physiological states (e.g., fasting, weight gain or loss, and hormonal status), differences in genetic make-up, changes in inflammation and acute-phase reactants, changes in metabolic states, other nonspecific changes, for example, cell death and tissue necrosis, and changes reflecting underlying chronic disease, for example, those caused by smoking and chronic lung disease, in contrast to lung cancer-specific changes.

12.3 Bioinformatics Algorithms in Biomarker Discovery

Unsuccessful application of proteomics biomarkers in clinical diagnosis indicates something amiss in the pipeline of biomarker discovery. Although sample preparation and diagnosis validation should be somehow responsible for the problem, bioinformatics algorithms used in the data preprocessing and biomarker discovery cannot be overruled due to their lack of consistency and reproduction.

Generally, the middle stage of the pipeline of biomarker discovery is composed of bioinformatics algorithms including baseline removal, normalization, denoising/smoothing, peak detection, peak alignment, feature selection, classification for biomarker candidates, and protein/peptide identification (Fig. 12.2 and Table 12.1). Because of the many different ways available to perform each step in the process of biomarker discovery, it often results in diverse outcomes from the use of different combinations of the algorithms. It is therefore natural to get less consistent and reproducible biomarker candidates [23]. Undoubtedly, the biomarker candidate identification in the middle stage of the pipeline needs a more robust strategy to result in a more standard biomarker discovery from proteomics data. We reviewed the main bioinformatics algorithms/methods used for data preprocessing and biomarker discovery. For every step in this stage, mathematical methods have been proposed and experimental backgrounds and the major challenges involved have been discussed (Table 12.1).

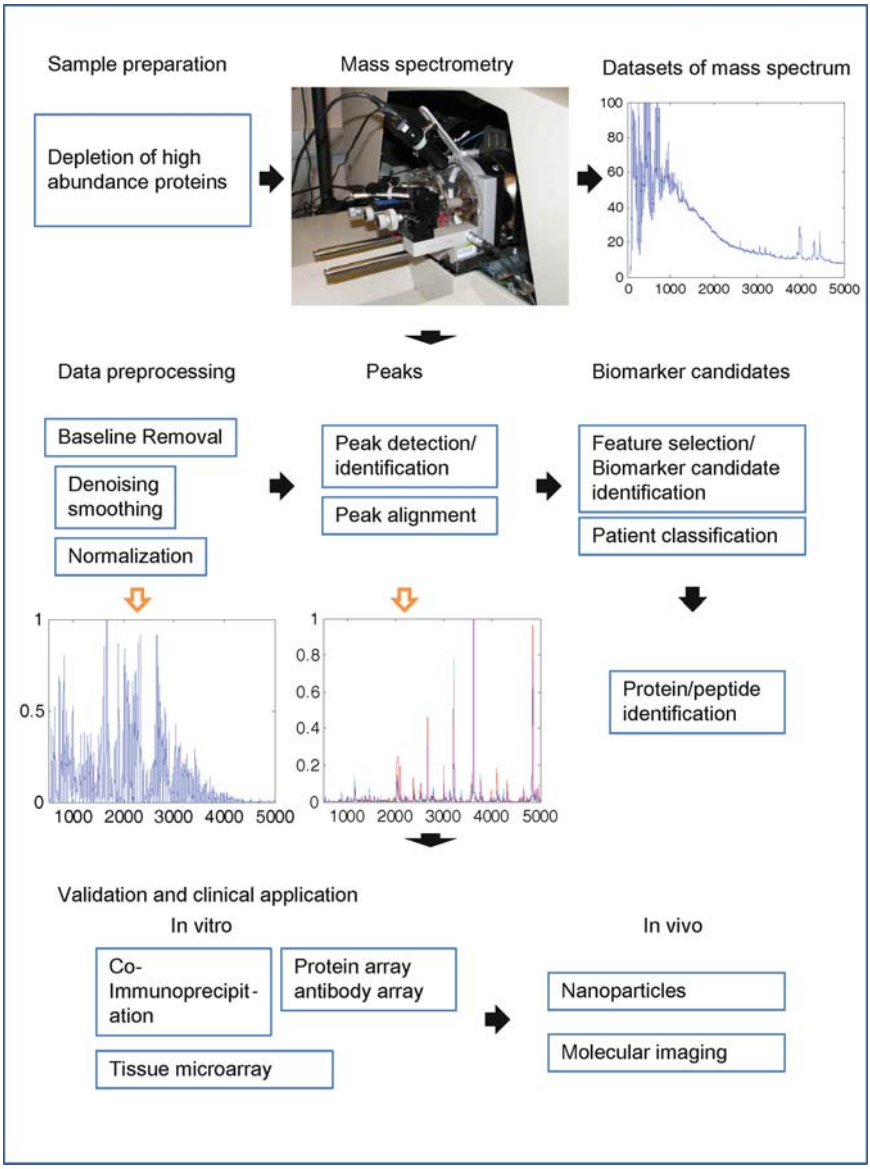

Fig. 12.2 The flowchart for proteomics biomarker discovery

12.3.1 Baseline Removal

Experimentally, the baseline in SELDI or MALDI is caused by a cloud of matrix molecules hitting the detector shortly after ionization (Malyarenko et al. 2005). As the first preprocessing step of SELDI or MALDI data, this critically influences

Table 12.1 The bioinformatics methods used in candidate proteomics biomarker discovery

The key steps in preprocessing	Experimental backgrounds	Algorithms in bioinformatics	Challenges
Baseline removal	Caused by the matrix molecules hitting the detector	Static or moving window sampling [45]	Whether the estimated signal is closer enough to true signal
		Wavelet [48]	Whether estimated baseline is closer enough to true baseline
		Manual method [49]	
Denoising/ smoothing	Chemical and random noises	Discrete wavelet transform (DWT) [51]	Choosing different wavelets Loss of peaks Complexity and time-consuming
		Matched filtration [53] Savitzky–Golay	Missing of low-abundant components of the sample Leave data noises in denoised signals
		Average moving [57]	Leave data noises in denoised signals Peak overlapping
Normalization	Identify and remove sources of systematic variation between spectrums	Dividing by a constant AUC [61] or unit length [64]	Structure noise
		Regression [65, 66] Quantile [64]	Seldom taking systematic bias into normalization process
Peak detection/ identification	Discerning true signals from noises	Window or binning methods [47, 77, 78]	Lack of interpretability Multiple charge states
		Thresholds for the signal/noise ratio [47, 72, 74, 79]	Mass-dependent sensitivity Chemical adducts and fragmentation
		Gaussian shape for LC-MS [80, 81]	Reproducibility of mass spectrums
		Decomposing overlap peaks [51, 78, 81–84]	Ion suppression effects Calibration
Peak alignment	The variation of corresponding peaks among the mass spectrums	Optimized warping [85–87]	Slight variation for the exact m/z and retention time of a peak
		Vectorized peaks [88]	The drifts for all detected peaks may be equal
		(Semi)supervised alignment using nonlinear regression methods [89]	One peptide may correspond to several peaks
		Hidden Markov models [90]	The peptides with similar m/z ratio may hard to be discerned from each other
		Statistical alignment [91]	Peaks only be found in few mass spectrum
		Clustering [92–94]	

(continued)

Table 12.1 (continued)

The key steps in preprocessing	Experimental backgrouds	Algorithms in bioinformatics	Challenges	
Biomarker candidates identification/feature selection	Biomarker candidate identification	Filter methods: t test [95, 96], F test [97], peak probability Contrast [98], Kolmogorov–Smirnov test [99], Correlation-based feature selection (CFS) [100]	Ignoring the feature dependence Ignoring the interaction with the classifier, i.e., the search in the feature subset space is separated from the search in the hypothesis space	It is a hard problem due to examining all possible models would require evaluating $2^p - 1$ models, where p is the number of variables included in the study
		Wrapper methods: genetic algorithms [101, 102], and nature inspired [61, 103]	Higher risk of overfitting than filter techniques and very computationally intensive	
		Embedded methods: random forest/ decision trees [96, 104], the weight vector of SVM [74, 105], and neural network [106]	Classifier dependence	
Clinical diagnosis/ classification	Biomarker discovery is aimed at finding a set of discriminatory proteins to diagnose different states with respect to a given disease	Generative approaches: LDA [107], QDA [107], Kernel density estimation [108], and K-nearest neighbor [96] Discriminative approaches: Logistic regression (LR) [109], neural network (NN) [110], support vector machine (SVM) [107], decision tree (DT) [109]	Hard to choose the learning approach for a given classification problem Different classification algorithms have their specific biases, but it is hard to identify the problem structure a priori	
Protein/peptide identification	Identify their corresponding protein from the characteristics of peptides digested enzymically from a protein	Database searching [111–114] De novo sequencing [115–122] Sequence tagging [123–125] Consensus of multiple engines [126]	Limited PTM Some methods used in some softwares are not published Contamination of the sample, imperfect fragmentation, simultaneous fragmentation of two different peptides, and low signal-to-noise ratio	

subsequent analysis steps. Current baseline removal algorithms of SELDI or MALDI data, which are based on mathematical morphology, result in biased signal estimates. Because of the parameterization of current algorithms for baseline removal, noise and spectral signal distributions bias the removal results, which may lead to seemingly interesting but ultimately irreproducible results in downstream analysis.

The process of baseline removal deals with the identification of the baseline and its removal from the mass spectrum. The baseline can be seen in the spectrum of a blank (zero protein) sample as a smooth and downward drifting curve moving from low m/z to high m/z. The baseline has the following three characteristics (Baggerly et al. 2003; Wang et al. 2003; Hilario et al. 2006): First, the amplitude of the baseline may be much larger than the signal so that the fold change estimate will be downward biased if we ignore the baseline. Second, the baseline is not flat in a spectrum, and therefore the bias will be heterogeneous across the spectrum. Third, the baseline varies from spectrum to spectrum, even between replicate sample runs, hence creating an unwanted source of variation.

Some algorithms used in baseline removal are based on mathematical morphology implementations and are also used in denoising/smoothing. First, either a static or a moving window sampling of the data is introduced. An assumption for effectively applying mathematical morphology is that most of the data points in a window (also called a structuring element) of the spectrum are nonpeaks. In this manner, the baselines are chosen as the intensities of the spectrum in low percentiles within this window (Baggerly et al. 2003). Next, in the algorithms for denoising/smoothing, such as moving average (MA), Savitzky–Golay (S-G) smoothing, Fourier filtering, and Wavelets, the Wavelets are used to baseline removal. The details about the Wavelets will be discussed in the next section. This method can reduce the data noises and remove the baseline of the data simultaneously (Perrin et al. 2001).

One challenge that occurs in baseline removal is automation of the algorithm, as the peak width of desired signal varies both within and across spectra. Jirasek et al. argued that the manual method guided by visual inspection should be still useful and important for baseline removal (Jirasek et al. 2004). Another problem for baseline removal is to determine the cutoff of the baseline. If the cutoff is too small, it underestimates the true baseline and leaks part of the noise into the signal estimation. Therefore, we focus on the following problems (Coombes 2005; Coombes et al. 2005; Tan et al. 2006) in the process of baseline removal. First, after baseline removal, the estimated signal should be closer to the true signal. Second, in the simulated data, estimated baseline should be closer to the true baseline. Third, after baseline removal, the estimated fold change should be closer to the true fold change. Last, the consistency of technical replicates should be improved after baseline removal.

12.3.2 Denoising/Smoothing

There are two types of noise in derived mass spectrums, that is, chemical and random. A mass spectrum is not only composed of peaks corresponding to the sample components, but also consists of noise from limited ion statistics, instabilities of ion source, thermal noise, and spikes (Andreev et al. 2003). Especially, the chemical noise comes from the MALDI matrix or mobile phase (ESI-MS). The data noises have a great effect on the accuracy of m/z values in the peak list in two ways. The first is MS peaks, representing components of solvent (ESI) or matrix (MALDI) or their contaminants, or some intense spikes, which could be mistaken for sample ions. The second is when the ratio of signal and noise of the sample peak is low, the centroid of the peak can be shifted, resulting in an inaccurate m/z value.

To detect the peaks in mass spectrum accurately, we must try to discern data noise from useful signals and remove it from the mass spectrum (Perrin et al. 2001; Gobom et al. 2002; Andreev et al. 2003; Baggerly et al. 2003; Wang et al. 2003; Jirasek et al. 2004; Rejtar et al. 2004; Bensmail et al. 2005; Coombes 2005; Coombes et al. 2005; Malyarenko et al. 2005; Hilario et al. 2006; Stolt et al. 2006; Tan et al. 2006). There are four main methods available to achieve this goal: discrete wavelet transform (DWT), matched Filtration using Fourier transform, Savitzky–Golay, and average moving.

12.3.2.1 Discrete Wavelet Transform

In the process of DWT (Coombes et al. 2005), an original signal $f(t)$ is decomposed into a set of basis functions called wavelets ψ. The vector of original signal $f(t)$ can be decomposed into an orthogonal matrix of wavelet functions W; thus, a vector of DWT coefficients w is achieved. By introducing the thresholds for DWT, the data noises can be removed. Soft threshold and hard threshold are two main approaches for the thresholding process. The soft threshold refines the coefficients that are less than threshold, and are set to zero, otherwise reduced by threshold to sign $(w_{ij})(|w_{ij}|-threshold)$. The hard threshold refines the coefficients that are less than threshold are set to zero, and are otherwise reduced by w_{ij}.

There are three problems in the DWT. The first is how to choose different wavelets to meet different requirements from various signal conditions. The second is the denoising efficiency that can be evaluated in statistics by computing the root mean square error (RMSE) between the denoised signal and the ideal one. The issue is that the derived wavelets by this optimized method may be more suitable for reducing the baseline, but it does not preserve the peaks in the mass spectrums. The last drawback is that it is complex and time-consuming.

12.3.2.2 Matched Filtration

The shape of the signal and the characteristics of the noise are two important factors for data denoising. If the input $X(t)$ can be represented as a sum of the signal of the

known shape, described by a function $S(t)$ and a random function $N(t)$ representing noise, then the maximum S/N can be derived when the input data are processed by a matched filter having the transfer function $H(f)$

$$H(f) = S^*(f)/P_{\mathrm{NN}}(f), \tag{12.1}$$

where

$$S^*(f) = \int_{-\infty}^{\infty} S(t) \exp(j2\pi f t)\, \mathrm{d}t$$

is the complex conjugate of the Fourier transform of the signal $S(t)$,

$$P_{\mathrm{NN}}(f) = \int_{-\infty}^{\infty} R_{\mathrm{NN}}(t) \exp(-j2\pi f t)\, \mathrm{d}t$$

is the power density spectrum of noise, R_{NN} is the autocorrelation function of the noise, t is time, and f is the frequency (Andreev et al. 2003).

In the matched filtration, the Gaussian function was assumed to characterize the peak shape. Thus, Gaussian function and second derivative of the Gaussian can be applied in this method. But the cross-correlation with the later can produce negative artifact peaks on both sides of the real MS peaks, which results in peak shape distortion and decreased mass accuracy (Andreev et al. 2003).

This method tries to find those peaks with high S/N ratio value and can improve the false positives, but it can increase the likelihood of false negatives, that is, missing of low abundant components of the sample.

12.3.2.3 Savitzky–Golay and Average Moving

The Savitzly–Golay smoothing filter and average moving are used simultaneously. The Savitzky–Golay method essentially performs a local polynomial regression (of degree k) on a distribution (of at least $k+1$ equally spaced points) to determine the smoothed value for each point. The main advantage of this approach is that it tends to preserve features of the distribution such as relative maxima, minima, and width, which are usually "flattened" by other adjacent averaging techniques (like moving averages, for example) (Gobom et al. 2002).

Savitzky–Golay and average moving, and matched filtration, comparing with DWT, still leave some data noises in the denoised signals. Band broadening also occurs, in which the second peak and the third peak were overlapped more seriously.

12.3.3 Normalization

The purpose of interspectrum normalization is to identify and remove sources of systematic variation between spectrum due to varying amounts of protein or

degradation over time in the sample or even variation in the instrument detector sensitivity (Marcuson et al. 1982; Rietjens et al. 1995; Steeves et al. 2000; Bolstad et al. 2003; Park et al. 2003; Alfassi 2004; Ressom et al. 2005; Callister et al. 2006; Wang et al. 2006; Arneberg et al. 2007). The general normalization methods to consider are the global normalization techniques. Such methods assume that the sample intensities are all related by a constant factor. A common choice for this rescaling coefficient is the spectrum median or the mean. The assumption behind the global normalization is that on an average, the number of proteins that are overexpressed is approximately equal to the number of proteins that are underexpressed, and the number of proteins whose expression level changes is few, relative to the total number of proteins (Fung and Enderwick 2002).

The mass spectrum data can be normalized by several ways, such as dividing a constant value, regression, and quantile (a nonparametric approach).

12.3.3.1 Dividing by a Constant Value

The constant value can be constant sum, unit length, or average AUC (area under the curve or total ion current) (Ressom et al. 2005), for example,

$$\mathbf{z}_i^{\mathrm{T}} = \frac{\mathbf{x}_i}{\sum_{j=1}^{M} x_{ij}} \quad i = 1, 2, \ldots, N \tag{12.2}$$

and

$$\mathbf{z}_i^{\mathrm{T}} = \frac{\mathbf{x}_i}{||\mathbf{x}_i||} \quad i = 1, 2, \ldots, N, \tag{12.3}$$

where $\mathbf{x}_i$ and $\mathbf{z}_i$ represent the profile for sample i before and after normalization, respectively, M is the number of points describing the profile, and N is the number of instrumental profiles acquired.

Equations (12.2) and (12.3) are normalized for constant sum and unit length, respectively (Arneberg et al. 2007).

12.3.3.2 Regression

The normalization technique assumes that systematic bias is linearly or nonlinearly dependent on the magnitude of peptide abundances (Bolstad et al. 2003; Park et al. 2003). In the former situation, linear regression was performed by applying least squares regression to the scatter plots. The resulting first-order regression equation is used to calculate each normalized peptide ratio:

$$m_i' = m_i - m_i^*, \tag{12.4}$$

where m_i^* is the predicted peptide ratio calculated from the regression equation. The derived value of a predicted peptide ratio represents the deviation from the abscissa of the regression line. Another way is to apply local regression and derive a nonlinear regression. It can be done by the LOWESS algorithm packaged to find the predicted value for each peptide ratio (Arneberg et al. 2007).

12.3.3.3 Quantile

This approach is based on the assumption that the distribution of peptide abundance in different samples is expected to be similar and can be accounted for by adjusting these distributions (Arneberg et al. 2007).

There are some challenges in applying these methods in normalization process. First, the structure noise, that is, noise increasing with signal size (heteroscedasticity), represents a major problem in the comparison of mass spectral profiles. The log transform or *n*th root transform is needed here to reduce the structure noise before normalization (Rietjens et al. 1995; Bolstad et al. 2003). Next, ideally, the most appropriate normalization technique is chosen after the causes of systematic bias are identified and characterized according to observed trends across the dynamic range of detection. However, this principle is seldom used due to the challenge in identifying and defining the wide range of possible contributions in both sample processing and analysis of the overall bias.

12.3.4 Peak Detection/Identification

Detecting peaks, that is, the signals from ionized proteins or peptides, from the mass spectrometry data, is in fact not an easy task. This is due to the multiple experimental factors, besides the chemical noises, which affect the signals (Gras et al. 1999; Prados et al. 2004; Listgarten and Emili 2005; Hilario et al. 2006; Shimizu et al. 2006; Albrethsen 2007; Fenselau 2007; America and Cordewener 2008; Ressom et al. 2008). The first experimental factor is the multiple charge states of an ionized protein or peptide, which means that the large peptide ions produced by ESI or MALDI often have different number of elemental charges and especially a broad distribution of charge states for large denaturized protein ions. After computing the mass/charge of the protein, its time points may not be unique. This can result in the difficulty in determining which time point corresponds to the protein and therefore affecting the detection of the peaks for the ionized protein. The second experimental factor is mass-dependent sensitivity of mass spectrum. Because of the fact that all ions of the same charge have the same kinetic energy after acceleration, the ions with heavier mass are slower and produce weaker signals. If reducing some data noise by cutoffs, the signals produced by these proteins might be accidentally removed. The third factor is the chemical adducts and fragmentation. The larger protein contaminated by some chemical adduct ions can produce a much broader

m/z value distribution than the expected pure protein. This results in the difficulty in identifying the centroid of corresponding peak of the protein. Another factor is the reproducibility of mass spectrums. In MALDI, the laser attacks the matrix compound including samples and thereby the signal intensity is affected by the laser power, the amount of sample, and the quality of the crystals. This leads to the diverse absolute intensities of repeated measurements. The fifth factor is the ion suppression effects. The signal intensity of a protein/peptide depends strongly on its chemical composition. Especially if the analyte concentration is beyond a certain threshold, analytes producing intense signals can suppress the signals of other analytes, which are less suitable for ionization. Thus, the signal intensity of certain analytes does not depend linearly on the initial concentration, but is influenced in a complex manner by the concentration of other analytes. This causes more complications in the discovery of biomarkers from multiple mass spectrums. The final hurdle is calibration, in which some of the parameters entering the process of transferring flying time to m/z value are only approximately known and lead to the slight shifts in the calculated masses. The experimental factors may be vital to peak identification from mass spectrum, and therefore these may result in wrongly identified biomarkers (identified by the m/z ratio).

Although the peak detection is an intermediate step in preprocessing the mass spectrum, it is an essential one for peptide identification and biomarker discovery. In fact, the peaks of a protein's peptides are first identified from mass spectrum and then the corresponding m/s ratios and intensities (retention time) were found for peak identification. The identified peptides are the fingerprints of the discovered protein, that is, biomarkers. Therefore, in the process of peak detection, the extent of overcoming the data noise for an algorithm determines the accuracy of the discovered biomarkers.

Most algorithms for peak detection considered the effects of the experimental factors on the mass spectrums and the signals of peaks.

- First, a window or binning in mass/charge axis is adopted in some algorithms (Yasui et al. 2003; Fushiki et al. 2006; Hilario et al. 2006). Because of the data denoising and smoothing of the signals, it is a continuous line whose horizontal line is m/z value and the vertical line is the intensity of the corresponding ion. Thus, detecting a peak is nothing but choosing a part of the signals instead of a single intensity point. Jarman et al. used a statistical test to check if the histogram within a sliding window represents a uniform distribution or has a peak. Another method similar to the window is the binning used to detect peaks, in which binning adds a fixed number of adjacent data points into one combined variable (Arneberg et al. 2007).
- Second, thresholds for signal to noise ratios exclude random peaks (Gras et al. 1999; Prados et al. 2004; Hilario et al. 2006; Karpievitch et al. 2007). Most of the peak detection methods adopt the threshold for signal to noise ratios excluding random peaks. The threshold for the signal to noise ratio can be obtained from statistical analysis of the noise. The distribution of the noisy peak intensities can be estimated for a certain mass window, and all intensities that have a low P-value with respect to this distribution can be considered as real

peaks. Another approach is to link the peak detection threshold directly to the identification or classification process. An alternative way to find the peaks and valleys is by starting with a straight line that connects the first and last point in the spectrum. The algorithm then finds the point in the raw spectrum that is farthest from the line.

- Third, Gaussian shape is approximately used for the peak shape in LC-MS (Li et al. 1997; Dijkstra et al. 2007). A real LC-MS signal consists of a sum of isolated signals plus noise and baseline where the noise has a different elution profile than peptides, which makes it possible to distinguish it from signals. Although the elution profile of a peptide is less well defined and less reproducible than its m/z signal, a Gaussian shape is usually a rather good approximation, but more flexible refinements were proposed.
- Finally, we deal with the overlap peaks (Li et al. 1997; Coombes et al. 2003, 2005; Yasui et al. 2003; Shackman et al. 2004; Zhang et al. 2008).

For different resolution of MALDI instruments, such as LC and FT-ICR, single isotopic peaks are either distinguishable or they melt into broader peaks containing several isotopes. On the other hand, for the high masses or low-resolution mass spectra the isotopic peaks may not be visible and collapse into a single broad peak, the shape of which may be distorted by fragmentation and chemical adducts.

Although the algorithms for peak detection are trying to overcome these hurdles from the experiments of mass spectrometry, they still encounter challenges in biomarker discovery. The first is that the complexity of peak detection caused by the experimental factors is hard to solve using any proposed algorithm so far. The next is lack of interpretability for the discovered biomarkers based on the selected peaks. It is not sure if the peaks really represent the real signal intensities and if the results are caused by the designs of these algorithms or the mistakes in mass spectrums.

12.3.5 Peak Alignment

Although the m/z measurements can be accurately obtained from properly calibrated instruments, the variation of corresponding peaks among the mass spectrums can still be found (Listgarten and Emili 2005). Elution patterns can become distorted (locally compressed and/or expanded in complex, nonlinear ways) by differences in chromatography performance due to changes in ambient pressure and temperature. Even under ideal conditions, MS duty cycles are finite and sampling is not necessarily constant, resulting in spectral capture at different time points across an eluting peak even between repeat analyses. In certain cases, drift may occur along the m/z axis as well, although this is far less of a problem than variations in time.

Some algorithms, such as correlation optimized warping (Bylund et al. 2002; Jaitly et al. 2006; Prince and Marcotte 2006), vectorized peaks (Hastings et al. 2002), (semi-) supervised alignment using nonlinear regression methods (Fischer et al. 2006) and Hidden Markov Models (Listgarten et al. 2007), and statistical alignment (Wang et al. 2007) or clustering (Silva et al. 2005; Lange

et al. 2007; Mueller et al. 2007), strongly benefit from high mass accuracy and stable retention times. Many of recent alignment algorithms (PLGS, SIEVE, VIPER, Pepper, MsInspect, Msight) depend on high-resolution m/z data. Other algorithms (MetAlign, CPM, Crawdad, MsMetrix) used data binning. And there are still other softwares, such as mzmine, metAlign, BinBase, xcms, MarkerLynx, BluFuse, SpecAligh, msInspect, Progenesis PG600, caMassClass, Xalign, msalign from Matlab, RTAlign, MS Align, LCMSWARP, CHromAlign, PETAl, MaekerView, MathDAMP, NameLess, CPM, meta-b, Chenomx Profiler, MS-Xelerator, OBI-Warp, Census, which can be used for peak alignment.

In peak alignment, we may meet with the following problems (America and Cordewener 2008): The exact m/z and retention time of a peak may vary slightly due to technical drift in MS and LC instruments (discussed in peak detection). The drifts for all detected peaks may not equal (caused by the experimental factors discussed in peak detection). Considering that one peptide may correspond to several peaks due to the experimental factors discussed in peak detection, a single peak in some mass spectrum may be detected as multiple peaks in the other mass spectrums. The peptides with similar m/z ratio values may be hard to be discerned from each other. Some peaks may only occur in very few mass spectra due to absence of that particular peptide in the other samples or may not be detected in low intensity. Thus, a perfect detailed alignment of all features seems to be a nonrealistic goal.

12.3.6 Biomarker Candidate Identification

Feature selection technique is usually used to identify biomarker candidates from the found peaks. In essence, feature selection needs to find the optimal model parameters for the feature subsets instead of just optimizing the parameters of the full feature subset. However, the feature selection introduces an additional layer of complexity in the modeling task. Thus, it is not difficult to see that feature selection is a hard problem. This is because examining all possible models would require evaluating $2^p - 1$ models, where p is the number of variables included in the study.

Feature selection is usually combined with classification problem in biomarker discovery. From the classification performance of the chosen subset of relevant features, we can determine whether the features in this subset should be chosen optimally. In the context of classification, feature selection techniques can be divided into three categories depending on the combination of the feature selection search with the construction of the classification model: filter methods, wrapper methods, and embedded methods. Filter methods include univariate filter, such as t test (Liu et al. 2002; Wu et al. 2003), F test (Bhanot et al. 2006), peak probability contrast (Tibshirani et al. 2004), Kolmogorov–Smirnov test (Yu et al. 2005), and multivariate filter, such as correlation-based feature selection (CFS) (Hauskrecht et al. 2005). Filter techniques access the relevance of features by looking only at the intrinsic properties of the data. A common disadvantage of filter methods is that they ignore the interaction with the classifier, that is, the search in the feature subset space is

separated from the search in the hypothesis space. Wrapper methods embed the model hypothesis search within the feature subset search, which includes deterministic methods, such as genetic algorithms (Petricoin and Liotta 2003; Li et al. 2004), and nature inspired methods (Ressom et al. 2005; Ressom et al. 2007). A common drawback of these techniques is that they have a higher risk of overfitting than filter techniques and are very computationally intensive, especially if building the classifier has high computational cost. In embedded methods, the search for an optimal subset of features is built into the classifier construction and can be considered as a search in the combined space of feature subsets and hypotheses. The embedded methods include random forest/decision trees (DTs) (Wu et al. 2003; Geurts et al. 2005), the weight vector of SVM (Prados et al. 2004; Zhang et al. 2006), and neural network (NN) (Ball et al. 2002).

12.3.7 Clinical Diagnosis

Biomarker discovery is aimed at finding a set of discriminatory proteins to diagnose different states with respect to a given disease. In this process, the classification methods have the potential to identify the performance in assigning a biological sample to one of several predefined classes or disease states, for example, in the simplest case, diseased vs. control. The biomarker candidates are those best features with high accuracies in classification process. Undoubtedly, the classification methods play a key role in identification of biomarker candidates. Diverse classification methods with different models and different parameters may result in distinct results in spite of same samples.

The classification algorithms can be divided into two classes, that is, generative approaches and discriminative approaches. Generative models are called so because they express a hypothesis about how the data were generated, such as linear discriminant analysis (LDA) (Wagner et al. 2003), quadratic discriminant analysis (QDA) (Wagner et al. 2003), Kernel density estimation (Lee et al. 2005), and K-nearest neighbor (Wu et al. 2003), in which LDA and QDA assume that the class densities are Gaussian with a set of parameters, and Kernel density estimation and K-nearest neighbor make no prior assumptions and estimate densities in a purely data-driven manner. Discriminative approaches build a direct mapping from inputs to class labels or model posterior class probabilities without modeling the underlying joint probability density, such as logistic regression (LR) (Rai et al. 2002), NN (Rogers et al. 2003), support vector machine (SVM) (Wagner et al. 2003), and DT (Rai et al. 2002), in which LR fails to deal with the data that are not linearly separable, and SVM and NN can classify the data and can be applied to the data that are nonlinearly separable.

There is an overwhelming number of classification algorithms, which can be combined in an exponential number of ways (Hilario et al. 2006). Which learning approach works well for a given classification problem is still an open question and will probably remain so for sometime. Different classification algorithms have their

specific biases, which should match the problem structure, that is, the concept that governs class assignment. Unfortunately the problem structure is not known a priori; in fact, it is precisely what should be discovered. Even in a circumscribed domain such as mass spectrometry, different learning algorithms could be appropriate for seemingly related problems, depending on the concept that underlies the data and how the features interact together to determine the class.

12.3.8 Protein/Peptide Identification

Protein identification is an indispensable task for applying the found candidate biomarkers (multiply charged ions) in drug discovery and clinical application. Eletrospray ionization (ESI) and tandem mass spectrometry (MS-MS) nowadays regularly perform protein/peptide identification. The multiply charged ions provided by ESI must be accounted for in the results, and the peptides enzymically digested from a protein in the MS-MS spectrums are also needed to be recovered. Computational methods are then proposed to identify the multiply charged ions through combination of these peptides. The most recognized peptide identification software packages can be classified into four categories: database searching (Perkins et al. 1999; Ma et al. 2003; Higdon et al. 2004; Itoh and Okamoto 2007), de novo sequencing (Fernandez-de-Cossio et al. 1995; Taylor and Johnson 1997, 2001; Dancik et al. 1999; Chen et al. 2001; Fischer et al. 2005; Frank and Pevzner 2005; Pitzer et al. 2007), sequence tagging (Huang et al. 2001; Shevchenko et al. 2001; Mackey et al. 2002), and consensus of multiple engines (Resing et al. 2004) (Fig. 12.3).

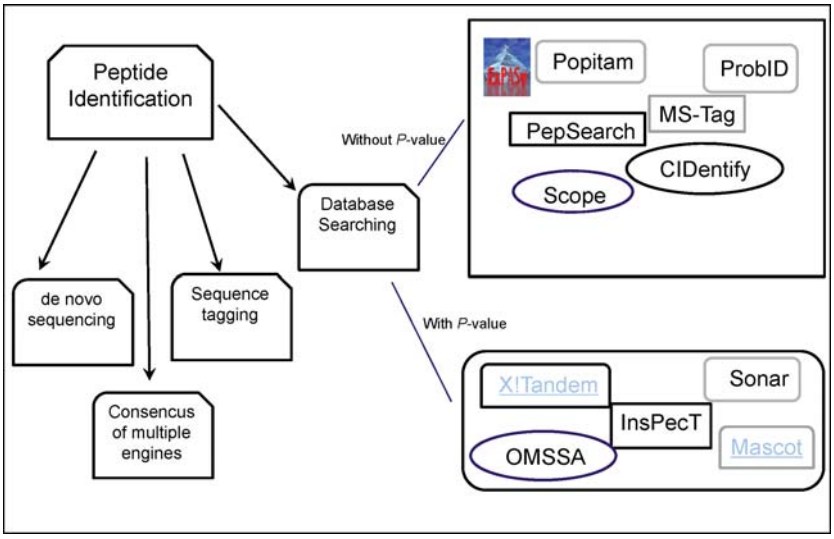

Fig. 12.3 The computational methods for protein/peptide identification

Database searching is the most important tool to identify protein. The peptide mass fingerprint (PMF), that is, the most abundant peaks in the spectrum, has the ability to uniquely define a protein. By searching a centralized database of sequenced protein, the homologous PMF of the predicted proteins derived from simulated transcription of sequenced genomes can be found. This method, however, does not have the ability to identify the proteins whose genomes are not yet sequenced and whose PTMs are not annotated. Robust identification depends on the complexity of the proteome searched and the presence of chemical noise and contaminants in the mass spectrum. In large proteomes, sensitivity can be greatly diminished.

When an appropriate database is not available, de novo sequencing is the only way to identify the peptide. With high-quality spectra, there is often enough information present to decode the sequences of unknown peptides directly, called as "de novo" sequencing. After the protein is digested by a small number of different enzymes and the de novo sequence of each different peptide set is found, the whole protein sequence can be recombined from the overlap.

Sequence-tagging approaches are to find the sequence of a peptide by searching a database with partial sequence information inferred from the MS-MS spectrum. Certainly, none of the peptide-sequencing programs is perfect. Researchers have started to use multiple programs to run the same dataset. The results of multiple engines are then combined to get fewer false positives, better coverage, and higher confidence.

The challenges in peptide identification are contamination of the sample, imperfect fragmentation, simultaneous fragmentation of two different peptides, post-translational modification (PTM), and low signal-to-noise ratio. Consequently, in practice, many y-ion and b-ion peaks might be absent from, and many other types of peaks might unexpectedly appear in, the spectrum. These can make MS–MS peptide identification significantly harder than it would appear to be.

12.4 Validation and Clinical Application

There is a big gap between biomarker discovery and biomarker validation (Rifai et al. 2006). Because of the challenges in blood proteomics and the algorithms in biomarker discovery, it is hard to find a consensus for the best biomarker discovery platform. Now, for the bioinformatics algorithms in biomarker discovery, the distinct combinations of them in different steps of biomarker discovery may result in different biomarker candidates although using the same sample. Such inconsistent and less reproducible biomarkers also inhibit the validation of them between labs and biomarker discovery platforms.

The gap between discovery and validation of proteomic biomarkers results in unusable diagnosis application of proteomic biomarkers. FDA, in recent years, only approves nearly one drug target discovered from proteomics per year (Fig. 12.4)

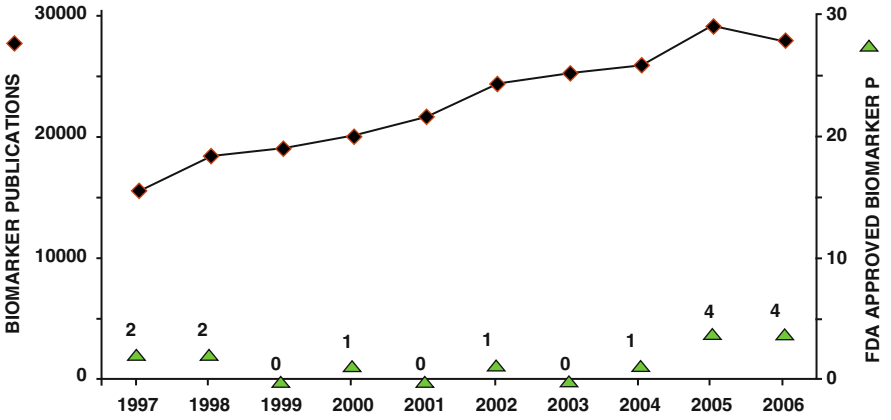

Fig. 12.4 The gap between the numbers of biomarker publications and FDA-approved biomarkers

(Ludwig and Weinstein 2005). Until now, none of proteomics biomarkers has been successfully applied in clinical application (Anderson and Anderson 2002).

The techniques required in the validation and clinical application of protein are immunoprecipitation (IP) (Su 2003), protein array (Stoll et al. 2002; Diamond et al. 2003; Bodovitz and Joos 2004), antibody array (Ng et al. 2007), tissue array (Shen et al. 2003; Radhakrishnan et al. 2008), nanoparticles (Kim et al. 2008), etc.

IP is a method that uses the antigen–antibody reaction principle to identify a protein that reacts specifically with an antibody from mixture of proteins so that its quantity or physical characteristics can be examined (Su 2003). An antibody (monoclonal or polyclonal) against a specific target antigen is allowed to form an immune complex with that target in a sample, such as a cell lysate. The immune complex is then captured on a solid support to which either Protein A or Protein G has been immobilized (Protein A or G binds to the antibody, which is bound to its antigen). The process of capturing this complex from the solution is referred to as precipitation. Any proteins not "precipitated" by the immobilized Protein A or G support are washed away. Finally, components of the bound immune complex (both antigen and antibody) are eluted from the support and analyzed by SDS-PAGE (gel electrophoresis), often followed by Western blot detection to verify the identity of the antigen. Immunoassays are expensive and slow to make, difficult to multiplex, and thus problematic for the new candidate biomarkers.

Protein arrays or antibody arrays are rapidly becoming established as a powerful means to detect proteins, monitor their expression levels, and investigate protein interactions and functions [131–134]. A mixture (e.g. of two tissue extracts) is applied to the array and the analytes of interest are captured by the specific ligand binders, followed by detection of binding. Similar to the comparison of samples from normal and diseased tissues on DNA arrays or on 2D gels, reference and test samples can be labeled with Cy3 and Cy5 fluors, mixed, gel filtered to remove unbound dyes, and then incubated on a chip of arrayed antibodies. Increased or decreased protein expression is assessed using a scanner, and up- or down-regulated proteins can be

identified from the ratios of the two dyes similar to the "traffic light" (red, yellow, green) system. There are a number of important technical challenges and bottlenecks in protein array technologies, some of which are unique to proteins while others are common to high-throughput methods in general, which need to be solved in order to achieve the maximum capability. They include the problems of obtaining global, functional protein expression for array construction and selection of ligand binders, aspects of protein coupling to surfaces, the sensitivity and dynamic range of detection systems, and standardization and data storage.

Tissue microarray (TMA) translates the convenience of DNA microarrays to tissues and evaluates the molecular targets in parallel [135, 136]. This approach allows simultaneous screening of large collectives of tumor specimens for any molecular alteration by relocating tissues from conventional histologic paraffin blocks, such that tissues from multiple patients or blocks could be seen on the same slide. The technology of TMA involves harvesting small disks of tissue from individual donor paraffin-embedded tissue blocks and placing them in a recipient block with defined array coordinates. These miniaturized collections of tissue spots result in a dramatic increase in throughput for in situ examination of gene status and gene expression from archival specimens. A single TMA block may yield information on the molecular characterization of up to 1,000 samples at a time. Although TMA looks promising, the major limitation of this technique is the tissue volume. The punches of 0.6-mm diameter from tumor are perceived as too small and potentially not representative of the entire area. This may be especially true of tumors that may be highly heterogeneous.

Nanoparticles are also used for protein detection [137]. A sensor array containing six noncovalent gold nanoparticle–fluorescent polymer conjugates has been created to detect, identify, and quantify protein targets. These gold particles serve as both selective recognition elements as well as quenchers for the polymer. The nanoparticle end groups carry additional hydrophobic, aromatic, or hydrogen-bonding functionality engineered to tune nanoparticle–polymer and nanoparticle–protein interactions. The challenge in this technique is the design of the polymers for gold nanoparticles. Since diverse polymers have different properties, such as monolayer and multilayer designs (for surface functionality and water solubility), detecting specific proteins needs specific polymers on the nanoparticles.

For the validation and clinical application of proteomic biomarkers, one problem is that proteomics discovery platforms are inefficient for the large sample sets required in verification or validation. Properly structured sets of ∼1,500 samples required in biomarker validation are unavailable from public sources and are expensive to develop (the "Zolg barrier") (Anderson and Anderson 2002). Next, most proteomic biomarker candidates are discovered from blood samples instead of tissue samples (Rifai et al. 2006). Just as what we discussed about the disadvantages of blood samples, such as the complexity and dynamic range of plasma, and the anticipated low relative abundance of many disease-specific biomarkers, the blood samples may not be suitable for biomarker discovery. In addition, most of proteomic biomarkers are validated in vitro instead of in vivo. From a biological point of view, the validation in vivo for the proteomics biomarkers discovered from tissues may be

more powerful for diagnosis application. Recently, the molecular imaging method has been proposed for biomarker validation in vivo (Pantaleo et al. 2008; Weissleder and Pittet 2008). To improve the accuracy of biomarker discovery, we should also find a new way in biomarker validation and clinical application.

Acknowledgments This research is funded by the Bioinformatics Core Research Grant at The Methodist Research Institute, Cornell University. Dr. Zhou is partially funded by The Methodist Hospital Scholarship Award. He and Dr. Wong are also partially funded by NIH grants R01LM08696, R01LM009161, and R01AG028928. The authors have declared no conflict of interest.

References

Adam BL, Qu Y et al (2002) Serum protein fingerprinting coupled with a pattern-matching algorithm distinguishes prostate cancer from benign prostate hyperplasia and healthy men. Cancer Res 62(13):3609–3614

Ahmed N, Barker G et al (2003) An approach to remove albumin for the proteomic analysis of low abundance biomarkers in human serum. Proteomics 3(10):1980–1987

Albrethsen J (2007) Reproducibility in protein profiling by MALDI-TOF mass spectrometry. Clin Chem 53(5):852–858

Alfassi ZB (2004) On the normalization of a mass spectrum for comparison of two spectra. J Am Soc Mass Spectrom 15(3):385–387

America AH, Cordewener JH (2008) Comparative LC-MS: a landscape of peaks and valleys. Proteomics 8(4):731–749

Anderson NL, Anderson NG (2002) The human plasma proteome: history, character, and diagnostic prospects. Mol Cell Proteomics 1(11):845–867

Anderson NL, Polanski M et al (2004) The human plasma proteome: a nonredundant list developed by combination of four separate sources. Mol Cell Proteomics 3(4):311–326

Andreev VP, Rejtar T et al (2003) A universal denoising and peak picking algorithm for LC-MS based on matched filtration in the chromatographic time domain. Anal Chem 75(22): 6314–6326

Arneberg R, Rajalahti T et al (2007) Pretreatment of mass spectral profiles: application to proteomic data. Anal Chem 79(18):7014–7026

Baggerly KA, Morris JS et al (2003) A comprehensive approach to the analysis of matrix-assisted laser desorption/ionization-time of flight proteomics spectra from serum samples. Proteomics 3(9):1667–1672

Ball G, Mian S et al (2002) An integrated approach utilizing artificial neural networks and SELDI mass spectrometry for the classification of human tumours and rapid identification of potential biomarkers. Bioinformatics 18(3):395–404

Bensmail H, Golek J et al (2005) A novel approach for clustering proteomics data using Bayesian fast Fourier transform. Bioinformatics 21(10):2210–2224

Bhanot G, Alexe G et al (2006) A robust meta-classification strategy for cancer detection from MS data. Proteomics 6(2):592–604

Bodovitz S, Joos T (2004) The proteomics bottleneck: strategies for preliminary validation of potential biomarkers and drug targets. Trends Biotechnol 22(1):4–7

Bolstad BM, Irizarry RA et al (2003) A comparison of normalization methods for high density oligonucleotide array data based on variance and bias. Bioinformatics 19(2):185–193

Brouwers FM, Petricoin EF III et al (2005) Low molecular weight proteomic information distinguishes metastatic from benign pheochromocytoma. Endocr Relat Cancer 12(2):263–272

Bylund D, Danielsson R et al (2002) Chromatographic alignment by warping and dynamic programming as a pre-processing tool for PARAFAC modelling of liquid chromatography-mass spectrometry data. J Chromatogr A 961(2):237–244
Callister SJ, Barry RC et al (2006) Normalization approaches for removing systematic biases associated with mass spectrometry and label-free proteomics. J Proteome Res 5(2):277–286
Chen T, Kao MY et al (2001) A dynamic programming approach to de novo peptide sequencing via tandem mass spectrometry. J Comput Biol 8(3):325–337
Cho SY, Lee EY et al (2005) Efficient prefractionation of low-abundance proteins in human plasma and construction of a two-dimensional map. Proteomics 5(13):3386–3396
Coombes KR (2005) Analysis of mass spectrometry profiles of the serum proteome. Clin Chem 51(1):1–2
Coombes KR, Fritsche HA Jr et al (2003) Quality control and peak finding for proteomics data collected from nipple aspirate fluid by surface-enhanced laser desorption and ionization. Clin Chem 49(10):1615–1623
Coombes KR, Tsavachidis S et al (2005) Improved peak detection and quantification of mass spectrometry data acquired from surface-enhanced laser desorption and ionization by denoising spectra with the undecimated discrete wavelet transform. Proteomics 5(16):4107–4117
Cox J, Mann M (2007) Is proteomics the new genomics? Cell 130(3):395–398
Dancik V, Addona TA et al (1999) De novo peptide sequencing via tandem mass spectrometry. J Comput Biol 6(3–4):327–342
Davis MT, Patterson SD (2007) Does the serum peptidome reveal hemostatic dysregulation? Ernst Schering Res Found Workshop 61:23–44
Diamandis EP (2003) Point: proteomic patterns in biological fluids: do they represent the future of cancer diagnostics? Clin Chem 49(8):1272–1275
Diamandis EP (2004) Analysis of serum proteomic patterns for early cancer diagnosis: drawing attention to potential problems. J Natl Cancer Inst 96(5):353–356
Diamond DL, Y Zhang et al (2003) Use of ProteinChip array surface enhanced laser desorption/ionization time-of-flight mass spectrometry (SELDI-TOF MS) to identify thymosin beta-4, a differentially secreted protein from lymphoblastoid cell lines. J Am Soc Mass Spectrom 14(7):760–765
Dijkstra M, Vonk RJ et al (2007) SELDI-TOF mass spectra: a view on sources of variation. J Chromatogr B Analyt Technol Biomed Life Sci 847(1):12–23
Ebert MP, Meuer J et al (2004) Identification of gastric cancer patients by serum protein profiling. J Proteome Res 3(6):1261–1266
Fenselau C (2007) A review of quantitative methods for proteomic studies. J Chromatogr B Analyt Technol Biomed Life Sci 855(1):14–20
Fernandez-de-Cossio J, Gonzalez J et al (1995) A computer program to aid the sequencing of peptides in collision-activated decomposition experiments. Comput Appl Biosci 11(4): 427–434
Fischer B, Roth V et al (2005) NovoHMM: a hidden Markov model for de novo peptide sequencing. Anal Chem 77(22):7265–7273
Fischer B, Grossmann J et al (2006) Semi-supervised LC/MS alignment for differential proteomics. Bioinformatics 22(14):e132–e140
Frank A, Pevzner P (2005) PepNovo: de novo peptide sequencing via probabilistic network modeling. Anal Chem 77(4):964–973
Fung ET, Enderwick C (2002) ProteinChip clinical proteomics: computational challenges and solutions. Biotechniques Suppl:34–38, 40–41
Fushiki T, Fujisawa H et al (2006) Identification of biomarkers from mass spectrometry data using a common peak approach. BMC Bioinformatics 7:358
Geho DH, Liotta LA et al (2006) The amplified peptidome: the new treasure chest of candidate biomarkers. Curr Opin Chem Biol 10(1):50–55
Geurts P, Fillet M et al (2005) Proteomic mass spectra classification using decision tree based ensemble methods. Bioinformatics 21(14):3138–3145
Gobom J, Mueller M et al (2002) A calibration method that simplifies and improves accurate determination of peptide molecular masses by MALDI-TOF MS. Anal Chem 74(15): 3915–3923

Gras R, Muller M et al (1999) Improving protein identification from peptide mass fingerprinting through a parameterized multi-level scoring algorithm and an optimized peak detection. Electrophoresis 20(18):3535–3550

Hanash SM, Pitteri SJ et al (2008) Mining the plasma proteome for cancer biomarkers. Nature 452(7187):571–579

Hastings CA, Norton SM et al (2002) New algorithms for processing and peak detection in liquid chromatography/mass spectrometry data. Rapid Commun Mass Spectrom 16(5):462–467

Hauskrecht M, Pelikan R et al (2005) Feature selection for classification of SELDI-TOF-MS proteomic profiles. Appl Bioinformatics 4(4):227–246

Higdon R, Kolker N et al (2004) LIP index for peptide classification using MS/MS and SEQUEST search via logistic regression. OMICS 8(4):357–369

Hilario M, Kalousis A et al (2006) Processing and classification of protein mass spectra. Mass Spectrom Rev 25(3):409–449

Hingorani SR, Petricoin EF et al (2003) Preinvasive and invasive ductal pancreatic cancer and its early detection in the mouse. Cancer Cell 4(6):437–450

Hoffmann P, Ji H et al (2001) Continuous free-flow electrophoresis separation of cytosolic proteins from the human colon carcinoma cell line LIM 1215: a non two-dimensional gel electrophoresis-based proteome analysis strategy. Proteomics 1(7):807–818

Hortin GL (2006) The MALDI-TOF mass spectrometric view of the plasma proteome and peptidome. Clin Chem 52(7):1223–1237

Huang L, Jacob RJ et al (2001) Functional assignment of the 20 S proteasome from *Trypanosoma brucei* using mass spectrometry and new bioinformatics approaches. J Biol Chem 276(30):28327–28339

Itoh SG, Okamoto Y (2007) Effective sampling in the configurational space of a small peptide by the multicanonical-multioverlap algorithm. Phys Rev E Stat Nonlin Soft Matter Phys 76(2, Part 2):026705

Jaitly N, Monroe ME et al (2006) Robust algorithm for alignment of liquid chromatography-mass spectrometry analyses in an accurate mass and time tag data analysis pipeline. Anal Chem 78(21):7397–7409

Jirasek A, Schulze G et al (2004) Accuracy and precision of manual baseline determination. Appl Spectrosc 58(12):1488–1499

Joos TO, Bachmann J (2005) The promise of biomarkers: research and applications. Drug Discov Today 10(9):615–616

Karpievitch YV, Hill EG et al (2007) PrepMS: TOF MS data graphical preprocessing tool. Bioinformatics 23(2):264–265

Kim YP, Oh YH et al (2008) Protein kinase assay on peptide-conjugated gold nanoparticles. Biosens Bioelectron 23(7):980–986

Lange E, Gropl C et al (2007) A geometric approach for the alignment of liquid chromatography-mass spectrometry data. Bioinformatics 23(13): i273–i281

Lee DS, Rudge AD et al (2005) A new model validation tool using kernel regression and density estimation. Comput Methods Programs Biomed 80(1):75–87

Lee HJ, Lee EY et al (2006) Biomarker discovery from the plasma proteome using multidimensional fractionation proteomics. Curr Opin Chem Biol 10(1):42–49

Li B, Robinson DH et al (1997) Evaluation of properties of apigenin and [G-3H]apigenin and analytic method development. J Pharm Sci 86(6):721–725

Li J, Zhang Z et al (2002) Proteomics and bioinformatics approaches for identification of serum biomarkers to detect breast cancer. Clin Chem 48(8):1296–1304

Li L, Umbach DM et al (2004) Application of the GA/KNN method to SELDI proteomics data. Bioinformatics 20(10):1638–1640

Listgarten J, Emili A (2005) Statistical and computational methods for comparative proteomic profiling using liquid chromatography-tandem mass spectrometry. Mol Cell Proteomics 4(4): 419–434

Listgarten J, Neal RM et al (2007) Difference detection in LC-MS data for protein biomarker discovery. Bioinformatics 23(2): e198–e204

Liu H, Li J et al (2002) A comparative study on feature selection and classification methods using gene expression profiles and proteomic patterns. Genome Inform 13:51–60
Ludwig JA, Weinstein JN (2005) Biomarkers in cancer staging, prognosis and treatment selection. Nat Rev Cancer 5(11):845–856
Ma B, Zhang K et al (2003) PEAKS: powerful software for peptide de novo sequencing by tandem mass spectrometry. Rapid Commun Mass Spectrom 17(20):2337–2342
Mackey AJ, Haystead TA et al (2002) Getting more from less: algorithms for rapid protein identification with multiple short peptide sequences. Mol Cell Proteomics 1(2):139–147
Malyarenko DI, Cooke WE et al (2005) Enhancement of sensitivity and resolution of surface-enhanced laser desorption/ionization time-of-flight mass spectrometric records for serum peptides using time-series analysis techniques. Clin Chem 51(1):65–74
Marcuson R, Burbeck SL et al (1982) Normalization and reproducibility of mass profiles in the detection of individual differences from urine. Clin Chem 28(6):1346–1348
McGuire JN, Overgaard J et al (2008) Mass spectrometry is only one piece of the puzzle in clinical proteomics. Brief Funct Genomic Proteomic 7(1):74–83
Miklos GL, Maleszka R (2001) Integrating molecular medicine with functional proteomics: realities and expectations. Proteomics 1(1):30–41
Mueller LN, Rinner O et al (2007) SuperHirn – a novel tool for high resolution LC-MS-based peptide/protein profiling. Proteomics 7(19):3470–3480
Ng JK, Ajikumar PK et al (2007) Spatially addressable protein array: ssDNA-directed assembly for antibody microarray. Electrophoresis 28(24):4638–4644
Pantaleo MA, Nannini M et al (2008) Conventional and novel PET tracers for imaging in oncology in the era of molecular therapy. Cancer Treat Rev 34(2):103–121
Park T, Yi SG et al (2003) Evaluation of normalization methods for microarray data. BMC Bioinformatics 4:33
Perkins DN, Pappin DJ et al (1999) Probability-based protein identification by searching sequence databases using mass spectrometry data. Electrophoresis 20(18):3551–3567
Perrin, C, Walczak B et al (2001) The use of wavelets for signal denoising in capillary electrophoresis. Anal Chem 73(20):4903–4917
Petricoin EF, Liotta LA (2003) Mass spectrometry-based diagnostics: the upcoming revolution in disease detection. Clin Chem 49(4):533–534
Petricoin EF III, Ornstein DK et al (2002a) Serum proteomic patterns for detection of prostate cancer. J Natl Cancer Inst 94(20):1576–1578
Petricoin EF, Ardekani AM et al (2002b) Use of proteomic patterns in serum to identify ovarian cancer. Lancet 359(9306):572–577
Petricoin EF, Belluco C et al (2006) The blood peptidome: a higher dimension of information content for cancer biomarker discovery. Nat Rev Cancer 6(12):961–967
Pitzer E, Masselot A et al (2007) Assessing peptide de novo sequencing algorithms performance on large and diverse data sets. Proteomics 7(17):3051–3054
Poon TC, Yip TT et al (2003) Comprehensive proteomic profiling identifies serum proteomic signatures for detection of hepatocellular carcinoma and its subtypes. Clin Chem 49(5):752–760
Powell K (2003) Proteomics delivers on promise of cancer biomarkers. Nat Med 9(8):980
Prados J, Kalousis A et al (2004) Mining mass spectra for diagnosis and biomarker discovery of cerebral accidents. Proteomics 4(8):2320–2332
Prince JT, Marcotte EM (2006) Chromatographic alignment of ESI-LC-MS proteomics data sets by ordered bijective interpolated warping. Anal Chem 78(17):6140–6152
Qu Y, Adam BL et al (2002) Boosted decision tree analysis of surface-enhanced laser desorption/ionization mass spectral serum profiles discriminates prostate cancer from noncancer patients. Clin Chem 48(10):1835–1843
Radhakrishnan R, Solomon M et al (2008) Tissue microarray – a high-throughput molecular analysis in head and neck cancer. J Oral Pathol Med 37(3):166–176
Rai AJ, Zhang Z et al (2002) Proteomic approaches to tumor marker discovery. Arch Pathol Lab Med 126(12):1518–1526

Ransohoff DF (2005) Bias as a threat to the validity of cancer molecular-marker research. Nat Rev Cancer 5(2):142–149
Rejtar T, Chen HS et al (2004) Increased identification of peptides by enhanced data processing of high-resolution MALDI TOF/TOF mass spectra prior to database searching. Anal Chem 76(20):6017–6028
Resing KA, Meyer-Arendt K et al (2004) Improving reproducibility and sensitivity in identifying human proteins by shotgun proteomics. Anal Chem 76(13):3556–3568
Ressom HW, Varghese RS et al (2005) Analysis of mass spectral serum profiles for biomarker selection. Bioinformatics 21(21):4039–4045
Ressom HW, Varghese RS et al (2007) Peak selection from MALDI-TOF mass spectra using ant colony optimization. Bioinformatics 23(5):619–626
Ressom HW, Varghese RS et al (2008) Classification algorithms for phenotype prediction in genomics and proteomics. Front Biosci 13:691–708
Rietjens IM, Steensma A et al (1995) Comparative biotransformation of hexachlorobenzene and hexafluorobenzene in relation to the induction of porphyria. Eur J Pharmacol 293(4):293–299
Rifai N, Gillette MA et al (2006) Protein biomarker discovery and validation: the long and uncertain path to clinical utility. Nat Biotechnol 24(8):971–983
Rogers MA, Clarke P et al (2003) Proteomic profiling of urinary proteins in renal cancer by surface enhanced laser desorption ionization and neural-network analysis: identification of key issues affecting potential clinical utility. Cancer Res 63(20):6971–6983
Rosty C, Christa L et al (2002) Identification of hepatocarcinoma-intestine-pancreas/pancreatitis-associated protein I as a biomarker for pancreatic ductal adenocarcinoma by protein biochip technology. Cancer Res 62(6):1868–1875
Sawyers CL (2008) The cancer biomarker problem. Nature 452(7187):548–552
Shackman JG, Watson CJ et al (2004) High-throughput automated post-processing of separation data. J Chromatogr A 1040(2):273–282
Shen S, Zhang PS et al (2003) Analysis of protein tyrosine kinase expression in melanocytic lesions by tissue array. J Cutan Pathol 30(9):539–547
Shevchenko A, Sunyaev S et al (2001) Charting the proteomes of organisms with unsequenced genomes by MALDI-quadrupole time-of-flight mass spectrometry and BLAST homology searching. Anal Chem 73(9):1917–1926
Shimizu A, Nakanishi T et al (2006) Detection and characterization of variant and modified structures of proteins in blood and tissues by mass spectrometry. Mass Spectrom Rev 25(5):686–712
Shin YK, Lee HJ et al (2006) Proteomic analysis of mammalian basic proteins by liquid-based two-dimensional column chromatography. Proteomics 6(4):1143–1150
Silva JC, Denny R et al (2005) Quantitative proteomic analysis by accurate mass retention time pairs. Anal Chem 77(7):2187–2200
Simpson RJ, Bernhard OK et al (2008) Proteomics-driven cancer biomarker discovery: looking to the future. Curr Opin Chem Biol 12(1):72–77
Steeves JB, Gagne HM et al (2000) Normalization of residual ions after removal of the base peak in electron impact mass spectrometry. J Forensic Sci 45(4):882–885
Stoll D, Templin MF et al (2002) Protein microarray technology. Front Biosci 7:c13–c32
Stolt R, Torgrip RJ et al (2006) Second-order peak detection for multicomponent high-resolution LC/MS data. Anal Chem 78(4):975–983
Su LK (2003) Co-immunoprecipitation of tumor suppressor protein-interacting proteins. Methods Mol Biol 223:135–140
Tam SW, Pirro J et al (2004) Depletion and fractionation technologies in plasma proteomic analysis. Expert Rev Proteomics 1(4):411–420
Tan CS, Ploner A et al (2006) Finding regions of significance in SELDI measurements for identifying protein biomarkers. Bioinformatics 22(12):1515–1523
Tang HY, Ali-Khan N et al (2005) A novel four-dimensional strategy combining protein and peptide separation methods enables detection of low-abundance proteins in human plasma and serum proteomes. Proteomics 5(13):3329–3342

Taylor JA, Johnson RS (1997) Sequence database searches via de novo peptide sequencing by tandem mass spectrometry. Rapid Commun Mass Spectrom 11(9):1067–1075
Taylor JA, Johnson RS (2001) Implementation and uses of automated de novo peptide sequencing by tandem mass spectrometry. Anal Chem 73(11):2594–2604
Thomas TM, Shave EE et al (2002) Preparative electrophoresis: a general method for the purification of polyclonal antibodies. J Chromatogr A 944(1–2):161–168
Tibshirani R, Hastie T et al (2004) Sample classification from protein mass spectrometry, by 'peak probability contrasts'. Bioinformatics 20(17):3034–3044
Villanueva J, Philip J et al (2004) Serum peptide profiling by magnetic particle-assisted, automated sample processing and MALDI-TOF mass spectrometry. Anal Chem 76(6):1560–1570
Vlahou A, Laronga C et al (2003) A novel approach toward development of a rapid blood test for breast cancer. Clin Breast Cancer 4(3):203–209
Wagner M, Naik D et al (2003) Protocols for disease classification from mass spectrometry data. Proteomics 3(9):1692–1698
Wang K, Johnson A et al (2005) TSE clearance during plasma products separation process by Gradiflow(TM). Biologicals 33(2):87–94
Wang MZ, Howard B et al (2003) Analysis of human serum proteins by liquid phase isoelectric focusing and matrix-assisted laser desorption/ionization-mass spectrometry. Proteomics 3(9):1661–1666
Wang P, Tang H et al (2006) Normalization regarding non-random missing values in high-throughput mass spectrometry data. Pac Symp Biocomput 315–326
Wang P, Tang H et al (2007) A statistical method for chromatographic alignment of LC-MS data. Biostatistics 8(2):357–367
Weissleder R, Pittet MJ (2008) Imaging in the era of molecular oncology. Nature 452(7187): 580–589
Whelan RJ, Sunahara RK et al (2004) Affinity assays using fluorescence anisotropy with capillary electrophoresis separation. Anal Chem 76(24):7380–7386
Won Y, Song HJ et al (2003) Pattern analysis of serum proteome distinguishes renal cell carcinoma from other urologic diseases and healthy persons. Proteomics 3(12):2310–2316
Wu B, Abbott T et al (2003) Comparison of statistical methods for classification of ovarian cancer using mass spectrometry data. Bioinformatics 19(13):1636–1643
Yasui Y, Pepe M et al (2003) A data-analytic strategy for protein biomarker discovery: profiling of high-dimensional proteomic data for cancer detection. Biostatistics 4(3):449–463
Yewdell JW (2003) Immunology. Hide and seek in the peptidome. Science 301(5638):1334–1335
Yu JS, Ongarello S et al (2005) Ovarian cancer identification based on dimensionality reduction for high-throughput mass spectrometry data. Bioinformatics 21(10):2200–2209
Zhang J, He S et al (2008) PeakSelect: preprocessing tandem mass spectra for better peptide identification. Rapid Commun Mass Spectrom 22(8):1203–1212
Zhang X, Lu X et al (2006) Recursive SVM feature selection and sample classification for mass-spectrometry and microarray data. BMC Bioinformatics 7:197
Zhukov TA, Johanson RA et al (2003) Discovery of distinct protein profiles specific for lung tumors and pre-malignant lung lesions by SELDI mass spectrometry. Lung Cancer 40(3):267–279

Index