BOPA : a bayesian hierarchical model for outlier expression detection

Hong, Zhaoping

2012

Hong, Z. P. (2012). BOPA : a bayesian hierarchical model for outlier expression detection.
Doctoral thesis, Nanyang Technological University, Singapore.

<https://hdl.handle.net/10356/49053>

<https://doi.org/10.32657/10356/49053>

**BOPA: A BAYESIAN
HIERARCHICAL MODEL FOR
OUTLIER EXPRESSION
DETECTION**

HONG ZHAOPING

Division of Mathematical Sciences
School of Physical and Mathematical Sciences

2012

BOPA: A BAYESIAN HIERARCHICAL MODEL FOR OUTLIER EXPRESSION DETECTION

HONG ZHAOPING

Division of Mathematical Sciences
School of Physical and Mathematical Sciences

A thesis submitted to Nanyang Technological University
in fulfillment of the requirement for the degree of
Doctor of Philosophy in Mathematical Sciences

2012

Acknowledgements

I would like, first and foremost, to thank my supervisor, Prof. Lian Heng, for the support and guidance shown to me in the development of this work and during my PhD study at Nanyang Technological University. His great insight into statistical methodology and hard work have set an example I hope to match some day.

I am grateful to the faculty of the Division of Mathematical Sciences at Nanyang Technological University. I have benefited a lot from various courses, seminars and individual discussions. I would also like to express my gratitude to the fellow students in the statistics group for their unbounded help, and many invaluable friendships.

Finally, I would like to thank my parents for their unwavering support during my time at Nanyang Technological University.

And, of course, Chun-Yin.

Abstract

DNA microarray technologies have the capability of simultaneously measuring the abundance of thousands of gene expressions in cells. A common task with microarrays is to determine which genes are differentially expressed under two different biological conditions of interest (e.g. cancerous against non-cancerous cells). It is often the case that there are thousands of genes for a single individual but relatively few individuals in the data set. Additionally, in many cancer studies, a gene may be expressed in some but not all of the disease samples, reflecting the complexity of the underlying disease. Traditional t -tests assume a mean shift for the tumor samples compared to normal samples and is thus not structured to capture partial differential expression. More powerful tests specially designed for this situation are needed to find genes with heterogeneous expressions associated with possible subtypes of the cancer. This thesis proposes a Bayesian model for cancer outlier profile analysis (BOPA). We build on the Gamma-Gamma model introduced in Newton et al. (2001); Kendzioriski et al. (2003) and Newton et al. (2004), by using a five-component mixture model to represent various differential expression patterns. The hierarchical mixture model explicitly accounts for outlier expressions and inferences are based on samples from posterior distributions generated from a Markov chain Monte Carlo algorithm. We present simulation and real-life datasets analysis to demonstrate our proposed methodology.

List of Works

Below is the list of work done, in chronological order, during my PhD studies in NTU.

1. Hong, Z. and Lian, H., “Inference of Genetic Networks from Time Course Expression Data Using Functional Regression with Lasso Penalty”, *Communications in Statistics-Theory and Methods*, Vol.40, No.10, pp. 1768-1779, 2011.
2. Hong, Z. and Lian, H., “Time-varying Coefficients Estimation in Differential Equation Models with Noisy Time-varying Covariates”, *Journal of Multivariate Analysis*, Vol.103, No.1, pp. 58-67, 2012.
3. Hong, Z. and Lian, H., “BOPA: A Bayesian Hierarchical Model for Outlier Expression Detection”, *submitted*.
Also appears on The 2011 International Conference on Applied Statistics, Taipei, Taiwan, May 2011.
4. Lian, H. and Hong, Z., “Sparse-smooth Regularized Singular Value Decomposition”, *submitted*.

Contents

Acknowledgements	iii
Abstract	v
List of Works	vi
1 Introduction	1
1.1 A Quick Introduction to Elements of Biology	2
1.2 Microarray Technology	4
1.3 Aims of Microarray Analysis	7
1.4 Research Problems	7
1.5 Innovative Aspects	8
1.6 Thesis Outline	9
2 Existing Statistical Methods	11
2.1 Literature Review	12
2.2 Methods to be Compared	14
2.3 False Discovery Rate	18
2.4 Summary of the Chapter	19

3	Introduction to Markov Chain Monte Carlo	21
3.1	Introduction to Markov Chains	22
3.2	The Metropolis and Metropolis-Hastings Algorithms	25
3.3	The Gibbs Sampler	28
3.4	Burning-in and Choice of Proposal Distribution	31
3.5	Summary of the Chapter	32
4	BOPA: A Five-state Bayesian Hierarchical Model	33
4.1	The Model	34
4.2	Marginal and Conditional Distributions	38
4.3	MCMC Sampler	43
4.4	Gene Selection	47
4.5	Summary of the Chapter	49
5	Simulations and Real Data Analysis	51
5.1	Simulations	51
5.2	Real-life Datasets	64
5.2.1	Breast Cancer Data Example	64
5.2.1.1	Full Data Set Analysis	64
5.2.1.2	Subset Data Analysis	66
5.2.1.3	Simulated Data Analysis	68
5.2.2	Prostate Cancer Data Example	70
5.3	Summary of the Chapter	72
6	Conclusion	75
6.1	Summary of Our Research	75
6.2	Limitations and Future Work	77
	Appendix	79

List of Figures

1.1	Comparison of cDNA and oligonucleotide microarray platforms (Wang et al., 2009).	5
4.1	The proposed hierarchical Bayesian model is represented as directed graphical model.	39
4.2	Possible transitions of the MCMC sampler among the five states. State 0 represents the non-DE state. States 1 and 3 represent up-regulated and down-regulated expression patterns respectively, while states 2 and 4 represent partially up-regulated and partially down-regulated expression patterns respectively.	44
5.1	ROC curves for simulation studies of scenario I with different sample sizes: true positive rate versus false positive rate. . .	55
5.2	ROC curves for simulation studies of scenario II with different sample sizes: true positive rate versus false positive rate.	56

5.3	ROC curves for simulation studies of scenario III with different sample sizes: true positive rate versus false positive rate.	57
5.4	Simulation results of scenario I with different sample sizes: comparisons of detecting power for genes with different expressed states, controlling the FDR below 0.05.	61
5.5	Simulation results of scenario II with different sample sizes: comparisons of detecting power for genes with different expressed states, controlling the FDR below 0.05.	62
5.6	Simulation results of scenario III with different sample sizes: comparisons of detecting power for genes with different expressed states, controlling the FDR below 0.05.	63
5.7	Rank of genes by coefficient of variation (CV) against rank of genes by mean: $G = 3170$ genes from $m = 8$ BRCA2 mutation positive tumors. For each gene, we compute the sample mean and CV which is the sample standard deviation divided by the sample mean.	65
5.8	ROC curves of the breast cancer simulated data set: (a) averaged false discovery rate. The x -axis is the positive rate: the proportion of genes called significant. (b) averaged successfully detection rate as a function of proportion of partially DE components.	68
5.9	Differentially expressed gene detection of prostate cancer microarray data: plotted are the $\log_2$ based intensity of those genes which are associated with prostate cancer. Those methods that can detect the gene are list in the parentheses. . . .	74

A.1	Comparison of trace plots for a gene that is partially up-regulated (state 2). (a)(b): Trace plot of Z_i when $K = 1$ and $K = 3$ respectively. (c)(d): Trace plot of $\lambda_{i2}/\lambda_{i1}$ when $K = 1$ and $K = 3$ respectively. The true value for $\lambda_{i2}/\lambda_{i1}$ is 0.5.	81
A.2	Comparison of trace plots for a gene that is partially down-regulated (state 4). (a)(b): Trace plot of Z_i when $K = 1$ and $K = 3$ respectively. (c)(d): Trace plot of $\lambda_{i2}/\lambda_{i1}$ when $K = 1$ and $K = 3$ respectively. The true value for $\lambda_{i2}/\lambda_{i1}$ is 2.	82

List of Tables

4.1	Definition of latent gene-specific state variable $Z_i(i = 1, \dots, G)$. Each Z_i takes value in $\mathcal{S} = \{0, 1, 2, 3, 4\}$, representing the five different gene expressed states for tumor samples. We use λ_{i1} and λ_{i2} to denote the inverse of the mean value of genes from normal samples and outlier tumor samples respectively.	36
5.1	Operating characteristics of simulation studies with $m=n=4$. $P(DE)$ is the true proportion of differentially expressed genes simulated. $\hat{DE}/G$ is the proportion of differentially expressed genes detected by different methods. Sens. is the proportion of genes that are detected out of the DE genes. Spec. is the proportion of genes that were not detected as DE out of the non-DE genes. FDR is the realized false discovery rate and FNDR is the realized false non-discovery rate.	58

5.2	Operating characteristics of simulation studies with $m=n=10$. $P(DE)$ is the true proportion of differentially expressed genes simulated. $\hat{DE}/G$ is the proportion of differentially expressed genes detected by different methods. Sens. is the proportion of genes that are detected out of the DE genes. Spec. is the proportion of genes that were not detected as DE out of the non-DE genes. FDR is the realized false discovery rate and FNDR is the realized false non-discovery rate.	59
5.3	Operating characteristics of simulation studies with $m=n=24$. $P(DE)$ is the true proportion of differentially expressed genes simulated. $\hat{DE}/G$ is the proportion of differentially expressed genes detected by different methods. Sens. is the proportion of genes that are detected out of the DE genes. Spec. is the proportion of genes that were not detected as DE out of the non-DE genes. FDR is the realized false discovery rate and FNDR is the realized false non-discovery rate.	60
5.4	DE genes detection for breast cancer data. Randomly choose four samples per group and report the average results based on 100 combinations. $\hat{DE}$: number of genes declared differentially expressed; % rep.: proportion of DE genes also declared significant when analyzing the full data set.	67

5.5	Operating characteristics of the breast cancer simulated data set. $\hat{DE}/G$ is the proportion of differentially expressed genes detected by different methods. Sens. is the proportion of genes that are detected out of the DE genes. Spec. is the proportion of genes that were not detected as DE out of the non-DE genes. FDR is the realized false discovery rate and FNDR is the realized false non-discovery rate.	69
5.6	Genes detected by t-test or BOPA under the target FDR level 0.01 and confirmed to be associated with prostate cancer. The last five columns also list the ranking of each gene by other methods. Those genes which are identified as DE by BOPA with 100% posterior probability are all ranked as 1. For ORT, genes are assigned the ranking 1742 if the ORT statistic equals zero. Those genes with OS statistic equal to zero are assigned a ranking of 1732.	73

Chapter 1

Introduction

Microarray technology has become a powerful and widely used technique in biology and medical research, which allows the simultaneous measurement of tens of thousands of gene expressions in cells. One of the important applications of the gene expression microarray is as a screening tool for detecting genes that are differentially expressed (DE) between different biological states in cancer studies. A particular challenge is that microarray experiments typically give us expression measurements on a large number of genes, say 10,000-20,000, but with few, if any, replicates for each gene. In addition, researchers have realized that changes of gene expression often occur in only a subset of cancer samples. These challenges that arise when examining microarray data have hampered the use of some classical statistical methods. In this thesis, we will present a Bayesian model for the analysis of differential expression in microarrays and develop computationally efficient methods for the estimation of various variables and parameters.

In this chapter, we first give a quick introduction to biology and microarray technology in Section 1.1 and Section 1.2. Particularly, we describe the

common problems in microarray data analysis in Section 1.3. After the introduction of microarrays, we present our research problems in Section 1.4 and contributions in Section 1.5. We conclude this introduction with an outline of the work that will be presented in this thesis in Section 1.6.

1.1 A Quick Introduction to Elements of Biology

Proteins are the structural components of cells and tissues and perform many key functions of biological systems. The ability to measure the presence and abundance of proteins in a particular cell under certain conditions is a central goal of molecular biology. Such knowledge would provide a very detailed description of the behavior of organisms at a cellular level and would in turn lead to a better understanding of various diseases and possible treatments. We first briefly review the process of protein synthesis.

The flow of genetic information is from DNA to RNA to proteins. This one-way process is the expression of genetic information in all cells and has been described as the central dogma of molecular biology (Watson and Crick, 1953).

The information encoded in DNA is ultimately used to construct proteins. The construction of proteins from DNA involves three distinct steps: translation, splicing and transcription. We will now briefly describe these three processes in turn.

1. **Transcription:** During this step, the double-stranded DNA is unwound and the information in one of the strands is copied into complementary preliminary messenger ribonucleic acid (mRNA).

2. **Splicing:** This stage involves removing the non-coding regions of the preliminary mRNA (known as introns). The remaining coding part of genes (exons) are then joined together. These different exons are combined to produce a final mRNA transcript.

3. **Translation:** In the final stage, mRNA is used as a template to assemble a chain of amino acids to form the protein. The order of amino acids is determined by three adjacent nucleotides, where each triplet is called a codon and codes one particular amino acid. As there are only 20 amino acids and 4^3 unique codons, this coding process is rather redundant.

The amount of mRNA molecules that are being transcribed by a particular gene is an approximate estimate of the level of expression of that gene.

Proteins in a cell are synthesized from genes and their life cycle can be roughly described as synthesis, functionality and degradation. There are many factors that can regulate the abundance of a particular protein, such as whether the respective gene is expressed, whether and how fast it is spliced, translated and modified, how long the half-life of the mRNA and the protein are. This suggests that the relationship between gene expression and the abundance of the respective protein is not straightforward. However, it has been proven difficult to measure protein abundance on a large scale. Therefore, much effort has been put into measuring their originators, the mRNAs. This measurement can be performed using microarrays, the analysis of which is the subject of this thesis.

1.2 Microarray Technology

DNA microarrays were first introduced in the mid-1990s and have for many years been central tools for large-scale measurement of mRNA abundance (Schulze and Downward, 2001). A gene expression microarray is made by placing tens of thousands of candidate gene sequences in discrete spots on a silicon chip or a glass slide. Each spot represents a known gene. There are two commonly available DNA microarray systems: cDNA microarrays and oligonucleotide microarrays, both of which rely on the same binding property of DNA.

cDNA microarrays

To measure gene expression, mRNA extracted from two different biological samples is reverse-transcribed into cDNA, labeled with dyes of different colors, and then hybridized to DNA sequences, each of which is spotted on a small region, or spot, on a glass slide. After hybridization, the dyes are excited individually by a laser and for each spot, a fluorescence reading can be made for both colors. Higher fluorescence indicates higher amounts of hybridized cDNA, which in turn indicates higher gene expression in the sample. The ratio of these readings provides us with a relative level of expression for our sample with respect to the control. These ratios are normally log-transformed (base 2) to preserve the symmetry between over and under expression. The downside of cDNA is that the absolute levels of gene expression can not be observed, but only one chip is needed for comparing different conditions.

Oligonucleotide microarrays

The construction of this type of microarray is slightly more complicated, but does not need a separate control sample and hence provides absolute

Figure 1.1: Comparison of cDNA and oligonucleotide microarray platforms (Wang et al., 2009).

rather than relative expression values. Expression of each gene is measured by comparing hybridization of the sample mRNA to a set of probes, composed of 11-20 pairs of oligonucleotides. The first type of probe in each pair is known as perfect match (PM) and is taken from the gene sequence. The second type is known as mismatch (MM) and is created by changing the middle base of the PM sequence. The goal of MM is to control for experimental variation and nonspecific binding of mRNA from other parts of genome. As with the cDNA arrays, our sample is labeled with a fluorescent dye and washed over the array. These microarrays give estimations of the absolute value of gene expression and therefore a comparison of two conditions requires the use of two separate microarrays. A comparison of

cDNA and oligonucleotide microarray platforms is given in Figure 1.1.

From images to numbers

Once the microarray has been hybridized, the amount of mRNA attached to the microarray is quantified using a laser scanner. This process results in a image where regions with more labeled mRNA have higher intensity compared to regions sparse of mRNA (Yang et al., 2001). From this point, a series of steps are necessary in order to come up with a data set of numbers:

Image analysis. In microarray experiments, the first quantified values are contained in the image files produced by the scanner. The pixel intensities, stored in these files, can be regarded as the raw data. Then image analysis tools are used for segmentation, which involves identifying the areas in those images that are representing expression information. Once the foreground and background regions are determined for each spot, the pixel values are summarized to give a single measurement for each region and background noise is subtracted.

Quality control. Quality control of the array involves analysis at three different levels. At the probe level, the quality of the spots is assessed (scratches, dark regions, spot size, magnitude of signal relative to noise). At the array level, some sources of error need to be evaluated such as array fabrication defects, problems with RNA extraction, failed labeling reaction, poor hybridization conditions and faulty scanning. At the gene level, the quality of the clones (contamination, mislabeling), or poor hybridization and printing is assessed.

Array normalization. The aim of the normalization is to account for systematic differences across different data sets and eliminate artifacts, so that the genes that are not really differentially expressed have similar values across the arrays. To do so, housekeeping genes (genes that are involved

in essential activity of the cell maintenance and survival, but not in cell function or proliferation) or spiked controls (for example adding non-human RNA to human samples) can be used. Alternatively, the normalization is achieved by centering arrays on their medians or means.

1.3 Aims of Microarray Analysis

The typical microarray experiment aims to identify differences in gene expression under different conditions. These can be between time points during a biological process, or between different tissue or tumor types. An obvious application is the identification of bio-markers corresponding to various disease states. For example, we may have gene expression measurements for some healthy prostate cells and similar measurements from tumor cells. Genes that have consistent differential expression across these two groups may be interesting targets for future biological research. Identification of these genes will certainly contribute to advancing our understanding of the molecular basis for human diseases and identifying novel therapeutic targets.

1.4 Research Problems

Microarray technology provides many opportunities to investigate cancer diseases using gene expressions. In gene expression studies, arguably the most important problem is to identify those genes that are associated with the underlying genomic mechanism of the disease, whose patterns of expression differ according to phenotype or experimental condition. Many different procedures have been proposed to tackle this problem (Baldi and

Long (2001); Ibrahim et al. (2002); Do et al. (2005); Lewin et al. (2006)). Although some methods work with multiple experimental conditions, this thesis focuses on the analysis of differential expressions under two conditions.

A particular challenge for microarray research is that the data are expensive to gather. The number of variables (genes) available in microarray studies tends to exceed the number of cases (subjects) by several orders of magnitude. Traditional statistical approaches to design and analysis were not developed in the context of such high dimensional and small sample problems. Recently, many researchers have come to the realization that in many cancer studies, some genes show increased or decreased expressions in disease samples, but only for a small number of those samples (Tomlins et al., 2005).

1.5 Innovative Aspects

To deal with the problems we identified in the last section related to cancer outlier detection, we will develop a novel procedure that overcomes these problems.

Here we are especially interested in detecting genes that are expressed in only a subset of the tumor samples. We choose the Bayesian hierarchical modeling framework as adopted by Kendzierski et al. (2003) and Newton et al. (2004). Bayesian approaches have several advantages over traditional methods and have been used widely in microarray studies. One of the most important aspects of Bayesian hierarchical modeling on microarray data is the sharing of information across different experimental units. This is especially important since most experiments are carried out with a small

number of biological replicates, making classical estimates of expression variabilities unstable.

Unlike the models proposed by Newton et al. (2004), we will use a five-component mixture model (also referred to as a five-state model) for different patterns of expressions across samples. Besides the usual three states used in some previous models to designate non-differential expression, up-regulated expression and down-regulated expression, we use two extra states for partially up/down-regulated expression. Previously proposed Bayesian methods for differential expression, such as Newton et al. (2004), Gottardo et al. (2006) and Lo and Gottardo (2007), do not take into account partial expression profiles and thus are expected to have low power in detecting outlier expression. As far as we know, besides partially up-regulated and partially down-regulated expression, there is no other possibilities that have been considered in detecting DE genes under two conditions. So in this thesis, we only consider five different states for each gene.

While previous works favored inferences based on posterior modes found using the EM algorithm, we prefer a fully Bayesian approach to explore the full posterior distribution using a Markov chain Monte Carlo (MCMC) algorithm. Besides, our efforts to develop a new model for detecting partial differential expression results in a much more complicated computational problem which possibly excludes the usage of the EM algorithm.

1.6 Thesis Outline

This thesis is organized into six chapters as follows.

In Chapter 2, we review previous methods for detecting differentially expressed genes. We also introduce in detail six methods which will be

compared with our proposed method in Chapter 5.

In Chapter 3, we give a brief introduction of Markov chain Monte Carlo (MCMC), which will be used to estimate the unknown parameters and latent variables contained in our newly developed method.

Chapters 4 and 5 contain the main contributions of this thesis. In Chapter 4, we describe our approach for cancer outlier profiling, as well as the computational challenges posed by the model, and outline the MCMC algorithm for posterior computation with detailed sampling steps. We also discuss Bayesian approaches to control the false discovery rate (FDR).

In Chapter 5, we perform some simulations to study the computational properties of the algorithm and to compare our model with several previous approaches. Interestingly, our analysis shows that the OS statistic proposed in Tibshirani and Hastie (2007) has serious problems in differential expression detection, which has not been noticed before. In addition, we evaluate the performance of various methods on two real-life data sets as well as on generated expression data.

In Chapter 6, we conclude the thesis by summarizing our contribution and reviewing our results. Also, we briefly outline some directions for the future work.

Chapter 2

Existing Statistical Methods

Microarrays allow simultaneous monitoring of thousands of gene expressions in cells. These present new statistical problems because the data are very highly dimensionalized with a very small number of replications. In past decades, many researchers have devoted their efforts to the study of the differential expression problem in microarray data analysis, in order to provide methodology that is not only sensitive in detecting DE genes but also flexible enough to account for the complex variability of microarray data (Kim et al., 2006; Lewin and Richardson, 2008). With microarray data, however, one needs to test possibly tens of thousands of features, thus creating a problem of multiplicity. In this Chapter, we describe a few widely used methods for testing differential expression, as well as the False Discovery Rate (FDR), which is a common error measure for controlling the number of false positives in microarrays. This chapter is organized as follows. In Section 2.1, we overview the techniques that had been used for detecting differentially expressed genes in the past. In Section 2.2, we describe six methods which will be used for comparison to our newly developed procedure. In Section 2.3, we introduce the FDR and review some

multiple testing procedures that are known to control errors. In Section 2.4, we summarize this chapter.

2.1 Literature Review

The most widely used method for detecting differential gene expression in comparative microarray studies is the two-sample t -statistic. Recently, many researchers have come to the realization that in many cancer studies, some genes show up- or down-expression patterns in a subset of disease samples. The study of Tomlins et al. (2005) shows that the t -statistic has low power in this case, and they introduced the so-called cancer outlier profile analysis (COPA). Their study shows clearly that COPA can perform better than the traditional t -test for cancer microarray datasets. More recently, several progresses have been made in this direction with the aim to design better statistics which account for the heterogeneous activation pattern of the cancer genes. Tibshirani and Hastie (2007) introduced a new statistic, which they called the outlier sum (OS). Later, Wu (2007) proposed an outlier robust t -statistic (ORT) and showed it usually outperformed the previously proposed ones in both simulation studies and applications to real data. Other contributions in this emerging field include Lian (2008), Hu (2008), van Wieringen et al. (2008) and Ghosh and Chinnaiyan (2009). Much like OS and ORT, the first two considered the few extreme gene expressions and compared their mean with the overall mean, but used a slightly different method in picking out these outliers. The latter two approaches are nonparametric in nature, and thus we expect that they will not work as well for small sample sizes when distribution functions cannot be estimated accurately.

In addition to the frequentist tests mentioned above, Bayesian methods recently show great promise, and can offer many advantages over more conventional approaches. Efron et al. (2001) discussed an empirical Bayes approach. They computed the posterior probability of differential expression by substituting estimates of relevant parameters based on the empirical distribution of observed transcription levels. Lönnstedt and Speed (2002) proposed an empirical Bayes normal mixture model for gene expression data, which was later extended to more general linear models by Smyth (2004) and Cui et al. (2005). Gottardo et al. (2006) developed a robust Bayesian hierarchical model, in which they used t -distribution to accommodate outlying data points. Rather than modeling the gene variability by a linear model, Newton et al. (2001) developed a hierarchical Gamma-Gamma (GG) method for detecting changes in gene expression in a single two-channel cDNA slide. Kendzierski et al. (2003) extended this to replicate chips with multiple conditions, and provided the option of using a hierarchical lognormal normal (LNN) model. Newton et al. (2004) presented a Bayesian hierarchical model using a mixture prior on the latent mean component representing different gene expressed states. Rossell (2009) extend it by relaxing the fixed-coefficient-of-variation assumption. Bayesian models are well-suited to the small sample sizes of microarray studies since they borrow information from all genes to estimate model parameters. They also provide a framework for incorporating all available information in a systematic manner, and explicitly include model and parameter variability.

2.2 Methods to be Compared

In this section, we describe six methods for determining differential expression in microarray data. These six methods will be compared to each other, and to our contributions in Chapter 5.

A comment on notation: suppose the experimental data consist of m samples from a normal group and n samples from a cancer group. Let x_{ij} be the measurement for gene $i = 1, 2, \dots, G$ and sample $j = 1, 2, \dots, m$ in the normal group and y_{ij} be the measurement for gene $i = 1, 2, \dots, G$ and sample $j = 1, 2, \dots, n$ in the cancer group. The six methods are as follows:

1. *T-statistic*

The standard t -statistic (up to a multiplication factor independent of the genes under study) for two-sample test of differences in means is defined for each gene i by

$$T_i = \frac{\bar{x}_i - \bar{y}_i}{s_i}, \quad (2.2.1)$$

where $\bar{x}_i = \sum_j x_{ij}/m$ is the average expression of gene i in normal samples, $\bar{y}_i = \sum_j y_{ij}/n$ is the average expression of gene i in cancer samples, and s_i is the usual pooled standard deviation estimate

$$s_i^2 = \frac{\sum_{1 \leq j \leq m} (x_{ij} - \bar{x}_i)^2 + \sum_{1 \leq j \leq n} (y_{ij} - \bar{y}_i)^2}{n + m - 2}.$$

The t -statistic is powerful when the alternative distribution is such that $y_{ij}, j = 1, 2, \dots, n$ all come from a distribution with a higher mean. For most cancer types, heterogeneous activation patterns make t -statistic inefficient for detecting those expression profiles.

2. *Outlier sum statistic (OS)*

In order to look for a better way of detecting changes that occur

in a small number of cancer samples, Tibshirani and Hastie (2007) proposed the “outlier-sum” statistic (OS) as follows:

$$OS_i = \frac{\sum_{y_{ij} \in O_i} (y_{ij} - \text{med}_i)}{\text{mad}_i}, \quad (2.2.2)$$

where O_i is the set of outliers disease samples for gene i defined by the following criterion:

$$O_i = \{y_{ij} : y_{ij} > q_{75}(\{x_{ij}\}_{1 \leq j \leq m}, \{y_{ij}\}_{1 \leq j \leq n}) + \text{IQR}(\{x_{ij}\}_{1 \leq j \leq m}, \{y_{ij}\}_{1 \leq j \leq n})\}, \quad (2.2.3)$$

where $q_r(\cdot)$ is the r th percentile of the data, and med_i is the median of the pooled samples for gene i

$$\text{med}_i = \text{median}(\{x_{ij}\}_{1 \leq j \leq m}, \{y_{ij}\}_{1 \leq j \leq n}),$$

and mad_i is the median absolute deviation of the pooled samples given by:

$$\text{mad}_i = 1.4826 \times \text{median}(\{|x_{ij} - \text{med}_i|\}_{1 \leq j \leq m}, \{|y_{ij} - \text{med}_i|\}_{1 \leq j \leq n}),$$

and $\text{IQR}(\cdot)$ is the interquartile range of the expression data

$$\begin{aligned} \text{IQR}(\{x_{ij}\}_{1 \leq j \leq m}, \{y_{ij}\}_{1 \leq j \leq n}) = & q_{75}(\{x_{ij}\}_{1 \leq j \leq m}, \{y_{ij}\}_{1 \leq j \leq n}) \\ & - q_{25}(\{x_{ij}\}_{1 \leq j \leq m}, \{y_{ij}\}_{1 \leq j \leq n}). \end{aligned}$$

3. Outlier robust t -statistic (ORT)

Wu (2007) argued that it might not be inefficient to use the median or the median absolute deviation over all samples, and instead proposed a modified version called the outlier robust t -statistic (ORT). The main difference is that outliers are defined relative to the normal sample instead of the pooled sample. So in their definition,

$$O_i = \{y_{ij} : y_{ij} > q_{75}(\{x_{ij}\}_{1 \leq j \leq m}) + \text{IQR}(\{x_{ij}\}_{1 \leq j \leq m})\}. \quad (2.2.4)$$

Similary, med_i in OS was replaced by med_{ix} and mad_i was replaced by

$$\text{mad}_i = 1.4826 \times \text{median}(\{|x_{ij} - \text{med}_{ix}|\}_{1 \leq j \leq m}, \{|y_{ij} - \text{med}_{iy}|\}_{1 \leq j \leq n}),$$

where med_{ix} and med_{iy} are the medians of normal and cancer samples respectively.

4. *Bayesian hierarchical model by Newton et al. (2004)*

Newton et al. (2004) proposed a method for detecting DE genes between two groups using a Bayesian hierarchical Gamma-Gamma model. Let μ_{i1} and μ_{i2} denote the gene-specific latent mean levels of $\{x_{ij}\}_{1 \leq j \leq m}$ and $\{y_{ij}\}_{1 \leq j \leq n}$. They modeled the gene expression values as Gamma random variables conditional on the mean components:

$$\begin{aligned} x_{ij} | \mu_{i1} &\sim \text{Gamma}(a_1, \frac{a_1}{\mu_{i1}}), \quad i = 1, \dots, G, j = 1, \dots, m \\ y_{ij} | \mu_{i2} &\sim \text{Gamma}(a_2, \frac{a_2}{\mu_{i2}}), \quad i = 1, \dots, G, j = 1, \dots, n \end{aligned}$$

where the shape parameters a_1 and a_2 are constant across genes within a condition (hence constant coefficients of variation), but may vary between conditions. Next, they considered expectations μ_{i1}, μ_{i2} to be a random pair from a bivariate distribution f , which is a discrete mixture over three hypotheses: non-DE, over-expressed and under-expressed in the cancer samples. Hence, the joint distribution of the latent mean levels for a single gene i can be written as a three-component mixture model:

$$\begin{aligned} f(\mu_{i1}, \mu_{i2}) &= p_0 \pi(\mu_{i1}) 1\{\mu_{i1} = \mu_{i2}\} \\ &+ 2p_1 \pi(\mu_{i1}) \pi(\mu_{i2}) 1\{\mu_{i1} < \mu_{i2}\} \\ &+ 2p_2 \pi(\mu_{i1}) \pi(\mu_{i2}) 1\{\mu_{i1} > \mu_{i2}\}, \end{aligned}$$

where p_0, p_1 and p_2 denote the marginal proportion of genes satisfying each of the three hypotheses, π is the density of inverse Gamma distribution and $1\{\cdot\}$ is the indicator function. These unknown parameters can be estimated by maximizing the marginal likelihood of the data using the EM algorithm.

5. *Bayesian robust inference for differential gene expression (BRIDGE)*

Gottardo et al. (2006) developed a Bayesian framework for testing for differential expression in microarrays in a way that is robust to outlying measurements due to experimental errors. They considered the measurements from each gene as the sum of a sample effect and an error term. The effect of sample j on each gene i is assumed to be a random effect with a mixture of two singular normal distributions, while the errors are modeled explicitly using a t -distribution, which accounts for outliers. Parameter estimation is carried out using Markov chain Monte Carlo (MCMC) algorithms. An R package called `bridge` implementing the method is available from Bioconductor at www.bioconductor.org.

6. *GaGa model*

Rossell (2009) modified the Gamma-Gamma hierarchical model introduced by Kendzierski et al. (2003) and Newton et al. (2004) by relaxing the constant coefficient of variation assumption. They introduced a gene-specific shape parameter a_i and assumed:

$$x_{ij}|\mu_{i1} \sim \text{Gamma}(a_i, \frac{a_i}{\mu_{i1}}), \quad i = 1, \dots, G, \quad j = 1, \dots, m$$

$$y_{ij}|\mu_{i2} \sim \text{Gamma}(a_i, \frac{a_i}{\mu_{i2}}), \quad i = 1, \dots, G, \quad j = 1, \dots, n$$

with the following hierarchical prior

$$\mu_{ik}|a_0, \nu \sim \text{Inverse-Gamma}(a_0, \frac{a_0}{\nu}), \quad i = 1, \dots, G, \quad k = 1, 2$$

$$a_i|\beta, \lambda \sim \text{Gamma}(\beta, \frac{\beta}{\lambda}), \quad i = 1, \dots, G.$$

Both EM and fully Bayesian approaches can be used to estimate the unknown parameters. This method is available as an R package from the Bioconductor.

2.3 False Discovery Rate

As we can see, microarray data analysis is basically a multiple hypothesis testing problem with a large number of hypotheses and a small number of replicates. Thus, controlling the multiple tests error is an important problem in microarray data analysis. A common error measure for controlling the number of false positives in microarrays is the False Discovery Rate (FDR) introduced by Benjamini and Hochberg (1995). This is the expectation of the false discovery proportion: let V be the number of false positives and R be the number of rejections, then

$$q = FDR = E(V/R|R > 0) \cdot Pr(R > 0).$$

Typically, for the microarray data analysis, researchers seek to control the FDR at $q = 0.05$, meaning that on average 5% of the declared significant DE genes are false.

The stepwise p -values adjustment method of Benjamini and Hochberg (1995) is one of the most widely used methods for controlling the FDR. Storey and Tibshirani (2003) also proposed a method to transform the p -values to q -values to find the threshold for controlling FDR. Here, the

q -value provides a measure of each feature's significance in terms of the FDR, automatically taking into account the fact that thousands of genes are simultaneously being tested (Storey and Tibshirani, 2003). Newton et al. (2004) demonstrated that the FDR can be estimated based on the posterior distribution of each gene within the Bayesian framework.

2.4 Summary of the Chapter

In this chapter, we reviewed frequentist and Bayesian methods for identifying differentially expressed genes. In this thesis, we focus on detecting the genes of differential expression only in a subset of cancer groups. Towards this target, we prefer Bayesian hierarchical modeling due to the advantages that have been discussed in Section 2.1. Inference in Bayesian models can be made in either an empirical or fully Bayesian framework. In the case of fully Bayesian models, Markov chain Monte Carlo (MCMC) is frequently used to estimate the posterior distribution of the model. In the following chapter, we introduce the MCMC algorithms used in this thesis.

Chapter 3

Introduction to Markov Chain Monte Carlo

The Markov chain Monte Carlo (MCMC) method has been widely used in modern Bayesian computing. Only for the simplest Bayesian models can we recognize the analytical forms of the posterior distributions and summarize inferences directly. The most common case is when conjugate prior distributions have been assumed. For more complicated problems such as Bayesian hierarchical models, posterior densities are too difficult to work with directly. Rather than deriving the analytic form of the posterior distribution, MCMC methods substitute a set of repetitive calculations that, in effect, simulate draws from this distribution. These draws are then used to approximate expectations of quantities of interest. MCMC approaches are so-named because one uses the previous sample values to randomly generate the next sample value, generating a Markov chain. So before introducing Markov chain Monte Carlo methods in more detail, it is useful to discuss some general properties of Markov chains (Section 3.1). In Section 3.2, we introduce the basic Metropolis algorithm, and the Metropolis-Hastings

generalization. In Section 3.3, we describe the Gibbs sampler which is a special case of Metropolis-Hastings algorithm. Finally, we discuss burning-in the sampler and how to choose a proposal distribution in Section 3.4. We conclude this chapter with a summary in Section 3.5. The introduction is mainly based on Gamerman and Lopes (2006), Gelman et al. (2004) and Walsh (2004).

3.1 Introduction to Markov Chains

The purpose of this section is to give an introduction to Markov chains which are required for developing MCMC methods. Let $\{\theta^0, \theta^1, \dots, \theta^t, \dots\}$ be a sequence of random variables with the range defined on a discrete state space $\mathcal{S}$. Then this sequence of random variables is a Markov chain if θ^{t+1} depends on all previous θ s only through its immediate predecessor θ^t , i.e.,

$$P(\theta^{t+1} = s_{t+1} | \theta^0 = s_0, \dots, \theta^t = s_t) = P(\theta^{t+1} = s_{t+1} | \theta^t = s_t) \quad (3.1.1)$$

for all $t = 0, 1, \dots$ and all $s_0, s_1, \dots, s_{t+1} \in \mathcal{S}$. Thus, we can think of a Markov chain applied to sampling as a mechanism that traverses randomly through a target distribution without having any memory of where it has been. Where it moves next is entirely dependent on where it is now.

We can then specify the Markov chain by giving the marginal distribution for θ^0 —the initial probabilities of the various states—and the conditional distributions for θ^{t+1} given the values for θ^t —the transition probabilities for one state to follow another state. Let $\boldsymbol{\pi}(0)$ denote the row vector of the initial probabilities of the state space, and $P(i, j)$ denote the transition probability that a process at state s_i moves to state s_j in a single step,

$$P(i, j) = P(i \rightarrow j) = P(\theta^{t+1} = s_j | \theta^t = s_i). \quad (3.1.2)$$

Using the transition probabilities, we can express the probability that the chain is in state s_i at time $t + 1$, denoted by $\pi_i(t + 1)$, in terms of the marginal probability for the previous variable in the form

$$\begin{aligned}\pi_i(t + 1) &= P(\theta^{t+1} = s_i) \\ &= \sum_k P(\theta^{t+1} = s_i | \theta^t = s_k) \cdot P(\theta^t = s_k) \\ &= \sum_k P(k \rightarrow i) \pi_k(t) = \sum_k P(k, i) \pi_k(t).\end{aligned}\quad (3.1.3)$$

Let $\boldsymbol{\pi}(t)$ denote the row vector of the state space probabilities at step t and $\mathbf{P}$ denote the matrix whose ij -th element is $P(i, j)$. Then Equation 3.1.3 can be written as

$$\boldsymbol{\pi}(t + 1) = \boldsymbol{\pi}(t)\mathbf{P} \quad (3.1.4)$$

Using the matrix form, we immediately have

$$\boldsymbol{\pi}(t) = \boldsymbol{\pi}(t - 1)\mathbf{P} = (\boldsymbol{\pi}(t - 2)\mathbf{P})\mathbf{P} = \boldsymbol{\pi}(t - 2)\mathbf{P}^2 = \cdots = \boldsymbol{\pi}(0)\mathbf{P}^t \quad (3.1.5)$$

A question of particular interest in the context of MCMC is the question of the long-term behavior of a Markov chain. We hope that the distribution of the chain converges to a well defined and unique limit and to exercise some control over what that limit shall be. A distribution is said to be invariant (or stationary) with respect to a Markov chain if each step in the chain leaves that distribution invariant. More formally, the distribution given by the probabilities $\boldsymbol{\pi}^*$ is stationary with respect to the Markov chain with transition matrix $\mathbf{P}$ if

$$\boldsymbol{\pi}^* = \boldsymbol{\pi}^*\mathbf{P} \quad (3.1.6)$$

The conditions for a stationary distribution is that the chain is irreducible and aperiodic. A Markov chain is said to be irreducible if, from all starting points, the Markov chain can reach any non-empty set with positive

probability, in some number of iterations. Likewise, a chain is said to be aperiodic if the chain doesn't oscillate between different sets of states in a regular periodic movement.

A sufficient (but not necessary) condition for ensuring that the required distribution π^* is stationary is to choose the transition probabilities to satisfy the property of detailed balance, defined by,

$$\pi_i^* P(i, j) = \pi_j^* P(j, i) \quad (3.1.7)$$

for all i and j . It is easily seen that a transition probability that satisfies detailed balance with respect to a particular distribution will leave that distribution stationary, because the j th element of $\pi \mathbf{P}$ is

$$(\pi \mathbf{P})_j = \sum_i \pi_i P(i, j) = \sum_i \pi_j P(j, i) = \pi_j \sum_i P(j, i) = \pi_j \quad (3.1.8)$$

which implies $\pi = \pi \mathbf{P}$. A Markov chain that respects detailed balance is said to be reversible.

In many applications we will need to sample from a distribution over continuous spaces, and hence will need to construct Markov chain having these as state spaces. The basic idea of discrete-state Markov chain can be generalized to a continuous state Markov process. Let $\pi_t(x)$ be the probability density for state x at time t and $p(x, y)$ be the probability density for a transition to state y from a given state x . Then the continuous extension of Equation 3.1.3 becomes

$$\pi_{t+1}(y) = \int \pi_t(x) p(x, y) dx \quad (3.1.9)$$

At equilibrium, the stationary distribution satisfies

$$\pi^*(y) = \int \pi^*(x) p(x, y) dx \quad (3.1.10)$$

The detailed balance 3.1.7 remains valid.

3.2 The Metropolis and Metropolis-Hastings Algorithms

In this section, we present the basic Metropolis algorithm and its generalization to the Metropolis-Hastings algorithm.

Metropolis algorithm

The Metropolis algorithm (Metropolis and Ulam, 1949; Metropolis et al., 1953) is named after its inventor, the American physicist and computer scientist Nicholas C. Metropolis. This sampler is simple but practical, and it can be used to obtain random samples from any arbitrary complicated target distribution of any dimension that is known up to a normalizing constant.

Suppose our goal is to draw T samples from some distribution $\pi(\theta)$ where $\pi(\theta) = p(\theta)/K$. The normalizing constant K is usually unknown, and very difficult to compute. To use the Metropolis algorithm, we need to set an initial value θ^0 and a symmetric proposal density $q(\theta^t, \theta^{t+1})$, which is the probability of visiting a value of θ^{t+1} given a previous value of θ^t . For the $(t + 1)$ th iteration, the algorithm generates a sample from $q(\theta^t, \cdot)$, and it makes a decision to either accept or reject the new sample. If the new sample is accepted, the algorithm repeats itself by starting at the new sample. If the new sample is rejected, the algorithm repeats with $\theta^{t+1} = \theta^t$. Many samples can be generated from π by repeating in this way.

The proposal distribution $q(\theta^t, \theta^*)$ should be an easy distribution from which to sample, and it must be symmetric, i.e., $q(\theta^t, \theta^*) = q(\theta^*, \theta^t)$, meaning that the likelihood of jumping to θ^* from θ^t is the same as the likelihood

of jumping back to θ^t from θ^* . The most common choice of the proposal distribution is the normal distribution $N(\theta^t, \sigma^2)$ with a fixed and user-specified σ^2 . The Metropolis algorithm can be summarized as follows:

Metropolis algorithm

1. Set $t = 0$, and choose a starting point θ^0 satisfying $p(\theta^0) > 0$.
2. Using current θ value, sample a candidate point θ^* from the proposal distribution $q(\theta^t, \cdot)$.
3. Calculate the following ratio:

$$\alpha = \min \left(\frac{p(\theta^*)}{p(\theta^t)}, 1 \right).$$

4. Sample u from the uniform distribution $U(0, 1)$.
 5. Set $\theta^{t+1} = \theta^*$ if $u < \alpha$; otherwise set $\theta^{t+1} = \theta^t$.
 6. Set $t = t + 1$. If $t < T$, the number of desired samples, return to step 2. Otherwise, stop.
-

This algorithm generates a Markov chain $(\theta^0, \theta^1, \dots, \theta^k, \dots)$, as the transition probabilities from θ^t to θ^{t+1} depend only on θ^t and not $(\theta^0, \dots, \theta^{t-1})$. Following a sufficient period (of, say, k steps), the chain approaches its stationary distribution and samples from the vector $(\theta^{k+1}, \dots, \theta^T)$ are samples from $p(\theta)$.

Metropolis-Hastings algorithm

Hastings (1970) generalized the Metropolis algorithm by using an asymmetric proposal distribution $q(\theta^t, \theta^*) \neq q(\theta^*, \theta^t)$. The difference in its implementation boils down to setting the acceptance probability for a candidate point as

$$\alpha = \min \left(\frac{p(\theta^*)q(\theta^*, \theta^t)}{p(\theta^t)q(\theta^t, \theta^*)}, 1 \right). \quad (3.2.1)$$

The other steps remain the same, resulting in the so-called Metropolis-Hastings algorithm.

Why does the Metropolis-Hastings algorithm work?

To prove that the Metropolis-Hastings sampling generates a Markov chain whose stationary distribution is that target distribution $p(\theta)$, it is sufficient to show that the Metropolis-Hastings transition probabilities satisfy the detailed balance equation 3.1.7 with $p(\theta)$.

Under the Metropolis-Hastings algorithm, we draw samples from $q(\theta, \theta^*)$ and then accept the move with probability $\alpha(\theta, \theta^*)$, so that the transition probability is given by

$$P(\theta, \theta^*) = q(\theta, \theta^*)\alpha(\theta, \theta^*) = q(\theta, \theta^*) \cdot \min \left(\frac{p(\theta^*)q(\theta^*, \theta)}{p(\theta)q(\theta, \theta^*)}, 1 \right) \quad (3.2.2)$$

The detailed balance can be verified as follows,

$$\begin{aligned} P(\theta, \theta^*)p(\theta) &= q(\theta, \theta^*) \cdot \min \left(\frac{p(\theta^*)q(\theta^*, \theta)}{p(\theta)q(\theta, \theta^*)}, 1 \right) \cdot p(\theta) \\ &= \min \{ p(\theta^*)q(\theta^*, \theta), p(\theta)q(\theta, \theta^*) \} \\ &= p(\theta^*)q(\theta^*, \theta) \cdot \min \left(\frac{p(\theta)q(\theta, \theta^*)}{p(\theta^*)q(\theta^*, \theta)}, 1 \right) \\ &= P(\theta^*, \theta)p(\theta^*). \end{aligned} \quad (3.2.3)$$

3.3 The Gibbs Sampler

The Gibbs sampler (introduced in the context of image processing by Geman and Geman (1984)), is a particular MCMC algorithm that has been found useful in many multidimensional problems. The key to the Gibbs sampler is that one only considers univariate conditional distributions — the distribution when all of the random variables but one are assigned fixed values. Such conditional distributions are far easier to simulate from than complex joint distributions and usually have simple forms. Thus, the sampler can be efficient when the parameters are not highly dependent on each other and the full conditional distributions are easy to sample from.

To introduce the Gibbs sampler, suppose the parameter vector Θ has been divided into d components or subvectors, $\Theta = (\theta_1, \dots, \theta_d)$. Each iteration of the Gibbs sampler cycles through the subvectors of Θ , drawing each subset conditional on the value of all the others. There are thus d steps in iteration t . Suppose we have chosen some initial state for the sampler. Then at each iteration t , an ordering of the d subvectors of Θ is chosen and, in turn, each θ_j^t is sampled from the conditional distribution given all the other components of Θ :

$$p(\theta_j | \Theta_{-j}^{t-1}),$$

where Θ_{-j}^{t-1} denotes all the components of Θ , except for θ_j , set at their current values:

$$\Theta_{-j}^{t-1} = (\theta_1^t, \dots, \theta_{j-1}^t, \theta_{j+1}^{t-1}, \dots, \theta_d^{t-1}).$$

Thus, each subvector θ_j is updated conditional on the latest values of the other components of Θ , which are the iteration t values for the components already updated and the iteration $t - 1$ values for the others.

Gibbs Sampling

1. Set $t = 0$, and choose an arbitrary initial value of $\Theta^0 = \{\theta_1^0, \theta_2^0, \dots, \theta_d^0\}$.
 2. Generate each component of Θ as follows:
 - Sample $\theta_1^{t+1} \sim p(\theta_1 | \theta_2^t, \theta_3^t, \dots, \theta_d^t)$.
 - Sample $\theta_2^{t+1} \sim p(\theta_2 | \theta_1^{t+1}, \theta_3^t, \dots, \theta_d^t)$.
 - $\vdots$
 - Sample $\theta_j^{t+1} \sim p(\theta_j | \theta_1^{t+1}, \dots, \theta_{j-1}^{t+1}, \theta_{j+1}^t, \dots, \theta_d^t)$.
 - $\vdots$
 - Sample $\theta_d^{t+1} \sim p(\theta_d | \theta_1^{t+1}, \theta_2^{t+1}, \dots, \theta_{d-1}^{t+1})$.
 3. Set $t = t + 1$. If $t < T$, the number of desired samples, return to step 2. Otherwise, stop.
-

For example, suppose we have a distribution $p(\theta_1, \theta_2, \theta_3)$ over three variables, and at step t of the algorithm we have selected values θ_1^t, θ_2^t and θ_3^t . We first update θ_1^t with a sample θ_1^{t+1} drawn from the conditional distribution

$$p(\theta_1 | \theta_2^t, \theta_3^t). \quad (3.3.1)$$

Next, we replace θ_2^t by a value θ_2^{t+1} obtained by sampling from the conditional distribution

$$p(\theta_2 | \theta_1^{t+1}, \theta_3^t), \quad (3.3.2)$$

so that the new value for θ_1 is used straight away in subsequent sampling steps. Then we replace θ_3 by a new value θ_3^{t+1} obtained by sampling from the conditional distribution

$$p(\theta_3 | \theta_1^{t+1}, \theta_2^{t+1}) \quad (3.3.3)$$

and so on, cycling through the three variables in turn. Repeat this process T times, generating a Gibbs sequence of length T . One iteration of all the univariate distributions at time t is often called a scan of the sampler.

Interpretation of the Gibbs sampler as a special case of the Metropolis-Hastings algorithm

Gibbs sampling can be viewed as a special case of the Metropolis-Hastings algorithm in the following way. We first define iteration $t + 1$ to consist of a series of d steps, with step j of iteration $t + 1$ corresponding to an update of the subvector θ_j conditional on all the other elements of Θ . Then the proposal distribution, $q_{j,t+1}(\cdot, \cdot)$, at step j of iteration $t + 1$ only jumps along the j th subvector, and does so with the conditional posterior density of θ_j given Θ_{-j}^t :

$$q_{j,t+1}(\Theta^t, \Theta^*) = \begin{cases} p(\theta_j^* | \Theta_{-j}^t) & \text{if } \Theta_{-j}^* = \Theta_{-j}^t \\ 0 & \text{otherwise} \end{cases}.$$

The only possible jumps are to parameter vectors Θ^* that match Θ^t on all components other than the j th. Under this proposal distribution, the ratio 3.2.1 at the j th step of iteration $t + 1$ is

$$\begin{aligned}
\alpha &= \min \left(\frac{p(\Theta^*) \cdot q_{j,t+1}(\Theta^*, \Theta^t)}{p(\Theta^t) \cdot q_{j,t+1}(\Theta^t, \Theta^*)}, 1 \right) \\
&= \min \left(\frac{p(\Theta^*) \cdot p(\theta_j^t | \Theta_{-j}^*)}{p(\Theta^t) \cdot p(\theta_j^* | \Theta_{-j}^t)}, 1 \right) \\
&= \min \left(\frac{p(\theta_j^* | \Theta_{-j}^*) \cdot p(\Theta_{-j}^*) \cdot p(\theta_j^t | \Theta_{-j}^*)}{p(\theta_j^t | \Theta_{-j}^t) \cdot p(\Theta_{-j}^t) \cdot p(\theta_j^* | \Theta_{-j}^t)}, 1 \right) \quad (3.3.4) \\
&= \min \left(\frac{p(\Theta_{-j}^*)}{p(\Theta_{-j}^t)}, 1 \right) \\
&= 1,
\end{aligned}$$

and thus every jump is accepted. The second line above follows from the first because, under this jumping rule, Θ^* differs from Θ^t only in the j th component. The third and fourth line follow from the second by applying the rules of conditional probability to $\Theta = (\theta_j, \Theta_{-j})$ and noting that $\Theta_{-j}^* = \Theta_{-j}^t$.

3.4 Burning-in and Choice of Proposal Distribution

In any practical application of MCMC, there are a number of important decisions to be made. These include the number of initial burn-in iterations discarded and the proposal distribution.

Burn-in refers to the practice of discarding an initial portion of a Markov chain sample so that the effect of initial values on posterior inference is minimized. Typically the first 1000 to 5000 elements are thrown out, and then one of many convergence tests (Gelman and Rubin, 1992; Geweke, 1992; Raftery and Lewis, 1992) is used to assess whether stationarity has indeed been reached.

The specific choice of proposal density q can have a marked effect on the performance of the sampler. For continuous state spaces, a common choice is a Gaussian centered on the current value, leading to an important trade-off in determining the variance, σ_q^2 , of the proposal density. If the variance parameter is too large, moves are large, but rejection rate will be high. This leads to high autocorrelation and very poor mixing, requiring much longer chains. However, if the standard deviation is small, then the proportion of accepted transitions will be high, but the posterior density will take longer to explore. Roberts and Rosenthal (2001) suggest acceptance rates should be between 0.2 and 0.4.

3.5 Summary of the Chapter

This chapter provided a brief introduction to MCMC. First, we presented the essential theory of Markov chains for the development of Monte Carlo methods. We described two general MCMC algorithms—Metropolis and Gibbs sampling. These methods form the basis for the posterior distribution inferences which will be discussed in the next chapter. After that, we also discussed burning-in the sampler and how to choose a proposal distribution. In the next chapter, we introduce our newly developed Bayesian hierarchical model for cancer outlier expression detection.

Chapter 4

BOPA: A Five-state Bayesian Hierarchical Model

From the observations of previous chapters, we notice that there are many challenges in the microarray outlier expression detection. As we discussed in Chapter 2, a microarray data set normally has a small number of samples due to experimental costs. Thus, the number of measurements per gene is usually quite small compared to the number of genes. This has made some traditional frequentist methods inefficient when detecting those differentially expressed genes. On the other hand, many researchers have realized that in many cancer studies, some genes show a system increase or decrease in expression, but only for a small number of disease samples. Existing methods are hampered by the small proportion of differentially expressed and will not have the sensitivity to identify the genes of interest.

In this chapter, we propose a Bayesian hierarchical model which contains five states, based on Newton et al. (2004). The new method contains features that (1) make most use of the information across different genes and samples, (2) take into account partial differentially expressed patterns

34 Chapter 4. BOPA: A Five-state Bayesian Hierarchical Model

and (3) reduce the complexity of model fitting by introducing latent variables. We organize this chapter as follows. We begin by presenting a novel Bayesian hierarchical model (BOPA) in Section 4.1. After developing the model, we give the various marginal and conditional distributions induced by the BOPA model in Section 4.2. We implement our hierarchical model fitting using MCMC in Section 4.3. We then discuss how to find DE genes based on the samples from posterior distributions thus obtained in Section 4.4. This chapter is concluded by a short summary in Section 4.5.

4.1 The Model

We consider the simple two-group microarray data for detecting cancer genes. We assume there are m normal samples and n cancer samples. The gene expressions for the normal samples are denoted by $\mathbf{X} = \{x_{ij}\}$ for genes $i = 1, 2, \dots, G$ and samples $j = 1, 2, \dots, m$, while $\mathbf{Y} = \{y_{ij}\}$ denotes the expressions for cancer samples with $i = 1, 2, \dots, G$ and $j = 1, 2, \dots, n$. We model these expression values as Gamma random variables with gene-specific means:

$$x_{ij} \sim \text{Gamma}(a_j, a_j \lambda_{ij}^x), \text{ with } E(x_{ij}) = 1/\lambda_{ij}^x,$$

$$y_{ij} \sim \text{Gamma}(b_j, b_j \lambda_{ij}^y), \text{ with } E(y_{ij}) = 1/\lambda_{ij}^y.$$

We assume expressions for normal samples are homogeneous so that $\lambda_{ij}^x \equiv \lambda_{i1}$ for $j = 1, \dots, m$, but the true expressions for different cancer samples could be different. Gamma models for observed expression values have been used in many previous studies on parametric modeling of differential expression (Kendzioriski et al., 2003; Newton et al., 2004; Jensen et al., 2009) and the implicit assumption of constant coefficient of variation in

the Gamma model has been checked for many real-life datasets. In the above we have used sample-specific shape parameters $\mathbf{a} = (a_1, \dots, a_m)$ and $\mathbf{b} = (b_1, \dots, b_n)$ in the Gamma distributions. A more parsimonious choice may be to constrain the shape parameters to be experimental-condition-specific as in Jensen et al. (2009). In typical genomic problems, with a large number of genes assayed for each sample, we usually find that we can estimate these shape parameters rather accurately and thus we use sample-specific shape parameters for full flexibility.

On the next level, marginally, the latent expression parameters λ_{ij}^x and λ_{ij}^y follows a Gamma distribution

$$\lambda_{ij}^x, \lambda_{ij}^y \sim \text{Gamma}(a_0, a_0 x_0).$$

This assumption allows for information sharing across genes and samples. Were we instead to fix λ_{ij}^x s and λ_{ij}^y s as constants, there would be no information sharing and potentially a loss in efficiency. The above assumption only specifies the marginal distribution of these parameters. One constraint is already mentioned above: λ_{ij}^x are all the same for $j = 1, \dots, m$. Other constraints on these parameters depend on another latent gene-specific state variable Z_i taking values in $\mathbf{S} = \{0, 1, 2, 3, 4\}$ defined as in Table 4.1 with additional parameters $\mathbf{p} = \{p_0, \dots, p_4\}$ specifying the state probabilities as $\Pr(Z_i = s) = p_s, s \in \mathbf{S}$, independently for each gene. The different parameters λ_{i1} and λ_{i2} are generated independently up to the constraints above. For example, if $Z_i = 1$ then the joint density of $(\lambda_{i1}, \lambda_{i2})$ is $2\pi(\lambda_{i1})\pi(\lambda_{i2})1\{\lambda_{i1} > \lambda_{i2}\}$, where π is the $\text{Gamma}(a_0, a_0 x_0)$ density. Thus the joint distribution of $(\lambda_{i1}, \lambda_{i2})$ is a mixture with five components.

To fully specify the distribution of latent parameters λ_{ij}^y when $Z_i = 2$ or 4, we need to introduce a gene-specific parameter $0 < \pi_i < 1$. Conditional

36 Chapter 4. BOPA: A Five-state Bayesian Hierarchical Model

Table 4.1: Definition of latent gene-specific state variable Z_i ($i = 1, \dots, G$). Each Z_i takes value in $\mathcal{S} = \{0, 1, 2, 3, 4\}$, representing the five different gene expressed states for tumor samples. We use λ_{i1} and λ_{i2} to denote the inverse of the mean value of genes from normal samples and outlier tumor samples respectively.

Z_i	Mean Components	Gene Expression Patterns for Tumor Samples
0	$\lambda_{ij}^y = \lambda_{i2} = \lambda_{i1}$	non-differential expression
1	$\lambda_{ij}^y = \lambda_{i2} < \lambda_{i1}$	up-regulated expression
2	$\lambda_{ij}^y = \lambda_{i2}$ or λ_{i1} with $\lambda_{i2} < \lambda_{i1}$	partially up-regulated expression
3	$\lambda_{ij}^y = \lambda_{i2} > \lambda_{i1}$	down-regulated expression
4	$\lambda_{ij}^y = \lambda_{i2}$ or λ_{i1} with $\lambda_{i2} > \lambda_{i1}$	partially down-regulated expression

on the event $Z_i = 2$ or 4, for each tumor sample independently, we assume $\lambda_{ij}^y = \lambda_{i2}$ with probability π_i and $\lambda_{ij}^y = \lambda_{i1}$ with probability $1 - \pi_i$. We assign a prior distribution $\text{Beta}(u, v)$ for π_i independently for each gene. We use $u = 2$ and $v = 2$ as a weak prior centered at $1/2$. We have also tried other priors such as $\text{Beta}(1, 1)$ and $\text{Beta}(5, 5)$ in our simulations and the results are quite robust to these different choices.

Finally, we use the following non-informative prior distribution for the unknown parameters $\Theta = \{\mathbf{a}, \mathbf{b}, a_0, x_0, \mathbf{p}\}$:

- $a_j, b_j, a_0, x_0 \sim \text{Uniform}(0, B)$ with a large constant B ($B = 1000$ in our current implementation),
- $\mathbf{p} = (p_0, \dots, p_4) \sim \text{Dirichlet}(d_0, \dots, d_4)$, where we use a small number $d_s = 0.1$ ($s = 0, 1, \dots, 4$) in our implementation.

When Z_i is constrained to only take values in $\{0, 1, 3\}$, we in effect revert back to the parametric model presented in Newton et al. (2004). They used an EM algorithm in which the latent parameters λ_{ij} can be integrated out of the model. Fitting of our model is complicated by the additional states associated with partial differential expression. In particular, when $Z_i = 2$ or 4, it is difficult to directly integrate out λ_{ij} since it is unknown whether the gene expression for a particular tumor sample has a differential expression. Thus we introduce the allocation vector $\mathbf{g}_i = (g_{i1}, \dots, g_{in})$ for each gene i . We distinguish the cases $Z_i = 2$ (partially up-regulated expression) and $Z_i = 4$ (partially down-regulated expression):

- When $Z_i = 2$, we set $g_{ij} = 1$ if $\lambda_{ij}^y = \lambda_{i1}$ (non-outlier) and $g_{ij} = 2$ if $\lambda_{ij}^y = \lambda_{i2}$ (up-regulated expression).
- When $Z_i = 4$, we set $g_{ij} = 1$ if $\lambda_{ij}^y = \lambda_{i1}$ (non-outlier) and $g_{ij} = 0$ if $\lambda_{ij}^y = \lambda_{i2}$ (down-regulated expression).

By the description of the generative model for λ_{ij}^y above, we have

$$\begin{aligned} p(\mathbf{g}_i | Z_i = 2 \text{ or } 4) &= \int_0^1 \pi_i^{n_{i2}} (1 - \pi_i)^{n_{i1}} p(\pi_i) d\pi_i \\ &= \int_0^1 \pi_i^{n_{i2}} (1 - \pi_i)^{n_{i1}} \frac{1}{\beta(u, v)} \pi_i^{u-1} (1 - \pi_i)^{v-1} d\pi_i \quad (4.1.1) \\ &= \frac{\beta(n_{i2} + u, n_{i1} + v)}{\beta(u, v)}, \end{aligned}$$

where n_{i1} is the number of non-differentially expressed tumor samples, n_{i2} is the number of differentially expressed tumor samples (note n_{i1} and n_{i2} implicitly depend on $\mathbf{g}_i$), with $n_{i1} + n_{i2} = n$, and $p(\pi_i)$ is the Beta(u, v) prior density. This integral can be easily evaluated to get a closed-form expression involving Beta functions.

For convenience, we also define an allocation vector for the other three states:

38 Chapter 4. BOPA: A Five-state Bayesian Hierarchical Model

- $Z_i = 0 \Rightarrow g_{ij} = 1, j = 1, \dots, n,$
- $Z_i = 1 \Rightarrow g_{ij} = 2, j = 1, \dots, n,$
- $Z_i = 3 \Rightarrow g_{ij} = 0, j = 1, \dots, n,$

with $p(\mathbf{g}_i|Z_i) = 1$ if and only if $\mathbf{g}_i$ satisfies the constraints above depending on the value of Z_i .

With the aid of latent variables $(Z_i, \mathbf{g}_i)$ defined above, we can reduce the complexity of model fitting by integrating out the latent parameters $\lambda_{ij}^x, \lambda_{ij}^y$ in our hierarchical model resulting in the following likelihood:

$$p(\mathbf{X}, \mathbf{Y}, \mathbf{Z}, \mathbf{g}|\Theta) = \prod_{i=1}^G p(\mathbf{x}_i, \mathbf{y}_i|\mathbf{g}_i, Z_i, \Theta)p(\mathbf{g}_i|Z_i)p(Z_i|\Theta) \quad (4.1.2)$$

where $\mathbf{x}_i, \mathbf{y}_i$ are all observed expression values for gene i . Thanks to the introduction of the allocation vector, the collapsed distribution $p(\mathbf{x}_i, \mathbf{y}_i|\mathbf{g}_i, Z_i, \Theta)$ can now be evaluated in closed form in the same way as illustrated in the appendix of Newton et al. (2004). We also give out details of the induction in the next section. Note that by introducing the allocation vector and integrating out λ_{ij} , reversible jump MCMC is not required for transitioning between different states. We give out a graphical representation for our proposed hierarchical Bayesian model in Figure 4.1.

4.2 Marginal and Conditional Distributions

In this section we present various marginal and conditional distributions which will be used in the later MCMC inference. To simplify the derivation, we suppress in the notation the gene subscript i . Also, we suppose $b_0 = a_0$ and $y_0 = x_0$.

Figure 4.1: The proposed hierarchical Bayesian model is represented as directed graphical model.

First, the posterior distribution of mean component λ_1 given gene expressions from normal samples is:

$$\begin{aligned}
 p(\lambda_1 | \mathbf{x}) &= \frac{p(\mathbf{x} | \lambda_1) p(\lambda_1)}{p(\mathbf{x})} \\
 &\propto p(\mathbf{x} | \lambda_1) p(\lambda_1) \\
 &= \frac{(a_0 x_0)^{a_0} (\lambda_1)^{a_0 - 1} \exp(-a_0 x_0 \lambda_1)}{\Gamma(a_0)} \prod_{j=1}^m \frac{(a_j \lambda_1)^{a_j} (x_j)^{a_j - 1} \exp(-a_j \lambda_1 x_j)}{\Gamma(a_j)} \\
 &\propto \frac{\left(\sum_{j=0}^m a_j x_j \right)^{\sum_{j=0}^m a_j} \lambda_1^{\sum_{j=0}^m a_j - 1} \exp[-\lambda_1 (\sum_{j=0}^m a_j x_j)]}{\Gamma(\sum_{j=0}^m a_j)}
 \end{aligned} \tag{4.2.1}$$

40 Chapter 4. BOPA: A Five-state Bayesian Hierarchical Model

Thus,

$$\lambda_1 | \mathbf{x} \sim \text{Gamma}\left(\sum_{j=0}^m a_j, \sum_{j=0}^m a_j x_j\right).$$

Similarly, we obtain the posterior distribution of mean component λ_2 given gene expressions from cancer samples as:

$$\lambda_2 | \mathbf{y} \sim \text{Gamma}\left(\sum_{j=0}^n b_j, \sum_{j=0}^n b_j y_j\right).$$

The predictive distribution of measurements on a gene given the hyperparameters a_0 and x_0 is:

$$\begin{aligned} p(\mathbf{x} | a_0, x_0) &= \int_0^\infty p(\mathbf{x} | \lambda_1) p(\lambda_1 | a_0, x_0) d(\lambda_1) \\ &= \int_0^\infty \frac{(a_0 x_0)^{a_0} (\lambda_1)^{a_0-1} \exp(-a_0 x_0 \lambda_1)}{\Gamma(a_0)} \\ &\quad \prod_{j=1}^m \frac{(a_j \lambda_1)^{a_j} (x_j)^{a_j-1} \exp(-a_j \lambda_1 x_j)}{\Gamma(a_j)} d(\lambda_1) \\ &= \frac{x_0 \Gamma(\sum_{j=0}^m a_j)}{(\sum_{j=0}^m a_j x_j)^{\sum_{j=0}^m a_j}} \prod_{j=0}^m \left[\frac{a_j^{a_j} x_j^{a_j-1}}{\Gamma(a_j)} \right] \\ &\quad \int_0^\infty \frac{(\sum_{j=0}^m a_j x_j)^{\sum_{j=0}^m a_j} \lambda_1^{\sum_{j=0}^m a_j-1} \exp[-\lambda_1 (\sum_{j=0}^m a_j x_j)]}{\Gamma(\sum_{j=0}^m a_j)} d(\lambda_1) \\ &= \frac{x_0 \Gamma(\sum_{j=0}^m a_j)}{(\sum_{j=0}^m a_j x_j)^{\sum_{j=0}^m a_j}} \prod_{j=0}^m \left[\frac{a_j^{a_j} x_j^{a_j-1}}{\Gamma(a_j)} \right]. \end{aligned} \tag{4.2.2}$$

The likelihood function of measurement of a gene for which the cancer

samples are over expressed is

$$\begin{aligned}
 p(\mathbf{x}, \mathbf{y} | \lambda_1 > \lambda_2) &= \int_0^\infty \int_0^\infty p(\mathbf{x} | \lambda_1) p(\mathbf{y} | \lambda_2) 2p(\lambda_1) p(\lambda_2) 1[\lambda_1 > \lambda_2] d\lambda_1 d\lambda_2 \\
 &= \int_0^\infty 2p(\mathbf{y} | \lambda_2) p(\lambda_2) I(\lambda_2) d\lambda_2,
 \end{aligned}
 \tag{4.2.3}$$

where

$$\begin{aligned}
 I(\lambda_2) &= \int_{\lambda_2}^\infty p(\mathbf{x} | \lambda_1) p(\lambda_1) d\lambda_1 \\
 &= \int_{\lambda_2}^\infty \frac{(a_0 x_0)^{a_0} (\lambda_1)^{a_0-1} \exp(-a_0 x_0 \lambda_1)}{\Gamma(a_0)} \prod_{j=1}^m \frac{(a_j \lambda_1)^{a_j} (x_j)^{a_j-1} \exp(-a_j \lambda_1 x_j)}{\Gamma(a_j)} d(\lambda_1) \\
 &= h(\mathbf{x}) \int_{\lambda_2}^\infty \frac{(\sum_{j=0}^m a_j x_j)^{\sum_{j=0}^m a_j} \lambda_1^{\sum_{j=0}^m a_j-1} \exp[-\lambda_1 (\sum_{j=0}^m a_j x_j)]}{\Gamma(\sum_{j=0}^m a_j)} d(\lambda_1) \\
 &= h(\mathbf{x}) \int_{\lambda_2 (\sum_{j=0}^m a_j x_j)}^\infty \frac{\psi_1^{\sum_{j=0}^m a_j-1} \exp[-\psi_1]}{\Gamma(\sum_{j=0}^m a_j)} d(\psi_1),
 \end{aligned}
 \tag{4.2.4}$$

and $h(\mathbf{x})$ is just $p(\mathbf{x} | a_0, x_0)$ in 4.2.2, and $\psi_1 = \lambda_1 (\sum_{j=0}^m a_j x_j)$.

42 Chapter 4. BOPA: A Five-state Bayesian Hierarchical Model

Thus,

$$\begin{aligned}
 p(\mathbf{x}, \mathbf{y} | \lambda_1 > \lambda_2) &= 2h(\mathbf{x}) \int_0^\infty \int_{\lambda_2(\sum_{j=0}^m a_j x_j)}^\infty p(\mathbf{y} | \lambda_2) p(\lambda_2) \frac{\psi_1^{\sum_{j=0}^m a_j - 1} \exp(-\psi_1)}{\Gamma(\sum_{j=0}^m a_j)} d(\psi_1) d\lambda_2 \\
 &= 2h(\mathbf{x}) h(\mathbf{y}) \int_0^\infty \int_0^{\psi_1 / (\sum_{j=0}^m a_j x_j)} p(\psi_1) \frac{p(\mathbf{y} | \lambda_2) p(\lambda_2)}{h(\mathbf{y})} d\lambda_2 d(\psi_1) \\
 &= 2h(\mathbf{x}) h(\mathbf{y}) \int_0^\infty p(\psi_1) \left(\int_0^{\psi_1 / (\sum_{j=0}^m a_j x_j)} p(\lambda_2 | \mathbf{y}) d\lambda_2 \right) d(\psi_1),
 \end{aligned} \tag{4.2.5}$$

where $p(\psi_1)$ is a Gamma density function with shape $\sum_{j=0}^m a_j$ and rate 1, and $p(\lambda_2 | \mathbf{y})$ is the posterior for λ_2 , which is also a Gamma distribution with shape $\sum_{j=0}^n b_j$. Let $\psi_2 = \lambda_2(\sum_{j=0}^n b_j y_j)$, then $\psi_2 \sim \text{Gamma}(\sum_{j=0}^n b_j, 1)$. Adjusting rates by a change of variable, we have

$$\begin{aligned}
 p(\mathbf{x}, \mathbf{y} | \lambda_1 > \lambda_2) &= 2h(\mathbf{x}) h(\mathbf{y}) \int_0^\infty \int_0^{r\psi_1} p(\psi_1) p(\psi_2) d\psi_2 d\psi_1 \\
 &= 2h(\mathbf{x}) h(\mathbf{y}) P[\psi_1 > \psi_2 / r] \\
 &= 2h(\mathbf{x}) h(\mathbf{y}) P[B > (1 - B)/r] \\
 &= 2h(\mathbf{x}) h(\mathbf{y}) P[B > 1/(1 + r)],
 \end{aligned} \tag{4.2.6}$$

where ψ_1 and ψ_2 are i.i.d. Gamma variables with shape $\sum_{j=0}^m a_j$ and $\sum_{j=0}^n b_j$, and

$$r = (\sum_{j=0}^n b_j y_j) / (\sum_{j=0}^m a_j x_j)$$

and $B = \frac{\psi_1}{\psi_1 + \psi_2} \sim \text{Beta}(\sum_{j=0}^m a_j, \sum_{j=0}^n b_j)$. Similarly we can derive $p(\mathbf{x}, \mathbf{y} | \lambda_1 < \lambda_2)$.

4.3 MCMC Sampler

Combining the likelihood (4.1.2) with the prior on the parameters, we intend to explore the posterior distribution $p(\mathbf{Z}, \mathbf{g}, \Theta | \mathbf{X}, \mathbf{Y}) \propto p(\mathbf{X}, \mathbf{Y}, \mathbf{Z}, \mathbf{g} | \Theta) p(\Theta)$. We implement our hierarchical model fitting using Markov chain Monte Carlo algorithm to be discussed in this section.

We alternately sample from the posterior of $(\mathbf{Z}, \mathbf{g})$ given current values of the parameters Θ and sample Θ given current values of latent variables $(\mathbf{Z}, \mathbf{g})$. Detailed formula for each step is presented in the following. Sampling of Θ can be achieved based on simple full conditional distribution for sampling $\mathbf{p}$, and based on Metropolis steps for sampling (a_0, x_0, a_j, b_j) . Given Θ , sampling of latent variables can be done one gene at a time as is obvious from the product form of the likelihood (4.1.2).

First, we discuss the algorithm details for the sampling of latent variables for a fixed gene i . Suppose the current latent variables are $(Z_i, \mathbf{g}_i)$. We use the Metropolis-Hastings algorithm to propose a new set of latent variables $(Z_i^*, \mathbf{g}_i^*)$. First we need to consider the proposal distributions. For the proposal of Z_i^* , we use a simple state transition system as shown in Figure 4.2. For example, when currently $Z_i = 0$, we propose to move to any of the other four states with equal probability. While if $Z_i = 1$, we propose to move to either $Z_i^* = 0$ or $Z_i^* = 2$. These transition probabilities are denoted by p_{s,s^*} when transitioning from $Z_i = s$ to $Z_i^* = s^*$. We do not consider transitions between more distinct states such as from $Z_i = 1$ to $Z_i^* = 3$ (from up-regulation to down-regulation) since these types of transitions represent large jumps and will rarely be accepted and thus represent a waste of computation time. Constraining possible transitions between states also makes implementation an easier task. When moving between states 0, 1 and

Figure 4.2: Possible transitions of the MCMC sampler among the five states. State 0 represents the non-DE state. States 1 and 3 represent up-regulated and down-regulated expression patterns respectively, while states 2 and 4 represent partially up-regulated and partially down-regulated expression patterns respectively.

3, no proposal of $\mathbf{g}_i^*$ is required since it is entirely determined by the state as defined in Section 4.1. Complications occur when a state representing partially differential expression is involved. We illustrate our algorithm with the proposed transition $Z_i = 0 \rightarrow Z_i^* = 2$. Implicitly, we have $\mathbf{g}_i = (1, 1, \dots, 1)$.

In general, denoting the proposal distribution of $\mathbf{g}_i^*$ by $Q(\mathbf{g}_i^*|Z_i^*)$ which might depend on other variables besides the proposed state Z_i^* , the acceptance of the proposal is based on the value of the ratio:

$$\begin{aligned}
 \alpha &= \frac{p_{2,0}Q(\mathbf{g}_i|Z_i = 0)p(Z_i^* = 2, \mathbf{g}_i^*|\mathbf{x}_i, \mathbf{y}_i, \Theta)}{p_{0,2}Q(\mathbf{g}_i^*|Z_i^* = 2)p(Z_i = 0, \mathbf{g}_i|\mathbf{x}_i, \mathbf{y}_i, \Theta)} \\
 &= \frac{p_{2,0} \cdot p_2 \cdot p(\mathbf{g}_i^*|Z_i^* = 2) \cdot p(\mathbf{x}_i, \mathbf{y}_i|Z_i^* = 2, \mathbf{g}_i^*, \Theta)}{p_{0,2} \cdot Q(\mathbf{g}_i^*|Z_i^* = 2) \cdot p_0 \cdot p(\mathbf{x}_i, \mathbf{y}_i|Z_i = 0, \mathbf{g}_i, \Theta)},
 \end{aligned} \tag{4.3.1}$$

where, as a reminder, $p(\mathbf{g}_i^*|Z_i^* = 2)$ is the prior probability of the allocation

vector defined by the integral (4.1.1). Given the proposed $\mathbf{g}_i^*$, the probability $p(\mathbf{x}_i, \mathbf{y}_i | Z_i^* = 2, \mathbf{g}_i^*, \Theta)$ can be computed in closed form as discussed in 4.2.6.

Suppose one uses the prior (4.1.1) as the proposal distribution $Q(\mathbf{g}_i^* | Z_i^* = 2)$, the prior distribution and proposal distribution of $\mathbf{g}_i^*$ in the numerator and the denominator conveniently cancel each other. However, we do not expect this simple proposal to perform well, because the allocation vector is set randomly without reference to the observed data, and the proposal is unlikely to be accepted. In other words, the proposal $\mathbf{g}_i^*$ from the prior distribution is likely to force the term $p(\mathbf{x}_i, \mathbf{y}_i | Z_i^* = 2, \mathbf{g}_i^*, \Theta)$ to be very small even with a moderate number of tumor samples n .

Instead, we adopt the idea of Jain and Neal (2004) proposed in the context of Dirichlet process clustering and use a proposal distribution taking into account the observed expression values. Starting from an arbitrary fixed allocation vector $\mathbf{g}_i$ on tumor samples, which might be different from the current allocation vector, we use a simple Gibbs scan over components of $\mathbf{g}_i$ and for each component j , draw a new value for g_{ij} from the conditional distribution

$$p(g_{ij} = c | Z_i = 2) \propto \int p(y_{ij} | g_{ij}, \lambda_{ic}, b_j) p_{-j,c}(\lambda_{ic}) d\lambda_{ic}, \quad c = 1, 2,$$

where $p_{-j,c}(\cdot)$ is the posterior distribution of λ_{ic} based on all observed expressions for which $g_{ik} = c, k \neq j$, and when $c = 1$ these also include the observed expressions on normal samples. Thus, using the above probability as the proposed update in turn for each component of $\mathbf{g}_i$, the proposal probability for a full sequential sampling scan is a product of the conditional probabilities of each individual update. In our algorithm, this product is the proposal probability, $Q(\mathbf{g}_i^* | Z_i^* = 2)$.

However, when a single sequential Gibbs scan is used, the proposed

46 Chapter 4. BOPA: A Five-state Bayesian Hierarchical Model

allocation vector may still not be sensible. To improve the efficiency of the algorithm, we perform K iterations of Gibbs sampling ($K = 3$ in our implementation). When calculating the proposal probability, we only need to use the product of the probabilities from the last scan instead of taking product probabilities from all K Gibbs scans. The reason is the same as in Jain and Neal (2004): the calculation of acceptance is valid starting from any fixed allocation vector $\mathbf{g}_i$, thus should be valid even for a random starting allocation vector. We will discuss in more detail the effects of using multiple Gibbs scans on posterior samples in Appendix A.

Next, we outline the steps of sampling from the posterior distribution of $Z, \mathbf{g}, \mathbf{a}, \mathbf{b}, a_0, x_0, \mathbf{p}$:

1. For $i = 1, \dots, G$, use a Metropolis-Hastings algorithm to sample $Z_i, \mathbf{g}_i$ conditional on all other variables. (1) Propose a move $Z_i \rightarrow Z_i^*$ following a uniform distribution on all moves obeying the constraints represented in Figure 4.2. (2) Propose an allocation vector $\mathbf{g}_i^*$. When Z_i^* is 0, 1, 3, this step is trivial. Otherwise use the sequential Gibbs scan to update $\mathbf{g}_i^*$ as mentioned before. (3) Accept the proposed move with probability

$$\begin{aligned} \alpha &= \frac{p_{Z_i^*, Z_i} Q(\mathbf{g}_i | Z_i) p(Z_i^*, \mathbf{g}_i^* | \mathbf{x}_i, \mathbf{y}_i, \Theta)}{p_{Z_i, Z_i^*} Q(\mathbf{g}_i^* | Z_i^*) p(Z_i, \mathbf{g}_i | \mathbf{x}_i, \mathbf{y}_i, \Theta)} \\ &= \frac{p_{Z_i^*, Z_i} \cdot Q(\mathbf{g}_i | Z_i) \cdot p_{Z_i^*} \cdot p(\mathbf{g}_i^* | Z_i^*) \cdot p(\mathbf{x}_i, \mathbf{y}_i | Z_i^*, \mathbf{g}_i^*, \Theta)}{p_{Z_i, Z_i^*} \cdot Q(\mathbf{g}_i^* | Z_i^*) \cdot p_{Z_i} \cdot p(\mathbf{g}_i | Z_i) \cdot p(\mathbf{x}_i, \mathbf{y}_i | Z_i, \mathbf{g}_i, \Theta)} \end{aligned} \quad (4.3.2)$$

where Q is the proposal distribution for the allocation vector.

2. sample $\mathbf{p} = (p_0, \dots, p_4)$ from $\text{Dirichlet}(N_0 + d_0, \dots, N_4 + d_4)$, where N_s ($s \in \mathbf{S}$) is the number of genes that has been assigned to state s , and $d_s = 0.1$ is the parameter for the prior Dirichlet distribution.

3. Sample each of the shape parameters a_j or b_j using a Metropolis step. To fix ideas, we consider sampling of b_1 , the shape parameter associated with the first tumor sample. In order to do that, we first sample latent parameters $\lambda_{i1}^y, i = 1, \dots, G$ from the conditional distribution $p(\lambda_{i1}^y | \mathbf{g}_i, \mathbf{X}, \mathbf{Y}, \Theta)$. Using a Metropolis step, we propose a move $b_1 \rightarrow b_1^* = b_1 e^\epsilon, \epsilon \sim N(0, \sigma_{b_1}^2)$ which is accepted with probability

$$\min \left[\prod_{i=1}^G p(y_{i1} | b_1^*, \lambda_{i1}^y) 1\{b_1^* \leq B\} / \prod_{i=1}^G p(y_{i1} | b_1, \lambda_{i1}^y), 1 \right]$$

4. Sample a_0 from the conditional distribution given latent λ_{ij} . Denoting all distinct sampled values λ_{ij} by $\mathbf{\Lambda}$. Using a Metropolis step, we propose a move $a_0 \rightarrow a_0^* = a_0 e^\epsilon, \epsilon \sim N(0, \sigma_{a_0}^2)$, which is accepted with probability

$$\min \left[\prod_{i=1}^G p(\mathbf{\Lambda} | a_0^*, x_0) 1\{a_0^* \leq B\} / \prod_{i=1}^G p(\mathbf{\Lambda} | a_0, x_0), 1 \right]$$

Similarly we sample x_0 from its conditional distribution given $\mathbf{\Lambda}$.

Note that for our sampler, although we integrate out λ_{ij} when sampling latent variables $(Z_i, \mathbf{g}_i)$, these variables are required in some other sampling steps. Thus our sampler is partially collapsed in the sense of van Dyk and Park (2008).

4.4 Gene Selection

After fitting our BOPA model, gene-specific inference is primarily based on the posterior distribution of Z_i . Our MCMC algorithm provides easy estimates for different quantities and parameters in our model. For example,

48 Chapter 4. BOPA: A Five-state Bayesian Hierarchical Model

one can assign to each gene a state that maximizes the posterior probability (which can be estimated from sampling frequencies for each state):

$$s_i = \arg \max_{s \in \mathbf{S}} p(Z_i = s | \mathbf{X}, \mathbf{Y}).$$

The full posterior distribution of Z_i actually tells us all the uncertainties associated with each gene. If a short list of DE genes are desired, one possible decision rule is to declare a gene to be differentially expressed if $P(Z_i \neq 0 | \mathbf{X}, \mathbf{Y}) \geq C$ for some threshold $0 < C < 1$, and Newton et al. (2004) suggested to estimate the false discovery rate (FDR) with

$$FDR(C) = \frac{\sum_i p(Z_i = 0 | \mathbf{X}, \mathbf{Y}) 1\{p(Z_i \neq 0 | \mathbf{X}, \mathbf{Y}) \geq C\}}{\sum_i 1\{p(Z_i \neq 0 | \mathbf{X}, \mathbf{Y}) \geq C\}}. \quad (4.4.1)$$

Conversely, in order to control the false discovery rate at a certain level, we can choose the smallest C on a fine grid of threshold values with estimated false discovery rate below that level.

In a Bayesian framework, thresholding of posterior state probability actually is the optimal decision rule for minimizing a loss function to trade off false discovery rate and false negative rate (Müller et al., 2007). This decision-theoretic perspective can lead to other loss functions that might be more appropriate in genomic studies. For example, the loss incurred from a false negative should depend on the magnitude of the differences in expression between the two experimental conditions, with a false negative on a significantly differentially expressed gene being of great concern. These considerations would lead to the following loss function

$$L(\delta_i, \eta_i) = \delta_i + k(1 - \delta_i)\eta_i,$$

where the decision is $\delta_i = 0$ or 1 if we decide the gene is non-differentially expressed and differentially expressed respectively, $\eta_i = 1$ if i -th gene is

non-differentially expressed and $\eta_i > 1$ is the ratio between the higher latent mean and the lower latent mean when it is differentially expressed, $k > 0$ is a constant that determines the trade off between false positive and false negative. More complicated loss functions can be constructed to take into account the loss involved when the incorrect non-null state is assigned to a gene. Since these types of loss functions typically involve a more complicated task of elicitation of loss parameters such as k , they are thus not used in our implementation.

4.5 Summary of the Chapter

In this chapter, we proposed a novel Bayesian hierarchical model for detecting DE genes based on the Gamma distribution. This model allows one to capture those partially differential expressed genes. This is achieved by extending the Gamma-Gamma model of Newton et al. (2004) while still maintaining closed-form inference. Also, we discussed how to use MCMC to estimate the parameters. Finally we described gene selection based on the posterior probabilities of differential expression calculated from the MCMC sampler. In the next chapter, we will investigate the performance of our methodology through simulation studies and real data analysis.

Chapter 5

Simulations and Real Data Analysis

We now evaluate the performance of our BOPA method through many simulated examples and two real-life datasets. We first use a series of simulations to study the performance of our methodology in Section 5.1. Next, we conduct experiments on two publicly available microarray datasets in Section 5.2. Throughout this chapter, we compare performances to the six methods presented in Chapter 2. All algorithms were implemented in the R programming language.

5.1 Simulations

We carried out some simulation studies to assess the detection performance of our methodology. For each simulated dataset, we generated $G = 1000$ genes, with $m = n = 4, 10$, or 24 replicates and a Gamma observation component with shape parameters generated independently for each sample $a_j, b_j \sim \text{Uniform}(10, 50), j = 1, \dots, m(n)$. For the distribution of the gene-specific parameters λ_{i1} and λ_{i2} , we adopted the same three scenarios as in Newton et al. (2004):

Scenario I Gamma with $a_0 = 0.5$ and $a_0 x_0 = 0.25$.

Scenario II Uniform on $5 \leq -\log_2 \sqrt{\lambda_{i1}\lambda_{i2}} \leq 11$ and $-2 \leq \log_2(\lambda_{i2}/\lambda_{i1}) \leq 2$ for DE genes and uniform on $5 \leq -\log_2 \lambda_{i1} \leq 11$ for non-DE genes.

Scenario III Uniform on $5 \leq -\log_2 \sqrt{\lambda_{i1}\lambda_{i2}} \leq 11$ and $-1 \leq \log_2(\lambda_{i2}/\lambda_{i1}) \leq 1$ for DE genes and uniform on $5 \leq -\log_2 \lambda_{i1} \leq 11$ for non-DE genes.

For each scenario above, we consider both $\mathbf{p} = (0.9, 0.025, 0.025, 0.025, 0.025)$ and $\mathbf{p} = (0.8, 0.05, 0.05, 0.05, 0.05)$, where partially DE genes π_i are independently generated from distribution $\text{Beta}(2, 2)$. We apply seven testing procedures, including our method (MCMC5), a three-state Bayesian model for which the partial DE states are removed (MCMC3), as well as BRIDGE (Gottardo et al., 2006), GaGa (Rossell, 2009), ORT (Wu, 2007), OS (Tibshirani and Hastie, 2007) and simple t -test (after log-transformation).

Figures 5.1-5.3 show the receiver operating characteristic (ROC) curves for the seven DE gene detection procedures under the three scenarios. Such curves can be used to evaluate the performance of various methods where we are trying to balance the number of true positives and the number of false positives. To create such curves, we first count the number of true positives (TP) and false positives (FP) for a range of number of declared DE genes cutoffs. We then plot the fraction of true positives out of the positives (true positive rate) vs. the fraction of false positives out of the negatives (false positive rate). The larger the area under the resulting curve, the better the performance of the model. We make several observations based on these figures. First, MCMC5 performs best for most of the simulations, especially when the sample size is small. Second, the worst performer under our simulation is OS. Surprisingly, the ROC curves show that OS is even worse than random guessing under our simulation setup. Appendix B contains

a detailed explanation for this. Thus one need to be very cautious when applying OS statistic. Third, the performance of the t -test can be close to that of MCMC5 when $m = n = 24$, but we see MCMC5 performs better than t -test when the false positive rate is small, which is arguably the more interesting case. Even if MCMC5 is not significantly better than other approaches under some scenarios, we show using our real data later that at least it is complementary to other procedures since it can detect some interesting genes that are missed by other procedures.

Furthermore, in Tables 5.1-5.3, we show some operating characteristics of the different procedures under the same simulation setup. The results reported are based on controlling FDR below 0.05. For the OS, ORT, and t -statistic, the p -values for all genes are calculated based on the rank of the statistic among those on the null genes (note that we know the null genes only in simulation). We also tried permutation tests and the results are similar. Then the p -values are transformed to q -values (Storey and Tibshirani, 2003) to find the threshold for controlling FDR, which is available in the `qvalue` package within R. For the other four procedures (MCMC5, MCMC3, BRIDGE and GaGa), we rank the genes and control the FDR based on the estimated posterior distribution as described in Section 4.4. For the target FDR level, we count the number of true positives (TP), false positives (FP), true negatives (TN) and false negatives (FN). Then we evaluate the proportion of DE genes ($\hat{DE}/G = (TP+FP)/G$), the sensitivity (Sens.) = $TP/(TP+FN)$, the specificity (Spec.) = $TN/(TN+FP)$, the realized false discovery rate (FDR) = $FP/(TP+FP)$ and the realized false non-discovery rate (FNDR) = $FN/(TN+FN)$. From Tables 5.1-5.3, we observe that the FDR controlling procedure is not effective for ORT and OS, with realized FDR frequently larger than 0.05, while for other procedures the FDR is

well controlled except for a few cases. Besides, our MCMC5 procedure can generally detect a larger number of differentially expressed genes and at the same time achieve similar or better sensitivity and specificity compared to other procedures.

Finally we compared the ability of these methods to accurately discover DE genes of different expressed states (Figures 5.4-5.6). We notice that MCMC5, MCMC3, BRIDGE and GaGa have similar detecting power for up-regulated expressed genes ($Z = 1$) and down-regulated expressed genes ($Z = 3$). But MCMC5 performs significantly better than other competitors when detecting partially DE genes ($Z = 2$ and $Z = 4$) for all three scenarios. This indicates the sensitivity and reliability of MCMC5 in the analysis of partially DE genes. It is also interesting to note that OS does better for those genes under partially differentially expressed states ($Z = 2$ and $Z = 4$) than for those under states $Z = 1$ and $Z = 3$, which contradicts our intuition. It will be observed again in the next section. A possible explanation for this phenomenon is given in Appendix B.

Figure 5.1: ROC curves for simulation studies of scenario I with different sample sizes: true positive rate versus false positive rate.

Figure 5.2: ROC curves for simulation studies of scenario II with different sample sizes: true positive rate versus false positive rate.

Figure 5.3: ROC curves for simulation studies of scenario III with different sample sizes: true positive rate versus false positive rate.

Table 5.1: Operating characteristics of simulation studies with $m=n=4$. $P(DE)$ is the true proportion of differentially expressed genes simulated. $\hat{DE}/G$ is the proportion of differentially expressed genes detected by different methods. Sens. is the proportion of genes that are detected out of the DE genes. Spec. is the proportion of genes that were not detected as DE out of the non-DE genes. FDR is the realized false discovery rate and FNDR is the realized false non-discovery rate.

Scenario I	$P(DE)=0.1$					$P(DE)=0.2$				
	$\hat{DE}/G$	Sens.	Spec.	FDR	FNDR	$\hat{DE}/G$	Sens.	Spec.	FDR	FNDR
MCMC5	0.088	0.850	0.997	0.034	0.016	0.161	0.770	0.991	0.044	0.055
MCMC3	0.068	0.650	0.997	0.044	0.038	0.138	0.665	0.994	0.036	0.078
BRIDGE	0.057	0.560	0.999	0.018	0.047	0.118	0.580	0.998	0.017	0.095
GaGa	0.059	0.580	0.999	0.017	0.045	0.122	0.600	0.998	0.016	0.091
ORT	0.056	0.520	0.996	0.071	0.051	0.097	0.450	0.991	0.072	0.122
OS	0.008	0.070	0.999	0.125	0.094	0.032	0.145	0.996	0.094	0.177
t -test	0.043	0.420	0.999	0.023	0.061	0.072	0.350	0.998	0.028	0.140
Scenario II	$P(DE)=0.1$					$P(DE)=0.2$				
	$\hat{DE}/G$	Sens.	Spec.	FDR	FNDR	$\hat{DE}/G$	Sens.	Spec.	FDR	FNDR
MCMC5	0.065	0.650	1	0	0.037	0.134	0.670	1	0	0.076
MCMC3	0.052	0.520	1	0	0.051	0.113	0.565	1	0	0.098
BRIDGE	0.031	0.310	1	0	0.071	0.062	0.310	1	0	0.147
GaGa	0.041	0.410	1	0	0.061	0.089	0.440	0.999	0.011	0.123
ORT	0.024	0.220	0.998	0.083	0.080	0.050	0.230	0.995	0.080	0.162
OS	0.004	0.030	0.999	0.250	0.097	0.019	0.085	0.998	0.105	0.187
t -test	0.022	0.200	0.998	0.091	0.082	0.064	0.315	0.999	0.016	0.146
Scenario III	$P(DE)=0.1$					$P(DE)=0.2$				
	$\hat{DE}/G$	Sens.	Spec.	FDR	FNDR	$\hat{DE}/G$	Sens.	Spec.	FDR	FNDR
MCMC5	0.023	0.230	1	0	0.079	0.078	0.385	0.999	0.013	0.133
MCMC3	0.013	0.130	1	0	0.088	0.059	0.295	1	0	0.150
BRIDGE	0.007	0.070	1	0	0.094	0.025	0.125	1	0	0.179
GaGa	0.014	0.130	0.999	0.071	0.088	0.045	0.225	1	0	0.162
ORT	0.006	0.050	0.999	0.167	0.096	0.019	0.085	0.998	0.105	0.187
OS	0.006	0.050	0.999	0.167	0.096	0.001	0	0.999	1	0.200
t -test	0.008	0.070	0.999	0.125	0.094	0.028	0.130	0.998	0.071	0.179

Table 5.2: Operating characteristics of simulation studies with $m=n=10$. $P(DE)$ is the true proportion of differentially expressed genes simulated. $\hat{DE}/G$ is the proportion of differentially expressed genes detected by different methods. Sens. is the proportion of genes that are detected out of the DE genes. Spec. is the proportion of genes that were not detected as DE out of the non-DE genes. FDR is the realized false discovery rate and FNDR is the realized false non-discovery rate.

Scenario I	$P(DE)=0.1$					$P(DE)=0.2$				
	$\hat{DE}/G$	Sens.	Spec.	FDR	FNDR	$\hat{DE}/G$	Sens.	Spec.	FDR	FNDR
MCMC5	0.093	0.890	0.996	0.043	0.012	0.197	0.930	0.986	0.056	0.017
MCMC3	0.082	0.820	1	0	0.020	0.181	0.880	0.994	0.028	0.029
BRIDGE	0.075	0.750	1	0	0.027	0.142	0.710	1	0	0.068
GaGa	0.081	0.800	0.999	0.012	0.022	0.174	0.855	0.996	0.017	0.035
ORT	0.077	0.720	0.994	0.065	0.030	0.169	0.790	0.986	0.065	0.051
OS	0.010	0.090	0.999	0.100	0.092	0.029	0.135	0.998	0.069	0.178
<i>t</i> -test	0.063	0.590	0.996	0.064	0.044	0.137	0.650	0.991	0.051	0.081
Scenario II	$P(DE)=0.1$					$P(DE)=0.2$				
	$\hat{DE}/G$	Sens.	Spec.	FDR	FNDR	$\hat{DE}/G$	Sens.	Spec.	FDR	FNDR
MCMC5	0.083	0.810	0.998	0.024	0.021	0.164	0.810	0.998	0.012	0.046
MCMC3	0.073	0.730	1	0	0.029	0.142	0.710	1	0	0.068
BRIDGE	0.055	0.550	1	0	0.048	0.103	0.515	1	0	0.108
GaGa	0.076	0.760	1	0	0.026	0.139	0.695	1	0	0.071
ORT	0.074	0.690	0.994	0.068	0.034	0.145	0.675	0.988	0.069	0.076
OS	0.010	0.090	0.999	0.100	0.092	0.022	0.100	0.998	0.091	0.184
<i>t</i> -test	0.059	0.570	0.998	0.034	0.046	0.116	0.570	0.998	0.017	0.097
Scenario III	$P(DE)=0.1$					$P(DE)=0.2$				
	$\hat{DE}/G$	Sens.	Spec.	FDR	FNDR	$\hat{DE}/G$	Sens.	Spec.	FDR	FNDR
MCMC5	0.046	0.450	0.999	0.022	0.058	0.120	0.600	1	0	0.091
MCMC3	0.036	0.350	0.999	0.028	0.067	0.094	0.470	1	0	0.117
BRIDGE	0.029	0.290	1	0	0.073	0.078	0.390	1	0	0.132
GaGa	0.036	0.360	1	0	0.066	0.089	0.445	1	0	0.122
ORT	0.030	0.280	0.998	0.067	0.074	0.100	0.465	0.991	0.070	0.119
OS	0.009	0.080	0.999	0.111	0.093	0.006	0.025	0.999	0.167	0.196
<i>t</i> -test	0.032	0.310	0.999	0.031	0.071	0.095	0.455	0.995	0.042	0.120

Table 5.3: Operating characteristics of simulation studies with $m=n=24$. $P(DE)$ is the true proportion of differentially expressed genes simulated. $\hat{DE}/G$ is the proportion of differentially expressed genes detected by different methods. Sens. is the proportion of genes that are detected out of the DE genes. Spec. is the proportion of genes that were not detected as DE out of the non-DE genes. FDR is the realized false discovery rate and FNDR is the realized false non-discovery rate.

Scenario I	$P(DE)=0.1$					$P(DE)=0.2$				
	$\hat{DE}/G$	Sens.	Spec.	FDR	FNDR	$\hat{DE}/G$	Sens.	Spec.	FDR	FNDR
MCMC5	0.094	0.880	0.993	0.064	0.013	0.201	0.945	0.985	0.060	0.014
MCMC3	0.088	0.860	0.998	0.023	0.015	0.187	0.910	0.994	0.027	0.022
BRIDGE	0.079	0.770	0.998	0.025	0.025	0.177	0.875	0.998	0.011	0.030
GaGa	0.088	0.850	0.997	0.034	0.016	0.196	0.935	0.989	0.046	0.016
ORT	0.080	0.750	0.994	0.063	0.027	0.189	0.885	0.985	0.064	0.028
OS	0.019	0.170	0.998	0.105	0.085	0.050	0.230	0.995	0.080	0.162
t -test	0.077	0.770	0.998	0.025	0.025	0.169	0.805	0.990	0.047	0.047
Scenario II	$P(DE)=0.1$					$P(DE)=0.2$				
	$\hat{DE}/G$	Sens.	Spec.	FDR	FNDR	$\hat{DE}/G$	Sens.	Spec.	FDR	FNDR
MCMC5	0.091	0.880	0.997	0.033	0.013	0.184	0.915	0.999	0.005	0.021
MCMC3	0.084	0.840	1	0	0.018	0.166	0.825	0.999	0.006	0.042
BRIDGE	0.073	0.730	1	0	0.029	0.148	0.740	1	0	0.061
GaGa	0.090	0.870	0.997	0.033	0.014	0.177	0.860	0.994	0.028	0.034
ORT	0.093	0.870	0.993	0.065	0.014	0.183	0.855	0.985	0.066	0.036
OS	0.020	0.180	0.998	0.100	0.084	0.036	0.165	0.996	0.083	0.173
t -test	0.078	0.760	0.998	0.026	0.026	0.163	0.775	0.990	0.049	0.054
Scenario III	$P(DE)=0.1$					$P(DE)=0.2$				
	$\hat{DE}/G$	Sens.	Spec.	FDR	FNDR	$\hat{DE}/G$	Sens.	Spec.	FDR	FNDR
MCMC5	0.072	0.700	0.998	0.028	0.032	0.149	0.740	0.999	0.007	0.061
MCMC3	0.063	0.630	1	0	0.040	0.141	0.700	0.999	0.007	0.070
BRIDGE	0.060	0.590	0.999	0.017	0.044	0.115	0.570	0.999	0.009	0.097
GaGa	0.066	0.640	0.998	0.030	0.039	0.142	0.695	0.996	0.021	0.071
ORT	0.065	0.610	0.996	0.062	0.042	0.140	0.655	0.989	0.064	0.080
OS	0.002	0.010	0.999	0.500	0.099	0.035	0.160	0.996	0.086	0.174
t -test	0.069	0.670	0.998	0.029	0.035	0.147	0.700	0.991	0.048	0.070

Figure 5.4: Simulation results of scenario I with different sample sizes: comparisons of detecting power for genes with different expressed states, controlling the FDR below 0.05.

Figure 5.5: *Simulation results of scenario II with different sample sizes: comparisons of detecting power for genes with different expressed states, controlling the FDR below 0.05.*

Figure 5.6: Simulation results of scenario III with different sample sizes: comparisons of detecting power for genes with different expressed states, controlling the FDR below 0.05.

5.2 Real-life Datasets

In this section, we apply the earlier methods to two cancer datasets. One is to compare the gene expression of two types of breast cancer tumors (Hedenfalk et al., 2001). The other is to identify genes that are differentially expressed between the normal and prostate cancer tissues (Singh et al., 2002).

5.2.1 Breast Cancer Data Example

We first consider a cDNA microarray dataset from Hedenfalk et al. (2001), involved in a breast cancer study. One aim of the study is to find differentially expressed genes between BRCA1 mutation positive tumors and BRCA2 mutation positive tumors. The dataset consists of 3226 genes for $m = 8$ BRCA2 tumors and $n = 7$ BRCA1 tumors. Following the preprocessing procedure described by Storey and Tibshirani (2003), we eliminated those genes with one or more measurement exceeding 20. This leaves us with $G = 3170$ genes for further analysis.

In Figure 5.7, we plot genes from eight BRCA2 tumors ranked by coefficient of variation (CV) against those genes ranked by mean. It can be seen that there is no strong systematic relationship between the CV and the mean, which shows the reasonableness of our model assumption of constant CV for each sample (Newton et al., 2004; Jensen et al., 2009).

5.2.1.1 Full Data Set Analysis

We implemented our model using the MCMC sampler outlined in Section 4.3. Five chains were run for 10,000 iterations with initial Z_i all assigned to state 0. For convergence diagnostics, we used the methods developed by

Figure 5.7: *Rank of genes by coefficient of variation (CV) against rank of genes by mean: $G = 3170$ genes from $m = 8$ BRCA2 mutation positive tumors. For each gene, we compute the sample mean and CV which is the sample standard deviation divided by the sample mean.*

Gelman and Rubin (1992) and Geweke (1992), which are available from the `boa` package within R (Smith, 2007). We examined many different parameter values using the package and the diagnostic statistics suggest that the chains have converged after 5,000 iterations. Then the initial 5,000 iterations were discarded as burn-in, and our further analysis is based on the remaining iterations.

Using MCMC5 we estimated $\mathbf{p} = (.7488, .0006, .1244, .0300, .0962)$, which indicated there were many genes expressed differently between breast tu-

mors with BRCA1 mutations and those with BRCA2 mutations. This result is consistent with the claims made in Hedenfalk et al. (2001). MCMC5, BRIDGE and GaGa identified 193, 75 and 132 genes, respectively, which are most likely DE when controlling the FDR at the 5% level. However, MCMC3, *t*-test, ORT and OS (combined with permutation test for calculating *p*-values) seemed very conservative with only 26, 32, zero and zero genes that declared differentially expressed.

By searching through some existing literature, we find that many of the genes detected by MCMC5 are related to DNA repair and apoptosis. For example, the c-Myc that promotes apoptosis (Juin et al., 2002) is partially up-regulated in BRCA1 tumors with 98% posterior probability. The ATDC is directly involved in DNA repair (Ronen and Glickman, 2001) and it is up-regulated with 99.8% posterior probability in BRCA1 related tumors. But both BRIDGE and GaGa approaches regard it as a non-differential gene with approximately 100% posterior probability. The VEGF is down-regulated in BRCA1 tumors with 100% posterior probability and it is associated with inhibition of tumor cell apoptosis (Pidgeon et al., 2001). However, this gene is ranked at 572, 2340 and 2635 by *t*-test, OS and ORT respectively. Thus it is more difficult to detect VEGF by these three methods.

5.2.1.2 Subset Data Analysis

In order to evaluate the sensitivity and reliability of the seven methods under small sample sizes, we randomly chose four replicates from each group and applied each method to find out the number of DE genes controlling the FDR below 0.05. This process was repeated 100 times. Since neither OS

Table 5.4: *DE genes detection for breast cancer data. Randomly choose four samples per group and report the average results based on 100 combinations. $\hat{DE}$: number of genes declared differentially expressed; % rep.: proportion of DE genes also declared significant when analyzing the full data set.*

	4 replicates per group		full data
	$\hat{DE}$	% rep.	$\hat{DE}$
MCMC5	37.8	0.796	193
MCMC3	11	0.103	26
BRIDGE	14.7	0.384	75
GaGa	26.9	0.713	132
<i>t</i> -test	0.4	0.028	32

nor ORT can detect a DE gene for the full data set, we focus our discussion on the remaining five methods.

Table 5.4 lists the average number of genes declared DE when analyzing a subset of 4 replicates per group, as well as the full data set. We also show in Table 5.4 the percentages of reproducibility, indicating how often DE genes that were discovered under small sample sizes were found again when analyzing the full data expression. The MCMC5 method detected more DE genes than the other competitors for both sample sizes and presented higher reproducibility. MCMC3 and *t*-test appear to be conservative and show low reproducibility under small sample size. It is a potential concern that the proposed five-component mixture model might be liberal. However, our simulation results in Table 5.1-5.3 suggest that MCMC5 method is not liberal since the specificity is not lower than other methods.

Figure 5.8: ROC curves of the breast cancer simulated data set: (a) averaged false discovery rate. The x-axis is the positive rate: the proportion of genes called significant. (b) averaged successfully detection rate as a function of proportion of partially DE components.

5.2.1.3 Simulated Data Analysis

We now attempt to apply our approach to the analysis of simulated datasets from real data. This is based on taking samples from the posterior distribution of the MCMC5 model fit to the breast cancer data. For each gene, let the simulated means consistent with the sample means and the expression measurements were sampled from Gamma distribution with estimated parameters. We assigned each gene a DE status according to the DE posterior probability. Hence the resulting data preserves to a large extent the distribution in the underlying dataset. In total, 100 simulated data sets were generated and we report the average results.

Table 5.5: Operating characteristics of the breast cancer simulated data set. $\hat{DE}/G$ is the proportion of differentially expressed genes detected by different methods. Sens. is the proportion of genes that are detected out of the DE genes. Spec. is the proportion of genes that were not detected as DE out of the non-DE genes. FDR is the realized false discovery rate and FNDR is the realized false non-discovery rate.

	$\hat{DE}/G$	Sens.	Spec.	FDR	FNDR
MCMC5	0.095	0.358	0.998	0.011	0.177
MCMC3	0.069	0.274	0.999	0.005	0.196
BRIDGE	0.057	0.227	0.999	0.007	0.206
GaGa	0.057	0.218	0.996	0.049	0.208
ORT	0.026	0.099	0.998	0.046	0.232
OS	0.008	0.031	0.999	0.063	0.245
t -test	0.023	0.092	0.999	0.012	0.233

In order to allow easy overall comparison, we ranked the genes by significance level, and then computed the false discovery rate (FDR) as a function of the proportion of genes declared significant. The average FDR curves are plotted in Figure 5.8a. As seen, our proposed MCMC5 method exhibited the lowest FDR when detecting the same number of DE genes. Both OS and t -test had actually worst global FDRs, which has been observed in the simulation studies in Section 5.1.

Again, we investigated the operating characteristics of different methods using a FDR of 5% level. Results are shown in Table 5.5, from which we see MCMC5 can detect more DE genes than other procedures with a well-controlled FDR level. For this breast cancer simulated dataset, we have

kept track of the known DE status of each simulated gene. To investigate the performance of these testing procedures when detecting partially DE genes, we computed the mean successfully detection rate as a function of the proportion of differentially expressed components of a gene. The resulting plot given in Figure 5.8b highlights several findings. First, as expected, MCMC5 obtained substantially more power than other procedures when the proportion was less than 80%. The advantage is noticeable for those genes with partially DE not exceeding 50%. Second, the power of BRIDGE and GaGa increased rapidly and outperformed MCMC5 slightly when the proportion reached 100%. Third, ORT and t -test did very badly when dealing with genes of small proportion partially differentially expressed. But t -test showed marked improvement when the proportion increased, while ORT did not show as much improvement as t -test. Finally, the detection rate of OS reduced when the proportion went from 50% to 100%, leading to overall high false discovery rate, as indicated in the FDR curves (Figure 5.8a). We present reasons about the poor performance of OS in Appendix B.

5.2.2 Prostate Cancer Data Example

We also applied our approach to the study of Singh et al. (2002), where Affymetrix arrays were used to compare the gene expressions from prostate cancer tissues with normal prostate tissues. They measured expressions of 12625 genes from 52 tumor and 50 normal prostate tissue samples. The raw data set contains 12,625 genes and it's really a burden for us to do data analysis using the proposed BOPA model. So we filtered the genes and kept those genes which are most possible differentially expressed when comparing control and cancer samples. We used "threestep" in Bioconduc-

tor to normalize the data and then applied a log transform. Furthermore, we filtered the genes using the following two criteria retaining genes whose: (1) expression measurements are above 100 in at least 20% of the samples; and have (2) interquartile ranges across the samples on the base 2 logarithmic scale which are at least 0.1. In the end we were left with 2204 genes for further analysis.

Using BOPA, we identified 200 genes with 100% posterior probability of being differentially expressed, which are all ranked as 1. Table 5.6 lists the genes called significant by the BOPA method or t -test when the FDR is controlled at 0.01, and confirmed to be related to prostate cancer. These genes are identified by a search on PubMed. The t -test identified 290 genes as DE, 9 of which are confirmed to be related to prostate cancer, while our method identified 255 with 12 of which confirmed. Almost all the 9 genes (except CAMKK2) identified by t -test are also identified by BOPA. The rankings of these genes using other different methods are also shown in Table 5.6. For OS, 473 genes had OS statistic value exactly equal to zero, and all these genes are assigned the ranking 1732, which explains the many genes with equal ranking for OS seen in the table. Similarly, those genes with ORT statistic equal to zero were assigned a ranking of 1742. We observe that several genes with high posterior probability by the Bayesian method (e.g. CDH1, KLK11, HSP90AA1 and IGFBP3) have large ranks by other methods. GaGa found 504 significant genes, whereas BRIDGE identified only 78 DE genes which is too conservative for this prostate cancer dataset.

In Figure 5.9 we show the expression profiles of the eight genes that were identified by the BOPA method. The figure allows us to investigate

the feature of expression measurements of those genes that can be detected by our proposed method but not by other procedures. For those genes (e.g. CDH1, SFRP1, FOXA1, KLK3 and AMACR) with great differences in expression between cancer samples and normal samples, BOPA performed as well as other tested alternatives. However, as discussed in the previous sections, BOPA showed higher sensitivity when the gene expression level from cancer samples was very close to that from normal samples (e.g. KLK11, HSP90AA1 and IGFBP3). This suggests that our proposed method is a valuable option in the analysis of partially differential expressed genes.

5.3 Summary of the Chapter

In this chapter, we assessed the performance of our methodology by conducting a series of simulation studies and real data analyses. These results show that BOPA outperforms other existing methods for identifying those partially differential expression. The real data study provides the additional evidence to support our proposed method. In the next chapter, we conclude our research findings and describe the future work.

Table 5.6: *Genes detected by t -test or BOPA under the target FDR level 0.01 and confirmed to be associated with prostate cancer. The last five columns also list the ranking of each gene by other methods. Those genes which are identified as DE by BOPA with 100% posterior probability are all ranked as 1. For ORT, genes are assigned the ranking 1742 if the ORT statistic equals zero. Those genes with OS statistic equal to zero are assigned a ranking of 1732.*

Method	Genename	Rank	BOPA	BRIDGE	GaGa	ORT	OS
t-test	AMACR	2	1	6	26	6	929
	FASN	54	1	76	281	1742	1732
	FOXA1	67	1	1	1443	818	1732
	CTBP2	72	235	131	174	1051	1015
	KRT18	75	1	112	239	1742	1732
	KLK3	76	1	9	549	8	1732
	CAMKK2	94	1366	444	379	28	88
	P4HB	175	1	24	616	1742	1732
	SFRP1	257	1	108	1118	1742	1732
Method	Genename	Rank	t -test	BRIDGE	GaGa	ORT	OS
BOPA	AMACR	1	2	6	26	6	929
	KLK3	1	76	9	549	8	1732
	P4HB	1	175	24	616	1742	1732
	FASN	1	54	76	281	1742	1732
	FOXA1	1	67	1	1443	818	1732
	KRT18	1	75	112	239	1742	1732
	SFRP1	1	257	108	1118	1742	1732
	HSP90AA1	1	490	208	717	1742	1732
	CDH1	1	463	50	1433	1742	1732
	KLK11	206	2068	1421	2061	823	432
	IGFBP3	219	1812	1231	2068	1742	1732
	CTBP2	235	72	131	174	1051	1015

Figure 5.9: *Differentially expressed gene detection of prostate cancer microarray data: plotted are the log₂ based intensity of those genes which are associated with prostate cancer. Those methods that can detect the gene are list in the parentheses.*

Chapter 6

Conclusion

We conclude this work with a summary of research findings and experimental results in Section 6.1, and a brief outline of some possible study directions for the future we present in Section 6.2.

6.1 Summary of Our Research

A main challenge with microarray data is that there are tens of thousands of genes but with a small number of replicates. Thus there is strong need for powerful statistical methods to deal with such high-throughput data. Motivated by cancer outlier profiling problems, we developed a framework for detecting DE genes, in a way that is sensitive to partially differential expressed genes.

In this thesis, an extension of the Gamma-Gamma model introduced by Newton et al. (2004) has been described. Using Bayesian hierarchical modeling, we strengthen our inferences by borrowing information across different genes and samples. Another advantage of using Bayesian inference is that it appropriately incorporates the uncertainties involved at all

levels and automatically propagates the uncertainties through the model. Furthermore, one can also incorporate prior information and conveniently synthesize data from different sources. For example, if we have some knowledge about the percentage of differentially expressed genes, we can incorporate it into the prior distribution of $\mathbf{p}$. Besides many advantages enjoyed by general Bayesian analysis compared to classical frequentist tests, including full posterior exploration and uncertainty propagation, our model also automatically identifies genes that are only partially expressed. In addition, an efficient MCMC sampler has been proposed to explore the posterior distributions of various variables and parameters. This is a natural and elegant framework for cancer outlier profiling and it can foreseeably be extended to microarray data with more than two experimental conditions.

We also compared our model (BOPA) with three well-known frequentist tests, two Gamma-based models and a method based on normal linear models. By a series of simulation studies, we have shown that BOPA can detect more genes than other competitors, while still controlling the FDR under the desired levels. The real data analysis provided additional evidence that our method can improve detection performance for those partial expressions.

In summary, we have developed a novel Bayesian hierarchical model for cancer outlier expression detection. This model is efficient for identifying genes that are up- or down-regulated in a subset of cancer samples compared with normal samples.

6.2 Limitations and Future Work

Although this thesis claims original and significant contributions to the microarray data analysis, there are some limitations to the proposed techniques and there are some ways to improve on this research.

The most serious concern is the possible model misspecification. Although we did not attempt to use alternative parametric models here, it can be expected that other common gene expression distributions such as lognormal-normal models, perform similarly to the Gamma-Gamma model used here, as discussed for the three-state model presented in Kendzierski et al. (2003). Another concern involves the modeling decision that we treat gene expressions under one condition as homogeneous. There is reason to suspect that this is not true for many datasets. In this aspect we followed other existing approaches, such as OS and ORT. Extension of our model to the case where gene expressions can be heterogeneous under all conditions makes the modelling aspect more complicated. In this thesis, we focus on detecting DE genes under two conditions. Although some other detecting methods could be extent to dealing with multiple conditions, our proposed BOPA is complicated and it is not so easy to scale it up to multiple conditions.

The second problem is the computational complexity. In order to improve the computation speed, we actually run the MCMC algorithm on a computer cluster with 8 CPUs with the help of the `snowfall` package in R to parallelize the implementation. The computing time is mainly affected by increasing p and number of replicates m and n . For our simulations for example, it takes about 5 to 20 minutes for each generated dataset to produce the results. The case studies we presented are feasible computa-

tionally for data sets with gene dimensions on the order of 10^3 . Further approximations or new computational algorithms are needed to be able to apply the BOPA model to data sets with hundreds of thousands or millions of variables.

Another possible future work is to apply our method on the analysis of the next generation sequencing data. We first need to transform the proposed BOPA model into discrete model. Due to that discrete distribution has different nature than continuous ones, it remains to be seen whether it will work well in practice.

Appendix

A. Effectiveness of the Gibbs sampling scans

To see the effect of using Gibbs sampling scans for proposing the allocation vector after a gene is proposed to be assigned to a partial DE state, we focus on those genes with partial differential expressions. The important tuning parameter is K , the number of sequential scans used to obtain the final proposed allocation vector, where $K = 3$ is used in our implementation. Figure A.1 shows some trace plots of a simulated gene with partially up-regulated expression (state 2) during the first 1000 iterations. From the trace plot of Z_i with $K = 1$, we see that the sampler spends most of the time in state $Z_i = 1$. The reason is that with a small K , the proposed allocation vector is not sensible and thus the proposal is rarely accepted so the sampler failed to explore the posterior distribution efficiently. We also show the trace plot of $\lambda_{i2}/\lambda_{i1}$ (defined to be 1 when the gene is non-DE). As a consequence of the fact that the chain is stuck at the wrong state, the posterior estimate for this parameter is obviously upward biased. This problem is corrected when $K = 3$, where the sampler correctly identifies the differential expression pattern of the gene and the posterior estimate of

$\lambda_{i2}/\lambda_{i1}$ is close to the true value 0.5. In Figure A.2, we focus on another gene with partially down-regulated expression and we observe the same phenomenon: we would obtain wrong inferences based on the sampler with $K = 1$. Of course, using the prior as the proposal distribution for the allocation vector is even more disastrous. Empirically we find that $K = 3$ is good enough for all our simulations. Increasing K beyond 3 does not improve our simulation results and thus the added computational burden is not justified in practice.

B. Explanation for the bad performance of OS statistic

As before, the gene expressions for normal samples are denoted by x_{ij} for genes $i = 1, 2, \dots, G$ and samples $j = 1, 2, \dots, m$, while y_{ij} denote the expressions for cancer samples with $i = 1, 2, \dots, G$ and $j = 1, 2, \dots, n$. For simplicity, we only consider a one-sided test where the activated genes from cancer samples have a higher expression level.

The usual t -statistic (up to a multiplication factor independent of genes) for a two-sample test of differences in means is defined for each gene i by

$$T_i = \frac{\bar{x}_i - \bar{y}_i}{s_i},$$

where $\bar{x}_i = \sum_j x_{ij}/m$ is the average expression of gene i in normal samples, $\bar{y}_i = \sum_j y_{ij}/n$ is the average expression of gene i in cancer samples, and s_i is the usual pooled standard deviation estimate

$$s_i^2 = \frac{\sum_{1 \leq j \leq m} (x_{ij} - \bar{x}_i)^2 + \sum_{1 \leq j \leq n} (y_{ij} - \bar{y}_i)^2}{n + m - 2}.$$

The t -statistic is powerful when the alternative distribution is such that $y_{ij}, j = 1, 2, \dots, n$ all come from a distribution with a higher mean. For

Figure A.1: Comparison of trace plots for a gene that is partially up-regulated (state 2). (a)(b): Trace plot of Z_i when $K = 1$ and $K = 3$ respectively. (c)(d): Trace plot of $\lambda_{i2}/\lambda_{i1}$ when $K = 1$ and $K = 3$ respectively. The true value for $\lambda_{i2}/\lambda_{i1}$ is 0.5.

(a)

(b)

(c)

(d)

Figure A.2: Comparison of trace plots for a gene that is partially down-regulated (state 4). (a)(b): Trace plot of Z_i when $K = 1$ and $K = 3$ respectively. (c)(d): Trace plot of $\lambda_{i2}/\lambda_{i1}$ when $K = 1$ and $K = 3$ respectively. The true value for $\lambda_{i2}/\lambda_{i1}$ is 2.

most cancer types, heterogeneous activation patterns make the t -statistic inefficient for detecting those expression profiles. Let

$$O_i = \{y_{ij} : y_{ij} > q_{75}(\{x_{ij}\}_{1 \leq j \leq m}, \{y_{ij}\}_{1 \leq j \leq n}) + \text{IQR}(\{x_{ij}\}_{1 \leq j \leq m}, \{y_{ij}\}_{1 \leq j \leq n})\}$$

be the outliers from the cancer samples for gene i , where $q_r(\cdot)$ is the r th percentile of the data, $\text{med}_i = \text{median}(\{x_{ij}\}_{1 \leq j \leq m}, \{y_{ij}\}_{1 \leq j \leq n})$ is the median of the pooled samples for gene i , $\text{mad}_i = 1.4826 \times \text{median}(\{x_{ij} - \text{med}_i\}_{1 \leq j \leq m}, \{y_{ij} - \text{med}_i\}_{1 \leq j \leq n})$ is the median absolute deviation of the pooled samples, and $\text{IQR}(\cdot)$ is the interquartile range of the data. The OS statistic from Tibshirani and Hastie (2007) is then defined as

$$OS_i = \frac{\sum_{y_{ij} \in O_i} (y_{ij} - \text{med}_i)}{\text{mad}_i}.$$

We now identify a particular situation where OS fails. Suppose for a certain gene i , we have $x_{ij} \approx 1$ (that is, all expression values under the normal condition are very close to, although not exactly equal to, 1) and $y_{ij} \approx 2$. This should be a very easy case for DE detection, say using the t -statistic. However, since $q_{75}(\{x_{ij}\}_{1 \leq j \leq m}, \{y_{ij}\}_{1 \leq j \leq n}) \approx 2$ and $\text{IQR}(\{x_{ij}\}_{1 \leq j \leq m}, \{y_{ij}\}_{1 \leq j \leq n}) \approx 1$, we have O_i to be an empty set, resulting in $OS_i = 0$. Thus OS statistic surprisingly cannot detect this kind of extremely easy case of differential expression.

Finally, as a side note, Wu (2007) studied the ORT statistic, which is similar to OS statistic. The important difference is that outliers are defined relative to the normal sample instead of the pooled sample. So in their definition,

$$O_i = \{y_{ij} : y_{ij} > q_{75}(\{x_{ij}\}_{1 \leq j \leq m}) + \text{IQR}(\{x_{ij}\}_{1 \leq j \leq m})\}.$$

Similary, med_i in OS is replaced by med_{ix} and mad_i by $\text{median}(\{x_{ij} - \text{med}_{ix}\}_{1 \leq j \leq m}, \{y_{ij} - \text{med}_{iy}\}_{1 \leq j \leq n})$, where med_{ix} and med_{iy} are the medians of

normal and cancer samples respectively. It is easily seen that ORT addresses the problem of OS explained above.

Bibliography

- Baldi, P. and Long, A. D. “A Bayesian framework for the analysis of microarray expression data: regularized t-test and statistical inferences of gene changes.” *Bioinformatics*, 17(6):509–519 (2001).
- Benjamini, Y. and Hochberg, Y. “Controlling the false discovery rate: a practical and powerful approach to multiple testing.” *Journal of the Royal Statistical Society. Series B (Methodological)*, 57(1):289–300 (1995).
- Cui, X., Hwang, J., Qiu, J., Blades, N., and Churchill, G. “Improved statistical tests for differential gene expression by shrinking variance components estimates.” *Biostatistics*, 6(1):59–75 (2005).
- Do, K. A., Muller, P., and Tang, F. “A Bayesian mixture model for differential gene expression.” *Journal of the Royal Statistical Society Series C-Applied Statistics*, 54:627–644 (2005).
- Efron, B., Tibshirani, R., Storey, J., and Tusher, V. “Empirical Bayes analysis of a microarray experiment.” *Journal of the American Statistical Association*, 96(456):1151–1160 (2001).
- Gamerman, D. and Lopes, H. *Markov Chain Monte Carlo: Stochastic*

- Simulation for Bayesian Inference*. Chapman & Hall/CRC, 2nd edition (2006).
- Gelman, A., Carlin, J., Stern, H., and Rubin, D. *Bayesian Data Analysis*. Chapman & Hall/CRC, 2nd edition (2004).
- Gelman, A. and Rubin, D. B. “Inference from iterative simulation using multiple sequences.” *Statistical Science*, 7:457–472 (1992).
- Geman, S. and Geman, D. “Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images.” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6:721–741 (1984).
- Geweke, J. *Evaluating the accuracy of sampling-based approaches to the calculation of posterior moments*. Bayesian Statistics 4. (Bernardo, J. M., Berger, J. O., Dawid, A. P. and Smith, A. F. M., eds.) Oxford Press (1992).
- Ghosh, D. and Chinnaiyan, A. M. “Genomic outlier profile analysis: mixture models, null hypotheses, and nonparametric estimation.” *Biostatistics*, 10(1):60–69 (2009).
- Gottardo, R., Raftery, A., Yeung, K., and Bumgarner, R. “Bayesian robust inference for differential gene expression in microarrays with multiple samples.” *Biometrics*, 62(1):10–18 (2006).
- Hastings, W. “Monte Carlo sampling methods using Markov chains and their applications.” *Biometrika*, 57(1):97–109 (1970).
- Hedenfalk, I., Duggan, D., Chen, Y., Radmacher, M., Bittner, M., Simon, R., Meltzer, P., Gusterson, B., Esteller, M., Kallioniemi, O., Wilfond, B.,

- Borg, A., and Trent, J. “Gene-expression profiles in hereditary breast cancer.” *New England Journal of Medicine*, 344(8):539–548 (2001).
- Hu, J. H. “Cancer outlier detection based on likelihood ratio test.” *Bioinformatics*, 24(19):2193–2199 (2008).
- Ibrahim, J. G., Chen, M. H., and Gray, R. J. “Bayesian models for gene expression with DNA microarray data.” *Journal of the American Statistical Association*, 97(457):88–99 (2002).
- Jain, S. and Neal, R. M. “A split-merge Markov chain Monte Carlo procedure for the dirichlet process mixture model.” *Journal of Computational and Graphical Statistics*, 13(1):158–182 (2004).
- Jensen, S. T., Erkan, I., Arnardottir, E. S., and Small, D. S. “Bayesian testing of many hypotheses $\times$ many genes: A study of sleep apnea.” *The Annals of Applied Statistics*, 3(3):1080–1101 (2009).
- Juin, P., Hunt, A., Littlewood, T., Griffiths, B., Swigart, L., Korsmeyer, S., and Evan, G. “C-Myc functionally cooperates with BAX to induce apoptosis.” *Molecular and Cellular Biology*, 22(17):6158–6169 (2002).
- Kendzierski, C. M., Newton, M. A., Lan, H., and Gould, M. N. “On parametric empirical Bayes methods for comparing multiple groups using replicated gene expression profiles.” *Statistics in Medicine*, 22(24):3899–3914 (2003).
- Kim, S., Lee, J., and Sohn, I. “Comparison of various statistical methods for identifying differential gene expression in replicated microarray data.” *Statistical methods in medical research*, 15(1):3–20 (2006).

- Lewin, A. and Richardson, S. *Bayesian methods for microarray data*. Handbook of Statistical Genetics. (Balding, D. J., Bishop, M. and Cannings, C., eds.) John Wiley & Sons, Ltd, Chichester, UK (2008).
- Lewin, A., Richardson, S., Marshall, C., Glazier, A., and Aitman, T. “Bayesian modeling of differential gene expression.” *Biometrics*, 62(1):1–9 (2006).
- Lian, H. “MOST: detecting cancer differential gene expression.” *Biostatistics*, 9(3):411–418 (2008).
- Lo, K. and Gottardo, R. “Flexible empirical Bayes models for differential gene expression.” *Bioinformatics*, 23(3):328–335 (2007).
- Lönnstedt, I. and Speed, T. “Replicated microarray data.” *Statistica Sinica*, 12(1):31–46 (2002).
- Metropolis, N., Rosenbluth, A., Rosenbluth, M., Teller, A., and Teller, E. “Equation of state calculations by fast computing machines.” *The Journal of Chemical Physics*, 21(6):1087–1091 (1953).
- Metropolis, N. and Ulam, S. “The Monte Carlo method.” *Journal of American Statistical Association*, 44(247):335–341 (1949).
- Müller, P., Parmigiani, G., and Rice, K. *FDR and Bayesian multiple comparisons rules*. Bayesian Statistics 8. (J. M. Bernardo, M. J. Bayarri, J. O. Berger, A. P. Dawid, D. Heckerman, A. F. M. Smith and M. West, eds.). Oxford Univ. Press. (2007).
- Newton, M. A., Kendzierski, C. M., Richmond, C. S., Blattner, F. R., and Tsui, K. W. “On differential variability of expression ratios: Improv-

- ing statistical inference about gene expression changes from microarray data.” *Journal of Computational Biology*, 8(1):37–52 (2001).
- Newton, M. A., Noueiry, A., Sarkar, D., and Ahlquist, P. “Detecting differential gene expression with a semiparametric hierarchical mixture method.” *Biostatistics*, 5(2):155–176 (2004).
- Pidgeon, G., Barr, M., Harmey, J., Foley, D., and Bouchier-Hayes, D. “Vascular endothelial growth factor(VEGF) upregulates BCL-2 and inhibits apoptosis in human and murine mammary adenocarcinoma cells.” *British Journal of Cancer*, 85(2):273–278 (2001).
- Raftery, A. and Lewis, S. *How many iterations in the Gibbs sampler*. Bayesian Statistics 4. (Bernardo, J. M., Berger, J. O., Dawid, A. P. and Smith, A. F. M., eds.) Oxford Press (1992).
- Roberts, G. and Rosenthal, J. “Optimal scaling for various Metropolis-Hastings algorithms.” *Statistical Science*, 16(4):351–367 (2001).
- Ronen, A. and Glickman, B. “Human DNA repair genes.” *Environmental and Molecular Mutagenesis*, 37(3):241–283 (2001).
- Rossell, D. “GaGa: A parsimonious and flexible model for differential expression analysis.” *Annals of Applied Statistics*, 3(3):1035–1051 (2009).
- Schulze, A. and Downward, J. “Navigating gene expression using microarrays - a technology review.” *Nature Cell Biology*, 3(8):190–195 (2001).
- Singh, D., Febbo, P., Ross, K., Jackson, D., Manola, J., Ladd, C., Tamayo, P., Renshaw, A., D’Amico, A., Richie, J., Lander, E., Loda, M., Kantoff, P., Golub, T., and Sellers, W. “Gene expression correlates of clinical prostate cancer behavior.” *Cancer Cell*, 1(2):203–209 (2002).

- Smith, B. J. “boa: An R package for MCMC output convergence assessment and posterior inference.” *Journal of Statistical Software*, 21(11) (2007).
- Smyth, G. “Linear models and empirical Bayes methods for assessing differential expression in microarray experiments.” *Statistical applications in genetics and molecular biology*, 3(1):Article 3 (2004).
- Storey, J. D. and Tibshirani, R. “Statistical significance for genomewide studies.” *Proceedings of the National Academy of Sciences of the United States of America*, 100(16):9440–9445 (2003).
- Tibshirani, R. and Hastie, T. “Outlier sums for differential gene expression analysis.” *Biostatistics*, 8(1):2–8 (2007).
- Tomlins, S. A., Rhodes, D. R., Perner, S., Dhanasekaran, S. M., Mehra, R., Sun, X. W., Varambally, S., Cao, X. H., Tchinda, J., Kuefer, R., Lee, C., Montie, J. E., Shah, R. B., Pienta, K. J., Rubin, M. A., and Chinnaiyan, A. M. “Recurrent fusion of TMPRSS2 and ETS transcription factor genes in prostate cancer.” *Science*, 310(5748):644–648 (2005).
- van Dyk, D. A. and Park, T. “Partially collapsed Gibbs samplers: Theory and methods.” *Journal of the American Statistical Association*, 103(482):790–796 (2008).
- van Wieringen, W. N., van de Wiel, M. A., and van der Vaart, A. W. “A Test for Partial Differential Expression.” *Journal of the American Statistical Association*, 103(483):1039–1049 (2008).
- Walsh, B. “Markov chain Monte Carlo and Gibbs sampling.” <http://web.mit.edu/~wingated/www/introductions/mcmc-gibbs-intro.pdf> (2004).

-
- Wang, T., Fox, J., and Giraud, A. *The biology of gastric cancers*. Springer Verlag (2009).
- Watson, J. and Crick, F. “Molecular structure of nucleic acids - a structure for deoxyribose nucleic acid.” *Nature*, 171(4356):737–738 (1953).
- Wu, B. L. “Cancer outlier differential gene expression detection.” *Biostatistics*, 8(3):566–575 (2007).
- Yang, Y., Buckley, M., and Speed, T. “Analysis of cDNA microarray images.” *Briefings in Bioinformatics*, 2(4):341–349 (2001).