UNIVERSITY OF CALIFORNIA
RIVERSIDE

Algorithms for Determining Differentially Expressed Genes and Chromosome Structures From High-Throughput Sequencing Data

A Dissertation submitted in partial satisfaction
of the requirements for the degree of

Doctor of Philosophy

in

Computer Science

by

Yi-Wen Yang

December 2015

Dissertation Committee:

Dr. Tao Jiang, Chairperson
Dr. Marek Chrobak
Dr. Stefano Lonardi
Dr. Thomas Girke

The Dissertation of Yi-Wen Yang is approved:

Committee Chairperson

University of California, Riverside

Acknowledgments

Pursuing my PhD degree is a long but fulfilled journey. I have a lot of people to thank for at this time. First and foremost, my PhD supervisor Dr. Tao Jiang, who kept me on track, and provided valuable feedback and guidance. It is really my pleasure to be a member of Dr. Jiang's group. His suggestions are always helpful in my doctoral research. I sincerely appreciate all the training I had in the past four years. Moreover, I would like to thank my doctoral dissertation committee members, Dr. Marek Chrobak, Dr. Stefano Lonard and Dr. Thomas Girke for their wisdom. A special thanks to Dr. Thomas Girke for his priceless advice on the biological background of NGS experiments. I also appreciate my family and friends for all their support.

I dedicate this thesis to my family. To my mother and father, who made me who I am today. To my lovely wife, who believed in me, motivated me, and supported me throughout this journey.

ABSTRACT OF THE DISSERTATION

Algorithms for Determining Differentially Expressed Genes and Chromosome Structures From High-Throughput Sequencing Data

by

Yi-Wen Yang

Doctor of Philosophy, Graduate Program in Computer Science
University of California, Riverside, December 2015
Dr. Tao Jiang, Chairperson

Next-generation sequencing (NGS) technologies are able to sequence DNA or RNA molecules at unprecedented speed and with high accuracy. Recently, NGS technologies have been applied in a variety of contexts, *e.g.,* whole genome sequencing, transcript expression profiling, chromatin immunoprecipitation sequencing, and small RNA sequencing, to accelerate genomic researches. The size of NGS data is usually gigantic such that the data analysis in these applications of NGS largely relies on efficient computational methods. Due to the critical demand for high performance computational algorithms, in the past few years, my research interest was focused on designing novel algorithms to address challenges in NGS data analysis. The main theme of this dissertation includes algorithmic solutions to three crucial problems in NGS data analysis, two arising from differential expression analysis using high-throughput mRNA sequencing (RNA-Seq) and the other from chromosome structure capture using high-throughput DNA sequencing (Hi-C). (1) In differential expression

analysis of RNA-Seq data, long or highly expressed genes are more likely to be detected by most of existing computational methods. However, such bias against short or lowly expressed genes may distort down-stream data analysis at system biology level. To further improve the sensitivity to short or lowly expressed genes, we designed a new computational tool, called MRFSeq, to combine both gene coexpression and RNA-Seq data. The performance of MRFSeq was carefully assessed using simulated and real benchmark datasets and the experimental results showed that MRFSeq was able to provide more accurate prediction in calling differentially expressed genes than the other existing methods such that the distortion due to the bias against short and lowly expressed genes was significantly alleviated. (2) Most of the existing differential expression analysis tools are developed for comparing RNA-Seq samples between known biological conditions. However, differential expression analysis is also important to other biological researches where the predefined conditions of samples are not available as *a priori*. For example, differential expressed transcripts can be used as biomarkers to classify a cohort of cancer samples into subtypes such that better diagnosis and therapy methods can be developed for each subtype. So, the first computational method, called SDEAP, was proposed to identify differential expressed genes and their alternative splicing events without the requirement of the predefined conditions. SDEAP provided accurate predition in our experiments on simulated and real datasets. The utility of SDEAP was further demonstrated by classifying subtypes of breast cancer, cell types and the cycle phases of mouse cells. (3) Chro-

mosome structures in nucleus play important roles in biological processes of cells. The Hi-C technology allows biology researchers to reconstruct the three dimensional structures of chromosomes in nucleus of cells on a genome-wide scale and thus serves as a vital component in studies of chromosome structures. During the experimental steps of Hi-C, systematic biases may be introduced into Hi-C data. Hence, eliminating the systematic biases is essential to all the applications using Hi-C data. We developed an improved bias reduction algorithm, called GDNorm. By taking advantages of a Poisson regression model that explicitly formulates the causal relationship of Hi-C data, systematic biases and spatial distances in chromosome structures, our experimental results showed that GDNorm was able to remove the biases from Hi-C data such that the corrected Hi-C data could lead to accurate reconstruction of chromosome structures. In the near future, with the rapid accumulation of NGS data, we expect these efficient computational methods to become valuable tools for discovering novel biological knowledge and benefit numerous genomic researches.

Contents

List of Figures **xi**

List of Tables **xiv**

1 Introduction **1**
1.1 Differential Transcript Expression Analysis Based on High-Throughput mRNA Sequencing . . . 3
1.2 Chromosome Conformation Capture using High-Throughput Sequencing and Bias Reduction . . . 7
1.3 Organization of the Rest of the Dissertation . . . 9

2 Differential Gene Expression Analysis Using Coexpression and RNA-Seq Data **13**
2.1 Introduction . . . 13
2.2 Methods . . . 17
2.2.1 Terminology and Notations . . . 17
2.2.2 Markov Random Field Model . . . 19
2.2.3 Maximum *a Posteriori* Estimation . . . 23
2.2.4 Confidence Levels of Prediction . . . 25
2.2.5 RNA-Seq Datasets . . . 27
2.2.6 Evaluation Metrics . . . 28
2.3 Experimental Results . . . 29
2.3.1 Selection of Differential Gene Expression Analysis Methods . . . 29
2.3.2 Simulation Studies . . . 30
2.3.3 Performance on Real RNA-Seq Data . . . 35
2.3.4 Performance on Genes with Low Read Counts . . . 42
2.3.5 Comparison with Cuffdiff 2 . . . 44
2.3.6 Consistency of Predictions by DESeq and MRFSeq . . . 45
2.4 Conclusion . . . 47

3 SDEAP: A Splice Graph Based Differential Transcription Expression Analysis Tool for Population Data **53**
3.1 Introduction . . . 53
3.2 Methods . . . 58
3.2.1 Discovery of ASMs . . . 60
3.2.2 Evaluation of Expression Features Using ASMs . . . 64
3.2.3 Analysis of Background Variance . . . 66
3.2.4 Testing Differential Transcript Expression . . . 67
3.2.5 Evaluation Metrics . . . 71
3.3 Experimental Results . . . 72
3.3.1 Experiments on Simulated Data . . . 74
3.3.2 Experiments on Real Data . . . 85
3.4 Conclusion . . . 92

4 GDNorm: An Improved Poisson Regression Model for Reducing Biases in Hi-C Data **94**
4.1 Introduction . . . 94
4.2 Methods . . . 98
4.2.1 Genomic Features . . . 98
4.2.2 A Bias Correction Method Based on Gradient Descent . . . 99
4.3 Experimental Results . . . 106
4.3.1 Simulation Studies . . . 106
4.3.2 Performance on Real Hi-C Data . . . 111
4.4 Conclusion . . . 121

Bibliography **122**

List of Figures

1.1 The central dogma of molecular biology. Source: adapted from [6] . . 2
1.2 RNA-seq workflow. Source: adapted from [129] 5
1.3 Overvoew of the Hi-C protocol. Plot (a) shows a typical Hi-C experiment and plot (b) illustrates a contact frequency matrix and the corresponding chromosome structures reconstructed from the matrix. Source: adapted from [74] 12

2.1 The precision-sensitivity curves comparing the prdiction accuracy of all methods on the simulated datasets in the interval [0,10) of true DE genes. Clearly, MRFSeq has the best overall performance. 34
2.2 Performance assessment at various sequencing depths. The X-axis shows the number of used lanes and the Y-axis indicates various assessment measures. In the upper column of plots, the LR threshold b is set as 0.5 and in the lower column b=2.0. Plots (a) and (d) compare the precision scores at different sequence depths. Plots (b) and (e) depict the sensitivity scores while plots. (c) and (f) illustrate the F-scores. . 39
2.3 Comparison of the methods when different confidence thresholds are applied. Plots (a) and (b) show the precision-senitivity curves when the LR threshold b is set as 0.5 and 2.0, respectively. 40
2.4 The precision-sensitivity curves assess the prdiction performance of MRFSeq and Cuffdiff 2 on the MAQC dataset. The dotted line shows the sensitivity value corresponding to the common FDR threshold 0.1. Note that sensitivity increases with FDR, and thus the region to the left of the dotted line might be more interesting in practice. 45
2.5 Comparison of the average edge degrees of incorrectly inverted genes and all genes in the coexpression networks used in the simulated and MAQC dataset. S_{avg} is the average edge degree of all genes in the coexpression networks used in the simulation while $S_{i.i.}$ is the average edge degree of all incorrectly inverted genes. M_{avg} is the average edge degree of all genes in the coexpression networks used in the MAQC datasets while $M_{i.i.}$ is the average edge degree of the incorrectly inverted genes. 47

2.6 Precision-sensitivity curves for comparing the prediction accuracy of MRFSeq and SimpleNetwork on the MAQC dataset with the LR values (a) b=0.5 and (b) b=2. 49
2.7 Precision-sensitivity curves for comparing the prediction accuracy of MRFSeq_NOISeq, MRFSeq, DESeq, and NOISeq on the (a) simulated and (b) MAQC datasets. 52

3.1 A counterexample to the graph modular decomposition algorithm used in DiffSplice. Plot (a) shows the counterexample where the vertices of the two ASMs $H(v_1, v_6)$ and $H(v_1, v_7)$ n ot detected by the algorithm are highlighted in yellow and red, repectively. In plot (b), the reduced graph $H(u, v)/E_{max}$ is illustrated. 64
3.2 The dendrograms of the hierarchical clustering for the breast cancer dataset. Plots (a) and (b) depict the clustering by SDEAP and DEXUS$_{exon}$. The Y-axis is the measurement of similarity between the samples and X-axis are the labels of each sample. The HER2 samples are colored red, the TNBC samples green and the non-TNBC samples blue. The three red boxes in each dendrogram illustrates three clusters obtained by the corresponding method. 87
3.3 Hierarchical clustering of 12 mECS and 12 Pr cells based on the DTE genes predicted by SDEAP and DEXUS$_{exon}$. Plots (a) and (b) depict the dendrograms obtained by DEXUS$_{exon}$ and SDEAP, respectively. The Y-axis is the measurement of similarity between the samples and the X-axis shows the labels of the mESC and Pr cells. The mESC cells are colored red and the Pr cells blue. The two red boxs illustrate two clusters obtained from each clustering consistent with the cell types. 88
3.4 The PCA transformation of expression features and the hierarchical clstering of the mESC cells using the DTE features identified by SDEAP and DEXUS$_{exon}$. Plots (a) and (b) are the projections of predicted DTE features by SDEAP and DEXUS$_{exon}$. Every red dot is a cell in the G1 cell-cycle phase and every blue dot a cell in the G2/M phase. Cells in the S phase are represented by green dots. Plots (c) and (d) depict the dendrograms made from the DTE features predicted by SDEAP and DEXUS$_{exon}$. The Y-axis is the measurement of similarity between samples and the X-axis shows the labels of the mECS cells in the three cell-cycle phases. The labels are colored in the same away as in plots (a) and (b). 93

4.1 Alignment between the reference chromosome 3D structures and structures predicted by GDNorm$_{sde}$, HiCNorm$_{sde}$ and BACH on simulated data. The red curves indicate the predicted structures and blue curves the reference structures. The results of GDNorm$_{sde}$, HiCNorm$_{sde}$ and BACH are shown from left to right. The top row is for the helix and bottom for the random walk. The quality of each structural alignment is evaluated by an RMSD value. 106
4.2 Comparison of the predicted spatial distance values with the 10 greatest and 10 smallest systematic biases. For each structure prediction method studied, two sets of 10 distance values form the two boxes in a comparison group. The left box depicts the distribution of the distance values for contacts with the greatest systematic biases while the right shows the distribution of the distance values for contacts with the smallest systematic biases. Clearly, GDNorm$_{sde}$ produced the most consistent distance values and HiCNorm$_{sde}$ the least. 111
4.3 Comparison of the reproducibility between two biological replicates achieved by GDNorm, HiCNorm, YT, ICE, and BACH on the 23 chromosomes, chr1 to chr23 (chrX), in the GM06990 cell line at 1M resolution. The distribution of Spearman's correlation coefficients achieved by a bias reduction method is represented as a solid curve over the 23 chromosomes. Plot (a) illustrates the overall reproducibility and plot (b) shows the reproducibility of high contact frequencies (RHCF). . . 114
4.4 Comparison of the reproducibility in the mESC dataset. Plots (a) and (b) illustrate the overall reproducibility and RHCF of GDNorm, HiCNorm, YT, and ICE on the 20 chromosomes, chr1 to chr20 (chrX), in the mESC cell line at 40kb resolution, respectively. Here, the distribution of Spearman's correlation coefficients achieved by each bias reduction method is represented as a solid curve over the 20 chromosomes. Plots (c) and (d) show the overall reproducibility and RHCF of GDNorm and BACH at 1M resolution, respectively. 115
4.5 The running time of GDNorm and HiCNorm on the mESC data at four different resolutions. The Y-axis shows the running time in seconds and the X-axis indicates the number of genomic segments at each resolution. 120

List of Tables

2.1 The pathways used in the simulation study. Nodes and Edges represent the number of nodes and edges in the coexpression network respectively. 31

2.2 Comparison of different methods on simulated datasets. Levels shows the range of the abundance levels of DE genes. Avg is the average percentage of DE genes among the 10 test datasets at the level. Methods are the names of the methods 32

2.3 Comparison of the prediction accuracy on Griffith's dataset. TP is the number of true positives and PP is the number of predicted positives. 42

2.4 Comparison of the prediction results on genes with low read counts. $\text{RTP}_{l/h}$ is the ratio of true positives with low read counts over the true positives with high read counts. $\text{RPP}_{l/h}$ is the ratio of predicted positives with low read counts over the predicted positives with high read counts. 43

2.5 Comparison of the prediction accuracy with Cuffdiff 2. 45

2.6 Performance assessment of MRFSeq on gene coexpression networks obtained by adding random edges. Edge is the percentage of randomly added edges 50

2.7 Performance assessment of MRFSeq on the gene coexpression networks obtained by deleting random edges. Edge is the percentage of randomly deleted edges 50

3.1 Comparison of the two DTE analysis methods on simulated datasets from binary conditions. The configuration (n_1, n_2) indicates the number of replicates in each condition. AUC_{pr}, PRE and REC denote the area under the precision-recall curve, precision and recall scores, respectively, averaged over the 10 repetitions. The standard deviation of each score is included in the parentheses following the score. The last row, AVG, shows the the average of AUC_{pr}, PRE and REC in 6 experiments. 78

3.2 Comparison of the two DTE analysis methods on simulated datasets from 3 or more conditions. The configuration $(n_1, n_2, ...)$ indicates the number of replicates in each condition. Again, AUC_{pr}, PRE and REC denote the area under the precision-recall curve, precision and recall scores, respectively, averaged over the 10 repetitions. The standard deviation of each score is included in the parentheses following the score. The last row, AVG, shows the the averages of AUC_{pr}, PRE and REC in 6 experiments. 79
3.3 Comparison of the two DTE analysis methods on simulated single-cell RNA-Seq data. The configuration $(n_1, n_2, ...)$ indicates the number of replicates in each condition. Again, AUC_{pr}, PRE and REC denote the area under the precision-recall curve, precision and recall scores, respectively, averaged over the 10 repetitions. The standard deviation of each score is included in the parentheses following the score. The last row, AVG, shows the the averages of AUC_{pr}, PRE and REC in 4 experiments. 81
3.4 Comparison of the performance in differential splicing analysis. . . . 83
3.5 Comparison of the numbers of the manually selected and qPCR validated maker genes for the mESC and Pr cells in the DTE genes predicted by SDEAP and $DEXUS_{exon}$. The second column indicates the total numbers of manually selected or qPCR validated marker genes. The numbers of manually selected or validated maker genes that appear in the DTE genes predicted by the two methods are given in the third and fourth columns. 89

4.1 RMSD values of the predicted structures on noisy data. 110
4.2 Correlation between normalized contact frequencies at 40kb resolution and spatial distance measured by FISH experiments in the two biological replicates of the mESC data. 119
4.3 The running time on the GM06990 and mESC datasets. 120

Chapter 1

Introduction

Nucleic acid sequencing is to determine the order of nucleotides in given DNA or RNA molecules and has numerous applications in biology researches. In the past decade, the next generation sequencing (NGS) technologies allow us to sequence all DNA or RNA molecules in cells on a whole-genome scale by generating millions of sequenced cDNA fragments in parallel. Because of its efficiency, NGS has plenty of applications in biological researches such as assembling the sequences of large genomes [5, 126, 76], studying the variation between genomes of the same species [2], reporting the interaction features of DNA-binding proteins [94], and profiling genome-wide epigenetic modifications [72]. However, due to the size and complexity of NGS data, the data analysis of NGS data largely relies on efficienct computational tools. The demand for high performance computational algorithms has been highly emphasized in the literature [6]. Hence, in the past few years, my research interest was focused on innovating and improving computational algorithms for NGS data analysis. In this dissertation, three novel computational algorithms are proposed for differential

transcript expression analysis using high-throughput mRNA sequencing (RNA-Seq) and systematic bias reduction in high-throughput chromosome conformation capture (Hi-C) data. For better understanding the details of the proposed computational algorithms, the background knowledge of the RNA-Seq and Hi-C technologies is reviewed in the sections 1.1 and 1.2 respectively.

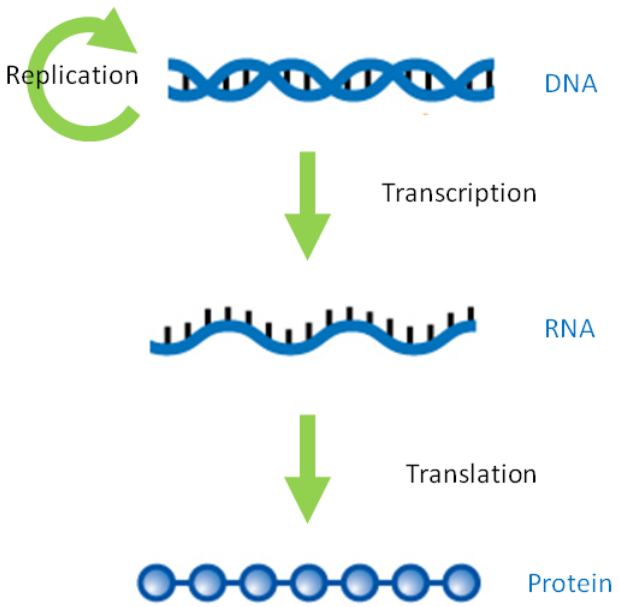

Figure 1.1: The central dogma of molecular biology. Source: adapted from [6]

1.1 Differential Transcript Expression Analysis Based on High-Throughput mRNA Sequencing

Proteins serve as basic function units in a cellular system and synthesized in cells by genes through transcription and translation. As illustrated in Figure 1.1, the process to produce proteins by transcription and translation is called the central dogma of molecular biology. In transcription, messenger RNAs are copied and edited from the exons in individual genes. In translation, amino acids of proteins are assembled using the complementary triplet (or codons) of the transcripted mRNAs. Note that exons of a gene may be joined together in various ways to produce different mRNA variants known as isoforms. The process of synthesizing the variants of mRNA is called alternative splicing. In the human genome more than 90% of the genes have multiple mRNA isoforms [69].

The transcriptome is the set of all RNA molecules, including mRNA, rRNA, tRNA, and other non-coding RNA transcribed in cells. Uncovering the complexity of transcriptome provides great insight into the biological processes of cellular activity and has long been of interest to biologists [129]. In recent years, RNA-Seq has taken a major role in the quantitative analysis of transcript expression and variant discovery and becomes a vital component for both discovery and quantification of transcripts in these genomic researches [123]. The overall workflow of RNA-Seq is illustrated in Figure 1.2. To measure the expression levels of transcripts, a collection of purified

RNAs is first sheared and converted into a cDNA fragments. The cDNA fragments are sequenced from either one or both ends on a high-throughput platform such as Illumina, SOLiD or Roche454. Every short sequence of the cDNA fragments obtained by the process is called a read. To measure abundance of transcript, then millions of reads are aligned to known reference genome sequences by computational alignment algorithms such as Tophat [121]. All mapped reads can be divided into three categories. The first are junctions, reads spanning multiple exons. The second are exonic reads, reads completely falling into a exonic region. The other is Poly-A reads, reads containing Poly-A tails. Note that the Poly-A tails of the mapped reads are usually truncated after being aligned. The proportion of reads matching a given transcript is used as quantification of the expression level of the transcript [78].

In addition to the quantification of transcript expression, studying the regulation of transcripts in biological processes of interests requires sensitive differential expression analysis to compare transcript expression in RNA-Seq samples. Earlier differential transcript expression analysis was mostly done at gene level. . However, most of the differential gene expression analysis tools have poor performance on lowly expressed or short genes. The estimation bias against lowly expressed or short genes may further propagate in downstream analyses at the systems biology level if it is not corrected. To obtain a better inference of differential gene expression, we propose a new efficient algorithm based on a markov random field (MRF) model, called MRF-Seq, that uses additional gene coexpression data to enhance the prediction power.

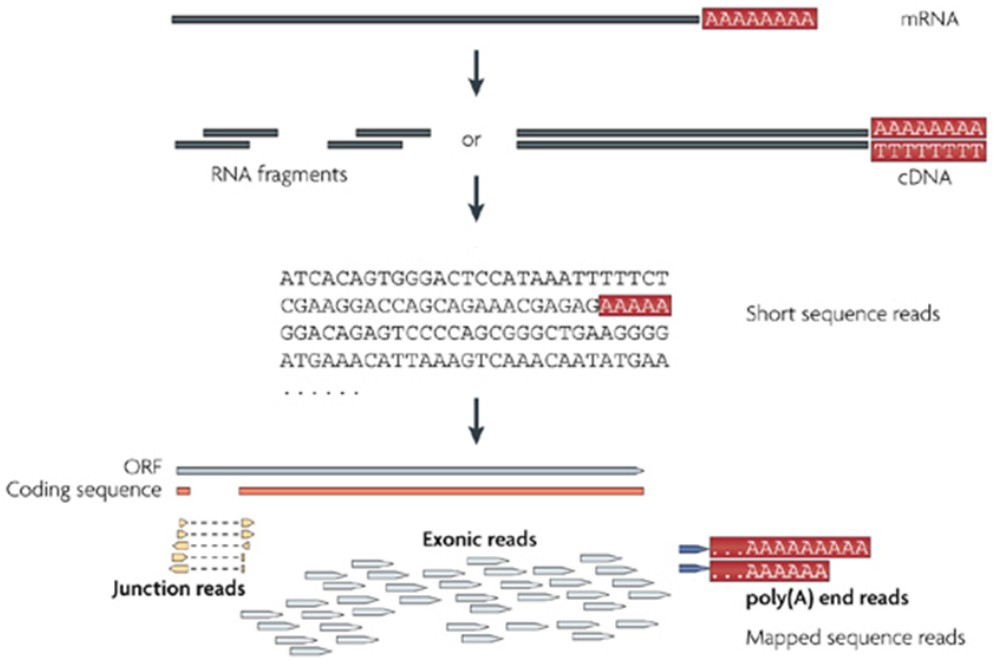

Figure 1.2: RNA-seq workflow. Source: adapted from [129]

As illustrated in Chapter 2, our main technical contribution is the careful selection of the clique potential functions in the MRF so its maximum *a posteriori* (MAP) estimation can be reduced to the well-known maximum flow problem and thus solved in polynomial time. The results from our simulated and real RNA-Seq experiments demonstrate that MRFSeq is more accurate and less biased than the existing methods based on RNA-Seq data alone and suggeste MRFSeq could serve as a vital component in many genomic researches.

Although, recently, several computational tools have been proposed for performing differential expression analysis at both gene and transcript levels [123], most of the differential expression analysis tools were developed to compare RNA-Seq samples of known biological conditions. The requirement of the predefined conditions restrict the applications of differential expression analysis to the case-control study. Differential expression analysis without predefined conditions is also critical to biological studies. For example, it can be used to discover biomarkers to classify cancer samples into subtypes such that better diagnosis and therapy methods can be developed for each subtype [66]. To the best of our knowledge, there is no method for performing differential transcript expression analysis without predefined conditions in the literature. Hence, we propose the first differential transcript expression analysis algorithm, called SDEAP, to compare transcript expression in RNA-samples without given predefined conditions. As demonstrated in Chapter 3, by taking advantage of a new graph modular decomposition algorithm on splice graphs for discovering alternative splicing events and a robust clustering approach to deal with data from an arbitrary number of conditions, SDEAP is able to provide accuracy prediction in differential expression analysis. Moreover, the prediction of SDEAP allows us to classify the subtypes of breast cancer and the cycle phases of mESC cells correctly.

1.2 Chromosome Conformation Capture using High-Throughput Sequencing and Bias Reduction

Three dimensional (3D) conformations of chromosomes in nuclei have known to be highly involved in many chromosomal mechanisms such that studying the variation of chromosome conformations becomes a important topic in epigenetics researches [74]. As a revolutionary tool, the Hi-C technology enables the study of chromosome structures at an unprecedentedly high throughput and resolution. The workflow of Hi-C technology is shown in Figure 1.3(a). Chromosomes of cells are firstly cross-linked with formaldehyde. DNA is digested with a restriction enzyme, e.g., NcoI or HindIII, that leaves a $5'$ overhang. The $5'$ overhang is filled with a biotinylated residue that results in a blunt-end fragment. Every blunt-end fragment between the cross-linked DNA fragments is then ligated under dilute conditions. Here, every ligated fragment represents a contact of two chromosome segments that are originally closed to each other in the nucleus. All ligated fragments are then sheared by sonication. Among all the sheared fragments, the fragments containing biotin are pulled down by antibodies and sequenced from their both ends using massively parallel DNA sequencing. To summarize the total amount of the contacts between every pair of genomic regions, the pair-end reads are aligned to the reference genome sequences. The number of mapped pair-end reads spanning two genomic regions, called contact frequency, is used to measure the spatial proximity of the two regions. In general, the

contact frequency of two genomic regions is assumed to be negatively proportional to the spatial proximity of the two regions. The contact frequencies of all the pairs of genomic regions are usually presented as a two dimensional matrix where the rows and columns correspond to the genomic regions as shown in Figure 1.3(b). The two dimensional matrix of contact frequencies can be alternatively considered as a distance matrix of genomic regions such that the three dimensional structure of chromosomes can be reconstructed from the distances of genomic regions. More experimental details of the Hi-C technology are discussed in [74]

During the experimental steps of Hi-C, systematic biases from different sources are introduced into the Hi-C data such that some parts of the genome are under or overrepresented in the terms of contact frequencies [135]. Regions with a high density of restriction fragments tend to be overrepresented in the read library. Fragments of various lengths have different propensity of forming ligation products with other fragments such that longer fragments appearing more frequently in true ligation events compared to shorter ones. The uniqueness of the genome sequence is called mappability. In Hi-C data, only uniquely mapped reads are used such that low-mappability (repetitive) regions contain fewer uniquely mapped reads than high-mappability regions. GC-content mainly affects the polymerase chain reaction amplification and results in different amplification efficiency for GC-rich and GC-poor sequences.

Removing the systematic biases from Hi-C data is essential to all applications of Hi-C data. Hence, we propose an improved Poisson regression model and an efficient

gradient descent based algorithm, GDNorm, for eliminating biases in Hi-C data. The details of the algorithm are presented in Chapter 4. GDNorm has been tested on both simulated and real Hi-C data. The experimental results show that GDNorm is able to conduct more comprehensive bias reduction and leads to better chromosome structure prediction when combined with a chromosome structure determination method such as ChromSDE. Moreover, the corrected Hi-C data obtained by GDNorm are well correlated to the spatial distance measured by florescent in situ hybridization (FISH) experiments. In addition to accurate bias reduction, GDNorm had the highest time efficiency on the real data.

1.3 Organization of the Rest of the Dissertation

The rest of this dissertation consists of three self-contained chapters. In each of the three chapters, I will review relevent biological background and previous work, define computational problems, present corresponding algorithmic solutions, and discuss experimental results. The two novel differential expression analysis tools, MRFSeq and SDEAP, are introduced in Chapter 2 and 3 respectively while the bias reduction tool GDNorm is presented in Chapter 4.

Chapter 2 starts with a comprehensive review of differential expression analysis at gene level in Section 2.1. The algorithm of MRFSeq is given in Section 2.2. The terms and notations used in our algorithms are defined in Section 2.2.1 while Section 2.2.2 provides the formulation of the Markov random field model and the design of its clique

potential functions. The parameter estimation of the Markov random field model is shown in Section 2.2.3 and 2.2.4. The experimental results are described in Section 2.3, which also contains a comparison between MRFSeq and existing differential expression analysis methods. In particular, Section 2.3.4 compares the performance of the methods on genes with low read counts and shows that MRFSeq achieves not only an overall significantly higher accuracy but also provides a less biased prediction. A few concluding remarks are given in Section 2.4.

In Chapter 3, related work on differential expression analysis with or without predefined biological conditions are reviewed in Section 3.1, where the motivation of designing a differential transcript expression analysis tool without biological conditions is emphasized. The main algorithm of SDEAP is illustrated in Section 3.2. Section 3.2.1 provides the graphical modular decomposition algorithm to locate alternative splice events in genes. The quantification of transcript expression in SDEAP is shown in Section 3.2.3. The details of the differential expression test procedure of SDEAP are presented in Section 3.2.3 and 3.2.4. The experimental results are demonstrated in Section 3.3. In Section 3.3.1, the performance of SDEAP is assessed by several benchmark datasets simulated using configurations from real RNA-Seq data. Moreover, SDEAP is used to identify differential expressed transcripts in real RNA-Seq datasets containing samples of different biological conditions. The predicted differential expressed transcripts are compared with the qPCR validation and used to classify the samples of different biological conditions in the datasets. The results of

the classification is presented and discussed in Section 3.3.2. The contributions and future work of SDEAP are concluded in Section 3.4.

Chapter 4 presents a comprehensive study on bias reduction for Hi-C data. Overview of the Hi-C experimental protocol is provided in Section 4.1 where systematic biases in Hi-C data and existing computational methods to remove the biases are also introduced. The details of the GDNorm algorithm are described in Section 4.2. The causal relationship between genomic features and the systematic biases is formulated in Section 4.2.1. Based on the formulation, in Section 4.2.2, a Poisson regression model is developed to estimate and remove the biases from Hi-C data. Several experimental results on simulated and real human and mouse data are presented in Section 4.3.1 and 4.3.2 respectively. The experimental results by different bias reduction methods are discussed compared in terms of accuracy, reproducibility and time efficiency. The contributions of GDNorm are concluded in Section 4.4.

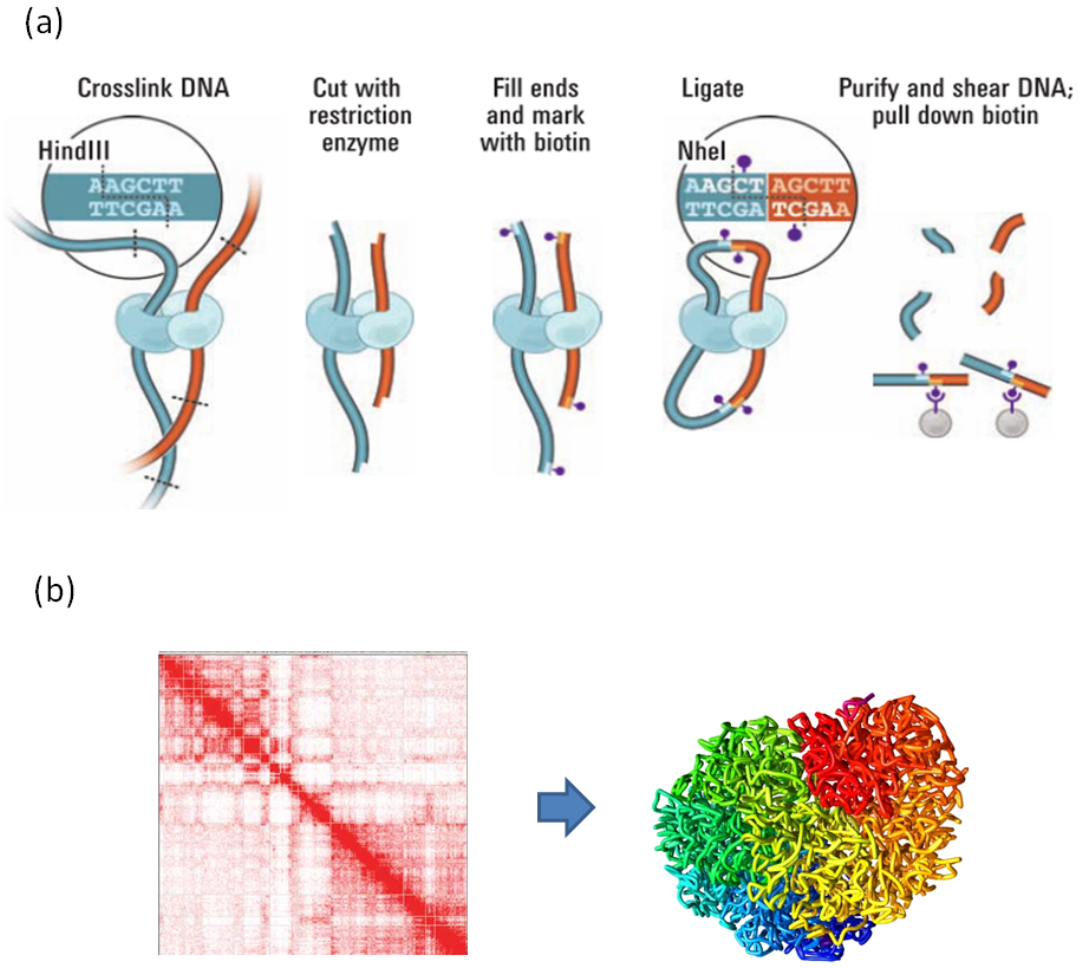

Figure 1.3: Overvoew of the Hi-C protocol. Plot (a) shows a typical Hi-C experiment and plot (b) illustrates a contact frequency matrix and the corresponding chromosome structures reconstructed from the matrix. Source: adapted from [74]

Chapter 2

Differential Gene Expression Analysis Using Coexpression and RNA-Seq Data

2.1 Introduction

Next generation sequencing technologies (NGS) have been widely used in genomics research. RNA-Seq, one of the most exciting applications of NGS, is used to reveal the complexity of tanscriptomes in biological systems [129]. Many unprecedented discoveries are being made by RNA-Seq, such as the inference of novel isoforms, characterization of the modes of antisense regulation and study of intergenic expression patterns [18, 84, 44, 122]. In recent years, RNA-Seq has taken a major role in the quantitative analysis of gene expression and transcript variant discovery. In the past decade most of these applications were dominated by microarray-based technologies. In these quantitative assays, RNA populations are partially sequenced and the obtained read sequences are aligned back to the reference genome. The aligned reads

are then assigned to genes based on the common regions that they share in the alignment. The number of reads assigned to a gene is called the *read count* of the gene which has been shown to be nearly linearly correlated with the expression level of a gene [78].

Differential gene expression analysis is to identify if genes express differently between biological conditions of interest. Given RNA-Seq read count data, detecting differentially expressed (DE) or equally expressed (EE) genes can be done by checking if the observed difference of the read counts is significant or not, *i.e.*, greater than some natural random variation. To test the significance of the difference between RNA-Seq read counts, the distribution of read counts was first assumed to be Poisson in [78, 128, 112]. However, the Poisson distribution may underestimate the variance of read counts and cause unexpected false positives in differential gene expression analysis [84, 102]. To solve the problem, negative binomial distributions were applied to RNA-Seq read data [4, 102, 103, 101] and have become the-state-of-the-art statistical model. Other than the methods based on the Poisson or negative binomial distributions, two data-driven probabilistic methods, baySeq [46] and NOISeq [118], have also been proposed. Moreover, given annotated or inferred mRNA transcripts (or isoforms) of genes, some statistical methods for detecting differential expression at the transcript level have been published recently [70, 123, 45, 143]. Since the expression level of a gene with known (or inferred) isoforms can be calculated by simply summing up the expression levels of its isoforms, these transcript-level methods can

be used as alternative methods for detecting differential expression of isoforms [123], although the accuracy of these methods clearly depends on the quality of the provided isoforms.

Although the statistical properties of RNA-Seq data have been well studied and taken into account in the above statistical methods, these methods suffer from the following issues. First, it has been observed that statistical power increases with read count values [93, 92, 137]. Note that the read count of a gene is proportional to the gene expression level multiplied by the gene length. As a result, long or highly expressed genes are more likely to be detected as DE genes compared with their short and/or lowly expressed counterparts. This bias in DE gene detection is unavoidable even when normalization or rescaling is applied to read count data [93, 137]. It is known that the selection bias on DE genes, if uncorrected, may lead to biased downstream analyses [93, 92, 137]. Second, the dependency among the expression of genes is not utilized in these methods. In gene expression analysis based on microarray data, the prior knowledge of gene coexpression patterns has been used to improve the performance of algorithms for detecting phenotype-related pathways [99], searching for significant pathway regulators [110], identifying differential gene expression patterns [53], and the classification of microarray data [100]. In particular, to obtain more accurate inference of DE genes, Wei and Li [133] proposed a *markov random field* (MRF) model that integrates the gamma-gamma model based on microarray data [87, 58] and gene coexpression networks extracted from KEGG pathways [56]

such that DE genes can be determined by the *maximum a posteriori* (MAP) estimation of the MRF model. Their experimental results demonstrate that the additional gene coexpression information can help detect more subtle changes of gene expression (*e.g.*, local disturbances within known pathways) and significantly improve the overall prediction accuracy of DE genes [133]. However, due to the difference between continuous microarray intensity values and discrete RNA-Seq read counts, The MRF model in [133] cannot be applied to RNA-Seq data immediately. Moreover, since the MAP estimation problem for an MRF model is generally NP-Hard [11], the MRF model in [133] was solved by a heuristic method, *iterated conditional modes* (ICM), which provides an approximately optimal prediction with no confidence scores.

In this work, we propose a novel MRF model, MRFSeq, combining RNA-Seq read counts with the prior knowlwdge of gene coexpression networks to infer DE genes. Different from the MRF model in [133], we choose the clique potential functions of the MRF model carefully so that the MAP estimation of DE genes can be reduced to the well-known maximum flow problem on flow networks based on the work of Kolmogorov and Zabih [64]. Since the maximum flow problem is polynomial-time solvable, our MRF model can be solved exactly in polynomial time. Moreover, we introduce a *loopy belief propagation* method [134, 82] to calculate the confidence of each inferred DE or EE gene. Our extensive experiments on simulated and real RNA-Seq data demonstrate that MRFSeq achieves a much improved overall estimation performance by gaining considerable sensitivity without losing precision. A detailed

analysis of the prediction results indicates that the DE genes predicted by MRFSeq are distributed more evenly across different values of read counts than those recovered by the existing methods using RNA-Seq data alone. Hence, MRFSeq can help alleviate the selection bias of DE genes against genes with low read counts. Our analysis further shows that most of the DE or EE genes that can be correctly predicted from RNA-Seq data alone are also correctly predicted by MRFSeq, implying that the use of the prior knowledge of gene coexpression does not introduce new biases in the differential analysis result. Moreover, we compare MRFSeq with a very recently published transcript-level method, Cuffdiff 2 [123], on the real RNA-Seq data using the annotated transcriptome from UCSC hg19 [81]. The comparison shows that MRFSeq is much more sensitive than Cuffdiff 2.

2.2 Methods

2.2.1 Terminology and Notations

Let $G = \{g_1, g_2, ..., g_n\}$ be the genes to be tested for differential expression and $X = \{x_1, x_2, ..., x_n\}$ the binary random variables such that each $x_i \in \{0, 1\}$ indicates the DE state of gene g_i. The random variable $x_i = 1$ if the gene g_i is a DE gene and $x_i = 0$ indicates that the gene is an EE gene. Two random variables x_i and x_j are assumed to be correlated if the two genes g_i and g_j form a pair of coexpressed genes. A configuration x is a 0-1 assignment to the random variables X. Assume that there

are p and q replicates in the two conditions, A and B, of interest, respectively. Let the read counts a_j^i and b_j^i be the number of the reads aligned to gene g_i in the j-th replicate of the conditions A and B, respectively. For each gene g_i, two sets of the read counts $R_{A,i} = \{a_1^i, a_2^i, ..., a_p^i\}$ and $R_{B,i} = \{b_1^i, b_2^i, ..., b_q^i\}$ are summarized from all the replicates of the two conditions A and B after mapping all the reads to the reference genome. Popular statistical measurements for the observed difference of read counts are the false discovery rates (FDR, *i.e.*, the p-value corrected for multiple testing [8]) and prior probability. The current statistical methods infer DE genes by checking independently for each gene if the difference measurement of its read count exceeds a certain threshold [92]. In our method, DE genes are determined by the configuration that maximize a likelihood function of both observed difference of read counts and gene coexpression while no prior knowledge of the thresholds is required. MRFSeq uses, but is not limited to, the FDR q_i from DESeq [4] as the difference measurement of the read counts $R_{A,i}$ and $R_{B,i}$, where $q_i \in [0, 1]$. To improve the computational efficiency of our algorithm, the FDR q_i is further discretized by binning the interval [0,1] into 20 intervals of the same length 0.05. Let $y_i \in \{1, 2, , 20\}$ denote the interval where the observed difference q_i belongs to and $Y = \{y_1, y_2, ..., y_n\}$ be the collection of all the discretized FDRs. The joint probability of the hidden variables X given its observed values Y is then formulated by an MRF model, a graphical model capable of capturing the statistical dependency of random variables [62], described in the next subsection. Given the joint probability of X conditional to Y, estimating the

DE states of the genes actually involves two inference problems. The first is the MAP estimation problem, i.e., searching for a configuration x^* such that $Pr(x^*|Y)$ is maximized. The algorithm for the MAP estimation problem will be discussed later in the section. The second is the *marginal probability* problem, *i.e.*, computing the probability $Pr(x_i|Y)$ as a confidence level of the configuration on each gene g_i. The loopy belief propagation method for the *marginal probability* problem is given in the supplementary materials.

2.2.2 Markov Random Field Model

Let $H = (V_x, E)$ be an undirected graph representing the coexpression network for G such that every node $v_{x_i} \in V_x$ is associated with the random variables $x_i \in X$ and every edge (i, j) shared by the nodes v_{x_i} and v_{x_j} encodes the dependency of the two correlated random variables x_i and x_j. Two variables x_i and x_j are assumed to be correlated if the two genes g_i and g_j are coexpressed. To determine which pair of the genes are the coexpressed genes, the correlation coefficient $c_{i,j}$ defined in COXPREdb [89] is used as the measurement of gene coexpression between the two genes g_i and g_j. Two genes are considered as a pair of coexpressed genes if $c_{i,j}$ is greater than a threshold ρ. We use $\rho = 0.5$ throughout this work because it is widely used in the literature [95, 131].

In our model, we think that the DE state of each gene should depend on its observed difference in read counts and the DE states of its coexpressed genes. In

other words, we can assume that every random variable is conditionally independent to the variables indexed by non-adjacent vertices in H. Hence, the following property is satisfied:

$$Pr(x_i|X) = Pr(x_i|x_j, v_{x_j} \in N(v_{x_j})), \tag{2.1}$$

where $N(v_{x_j})$ represents the neighbors of v_{x_i} in H. By the Hammersley-Clifford theorem [9], a joint distribution of the random variables X given Y can be factorized as a form of clique potential functions $T_C(C)$, the positive functions for configurations over cliques in the given graph H such that $Pr(X|Y) = \prod_{C \in H} T_C(C)$.

To model the pairwise dependency between coexpressed genes, we may use an MRF model consisting of only potential functions for cliques of sizes at most 2. This type of MRFs is called the *pairwise* MRFs [10] and will be used in our work. There are two types of potential functions adopted in our MRF model. One is the unary functions $\phi_i(x_i)$ that score how compatible the random variable x_i is with its observed evidence y_i. The other is the pairwise potential functions $\psi_{(i,j)}(x_i, x_j)$ that measure the statistical dependency between every pair of correlated variables x_i and x_j. By the definition of the potential functions, the joint distribution of X given Y can be written as:

$$Pr(X|Y) = \tfrac{1}{Z} \prod_{(i,j) \in E} \psi_{(i,j)}(x_i, x_j) \prod_{i=1}^{n} \phi_i(x_i), \tag{2.2}$$

where Z is the normalized term to assure that the joint probability $Pr(X|Y)$ sums up to 1. Let $P_{(1,i)} = Pr(x_i = 1|y_i)$ and $P_{(0,i)} = Pr(x_i = 0|y_i)$. The unary function

$\phi_i(x_i)$ is defined as follows:

$$\phi_i(x_i) = \begin{cases} P_{(1,i)}/P_{(0,i)}, & \text{if } P_{(1,i)} > P_{(0,i)}, x_i = 1 \\ P_{(0,i)}/P_{(1,i)}, & \text{if } P_{(0,i)} > P_{(1,i)}, x_i = 0 \\ 1, & \text{otherwise,} \end{cases} \tag{2.3}$$

To calculate the unary functions, the ratio between the two prior probabilities $Pr(x_i = 1|y_i)$ and $Pr(x_i = 0|y_i)$ should be given as a known parameter in our MRF model. To estimate the parameter, the read counts of four replicates (two per condition) for 10000 DE genes and 10000 EE genes are first synthesized. Our simulation of the read counts of the DE and EE genes follows the same steps as used in the simulation study of DESeq [4]. For the DE genes, the $\log_2$ fold change rate of the observed read counts between two conditions is randomly drawn from the normal distribution with mean 0 and variance 0.7. For the EE genes, the mean is set to be 0 and the variance 0.2. After the simulation of read counts, the discretized FDRs introduced previously are calculated as the observed difference in the synthesized read counts. Assume that there are m_{y_i} DE genes and n_{y_i} EE genes whose discretized FDR is y_i in this simulation. We further assume the equality of the two background probabilities of x_i holds, *i.e.*, $Pr(x_i = 1) = Pr(x_i = 0)$. By Baye's rule, the ratio of the prior probabilities is obtained as follows:

$$\frac{Pr(x_i = 0|y_i)}{Pr(x_i = 1|y_i)} = \frac{Pr(y_i|x_i = 0)Pr(x_i = 0)}{Pr(y_i|x_i = 1)Pr(x_i = 1)} = \frac{n_{y_i}}{m_{y_i}}, \tag{2.4}$$

Symmetrically, we have $Pr(x_i = 1|y_i)/Pr(x_i = 0|y_i) = m_{y_i}/n_{y_i}$.

For the pairwise function $\psi_{(i,j)}(x_i, x_j)$ of every pair of coexpressed genes g_i and g_j, the correlation coefficient $c_{i,j}$ defined in COXPREdb [89] is used as the measure of the statistical dependency between x_i and x_j. The pairwise potential functions are thus defined as follows:

$$\psi_{(i,j)}(x_i, x_j) = \begin{cases} e^{c_{i,j}}, & \text{if } x_i = x_j, \\ 1, & \text{otherwise,} \end{cases} \tag{2.5}$$

This completes the specification of the joint distribution of X. To facilitate the presentation of our algorithms, the joint distribution of X can be rewritten by taking negative logarithm on both sides of Eq. (2) as below:

$$E(X|Y) = -\gamma - \sum_{i=1}^{n} \alpha_i(x_i) - \sum_{(i,j)\in E} \beta_{(i,j)}(x_i, x_j), \tag{2.6}$$

where γ is a constant, $\alpha_i(x_i) = ln\phi_i(x_i)$ and $\beta_{(i,j)}(x_i, x_j) = ln\psi_{(i,j)}(x_i, x_j)$. $E(X|Y)$ is called the pseudo-energy function when each α_i is a unary term and each $\beta_{(i,j)}$ is a pairwise term of the energy. A configuration maximizing the joint probability $Pr(X|Y)$ is actually the configuration minimizing the pseudo-energy function $E(X|Y)$ [10].

2.2.3 Maximum *a Posteriori* Estimation

Different from the heuristic method, ICM, used to approximate the MAP of the MRF model of Wei and Li [133], we show in this subsection that, by designing the potential functions in MRFSeq carefully, the MAP estimation problem for MRFSeq is no longer an NP-Hard problem because it can be reduced to the maximum flow problem on flow networks and solved optimally in polynomial time.

A random variable x_i is said to be *inverted* by a configuration x if the state assignment to x_i violates its prior probability, *i.e.*, $x_i = 1$ if $Pr(x_i = 0|y_i) > Pr(x_i = 1|y_i)$ or $x_i = 0$ if $Pr(x_i = 1|y_i) > Pr(x_i = 0|y_i)$. For an inverted random variable x_i, $\alpha_i(x_i) = 0$ instead of $|ln\phi_i(1) - ln\phi_i(0)|$. We define $|ln\phi_i(1) - ln\phi_i(0)|$ as the cost of the inversion. Two correlated variables x_i and x_j are said to be *separated* by a configuration x if the assigned states of x_i and x_j are different, *i.e.*, $x_i \neq x_j$. For a pair of separated variables x_i and x_j, $\beta_{(i,j)}(x_i, x_j) = 0$ instead of $c_{i,j}$. The cost of the separation is $c_{i,j}$. Kolmogorov and Zabih [64] proved that when the pairwise term $\beta_{(i,j)}(x_i, x_j)$ of the pseudo-energy function $E(X|Y)$ is *submodular*, that is, the following property is satisfied:

$$\beta_{(i,j)}(0,0) + \beta_{(i,j)}(1,1) \geq \beta_{(i,j)}(0,1) + \beta_{(i,j)}(1,0), \tag{2.7}$$

searching for a configuration that minimizes the pseudo-energy function can be done by looking for a configuration minimizing the total the cost of inversion and sepa-

ration. That is, the MAP estimation problem on an MRF model can be reduced to the maximum flow (or minimum cut) problem over a flow newtwork H' such that a minimum cut of H' corresponds to a MAP estimation of the MRF model and the saturated capacity of the cut is exactly the total cost of the inversion and separation.

It is easy to verify that our paiwise term is submodular. $\beta_{(i,j)}(0,0) + \beta_{(i,j)}(1,1)$ sums up to $2c_{i,j}$, where $c_{i,j} \geq 0.5$, while $\beta_{(i,j)}(0,1) + \beta_{(i,j)}(1,0)$ is 0. The reduction from our MRF model whose graph representation is $H = (V_x, E)$ to the flow network $H' = (V_x \cup \{s,t\}, E')$ can be done as follows. The nodes of H' are the union of the nodes of H and two additional nodes, the source s and sink t. Every undirected edge (i,j) of H is transformed into two directed edges (i,j) and (i,j) with capacity $c_{i,j}$. For every node x_i, two directed edges (s,i) and (i,t) are added to E'. The capacity of the edge (s,i) is $|ln\phi_i(1) - ln\phi_i(0)|$ if $Pr(x_i = 1|y_i) > Pr(x_i = 0|y_i)$. Otherwise, the capacity of the edge (s,i) is 0. Symmetrically, the capacity of the edge (i,t) is $|ln\phi_i(1) - ln\phi_i(0)|$ if $Pr(x_i = 0|y_i) > Pr(x_i = 1|y_i)$. Otherwise, the capacity of the edge (i,t) is 0. After running a standard maximum flow algorithm, *e.g.*, the Edmond and Karp algorithm [32], on the flow network H', a minimum cut $Q = \{V_s \cup s, V_t \cup \{t\}\}$ is obtained, where V_s are the nodes adjacent to s and V_t the nodes adjacent to t. It represents a 0-1 assignment such that all the random variables corresponding to the nodes of V_s are assigned 1 and all the random variables corresponding to the nodes of V_t are assign 0. Then, a gene g_i is inferred as a DE gene if x_i is 1, or an EE gene otherwise.

2.2.4 Confidence Levels of Prediction

To calculate the marginal probabilities of the random variables X (as a way of estimating the confidence of our inferred configurations), Pearl proposed an exact inference algorithm for MRF models whose graph representations are trees [97]. However, to the best of our knowledge, there is no efficient way to calculate the marginal probability of a random variable x_i for MRF models that contain cycles. A popular heuristic algorithm, called the *loopy belief propagation* algorithm [82, 134], will be adopted in our work to approximate marginal probabilites.

The *belief* of a random variable x_i given the observed values Y is the marginal probability $Pr(x_i|Y)$. Loopy belief propagation [82, 134] is a heuristic algorithm to compute the belief of variables by iteratively passing and updating partially computed results, called *messages*, between variables until the belief converges. To be specific, let the function $m_{(i,j)}(x_j)$ denote the message passed from node v_i to node v_j. The message function $m_{(i,j)}(x_j)$ is defined as

$$\begin{aligned} &\rho(i,j) = \textstyle\prod_{v_k \in N(v_i)-\{v_j\}} m_{(k,i)}(x_i), \\ &m_{(i,j)}(x_j) = \alpha \textstyle\sum_{x_i} \psi_{(i,j)}(x_i, x_j)\phi_i(x_i)\rho(i,j), \end{aligned} \tag{2.8}$$

where α is a normalizing constant such that $m_{(i,j)}(0) + m_{(i,j)}(1) = 1$. Then, the belief

of each variable x_i can be written in a product form of the messages as follows:

$$Pr(x_i|Y) = \beta\phi_i(x_i) \prod_{v_j \in N(v_i)} m_{(j,i)}(x_i), \tag{2.9}$$

where β is a constant such that $Pr(x_i = 1|Y) + Pr(x_i = 0|Y) = 1$. In our implementation, the message functions are initialized to the uniform distribution functions. To perform the loopy belief propagation algorithm, a spanning tree of the given graph H is constructed at first and the postorder, $v_{p1}v_{p2}...v_{p_n}$, of the tree nodes is used as the order for propagating messages. At every iteration of i from 1 to n, every message $m_{(p_i,p_j)}(x_{p_j})$ associated with the edge (v_{p_1}, v_{p_2}) is updated by Eq. 2.8. The update is iteratively performed and it terminates when the change of the belief is smaller than 10^{-4}. Once the updating process converges, the marginal probabilities of the variables can be calculated by Eq. 2.9. The loopy belief propagation algorithm is used here as a secondary prediction algorithm that provides us with the marginal probability of each variable as the confidence of prediction. Because it is a heuristic algorithm for calculating the marginal probabilities, for a very small fraction of the genes considered (fewer than 1% in our experiments), the algorithm may yield a conflicting prediction result against the MAP estimation configuration, *e.g.*, $x_i = 1$ in the MAP estimation but the loopy belief propagation algorithm returns $Pr(x_i = 1|Y) < 0.5$. In such a case, the DE state of the concerned gene will be determined by the MAP estimation result.

2.2.5 RNA-Seq Datasets

Two publicly available human RNA-Seq datasets, the MAQC dataset [107, 15] and Griffth's dataset [45], will be used as the benchmark datasets to assess the performance of our selected differential gene expression analysis methods. Each of the dataset is associated with an additional qRT-PCR dataset to validate the DE states of genes. The MAQC dataset consists of two samples, Ambion's human brain reference RNA (brain) and Stratagene's human universal reference RNA (UHR). Each sample provides seven replicates and a total of 45 million single-end RNA-Seq reads of length 35 bps. The read counts for the MAQC dataset is obtained from 71 million uniquely mapped reads calibrated by ReCounts [39]. Griffith's dataset was made from the qRT-PCR validation for the DE or alternatively expressed genes highlighted by ALEXA-Seq [45]. It contains 96 and 198 million pair-end reads across two human colorectal cancer cell lines that only differ in fluorouracil resistance phenotypes. To equilibrate sequencing depth in both samples, as done in [118], the read library size is set to be about 100 million reads per condition. Raw RNA-Seq reads of the MAQC dataset were downloaded from the SRA database [67] while the RNA-Seq reads of Griffith's dataset were downloaded from the FTP site of the ALEXA-Seq website. The gene association across platforms was performed with BioMart [140]. Unmatched genes were discarded in downstream analysis steps. To obtain the read counts for Griffith's dataset, the raw RNA-Seq reads were aligned against the high-coverage assembly of the human genome UCSC hg19 [81] using Tophat [121] where

two mismatches were allowed and reads mapped to multiple locations were removed. Finally, the read counts for each gene in Griffith's dataset were summarized by using the R packages GenomicFeatures and RSamtools from Bioconductor along with the genome annotation information from Ensembl (version 60) [38] and only exonic reads. For a fair comparison, a pseudo read count, 1, was applied to all genes with zero read counts to avoid the divided-by-zero problem in some statistical calculations.

2.2.6 Evaluation Metrics

Following the assessment method of Bullard *et al.* [15], all our experimental results are evaluated in terms of precision (PRE), $\text{PRE} = \text{TP}/(\text{TP} + \text{FP}) \times 100\%$, and sensitivity (SEN), $\text{SEN} = \text{TP}/(\text{TP} + \text{FN}) \times 100\%$, where TP is the number of true positives, FP the number of false positives and FN the number of false negatives. To combine the two evaluation measures, the F-score (FS) [124], defined as $\text{FS} = [2 \times (\text{PRE} \times \text{SEN})/(\text{PRE} + \text{SEN})] \times 100\%$, is used as a measure of the overall performance of a prediction method in our tests.

2.3 Experimental Results

2.3.1 Selection of Differential Gene Expression Analysis Methods

To compare our method with the existing gene differential analysis methods, the same selection criteria proposed by Tarazona *et al.* [118] was followed. However, Fisher's exact test [36], which was compared in [118], was excluded here because its performance was shown to be far lower than those of the other methods. At the end, four methods including edgeR [101], DESeq [4], baySeq [46], and NOISeq [118] were selected to be compared in our tests. Note that NOISeq has two versions, NOISeq_real and NOISeq_sim, and the version NOISeq_real is used in our experiments because numbers of replicates in our simulated and real datasets are always greater than one. Some reasonable cutoff values are required in these methods (except MRFSeq) to decide the significance of a statistical difference measurement. To obtain comparable performance analysis scenarios, the cutoff values adopted in the literature are applied in our experiments. More specifically, the FDR 0.1 chosen in DESeq is used for DESeq and edgeR. We choose the probability 0.8 and 0.999, as done in the work of [118], for NOISeq and baySeq, respectively. Experiments at two levels of difficulty are conducted to compare our method MRFSeq with the other selected methods. At the first level, all read counts of the benchmark datasets are generated from the same distribution as assumed in the simulation studies of DESeq. At the second level, all

read counts of the genes are accumulated from the two real datasets, the MAQC and Griffith's datasets, and may contain low read counts. In addition to the comparisons with the gene-level methods, MRFSeq is also compared with the recently published transcript-level method Cuffdiff 2 on the two RNA-Seq datasets.

2.3.2 Simulation Studies

Simulation experiments

Our simulation experiments follow the framework in [133]. All gene sets associated with the 186 KEGG pathways in MSigDB [117] were downloaded. The coexpression networks of the 186 gene sets were then defined using COXPREdb [89] and they formed 186 undirected graphs. A gene set was discarded if the number of the edges in its coexpression network is less than the number of the nodes. After the filtration, 37 gene sets consisting of 2194 different genes were kept. The 37 coexpression networks listed in Table 2.1 were merged as a global network consisting of 2194 nodes and 8512 edges. All the methods are tested at five different abundance levels of true DE genes. The performance assessment is categorized into five classes, where each class represents a abundance level interval of 10% such that the five classes cover abundance levels of DE genes ranging from 0% to 50% as done in [133]. At each of the five levels, we randomly choose 10 combinations of the pathways to form the sets of true DE genes, while keeping the rest of the genes as true EE genes, such that the

Table 2.1: The pathways used in the simulation study. Nodes and Edges represent the number of nodes and edges in the coexpression network respectively.

Pathways	Nodes	Edges
KEGG ALLOGRAFT REJECTION	38	94
KEGG ALZHEIMERS DISEASE	169	791
KEGG ANTIGEN PROCESSING AND PRESENTATION	89	291
KEGG ASTHMA	30	40
KEGG AUTOIMMUNE THYROID DISEASE	53	120
KEGG CALCIUM SIGNALING PATHWAY	178	436
KEGG CARDIAC MUSCLE CONTRACTION	80	163
KEGG CELL ADHESION MOLECULES CAMS	134	244
KEGG CELL CYCLE	128	618
KEGG CITRATE CYCLE TCA CYCLE	32	40
KEGG COMPLEMENT AND COAGULATION CASCADES	69	139
KEGG DILATED CARDIOMYOPATHY	92	127
KEGG DNA REPLICATION	36	221
KEGG ECM RECEPTOR INTERACTION	84	141
KEGG ENDOCYTOSIS	183	201
KEGG FOCAL ADHESION	201	284
KEGG GRAFT VERSUS HOST DISEASE	42	113
KEGG HUNTINGTONS DISEASE	185	857
KEGG HYPERTROPHIC CARDIOMYOPATHY HCM	85	110
KEGG INTESTINAL IMMUNE NETWORK FOR IGA PRODUCTION	48	54
KEGG LEISHMANIA INFECTION	72	96
KEGG MAPK SIGNALING PATHWAY	267	321
KEGG MISMATCH REPAIR	23	56
KEGG NATURAL KILLER CELL MEDIATED CYTOTOXICITY	137	235
KEGG NEUROACTIVE LIGAND RECEPTOR INTERACTION	272	487
KEGG NUCLEOTIDE EXCISION REPAIR	44	52
KEGG OOCYTE MEIOSIS	114	188
KEGG OXIDATIVE PHOSPHORYLATION	135	854
KEGG PARKINSONS DISEASE	133	801
KEGG PRIMARY IMMUNODEFICIENCY	35	52
KEGG PROTEASOME	48	291
KEGG RIBOSOME	88	2545
KEGG SPLICEOSOME	128	703
KEGG SYSTEMIC LUPUS ERYTHEMATOSUS	140	296
KEGG TYPE I DIABETES MELLITUS	44	97
KEGG T CELL RECEPTOR SIGNALING PATHWAY	108	129
KEGG VIRAL MYOCARDITIS	73	116

Table 2.2: Comparison of different methods on simulated datasets. Levels shows the range of the abundance levels of DE genes. Avg is the average percentage of DE genes among the 10 test datasets at the level. Methods are the names of the methods

Levels	Avg (%)	Methods	PRE (%)	SEN (%)	FS (%)
[0,10)	5.7	**MRFSeq**	**75.55(11.0)**	**71.99(12.8)**	**73.36(10.3)**
		baySeq	66.23(10.2)	53.49(4.3)	59.02(6.4)
		DESeq	68.57(10.3)	47.78(4.7)	55.87(4.3)
		edgeR	63.07(12.8)	57.07(2.9)	59.11(4.8)
		NOISeq	50.04(17.3)	58.32(3.0)	52.29(9.3)
[10,20)	15.3	**MRFSeq**	**71.70(4.1)**	**72.10(7.3)**	**71.70(4.4)**
		baySeq	68.70(3.1)	61.50(0.8)	64.90(1.4)
		DESeq	73.60(3.0)	53.40(1.2)	61.80(0.7)
		edgeR	74.00(3.5)	54.90(1.6)	63.00(0.4)
		NOISeq	68.70(4.3)	55.90(0.7)	61.60(1.4)
[20,30)	22.2	**MRFSeq**	**74.50(3.6)**	**72.20(5.1)**	**73.20(3.0)**
		baySeq	70.00(1.8)	63.10(0.8)	66.40(1.0)
		DESeq	75.90(2.2)	53.60(0.5)	62.80(0.9)
		edgeR	77.10(2.3)	52.80(0.8)	62.60(0.4)
		NOISeq	75.90(2.5)	46.70(0.7)	57.80(0.6)
[30,40)	32.2	**MRFSeq**	**77.50(2.5)**	**68.10(4.4)**	**72.40(3.1)**
		baySeq	71.00(1.1)	66.40(1.0)	68.70(0.9)
		DESeq	78.90(1.4)	55.00(0.4)	64.80(0.6)
		edgeR	79.70(1.4)	51.70(0.5)	62.70(0.2)
		NOISeq	78.60(1.5)	45.20(0.7)	57.40(0.3)
[40,50)	41.7	**MRFSeq**	**83.70(2.0)**	**70.90(2.2)**	**76.70(1.8)**
		baySeq	75.10(1.6)	68.90(1.6)	71.90(1.6)
		DESeq	83.80(2.3)	55.70(0.3)	66.90(0.9)
		edgeR	84.70(2.1)	50.30(0.4)	63.10(0.3)
		NOISeq	83.80(2.3)	43.80(0.3)	57.50(0.4)

percentage of the true DE genes is within the range of the level. The 10 different combinations form 10 benchmark datasets and read counts are randomly obtained by following the same steps for simulating read counts used in DESeq. The simulated read counts range from 25 to 401. All the methods are applied to the 50 benchmark datasets. The complete assessment on all 5 intervals is presented in Table 2.2. For the convenience of the reader, the precision-sensitivity curves are also provided in Figure 2.1 .

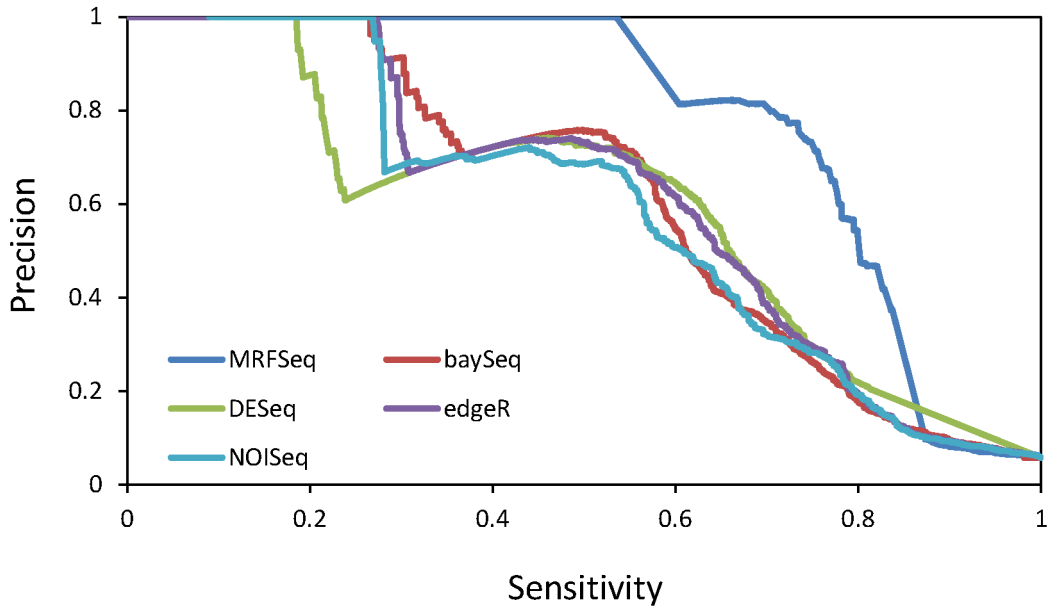

Figure 2.1: The precision-sensitivity curves comparing the prdiction accuracy of all methods on the simulated datasets in the interval [0,10) of true DE genes. Clearly, MRFSeq has the best overall performance.

Comparisons of the methods on simulated data

MRFSeq has clearly the best F-scores (i.e., the overall performance) and significantly improved sensitivity over the other methods. Its F-score is 14.2%, 6.8%, 6.8%, 3.7%, and 4.8% greater than the second best in the five interval while its improvement on sensitivity is 13.6%, 10.6%, 9.1%, 1.7%, and 2%, respectively. Although baySeq provides close sensitivity scores in the intervals [30,40) and [40,50), it fails to obtain comparable precision scores and hence has an inferior overall performance. While achieving a considerable improvement on sensitivity in the interval [0,10), MRFSeq

improves the precision by at least 6.9%. In the other four intervals, MRFSeq's precision is slightly lower than those of the other methods. The difference between the precision of MRFSeq and the best precision in these intervals is 2.3%, 1.4%, 2.2%, and 1%, respectively, which are actually smaller than the standard deviations. The standard deviations of the sensitivity and F-score of MRFSeq are greater than the standard deviations of the other methods. This is because the performance of MRFSeq is somewhat correlated to the topological distributions of the true DE genes on the coexpression network. The amount of improvement achieved by MRFSeq may vary depending on the topological distribution. Nevertheless, the simulation results demonstrate that coexpression data could help improve differential gene expression analysis by increasing the coverage of true DE genes significantly.

2.3.3 Performance on Real RNA-Seq Data

Experiments on the MAQC dataset

In addition to the previous simulation study, the performance for inferring DE genes is assessed on the MAQC dataset. Previously, [118] tested the selected methods on different numbers of replicates (or lanes), from 2 replicates to 7 replicates per condition, in the MAQC dataset to see how sequencing depth would affect the performance of the methods. The results indicated that increasing the sequencing depth would decrease the precision of all selected methods except NOISeq. To compare the performance and understand how the precision of MRFSeq would change as the

sequencing depth increases, the experiments designed by Tarazona *et al.* are used in our work. Different numbers of replicates are considered such that the read library size varies from 14 to 45 million reads in each of the two samples. The expression levels of the genes in the MAQC dataset were measured by the normalized threshold cycle values (CT) of qRT-PCR. To validate the true DE genes of the MAQC dataset, a gene is defined as a true DE gene if the $\log_2$ fold change ratio (LR) of its CT values is greater than a certain threshold b, *e.g.*, 0.5 or 2, while a gene is a true EE gene if its LR is smaller than threshold a, *e.g.* 0.2 [15]. Any gene whose LR is between the two thresholds a and b is considered as a borderline gene. In the previous studies, all borderline genes were discarded [15, 118]. Due to the detection limitation of qRT-PCR, lowly expressed genes may be absent in some of the qRT-PCR assays. A gene that was detected in at least one of the qRT-PCR assays would also be removed if it failed to appear in at least three fourths of the qRT-PCR assays [15]. Different from the previous studies, we do not throw away those borderline genes. To further test the inference power on genes with low read counts, lowly expressed genes are also kept in our experiments. This gives us a total of 836 genes. We define a gene as a true DE gene if its LR is larger than the threshold b. Otherwise, the gene is a true EE gene. There are 669 true DE genes when the threshold b is set to be 0.5 and 373 true DE genes when b is 2.0. The coexpression network of the 836 genes forms a graph of 836 nodes and 2426 edges. All the methods are tested at these two different abundance levels (or LR values) of DE genes. The prediction results are again assessed in terms

of precision, sensitivity and F-score as summarized in Figure 2.2.

Comparison of the performance on the MAQC dataset

Similar to the results in the simulation study, MRFSeq achieves significantly improved sensitivity scores and F-scores at both abundance levels of true DE genes. The improvement on sensitivity is at least 9.2% and 8.8% for all sequencing depths considered when b=0.5 and 2, respectively. While achieving the best sensitivity scores, the precision scores of MRFSeq are also comparable to the precision of the others except NOISeq who exhibits extremely high precision. Note that although NOISeq has the best precision among all methods, its sensitivity is much lower than the scores of the others and its overall performance (as measured by F-score) suffers from this. As the sequence depth increases, the precision of NOISeq remains stable while all other methods lose some precision. The precision of DESeq drops 4.0% and 4.7%, respectively, for the two values of b, when the number of replicates increases from two to seven. The decrease in precision is 5.4% and 6% for baySeq while edgeR loses 2.0% and 2.6%. At the same time, the precision of MRFSeq only decreases 1.6% and 2.8%. The relative small loss of the precision for MRFSeq can be explained by the fact that many false positives, if not predicted at a strong confidence level, could be eliminated by MRFSeq using the coexpression information. Hence, these results on the MAQC dataset show that coexpression information not only helps gaining more coverage of the true DE genes but also keeps precision relatively stable against the

increase of sequencing depth. Moreover, it could help reduce our reliance on deeply covered RNA-Seq data in differential gene expression analysis.

Taking confidence scores into consideration

Like the FDRs of DESeq and edgeR or the prior probability of baySeq and NOISeq, MRFSeq estimates the confidence (*i.e.*, marginal probability) for each predicted DE gene and a confidence threshold can be applied to select DE genes for the output (instead of following the MAP estimation algorithm). We are interested in the performance of MRFSeq on the MAQC dataset when different thresholds are applied to the confidence. To calculate the confidence scores, the loopy belief propagation algorithm is run on all 7 replicates in the MAQC dataset. To compare the performance of MRFSeq with the other methods, a precision-sensitivity curve where each point represents the precision and sensitivity under a certain threshold, is depicted for each of the selected methods, as done in [118]. To depict the precision-sensitivity curves for DESeq and edgeR the range of the FDR threshold from 10^{-6} to 1 is selected. Note that this range for FDR cutoffs covers all the threshold values used in practice and these FDR thresholds yield sensitivity values between 45% and 100%. For the other methods that do not use FDRs, equivalent thresholds that lead to sensitivity within the same range, *i.e.*, 45% to 100%, are applied to draw the precision-sensitivity curves. The precision-sensitivity curves in Figure 2.3 show that, in general, MRFSeq

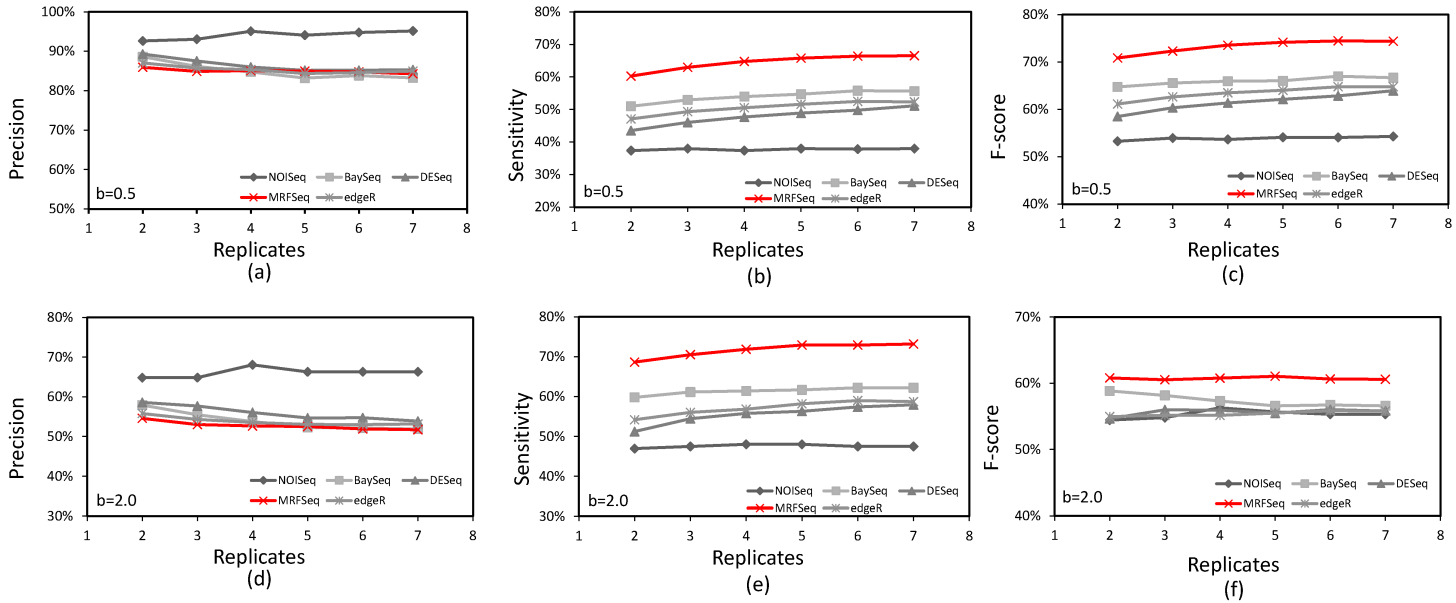

Figure 2.2: Performance assessment at various sequencing depths. The X-axis shows the number of used lanes and the Y-axis indicates various assessment measures. In the upper column of plots, the LR threshold b is set as 0.5 and in the lower column b=2.0. Plots (a) and (d) compare the precision scores at different sequence depths. Plots (b) and (e) depict the sensitivity scores while plots. (c) and (f) illustrate the F-scores.

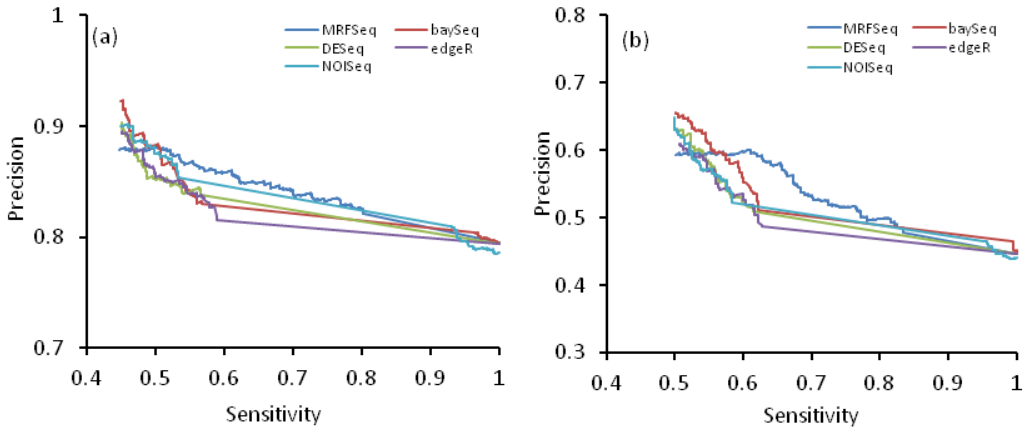

Figure 2.3: Comparison of the methods when different confidence thresholds are applied. Plots (a) and (b) show the precision-senitivity curves when the LR threshold b is set as 0.5 and 2.0, respectively.

provides more accurate confidence scores than the other methods. Note that unlike the MAP estimation algorithm, using the marginal probabilities obtained by the loopy belief propagation algorithm to infer DE genes requires additional knowledge to choose an approproate confidence (marginal probability) threshold. Besides, the loopy belief propagation algorithm is a heuristic and thus does not guarantee correct marginal probabilities. Hence, MRFSeq uses the MAP estimation to select DE genes and the loopy belief propagation algorithm only to estimate the confidence score of each selected DE gene.

Comparisons of the performance on Griffith's dataset

The qRT-PCR data of Griffith's dataset consists of 193 exons assays on 94 protein coding genes. Different to the LR of the MAQC dataset, a two-tailed t-test was

applied to identify the true DE genes from the qRT-PCR data of Griffith's dataset. A p-value of the t-test was considered significant if it is smaller than 0.05 [45]. Under this criterion, 83 true DE genes and 11 true EE genes are identified and used in testing the selected methods. The coexpression network of the 94 genes extracted from COXPREdb forms a graph of 94 nodes and 25 edges. The performance of the methods on Griffith's dataset is shown in Table 2.3. MRFSeq still has the best overall performance, although its improvement over the other methods is not as significant as on the MAQC data. Please see the supplementary materials for a detailed discussion. Its sensitivity is 1.2% better than the second best. The prediction accuracy of all the methods on this dataset is generally higher than those in the previous tests on the MAQC data. Because Griffith's dataset was made from the DE or alternatively expressed genes selected by ALEXA-Seq [45], the difference of read counts between the two conditions is more apparent and hence the inference task becomes easier. However, at the same time, the room for improvement gets smaller. There are only 25 pairs of coexpressed genes in this dataset according to COXPREdb[89]. The large independence of gene expression makes it hard for MRFSeq to achieve an improved performance. Nevertheless, MRFSeq still outperforms the other methods in the test, although the difference between the F-score of MRFSeq and the second best is not as significant as on the MAQC data.

Table 2.3: Comparison of the prediction accuracy on Griffith's dataset. TP is the number of true positives and PP is the number of predicted positives.

Methods	TP	PP	PRE(%)	SEN(%)	FS(%)
MRFSeq	**80**	**90**	**88.9**	**96.3**	**92.4**
baySeq	74	81	91.3	89.1	90.2
DESeq	79	89	88.7	95.1	91.8
edgeR	73	84	86.9	87.9	87.4
NOISeq	57	60	95.0	68.7	79.7

2.3.4 Performance on Genes with Low Read Counts

Genes with low read counts

To understand how the methods perform on genes with different read count levels, the prediction on the real datasets is further analyzed. The genes in the datasets are separated into two classes, genes with low read counts and genes with decent read counts. In [15], a gene is said to have a low read count if it has fewer than 10 reads in every replicate of the two conditions. Otherwise, the gene is said to have a decent read count. Since Griffith's dataset contains only genes with decent read counts, we consider the MAQC dataset only below. Among the 836 genes in the MAQC dataset, there are 453 genes with low read counts and 383 genes with decent read counts. The methods baySeq and NOISeq provide an additional option for normalizing gene lengths. These two methods with normalized gene lengths are denoted as baySeq$_{len}$ and NOISeq$_{len}$, respectively. To further study the effect of the normalization on genes with low read counts, baySeq$_{len}$ and NOISeq$_{len}$ are also applied to the MAQC dataset. By choosing a threshold of $b = 0.5$ for the LR values, the prediction results on genes with low read counts by different methods are compared in Table 2.4.

Table 2.4: Comparison of the prediction results on genes with low read counts. $RTP_{l/h}$ is the ratio of true positives with low read counts over the true positives with high read counts. $RPP_{l/h}$ is the ratio of predicted positives with low read counts over the predicted positives with high read counts.

Methods	[a]$RTP_{l/h}$(%)	[b]$RPP_{l/h}$(%)	PRE(%)	SEN(%)	FS(%)
MRFSeq	**43.1**	**42.7**	**84.8**	**38.8**	**53.3**
baySeq	6.8	6.2	100.0	5.5	10.4
$baySeq_{len}$	7.4	6.8	100.0	6.1	11.5
DESeq	12.5	13.0	82.6	11.0	19.4
edgeR	13.0	13.3	83.3	11.6	20.4
NOISeq	0.0	0.0	-	0.0	-
$NOISeq_{len}$	4.5	5.0	84.6	3.2	6.1

Significant improvement on genes with low read counts

On the genes with low read counts, the sensitivity of MRFSeq is 38.8% while the second best sensitivity is only 11.6%. Similarly, MRFSeq achieves an F-score of 53.3% while the second best F-score is only 20.4%. In addition to these significant improvements, the prediction of MRFSeq shows a more balanced pattern between genes with low read counts and genes with decent read counts. The $RTP_{l/h}$ of MRFSeq is 43.1% while its $RPP_{l/h}$ is 42.7%. The second best $RTP_{l/h}$ and $RPP_{l/h}$ are only 13.0% and 13.3% (obtained by edgeR). This result shows that all the other methods are quite biased against genes with low read counts. Most of their predicted DE genes are from the genes with decent read counts. After applying the normalization of gene lengths on genes with low read counts, the performance of baySeq and NOISeq is slightly improved. However, the length normalization does not really improve the overall performance on genes with low read counts much or correct the selection bias.

2.3.5 Comparison with Cuffdiff 2

Different from gene-level methods that use raw read counts, Cuffdiff 2 requires the mapping of reads to the given transcripts of genes as input to call differential gene expression [123]. To assess the performance of Cuffdiff 2 on the MAQC and Griffith's datasets, the RNA-Seq reads of the two real datasets are mapped to the annotated transcriptome UCSC hg19 using Tophat as done in [123]. The same threshold 0.1 for the FDR values is used to call DE genes for Cufflink 2. The prediction accuracies of MRFSeq and Cuffdiff 2 on the two datasets are summarized in Table 2.5, with the LR threshold $b = 2$ and the cutoff p-value 0.05 for the MAQC and Griffith's datasets, respectively. The precision-sensitivity curves also are illustrated in Figure 2.4. The table shows that MRFSeq has a signicantly better F-score (and thus overall performance) by achieving a higher sensitivity, while Cuffdiff 2 achieves a better precision. The precision-sensitivity curve also suggests that MRFSeq has a better overall performance than Cuffdiff 2 when we consider the full spectrum of FDR or restricting the FDR value to at most 0.1. A detailed analysis shows that Cuffdiff 2 predicts fewer true DE genes with relatively small LR values than MRFSeq. In the MAQC dataset, there are 290 true DE genes with the LR values from 0.5 to 2. The prediction of MRFSeq covers 171 of the 290 genes while Cuffdiff 2 can only detect 140 of the true DE genes. In Griffith's dataset, 9 true DE genes are associated with p-values, which measure the significance of the difference between the LR values, from 0.005 to 0.001. All of the 9 true DE genes are predicted by MRFSeq, but Cuffdiff 2

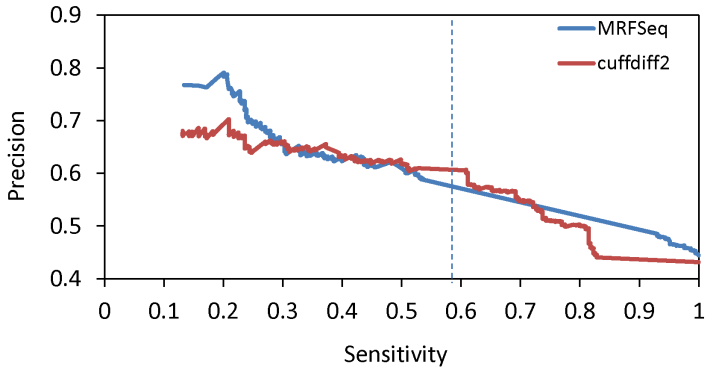

Figure 2.4: The precision-sensitivity curves assess the prdiction performance of MRF-Seq and Cuffdiff 2 on the MAQC dataset. The dotted line shows the sensitivity value corresponding to the common FDR threshold 0.1. Note that sensitivity increases with FDR, and thus the region to the left of the dotted line might be more interesting in practice.

Table 2.5: Comparison of the prediction accuracy with Cuffdiff 2.

Datasets	Methods	PRE(%)	SEN (%)	FS (%)
MAQC	**MRFSeq**	**46.9**	**95.7**	**63.0**
	Cuffdiff 2	59.3	61.3	60.3
Griffith's	**MRFSeq**	**84.6**	**93.9**	**89.0**
	Cuffdiff 2	96.9	37.2	53.8

could only identify 4 of the DE genes. This result is consistent with the discussion in [123]. In general, Cuffdiff 2 may report fewer DE genes with relatively low LR rates because of its control of variance in expression owing to fragment count uncertainty.

2.3.6 Consistency of Predictions by DESeq and MRFSeq

A gene is defined to be *incorrectly inverted* if its DE state is correctly predicted by using RNA-Seq data alone but incorrectly predicted by MRFSeq. Although our

above results demonstrate that utilizing the prior knowledge of gene coexpression significantly improves the overall accuracy of differential gene expression analysis and helps to alleviate the bias against genes with low read counts, it raises the question if the prior knowledge might introduce some new prediction biases. In this subsection, we estimate the number of incorrectly inverted genes in the prediction of MRFSeq compared with prediction by a popular RNA-Seq based method DESeq and analyze the types of genes in coexpression networks that are more likely to be incorrectly inverted. The detailed prediction results of DESeq and MRFSeq on our above simulation and real datasets are compared. In the 40 simulation benchmark datasets, only 3092 of the 73619 (4.2%) correctly predicted genes by DESeq are incorrectly inverted by MRFSeq. In the MAQC and Griffith's datasets, only 16 (3.5%) and 0 (0%) genes correctly predicted by DESeq are incorrectly inverted by MRFSeq, respectively. Generally, most of the correctly predicted genes by DESeq remain correct in the MRFSeq prediction. Moreover, we observe that the incorrectly inverted genes tend to have higher edge degrees in gene coexpression networks than the other genes. The comparison of the average edge degree of all genes and that of the incorrectly inverted genes in gene coexpression networks is shown in Figure 2.5. The significance of the difference between the edge degrees is confirmed by using one-tailed t-test [42]. The p-value of the t-tests on the simulation and MAQC datasets are 5.1×10^{-14} and 5.1×10^{-4}, respectively. However, since gene coexpression networks usually possess the well-known scale-free property, only a small number of genes have high edge de-

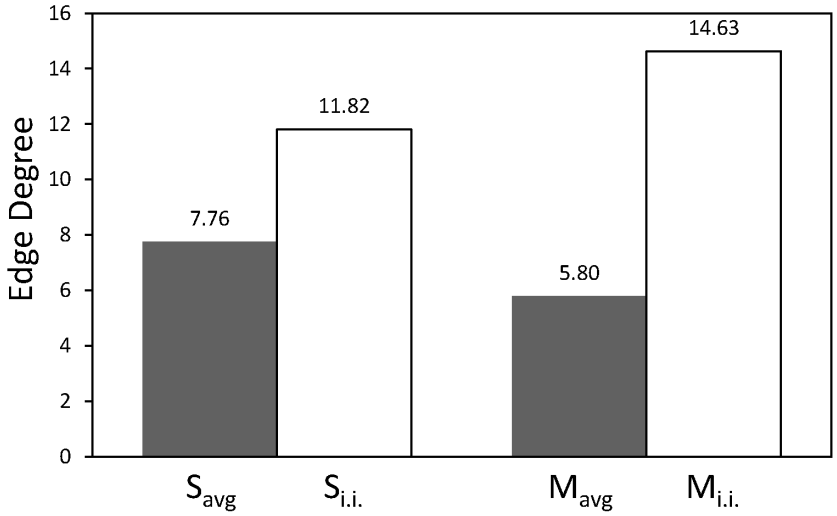

Figure 2.5: Comparison of the average edge degrees of incorrectly inverted genes and all genes in the coexpression networks used in the simulated and MAQC dataset. S_{avg} is the average edge degree of all genes in the coexpression networks used in the simulation while $S_{i.i.}$ is the average edge degree of all incorrectly inverted genes. M_{avg} is the average edge degree of all genes in the coexpression networks used in the MAQC datasets while $M_{i.i.}$ is the average edge degree of the incorrectly inverted genes.

grees [115, 17]. This property should limit the number of incorrectly inverted genes, and thus most of the DE or EE genes correctly predicted based on RNA-seq data alone (by, *e.g.*, DESeq) are well preserved in the result of MRFSeq.

2.4 Conclusion

In this work, we have proposed a new statistical method, MRFSeq, that combines both RNA-Seq data and coexpression information and obtains a MAP estimation of the differentially/equally expressed genes efficiently. The improvement benefits from

our graphical model is assessed by comparing MRFSeq with a simple method, called SimpleNetwork, that uses the median of the DESeq FDR values of each gene and its neighbors in the coexpression network as the predicted FDR for the gene. SimpleNetwork is run on the MAQC dataset and the precision-sensitivity curves of MRFSeq, SimpleNetwork and DESeq are shown in Figure 2.6. The results demonstrate that, even for a simple method like SimpleNetwork, the introduction of coexpression data improves the performance of calling DE genes. However, the accuracy of SimpleNetwork is much worse than that of our graphical model MRFSeq. This is because the correlation coefficients in the gene coexpression data are not fully utilized and the reliability of each predicted FDR value is not taken into account.

Using extensive experiments on both simulated and real data, we have shown that MRFSeq is able to take advantage of coexpression information and this additional piece of information can help provide a more accurate and less biased differential gene expression analysis. Clearly, our improved performance (especially on genes with low read counts) critically depends on the existence of a high quality gene coexpression network. To investigate how much the performance of MRFSeq will be deteriorated when the gene coexpression networks contains incorrect coexpression relationship, we modified the coexpression network of the MAQC dataset by randomly inserting 10% to 30% additonal edges into the network or deleting 10% to 30% random edges from the network. The weight of an added edge is randomly and uniformly drawn from 0.5 to 1. The performance of MRFSeq on the resulting networks are summarized

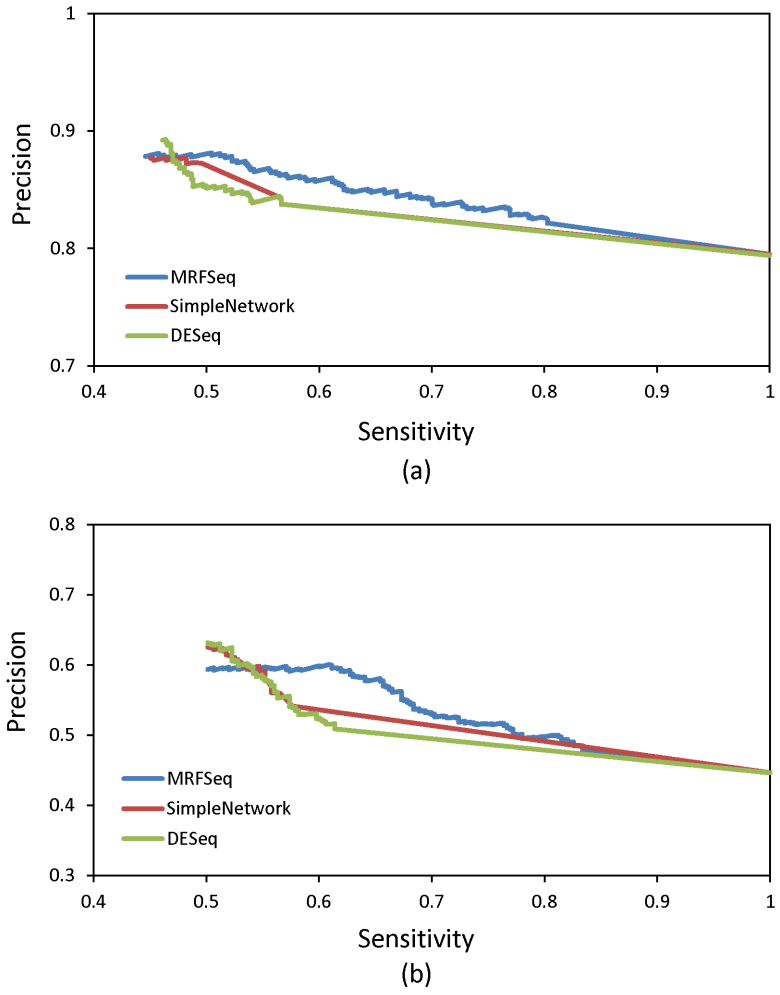

Figure 2.6: Precision-sensitivity curves for comparing the prediction accuracy of MRFSeq and SimpleNetwork on the MAQC dataset with the LR values (a) b=0.5 and (b) b=2.

Table 2.6: Performance assessment of MRFSeq on gene coexpression networks obtained by adding random edges. Edge is the percentage of randomly added edges

[a]Edges	[b]TP	[c]PP	PRE (%)	SEN (%)	FS (%)
0	445	528	84.28	66.51	74.35
10	447	537	83.24	66.51	73.94
20	457	595	76.80	68.31	72.31
30	456	624	74.67	68.16	71.26

Table 2.7: Performance assessment of MRFSeq on the gene coexpression networks obtained by deleting random edges. Edge is the percentage of randomly deleted edges

[a]Edges	[b]TP	[c]PP	PRE (%)	SEN (%)	FS (%)
0	445	528	84.28	66.51	74.35
10	441	520	83.24	65.91	74.17
20	429	505	84.95	64.12	73.07
30	419	493	84.98	62.63	72.11

in Table 2.6 and 2.7. Clearly, the deletion of random edges introduced mainly false negatives while the addition of random edges introduced more false positives.

Finally, MRFSeq uses the DE analysis results of DESeq. It would be interesting to study how MRFSeq performs when the DE analysis results of other tools are used instead. To demonstrate the flexibility of MRFSeq so that MRFSeq can incoporate with different differential gene expression analysis tools, the variant of MRFSeq, denoted as MRFSeq_NOISeq, that uses the results from NOISeq instead of DESeq is implemented. In MRFSeq_NOISeq, we set $Pr(x_i = 0|y_i) = y_i$, where y_i is the prior probability provided by NOISeq. Both MRFSeq and MRFSeq_NOISseq are run on a simulated dataset from the interval [0,10) of true DE genes and on the MAQC dataset with the LR threshold b=0.5. The precision-sensitivity curves in Figure 2.7 illustrate that the introduction of coexpression data using our probabilistic graph model can improve

the prediction accuracy of both DESeq and NOISeq. On the MAQC dataset where NOISeq outperforms DESeq, the performance of MRFSeq_NOISeq is better than that of MRFSeq. On the simulated dataset where DESeq outperforms NOISeq, the performance of MRFSeq is better than that of MRFSeq_NOISeq. We will make the version MRFSeq_NOISeq available to the user on the website in the near future. We plan to make MRFSeq flexible so it can be combined with any DE analysis tool in the near future. Our experiments used COXPREdb [89], which consists of coexpression data for seven model organisms. One could also consider using other sources of coexpression data such as ACT [77]), ATTED-II [88], CSB.DB [113], CoP[91], *etc.* or contructing custom gene coexpression networks from publicly available expression data such as GEO (`http://www.ncbi.nlm.nih.gov/geo/`), ENCODE (`http://encodeproject.org/ENCODE/`), modENCODE (`http://www.modencode.org/`), *etc.*, especially for organisms (or tissues) not covered by COXPREdb. Moreover, the threshold ρ used for extracting pairs of coexpressed genes from a given gene coexpression network may also have an impact on the performance of our algorithm. We set $\rho = 0.5$ empirically in our experiments based on the literature [95, 131] and some preliminary tests on the MAQC data. Clearly, a higher ρ may decrease the sensitivity of MRFSeq while a lower ρ may decrease the precision of MRFSeq. We plan to explore the impact of different coexpression networks (including the choice of ρ) on the performance of MRFSeq and study automatic methods for choosing an optimal ρ in future work.

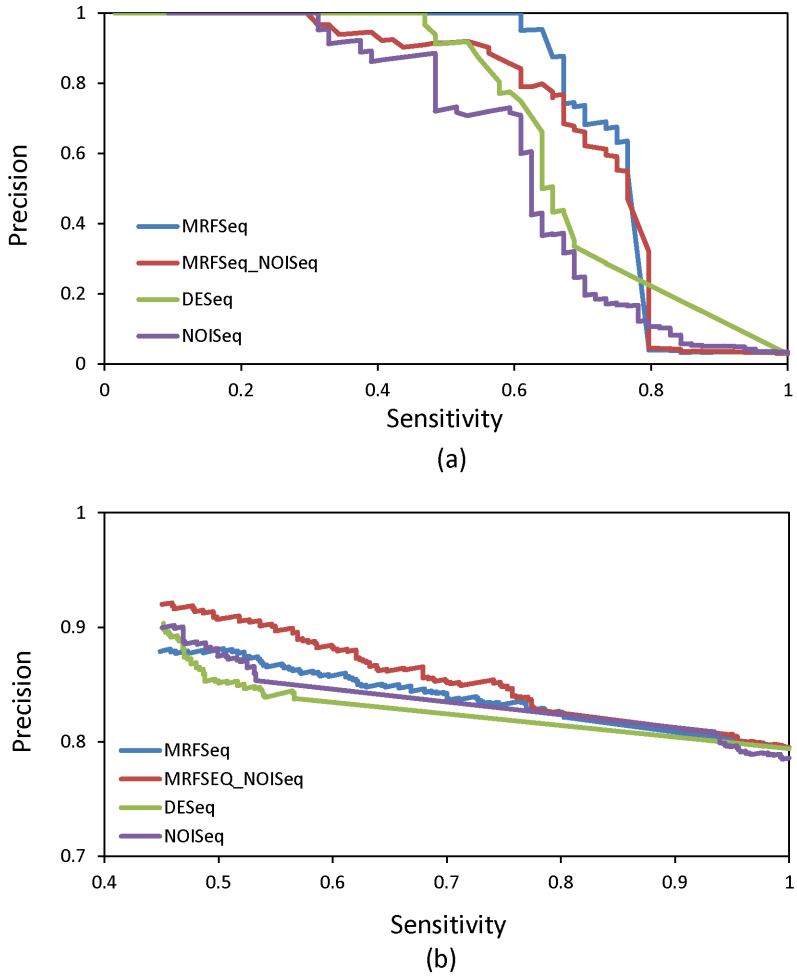

Figure 2.7: Precision-sensitivity curves for comparing the prediction accuracy of MRFSeq_NOISeq, MRFSeq, DESeq, and NOISeq on the (a) simulated and (b) MAQC datasets.

Chapter 3

SDEAP: A Splice Graph Based Differential Transcription Expression Analysis Tool for Population Data

3.1 Introduction

Understanding the relationship between genetic (or epigenetic) variation and transcriptional regulation is a major goal in many large-scale genomic analysis. [19, 123] In recent years, RNA-Seq has taken a major role in the quantitative analysis of transcript expression and variant discovery, and become a vital component of genomic and transcriptomic research [122]. Studying the regulation of transcripts in biological processes of interests requires sensitive and specific detection of changes in transcript abundance. Earlier differential expression methods such as DESeq and edgeR detect changes in the absolute abundance of genes [4, 101]. Recently, several differential transcript expression (DTE) analysis methods, such as Cuffdiff 2 [123], DEXSeq [3] and

ALEXA-Seq [45], report genes that have differentially expressed transcripts whose abundance values alter between biological conditions. Along with a list of differential expressed genes, DEXSeq and ALEXA-Seq provide differentially expressed exons and alternative splicing events, respectively, to indicate where the expression of transcripts diverges. Cuffdiff 2 further infers the absolute abundance of transcripts to portrait the comprehensive alternation in transcript expression. In addition to the DTE methods, differential splicing (DS) analysis methods, such as MISO [57], FDM [109], MATS [106], and DiffSplice [50], are focused on identifying difference in relative abundance of transcripts. Note that a change in the absolute abundance of a transcript may result from a change in the basal expression level of the corresponding gene or its splicing ratio. In other words, DTE methods should be able to discover DS events but not vice versa. Nevertheless, these DTE or DS methods provide precious transcriptomic information useful to many down-stream applications.

These DTE or DS methods are developed to compare the expression of transcripts between two biological conditions. The requirement of predefined biological conditions restricts their applications to case-control studies only. However, the DTE and DS analysis may also find many find important applications in population based studies, where predefined conditions are unavailable *a priori*. For example, a recent population study to improve the diagnosis and prognosis for breast cancer shows that triple-negative breast cancer can be further classified into six subtypes based on differential analysis of the expression profiles of patients [66]. Each of the six subtypes has

different sensitivities to targeted therapies. Moreover, to understand the functions and mechanisms during cell development or differentiation, differentially expressed transcripts are used to characterize cell types or specificity in a mixed population [13, 14, 120]. Due to the emergent demand for computational tools for DTE and DS analysis in population data, several methods have been proposed recently. SIBER and DEXUS test differential expression at the gene level by looking at the numbers of reads mapped to individual genes [119, 63]. An extended protocol of DESeq2 has been published recently to identify differentially expressed genes for single-cell RNA-Seq data [13]. SigFuge compares the areas under normalized read-depth curves to call DS genes [61]. To the best of our knowledge, there is no DTE analysis tool in the literature for population data without predefined biological conditions. Hence, in this paper, we present the first DTE analysis method, called SDEAP, that discovers genes with differentially expressed transcripts and the corresponding alternative splicing events on samples without predefined biological conditions. Note that alternative splicing events could be valuable in down-stream applications on their own right, *e.g.* as biomarkers in several cancer studies [139, 23, 127].

As observed in SigFuge [61], the numbers of reads mapped to individual exons can be used as the input of DEXUS to conduct DTE analysis and identify genes with differential transcript expression (*i.e.*, DTE genes). Although this modified version of DEXUS, denoted as DEXUS$_{exon}$, can regarded as a DTE analysis method, its prediction results do not directly suggest how and where transcription diverges

in the population. To address this problem, a graphical data structure, called the *splice graph*, is used to model the structures and expression of all transcripts of a gene such that alternative splicing events can be represented by decomposing the graph into *alternative splicing modules* (ASMs) as originally proposed in DiffSplice [50]. However, the graph modular decomposition algorithm proposed in DiffSplice is not used because we have found a counterexample, discussed in Section 3.2.1, to its correctness. A corrected algorithm is provided in this manuscript and implemented in SDEAP.

Generally, there are two main steps in differential expression analysis without predefined conditions [61]. The first is to cluster the individuals in a population based on some numerical features used to summarize the expression of each gene (or transcript), called expression features (*e.g.*, read counts of genes in DEXUS and areas under normalized read-depth curves in SigFuge), and then test the statistical significance of the difference between the features in different clusters for each gene. However, both DEXUS and SigFuge assume that the input population consists of only two groups and always cluster the individual into two clusters. This assumption is unrealistic in many applications and an incorrect partition of individuals may lead to unreliable conclusions of differential expression tests. Hence, in SDEAP, the numbers of clusters in population data are not predefined and learned from the data by using Dirichlet infinite mixture models. This robust clustering method helps improve the performance of our method, as demonstrated in our experiments. Moreover, methods

accounting for variability due to outliers across biological or technical replicates of an RNA-Seq experiment have been utilized to reduce the number of false positives in differential expression analysis [4, 101, 13]. To control false positives, a similar regression model proposed in DESeq and edgeR is used in SDEAP to dynamically estimate the observed variance due to outliers across individual genes.

To assess the prediction accuracy of SDEAP, several computational experiments on both simulated and real data are conducted to compare SDEAP with DEXUS$_{exon}$. SIBER is excluded from our comparisons because DEXUS has been proved to significantly outperform SIBER and, moreover, SIBER can only be run on large datasets of more than 50 RNA-Seq samples [119]. We simulated RNA-Seq data that reflect the variance and noise in real RNA-Seq data. In our simulated experiments, SDEAP showed better control of false positives, achieved the best overall performance and retained robustness on noisy data that contain outliers often seen in single-cell RNA-Seq data. More specifically, on simulated standard and single-cell RNA-Seq data, SDEAP outperformed DEXUS$_{exon}$ by 0.17 in the area under precision-recall curve (or AUC$_{pr}$) on average. On these data, DEXUS$_{exon}$ achieved lower accuracy with much higher false positive rates. Moreover, its performance dropped significantly when the population consists of groups with skewed sizes or the number of groups is greater than two. Although DS analysis is not the main purpose of SDEAP, we compared it with SigFuge in the detection of changes in relative abundance of transcripts by repeating the simulated experiments in SigFuge [61]. SDEAP discovered more DS

genes than SigFuge without producing any false positives. To further demonstrate the value of SDEAP in biological applications, we downloaded three real RNA-Seq dataset, one standard RNA-Seq dataset from breast cancer patients and two single-cell RNA-Seq datasets, and used the alternative splicing events found by SDEAP as biomarkers to classify cancer subtypes, cell types and cell-cycle phases. The classification of RNA-Seq samples using the alternative splicing events from SDEAP is much more consistent with the real biological conditions (*i.e.*, cancer subtypes, cell types and cell-cycle phases). SDEAP outperformed DEXUS$_{exon}$ by 0.28 and 0.13 in Jaccard index for classifying cancer subtypes and cell-cycle phases, respectively. The prediction results of the both methods are also compared to qPCR validations of gene expression. More validated DTE genes are covered by the prediction of SDEAP. These experimental results show that SDEAP performs DTE analysis well on real population data.

3.2 Methods

A splice graph is a data structure that represents the structures and abundance of the transcripts (or isoforms) of a gene. In the literature, there are slightly different definitions of splice graphs. Here, we follow the definition of splice graphs used in DiffSplice [50]. An expressed segment is an exonic region delimited by two exon boundaries. Note that an expressed segment does not necessarily correspond to a whole exon. An exon can be split into several expressed segments due to its alternative

splicing sites. A splice graph $G(V \cup \{s,t\}, E)$ of a gene g is a weighted and directed acyclic graph where every vertex $v \in V$ denotes an expressed segment R_v. For every pair of vertices u and v, there is a directed edge (u,v) from u to v if the expressed segment R_v immediately follows R_u in some transcript of the gene g. In addition to the vertices V representing expressed segments, two artificial vertices s and t are included in G to indicate the beginning and end of all transcripts of the gene g, respectively. The vertex s is connected to every vertex corresponding to the very first expressed segment of a transcript of the gene g and every vertex denoting the last expressed segment of a transcript is connected to t. Thus, every (s,t) path in G represents a transcript of g. In SDEAP, we assume that splice graphs are provided as the input. Given all RNA-Seq reads mapped to the gene g in an RNA-Seq sample, the weight of a vertex v, $w(v)$, is defined as the number of reads mapped to the region R_v and the weight of the edge (u,v), $w(u,v)$, is the number of reads that span the two expressed segments R_u and R_v.

A vertex $u \in V$ *pre-dominates* a vertex $v \in V$ if every path from the artificial vertex s to v contains u. The vertex u is called a pre-dominator of the vertex v. A vertex $w \in V$ *post-dominates* a vertex $v \in V$ if every path from v to the artificial t contains w. The vertex w is called a post-dominator of the vertex v. An ASM (or alternative splicing module) is an induced subgraph $H(s_1,t_1) = \{V_H, E_H, s_1, t_1\}$ of G with the entry s_1 and the exit t_1 outside H that satisfies the following conditions [50]: (1) (Single entry) All edges from $(G-H)$ to H come from s_1; (2) (Single exit)

All edges from H to $(G - H)$ go to t_1; (3) (Alternative paths) Let $d_+(u)$ denote the number of outgoing edges from the vertex u and $d_-(u)$ the number of incoming edges of u. Then $d_+(s_1) > 1$ and $d_-(t_1) > 1$; (4) (Minimality) There does not exist a vertex $v \in V_H$, such that v post-dominates s_1 or pre-dominates t_1 in $H(s,t)$. Moreover, an ASM $H_1(t_1, s_1)$ can be a subgraph of another ASM $H_2(t_2, s_2)$. If there is no ASM that contains H_1 and is contained by H_2, H_1 is said to be immediately contained by H_2. By the definition of ASMs, an ASM is allowed to be only immediately contained by one another ASM such that the containment of ASMs can be represented as a tree that is called the hierarchy tree T of ASMs. In the hierarchy tree T, $H_1(s_1, t_1)$ is said to be a child ASM of H_2 and H_2 is the parent ASM of H_1. The *abstraction* of an ASM $H_2(s_2, t_2)$ is a graph obtained by replacing every child ASM $H_1(s_1, t_1)$ of $H_2(s_2, t_2)$ with an artificial edge (s_1, t_1). An ASM path is a path from s_2 to t_2 in the abstraction of an ASM $H_2(s_2, t_2)$.

3.2.1 Discovery of ASMs

In this subsection, we present a modular decomposition algorithm to identify all ASMs of an input graph G and construct the hierarchy tree of the ASMs. In this algorithm, every ASM is discovered before its parent and then shrunk into an artificial edge right after being identified such that the parent of the shrunk ASM is known when the ASM that contains the artificial edge is being discovered. The discovery of an ASM $H_1(s_1, t_1)$ hinges on locating its entry s_1 and exit t_1. When the entry s_1

and exit t_1 are anchored, the ASM H_1 is the union of the paths from s_1 to t_1. The out-degree of every entry s of an ASM is greater than 1. For every vertex u with the out-degree $d_-(u) > 1$, u is a candidate of an entry of some ASM. Similarly, for every vertex v with the in-degree $d_+(v) > 1$, v is a candidate of the exit of an ASM. Given a candidate entry-exit pair (u, v), we check if u and v are the entry and exit of an ASM by verifying whether the union of the paths from vertex u to vertex v satisfies the four properties of an ASM. If the subgraph is an ASM $H(u, v)$, shrink $H(u, v)$ into an artificial edge connecting vertex u and v. For an ASM $H_1(s_1, t_1)$ and its parent ASM $H_2(s_2, t_2)$, as long as the pair (s_1, t_1) is tested before the pair (s_2, t_2), $H_1(s_1, t_1)$ is ensured to be identified before its parent ASM $H_2(s_2, t_2)$. This can be done by enumerating all candidate entry-exit pairs in the following order.

Assume that all vertices of the input splice graph G are sorted by topological sort [20]. Let β be the topological order of vertices and $\beta(u)$ the index of vertex u in the order. If there is a path from vertex u to vertex v, $\beta(u) > \beta(v)$. For every ASM $H_1(s_1, t_1)$ whose parent is $H_2(s_2, t_2)$, if all vertices are traversed in the order of β, the exit t_1 of H_1 must be traversed before the exit t_2 of its parent H_2, *i.e.*, $\beta(t_1) < \beta(t_2)$. Let $\bar{\beta}$ be the reverse of the topological order β. If all vertices of G are traversed in the order of $\bar{\beta}$, the entry s_1 of the ASM $H_1(s_1, t_1)$ is always traversed before the entry s_2 of the parent ASM $H_2(s_2, t_2)$, *i.e.*, $\bar{\beta}(s_1) < \bar{\beta}(s_2)$. Moreover, for any ASM $H(s_1, t_1)$, it follows from an ASM that $\beta(s_1) < \beta(t_1)$. Hence, all candidates of the entry are traversed in the order of $\bar{\beta}$. When a candidate entry u of some ASM is visited, for

every candidate exit v such that $\beta(v) > \beta(u)$, v is chosen in the order of β to pair up with u as a candidate entry-exit pair (u, v). In this order of enumerating candidate entry-exit pairs, for every ASM $H_1(s_1, t_1)$ and its parent $H_2(s_2, t_2)$, the candidate entry-exit pair (s_1, t_1) is always tested before (s_2, t_2). Thus, every ASM is guaranteed to be identified before its parent.

The time complexity of topological sorting is linear in the number of vertices and edges, *i.e.*, $O(|V| + |E|)$. Enumerating all candidate entry-exit pairs takes time $O(|V|^2)$. For every candidate entry-exit pair (u, v), the union of the $u - v$ paths is verified as an ASM by checking the conditions (1) and (2) in the ASM definition, which requires $O(|V|+|E|)$ time. Therefore, the time complexity of identifying ASMs from a splicing graph G is $O(|V|^3 + |V|^2|E|)$.

We have discovered a counterexample to the graph modular decomposition algorithm used in DiffSplice [50]. Before presenting the counterexamples, we first describe the algorithm below for completeness. Let the input splice graph be $G(V \cup \{s, t\}, E)$. A vertex $u \in V$ *pre-dominates* a vertex $v \in V$ if every path from the artificial vertex s to v contains u. The vertex u is called a pre-dominator of the vertex v. A vertex $w \in V$ *post-dominates* a vertex $v \in V$ if every path from v to the artificial t contains w. The vertex w is called a post-dominator of the vertex v. A vertex $u \in V$ *immediately* pre-dominates a vertex $v \in V$ if there is no vertex $p \in V, y \neq u$ that pre-dominates v on the paths from u to v. Similarly, A vertex $w \in V$ *immediately* post-dominates a vertex $v \in V$ if there is no vertex $q \in V, y \neq u$ that post-dominates v on the paths

from v to w. In the graph modular decomposition algorithm of DiffSplice [50], all edges are ordered such that edge (u, v) is said to be greater than (u', v'), denoted as $(u, v) > (u', v')$ if and only if there exists a directed path from u to u' and a directed path from v' to v. An edge (u, v) is called a *maximal* edge in a subgraph H of G if there is no edge in H greater than (u, v).

According to the pseudocode given in [50], the graph modular decomposition algorithm consist 3 steps and decomposes the input splice graph recursively in a top-down fashion. The first step is to calculate all immediate pre-dominators and post-dominators for every vertex. To achieve this, it enumerates every vertex u with $d_+(u) > 1$ (or $d_-(u) > 1$) as a candidate of the entry (or exit) of an ASM, respectively. The second step is to enumerate every candidate of the entry. For every candidate u of the entry, u is paired up with a candidate v of the exit where v is an immediate post-dominator of u. The subgraph $H(u, v)$ bounded by the vertices u and v is a candidate ASM. However, the exit of an ASM is not necessarily a post-dominator of the entry. Hence, in the third step, the maximal edges E_{max} of $H(u, v)$ are removed from $H(u, v)$. The above three steps are then repeated on the reduced subgraph $H(u, v)/E_{max}$ recursively until no more ASMs are detected.

A counterexample is given in Figure 3.1(a). The splice graph in the figure contains three ASMs: $H(v_1, v_8)$, $H(v_1, v_6)$ and $H(v_1, v_7)$. However, we can show that DiffSplice is unable to identify the ASMs $H(v_1, v_6)$ and $H(v_1, v_7)$. Note that vertex v_8 is the only post-dominator of vertex v_1. In the very first iteration, v_1 is paired up with v_8 such

that $E_{max} = \{(v_1, v_3), (v_1, v_5), (v_2, v_6), (v_4, v_7)\}$. Figure 3.1(b) shows $H(u, v)/E_{max}$ after deleting E_{max} from $H(v_1, v_8)$, which is used as the input graph in the second iteration. Except for v_1 and v_8, every vertiex now has only one in-coming and one out-going edge. No vertex will be considered as the entry of a new ASM and hence the ASM $H(v_1, v_6)$ and $H(v_1, v_7)$ will not be discovered. Therefore, the graph modular decomposition algorithm in DiffSplice fails to detect all ASMs in this counterexample.

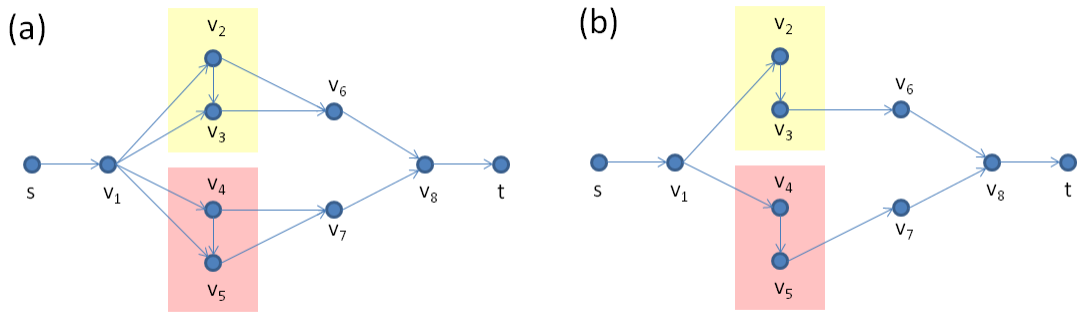

Figure 3.1: A counterexample to the graph modular decomposition algorithm used in DiffSplice. Plot (a) shows the counterexample where the vertices of the two ASMs $H(v_1, v_6)$ and $H(v_1, v_7)$ n ot detected by the algorithm are highlighted in yellow and red, repectively. In plot (b), the reduced graph $H(u, v)/E_{max}$ is illustrated.

3.2.2 Evaluation of Expression Features Using ASMs

Expression features are numerical features used to summarize the expression of transcripts (or genes). In SDEAP, the expression features concern expressed segments, junctions and ASM paths. If there are n RNA-Seq samples in a population data, every expression feature f has n abundance values, $F = \{f_1, f_2, ..., f_n\}$. If the number

of paths in an ASM is greater than 4, we simply use the abundance values of the expression segments and junctions in the ASM as its expression features. This is because observed that the estimation of the abundance of paths may suffer from non-identifability and our observation is consistent with the discussion in DiffSplice [50]. If the number of paths in an ASM is less than or equal to 4, the abundance values of the paths in the ASM are used as the expression features. Here, abundance is measured by the average RNA-Seq fragment coverage per thousand bps per million fragments (FPKM) [122] and are estimated as follows.

For an expressed segment, the FPKM is the number of fragments mapped to the expressed segment divided by the length of the segments in kilo bps and the size of the RNA-Seq fragment library in millions. For a junction, because the length of mapped reads is the length of the region where each junction read spans, the FPKM of a junction is the number of mapped reads divided by the read length and the size of the library. Given an ASM $H(u, v)$, let the ASM paths of $H(u, v)$ be $P = \{p_1, p_2, ..., p_N\}$ such that all expressed segments and junctions covered by the paths can be represented as a numerical matrix $A_{M\times N} = (a_{i,j}), 1 \leq i \leq M$ and $1 \leq N \leq j$, where each of the M rows represents an expressed segment or a junction and each of the N columns represents a path. If the path p_j includes an expressed segment j, $a_{i,j} = l_i$, where l_i is the length of the expressed segment. If the path p_j includes a junction j, $a_{i,j} = \hat{l}_i$, where $\hat{l}_i$ is the length of the RNA-Seq reads. Otherwise, $a_{i,j} = 0$. Let the abundance values (FPKMs) of the paths be

$X = \{x_1, x_2, ..., x_N\}$. Note that the first and last vertices and artificial edges of each path are not included in the rows of $A_{M \times N}$. All mapped reads are assumed to be evenly distributed on each of the paths. The expected number $\hat{r_i}$ of reads falling into the i-th expressed segment or junction is proportional to both the length of the expression feature and the sum of the expression levels of all paths containing the i-th expressed segment or junction such that $\hat{r_i} = \sum_{j=1}^{N} a_{i,j} x_j$. Let the observed number of reads falling into the i-th expressed segment or junction be r_i. The expression levels of the paths, $X = \{x_1, ..., x_N\}$, are then determined by using the abundance values X^* that minimizes the following residual sum of squares:

$$X^* = \arg\max_X \frac{1}{2}(r_i - \hat{r_i})^2 = \arg\max_X \frac{1}{2}(r_i - \sum_{j=1}^{N} a_{i,j} x_j)^2 \tag{3.1}$$

with respect to the constraints that $x_j \geq 0$ for all $1 \leq j \leq N$. In the implementation of SDEAP, an R package opt is used to solve the quadratic optimization problem by the L-BFGS-B algorithm [16].

3.2.3 Analysis of Background Variance

SDEAP chooses expression features with observed variance significantly greater than the background variance. In the literature [4, 101], the observed variance of expression features is postulated as due to technical noise and biological variation among biological conditions. Thus, the technical noise is usually used as the back-

ground variance. For an expression feature f, the expected variance ρ due to technical noise is modeled as a quadratic function of the observed mean μ_f such that $\rho(\mu_f) = \mu_f + \phi{\mu_f}^2$, where the parameter ϕ is called the dispersion of samples. When the biological conditions of the input samples are given, the dispersion ϕ is estimated by regression using the conditioned mean and variance $(\hat{\mu}_{f_A}, \hat{\rho}_{f_A})$ of each expressed feature f, where the conditioned sample mean $\hat{\mu}_{f_A}$ and variance $\hat{\rho}_{f_A}$ are the observed mean and variance in the samples of the same condition A. However, in our case, the biological conditions are not given *a priori* In this case, the quadratic function $\rho(\mu_f)$ is suggested to be fitted to the overall mean and variance $(\hat{\mu}_f, \hat{\rho}_f)$, where $\hat{\mu}_f$ is the average and $\hat{\rho}_f$ is the variance of the expression feature f in all samples, to approximate the real dispersion [14]. In the implementation of SDEAP, the model fitting is implemented by using the general linear regression model GLM package in R [29]. After the estimation of the dispersion, for every feature f, the expected variance $\rho(\mu_f)$ is used as the background variance at its expression level μ_f. In general, expression features with low background variance are preferred, *e.g.*, less than 0.3 [13]. An expression feature f is selected as an informative feature if $\hat{\rho}_f/\rho_f > \hat{\gamma}$ where $\hat{\gamma}$ is given as a user defined threshold, as employed in [4, 14].

3.2.4 Testing Differential Transcript Expression

Testing the difference of an informative feature f without given biological conditions includes two main steps: (1) clustering the instances of f in the population

and (2) testing the statistical significance of difference between the clusters of the instances. In SDEAP, the clustering is done by using Dirichlet infinite mixture models. Then, the one-way ANOVA test is used to provide the statistical measurement of significance [37].

To illustrate a Dirichlet infinite mixture model, we start from the Gaussian mixture model of fixed k components and then let k goes to infinity. When the input population data consist of n RNA samples $S = \{s_1, s_2, s_3, ..., s_n\}$, every expression feature f has n instances, $F = \{f_1, f_2, ..., f_n\}$, where f_i is the instance of feature f in the RNA-Seq sample s_i. Each of the instances, f_i, is assumed to follow the Gaussian distribution of some mean μ and variance ρ and denoted as $f_i \sim N(\mu, \rho)$. In the Gaussian mixture model, the instances of f are assumed to be generated from exactly k Gaussian distributions. Let $C = \{c_1, c_2, ..., c_n\}, c_i \in \{1, 2, ..., k\}$ be a set of component indices such that each index c_i indicates which component f_i belongs to. Let $\mu = \{\mu_1, \mu_2, ..., \mu_k\}$ and $\rho = \{\rho_1, \rho_2, ..., \rho_k\}$ be the means and variance of the k components. The likelihood function of C given F can be written as:

$$Pr(C, F|\pi, \mu, \rho) = \prod_{i=1}^{n} \sum_{j=1}^{k} I(c_i = j)\pi_j N(f_i|\mu_j, \rho_j) \tag{3.2}$$

where $\pi = \{\pi_1, \pi_2, ..., \pi_k\}$ such that π_j is the probability of instance f_i belonging to component j, I is an indicator function and $N(f_i|\mu_j, \rho_j)$ is the probability that f_i is drawn from a normal distribution with the mean μ_j and variance ρ_j. When k is

fixed, the clustering C of the feature instances and the model parameters μ, ρ can be determined by the parameters that maximize the likelihood function given in Eq. (2) by using the EM algorithm [27]. However, if k is not given as a prior and allowed to be infinitely large, Eq. (2) is intractable and the estimation of the likelihood function cannot be done by the EM approach. The MCMC algorithm is a well-known technique to get around the intractability [86] . The main idea of the MCMC algorithm is to sample parameters from the conditional posterior of the parameters and update each parameter in turn. To apply the MCMC algorithm, the posterior probability functions of $c_i = c_j$ and $c_i \neq c_j$ for all $i \neq j$ given the model parameters μ, ρ and the observed feature values F are required for sampling. To derive the posterior probability functions, the prior probability functions of the parameters are assumed as follows. The vector π is assigned a Dirichlet prior, $\pi \sim \text{Dirichlet}(\alpha/k, ..., \alpha/k)$, with k concentration parameters α/k where α is a hyper parameter given as a prior. Note that α/k approaches zero when k goes to infinity. By integrating the the mixing proportion π, the prior probability that $c_i = j$ given $c_1, ..., c_{i-1}$ is written as

$$Pr(c_i = j | c_1, ..., c_{i-1}) = \frac{n_{i,j} + \alpha/k}{i - 1 + \alpha}, \tag{3.3}$$

where $n_{i,j}$ is the number of the component index $c_{i'} = j$ given all the indicator $c_{i'}, i' < i$. If k approaches infinity, the conditional prior probability of f_i belonging to

component j, *i.e.*, $c_i = j$, is

$$Pr(c_i = j|c_1, ..., c_{i-1}) \rightarrow \frac{n_{i,j}}{i - 1 + \alpha} \tag{3.4}$$

Similarly, the conditional prior probability of f_i not belonging to any component $c_{i'}$ is

$$Pr(c_i \neq c_{i'}, i' < i|c_1, ..., c_{i-1}) \rightarrow \frac{\alpha}{i - 1 + \alpha} \tag{3.5}$$

By combining the likelihood function of Eq. (2) with Eq. (4) and Eq. (5), the conditional posterior probability functions for $c_i = c_j$ with the given model parameters, μ and ρ, and observed feature values F are

$$Pr(c_i = j|C_{-i}, F, \mu, \rho) \propto b\frac{n_{i,j}}{i - 1 + \alpha}N(f_i|\mu_j, \rho_j), \tag{3.6}$$

where b is a constant for normalization. The conditional posterior probability functions for $c_i \neq c_j, j \neq i$ are

$$Pr(c_i = j|C_{-i}, F, \mu, \rho) \propto b\frac{\alpha}{i - 1 + \alpha}\int N(f_i|\mu_j, \rho_j)dG_0, \tag{3.7}$$

where G_0 is the prior probability of μ and ρ. The component indicators C are determined by sampling from a Markov chain of the posterior probabilities with Eq. (6) and Eq. (7) as its equilibrium distribution. More detailed derivation of the prior and posterior probabilities of the parameters is discussed in [86]. In the implemen-

tation of SDEAP, a python package scikit-learn is used [98]. After the clustering of the feature values F, a one-way ANOVA test is performed to test if the clusters are indeed significantly different. The null hypothesis for ANOVA is that the mean is the same for all clusters of F. In SDEAP, the p-values from the ANOVA test are corrected to the false discovery rate (FDR), the rate of type I errors in multiple null hypothesis testing [8]. If the FDR value is small enough, *e.g.*, less than 0.1, we reject the null hypothesis and conclude that the feature f is different among the RNA-Seq samples. If an ASM has a differentially expressed feature, the ASM will be reported as an alternative splicing event. If a gene has a differentially expressed feature, the genes will be predicted as a DTE gene.

3.2.5 Evaluation Metrics

All our experimental results are evaluated in terms of precision (PRE), PRE = TP/(TP + FP), and recall (REC), REC = TP/(TP + FN), where TP is the number of true positives, FP the number of false positives and FN the number of false negatives. To combine the two evaluation measures, the area under the precision-recall curve (or AUC_{pr}) is used as a measure of the overall performance of a prediction method in our tests. To assess the similarity of In this manuscript, an R package PRROC is used to calculate the PRE, REC and AUC_{pr} scores [43]. To measure the similarity between clusters of RNA-Seq samples and real biological conditions, Jaccard indices are calculated as follows [111]. Given two clusterings, C and C', of the

input RNA-Seq samples, let s be the number of pairs of samples that belong to the same cluster in both clusterings, d_1 the number of pairs of samples that belong to the same cluster in clustering C but not in C', and d_2 the number of pairs of samples that belong to the same cluster in clustering C' but not in C. The Jaccard index J is then defined as $\mathrm{J} = \mathrm{s}/(\mathrm{s} + \mathrm{d}_1 + \mathrm{d}_2)$.

3.3 Experimental Results

SDEAP and DEXUS$_{exon}$ are tested on both simulated and real datasets. In our simulation study, several realistic configurations of real RNA-Seq data are considered. In the first simulation, bimodal RNA-Seq data are simulated, while data are generated from three or more overlapping groups in the second simulation. Noise unique to single-cell RNA-Seq is introduced in the third simulation to test the robustness of the methods in dealing with data with high background variance. To avoid biased assessment of prediction accuracy due to random sampling, the simulation experiment on every configuration is repeated 10 times and the average performance of each method is reported. The simulation performed in [61] is also repeated as the forth simulation experiment to assess the performance of SDEAP in calling DS genes. The results of SDEAP are then compared with those of SigFuge reported in [61]. In our experiments on real data, the predicted expression features of DTE genes are used to cluster the input RNA-Seq samples. In particular, the differentially expressed exons of each DTE gene predicted by DEXUS$_{exon}$ are used as the expression features of that

gene for DEXUS$_{exon}$, while differentially expressed ASMs are the expression features for SDEAP. To avoid biases due to the sizes of genes, if a predicted DTE gene has more than one differentially expressed feature, the feature with the greatest significance measurement, *i.e.*, FDR in SDEAP or I/NI scores in DEXUS$_{exon}$, is selected as the informative expression feature for the DTE gene. Given the expression features, all samples are clustered by a widely used hierarchical clustering package MADE4 in gene expression analysis [22]. The performance of SDEAP and DEXUS$_{exon}$ are compared by the similarity between the clustering and known biological conditions in the real dataset, because we believe the variance of expression features from correctly predicted DTE genes can reflect the biological conditions in the data. Note that SigFuge is not included in the comparisons of clustering results here because it is a DS analysis method and it only provides p-values that measure the variance of individual genes among samples. In other words, it is difficult to extract expression features from the prediction by SigFuge that can be used to cluster samples. Although our real data analysis will include two experiments on single-cell RNA-Seq data, the extended protocol of DESeq2 for single-cell RNA-Seq data is not compared in the experiments because it requires spike-in ERCC information and our single-cell RNA-Seq datasets do not contain spike-in ERCC information [13].

3.3.1 Experiments on Simulated Data

Simulation of RNA-Seq samples

A population is a collection of RNA-Seq samples with different biological conditions. In our simulation experiments, each biological condition is associated with an expression profile of transcripts. The expression profiles of transcripts are generated by different protocols for different study purposes and will be discussed later in each simulation experiment. Given an expression profile of transcripts, to synthesize RNA-Seq reads for an RNA-Seq sample, we first randomly draw a number of cDNA fragments from each transcript in the RNA-Seq library of the sample according to the negative binomial distribution. In our simulation, a moderate size, 40 million reads, RNA-Seq library is assumed. Hence, for every transcript, the mean of the negative binomial distribution is set as the product of the transcript expression value (FPKM), the length of the transcript in thousand bps and the size of the RNA-Seq library in millions (40), while the dispersion value of the distribution is set as $\phi = 0.179$ as done in the literature [61]. Then, paired-end RNA-Seq reads of 50 bps each are obtained from both ends of each synthesized cDNA fragment. However, in real RNA-Seq data, the observed variance of transcript expression is @articlePRROC, title=PRROC: computing and visualizing precision-recall and receiver operating characteristic curves in R, author=Graul,J. and others, journal=Bioinformatics, volume=31(15), pages=2595-2597, year=2015 significantly greater than the sample variance modeled by the negative binomial distribution [90]. Two studies based on real RNA-Seq data show that

approximately 5% genes in the same biological condition have significantly higher variance of transcript expression than expected due to outliers [90, 40]. To simulate datasets that reflect real RNA-Seq data as much as possible, 5% genes are selected as genes that contain outliers in their expression profiles. Extreme high values of transcript expression are usually detected in approximately 10% real RNA-Seq samples in the expression profiles [40]. To simulate extreme high values of transcript expression, we allow the cDNA fragments from the corresponding transcripts of the selected genes to have 10% probability of being amplified from 5 to 10 times as employed by [144]. In addition to the extreme high expression values of transcripts, a study shows the exons of lowly expressed transcripts are ubiquitously missing in every one of two technical or biological replicates [80]. To include the missing-value events, among the 5% selected genes, the transcripts of lowly expressed genes whose expression values are lower than 1.0 have a probability from 30% to 50% of being assigned zero cDNA fragments. During the synthesis of cDNA fragments, a positional profile that reflects positional biases due to complementary DNA fragmentation is used for each transcript as done in [73]. Afterwards, simulated RNA-Seq reads are used as the input data to evaluate the DTE analysis methods. Throughout our simulation experiments, the hg38 transcript annotation of the human genome from the UCSC genome browser is used as the annotation of transcripts for SDEAP and DEXUS$_{exon}$. Only genes that have at least two transcripts in the annotation are considered in our simulated datasets.

Performance on RNA-Seq data from two conditions

In this simulation experiment, RNA-Seq samples are generated from two expression profiles of transcripts and used to assess the performance of SDEAP and DEXUS$_{exon}$ on RNA-Seq data from two biological conditions. The two expression profiles that correspond to two biological conditions are created as follows. In the first of the two expression profiles of transcripts, the expression value (FPKM) of each transcript is randomly drawn from a log-normal distribution as done in the literature [73]. Every transcript with a decent FPKM (> 0.1) is regarded as an expressed transcript. Each genes without expressed transcripts is removed such that our benchmark datasets are comprised of 3089 genes. To create the second expression profile, among the 3089 genes, 308 ($\sim$10%) genes are chosen as DTE genes. All the DTE genes are evenly divided into three categories: up-regulated, down-regulated and differentially spliced genes. For each up-regulated gene, a detectable transcript is randomly selected and its abundance is increased by a factor of at least 4, a widely used threshold to define differential expression in the literature [136, 15]. Similarly, from each down-regulated gene, the abundance of a randomly selected transcript is decreased by a factor of at least 4. For each differentially spliced genes, the maximum and minimum abundance values of its transcripts are swapped. For the other 2781 EE genes, the abundance values of their transcripts remain the same in the both expression profiles.

We choose a specific number of replicates in each biological condition, n_1 and n_2, to evaluate the prediction accuracy of SDEAP and DEXUS$_{exon}$ on balanced and

unbalanced data of various group sizes. If $n_1 > n_2$, the n_1 RNA-Seq samples are called the majority group and the n_2 samples are called the minority group. The configurations (6,6), (50,50), (9,3), and (20,4) of (n_1, n_2) are chosen from the simulation experiments for DEXUS and SigFuge [63, 61]. To further study the performance on unbalanced data of greater group sizes, the unbalanced configurations (9,3) and (20,4) are multiplied by 4 to create another two configurations (36,12) and (80,16). The performance of both methods on all group size configurations is summarized in Table 3.1.

SDEAP clearly outperforms DEXUS$_{exon}$ in terms of the AUC$_{pr}$. The average AUC$_{pr}$ for SDEAP over all configurations is 0.809 and the average AUC$_{pr}$ for DEXUS$_{exon}$ is only 0.624. SDEAP outperforms DEXUS$_{exon}$ by at least 0.09 and 0.1 in the precision and recall scores, respectively. In general, increasing the number of samples benefits the accuracy of prediction. Both methods achieve the best performance on the balanced configuration (50,50) of the largest size. We observe that the precision scores of SDEAP are generally higher than those of DEXUS$_{exon}$. This is because the background variance estimation performed in SDEAP makes it less sensitive to the background noise. Notably, the prediction accuracy of DEXUS$_{exon}$ is somehow related to the proportion of the minority group and drops drastically, from 0.789 to 0.513 in the AUC$_{pr}$, when the proportion of the minority groups decreases. The performance of SDEAP is more robust with respect to the decrease of the proportion of the minority group until it drops down to 16.6% in the last two experiments with

configurations (20,4) and (80,16). In these experiments, the observed variance in the expression profiles of true DTE genes is close to the background variance due to the outliers.

Table 3.1: Comparison of the two DTE analysis methods on simulated datasets from binary conditions. The configuration (n_1, n_2) indicates the number of replicates in each condition. AUC_{pr}, PRE and REC denote the area under the precision-recall curve, precision and recall scores, respectively, averaged over the 10 repetitions. The standard deviation of each score is included in the parentheses following the score. The last row, AVG, shows the the average of AUC_{pr}, PRE and REC in 6 experiments.

	SDEAP			DEXUS_{exon}		
Configuration	AUC_{pr}	PRE	REC	AUC_{pr}	PRE	REC
(6, 6)	0.837 (0.02)	0.911 (0.03)	0.887 (0.03)	0.766 (0.02)	0.702 (0.03)	0.665 (0.03)
(50, 50)	0.838 (0.01)	0.838 (0.01)	0.977 (0.01)	0.789 (0.01)	0.749 (0.01)	0.873 (0.01)
(9, 3)	0.857 (0.02)	0.905 (0.03)	0.873 (0.03)	0.533 (0.02)	0.681 (0.03)	0.655 (0.03)
(36, 12)	0.858 (0.01)	0.896 (0.02)	0.893 (0.02)	0.579 (0.01)	0.680 (0.02)	0.715 (0.02)
(20, 4)	0.726 (0.01)	0.835 (0.02)	0.788 (0.02)	0.513 (0.01)	0.678 (0.02)	0.642 (0.02)
(80, 16)	0.737 (0.01)	0.833 (0.01)	0.831 (0.01)	0.565 (0.01)	0.705 (0.01)	0.701 (0.01)
AVG	0.809	0.869	0.874	0.624	0.699	0.708

Performance on RNA-Seq data from three or more conditions

To evaluate the performance of SDEAP and DEXUS_{exon} on population data from three or more biological conditions, we simulate RNA-Seq samples using three or five expression profiles of transcripts. In these simulation experiments, the expression of transcripts is generated from groups (*i.e.*, conditions) that largely overlap with each other. The experimental configurations for generating the mixture of expression values is set up to reflect the reality in some challenging practical applications, *e.g.*, RNA-Seq data sampled at serial time points [1]. Given the two expression profiles of

Table 3.2: Comparison of the two DTE analysis methods on simulated datasets from 3 or more conditions. The configuration $(n_1, n_2, ...)$ indicates the number of replicates in each condition. Again, AUC_{pr}, PRE and REC denote the area under the precision-recall curve, precision and recall scores, respectively, averaged over the 10 repetitions. The standard deviation of each score is included in the parentheses following the score. The last row, AVG, shows the the averages of AUC_{pr}, PRE and REC in 6 experiments.

	SDEAP			$DEXUS_{exon}$		
Configuration	AUC_{pr}	PRE	REC	AUC_{pr}	PRE	REC
(10,10,10)	0.701 (0.02)	0.886 (0.03)	0.681 (0.03)	0.545 (0.02)	0.631 (0.03)	0.483 (0.03)
(10,10,30)	0.645 (0.01)	0.907 (0.01)	0.506 (0.01)	0.466 (0.01)	0.485 (0.01)	0.272 (0.01)
(10,10,10,10,10)	0.625 (0.01)	0.887 (0.01)	0.486 (0.01)	0.446 (0.01)	0.467 (0.01)	0.255 (0.01)
(20,10,10)	0.767 (0.02)	0.894 (0.02)	0.85 (0.02)	0.641 (0.02)	0.696 (0.02)	0.662 (0.02)
(20,10,30)	0.677 (0.01)	0.886 (0.01)	0.633 (0.01)	0.507 (0.01)	0.590 (0.01)	0.422 (0.01)
(20,10,10,10,10,10)	0.669 (0.01)	0.887 (0.01)	0.612 (0.01)	0.487 (0.01)	0.581 (0.01)	0.403 (0.01)
AVG	0.681	0.891	0.628	0.515	0.575	0.416

transcripts in the previous simulation experiments for binary conditions, we generate three additional expression profiles of transcripts as follows. The third expression profile of transcripts is the average of the first and second profile, *i.e.*, the average of each transcript in the first and second profiles is calculated and assigned as the expression value of the transcript in the third expression profile. Similarity, the fourth expression profile is the average of the first and the third profile and the fifth expression profile is the average of the second and the third profile. Again, we consider different combinations of group sizes to study the performance of SDEAP and $DEXUS_{exon}$ on balanced and unbalanced data of multiple biological conditions. The prediction accuracy of SDEAP and $DEXUS_{exon}$ is assessed in Table 3.2.

Again, SDEAP shows better performance than $DEXUS_{exon}$. The AUC_{pr} scores of SDEAP are at least 0.126 higher than the score of $DEXUS_{exon}$ in every experi-

ment setting. Note that DEXUS$_{exon}$ yields very low recall scores, 0.272 and 0.255, on configurations (10, 10, 30) and (10, 10, 10, 10, 10). This may be due to its lack of ability to determine the correct numbers of clusters for the expression features. The assumption of binary conditions in DEXUS$_{exon}$ may likely lead it to two inseparable clusters of expression features that could negatively impact the statistical test for the significance of difference. Moreover, we find that the precision scores of SDEAP across different configurations of the group sizes are much higher than those of DEXUS$_{exon}$, which suggests that the control of false positives in SDEAP is better than that in DEXUS$_{exon}$. We also notice that the recall scores of SDEAP and DEXUS$_{exon}$ increase, from 0.506 to 0.612 and from 0.255 to 0.405, respectively, with the proportion of samples in the first two groups. This is because the observed variance of the expression features of true DTE genes increases with the proportion of the samples in the first two groups and when the observed sample variance of true DTE genes is significantly higher than the background variance, calling the true DE (*i.e.*, differentially expressed) genes becomes much easier for both methods.

Robustness on noisy data

Single-cell RNA-Seq serves as a fundamental tool to measure the expression of transcripts in individual cells and has numerous applications in biological research [14]. However, due to the low abundance of transcripts in an individual cell, the technical noise in single-cell (SC) RNA-Seq data is much higher than that in the

Table 3.3: Comparison of the two DTE analysis methods on simulated single-cell RNA-Seq data. The configuration $(n_1, n_2, ...)$ indicates the number of replicates in each condition. Again, AUC_{pr}, PRE and REC denote the area under the precision-recall curve, precision and recall scores, respectively, averaged over the 10 repetitions. The standard deviation of each score is included in the parentheses following the score. The last row, AVG, shows the the averages of AUC_{pr}, PRE and REC in 4 experiments.

	SDEAP			$DEXUS_{exon}$		
Configuration	AUC_{pr}	PRE	REC	AUC_{pr}	PRE	REC
(50,50)	0.615 (0.01)	0.649 (0.01)	0.837 (0.01)	0.406 (0.01)	0.376 (0.01)	0.483 (0.01)
(80,16)	0.432 (0.01)	0.556 (0.01)	0.561 (0.01)	0.271 (0.01)	0.025 (0.01)	0.025 (0.01)
(10,10,10)	0.414 (0.02)	0.621 (0.03)	0.662 (0.03)	0.250 (0.02)	0.186 (0.03)	0.198 (0.03)
(20,10,10)	0.474 (0.02)	0.673 (0.02)	0.698 (0.02)	0.304 (0.02)	0.222 (0.02)	0.231 (0.02)
AVG	0.484	0.624	0.689	0.308	0.202	0.234

standard RNA-Seq data. To evaluate the performance of SDEAP and $DEXUS_{exon}$ on simulated SC RNA-Seq data, technical noise unique to SC data is included in the simulation. In a typical SC data, the transcripts of a gene with moderate or high abundance in one cell may not be detected in another cell. The failure of detecting the transcripts is called a *dropout* event. In a previous study [14], 13.3% of the genes were observed to have experienced dropouts in a real SC RNA-Seq sample [59]. Hence, 13.3% genes in our simulations are selected as genes with dropout events. The probability that the transcripts of a gene are not detected in a sample is called the dropout rate of the gene. The dropout rate of a gene is known to be related to the read count of the gene [59]. To estimate the dropout rates in a real SC dataset, 12 SC RNA-Seq samples of the same cell type from mouse mESC cells are downloaded from the NCBI GEO database with accession code GSE42268 [105]. In each of the samples, the read counts of all genes (in millions) are calculated and normalized by

the total number of reads in the sample. The logarithm of the normalized read counts (LRC) from 0 to 2 is discretized by binning the interval [0,2] into 20 intervals of the same length 0.1. Then, all genes with LRC from 0 to 2 are first assigned to the 20 bins according to their LRCs. The remaining genes with LRC < 0 are assigned to the first bin and the genes with the LRC > 2 to the last bin. For every gene in a bin, the proportion of the samples where the gene has zero abundance is calculated and then the average of the proportion is used as the dropout rate for all the genes whose LRC is in the corresponding interval of the bin. In addition to dropout events, the observed dispersion rate in SC data is also higher than the dispersion estimated from standard RNA-Seq data. Thus, the dispersion rate is increased to 0.25 when generating the number of cDNA fragments for each transcript in our simulations. Here, we reuse the group sizes in some experiments on standard RNA-Seq data to study the prediction accuracy of SDEAP and DEXUS$_{exon}$ on balanced and unbalanced SC data. The results on four simulated SC dataset are summarized in Table 3.3.

Similar to the results in the previous simulations of standard RNA-Seq data, SDEAP significantly improves the precision and recall scores of DEXUS$_{exon}$ by at least 0.273 and thus achieves much better overall performance. Due to high technical noise, the performance of both methods declines, but in very different scales. The average precision scores of SDEAP decreases by 0.241 and its recall score drops only 0.133, while the average precision and recall scores of DEXUS$_{exon}$ drop 0.48 and 0.429, respectively. This shows that SDEAP is more robust with respect to the increased

technical noise. Note that among the four configurations, DEXUS$_{exon}$ has very low prediction accuracy, at most 0.222 and 0.231 in precision and recall, respectively, except on the balanced binary configuration (50,50). This suggests that DEXUS$_{exon}$ may not be suitable for treating SC data. The conclusion is consistent with the results of our later experiments on real SC data.

Table 3.4: Comparison of the performance in differential splicing analysis.

Setting No.	(ψ_1, ψ_2)	SDEAP	SigFuge	DEXUS$_{exon}$
1	(50, 50)	0	2	0
2	(50, 50)	100	89	13
2	(75, 25)	100	98	10
3	(50, 50)	100	99	23
3	(75, 25)	100	60	56

Detecting changes in relative abundance of transcripts

In order to study the effectiveness of SDEAP in DS analysis, we perform simulation experiments to compare its with performance with DEXUS$_{exon}$ and SigFuge. In our simulations, two hypothetical gene models concerning two transcripts, isoform t_1 and t_2, are considered as illustrated in Supplementary Figure S1. The first model is a three exon gene model that has a cassette exon excluded from isoform t_1 but retained in isoform t_2. The second model is a four exon gene model containing mutually exclusive cassette exons. The length of each exon is 400 bps. Three different experimental configurations are considered in the simulations. For each configuration, RNA-Seq

samples of binary groups are simulated. Let the relative abundance values of the two transcripts t_1 and t_2 be ψ_1 and ψ_2, respectively. In the first configuration, the three exon model is used where the relative abundance values, (ψ_1, ψ_2), are set as (0.5, 0.5) for both groups in order to evaluate the number of false positives in the prediction. In the second configuration, (ψ_1, ψ_2) is set as (0.75, 0.25) and (0.25, 0.75) for the first and second groups of the three exon model, respectively, in order to evaluate the number of true positives (or sensitivity). In the third configuration, the same abundance values of the two transcripts in the second configuration are used in the two sample groups but the gene model has four exons. In each configuration, two combinations of group sizes (n_1, n_2), (50, 50) and (75, 25), are considered. The numbers of RNA-Seq reads for each of the two transcripts are sampled from two negative binomial distributions with means $l_1 \times \alpha \times \psi_1$ and $l_2 \times \alpha \times \psi_2$, respectively, and dispersion $\phi = 0.179$ where $\alpha = 100$ is the fixed read coverage per bp, l_1 and l_2 are the lengths of the two transcripts. Then, for each sample and each isoform, 50bp reads are synthesized uniformly across the two isoforms in the gene models. The simulated reads are used as mapped reads and provided to the three methods as the input data. The experiment in each configuration is repeated 100 times and the results are summarized in Table 3.4.

In the first configuration, only SigFuge makes two false positive calls when DEXUS$_{exon}$ and SDEAP predict no false positives. In the second and third configurations, no true positive is missed in the prediction by SDEAP. SigFuge also provides high sensitivity

in the second configuration. However, its sensitivity drops in the third configuration when the sizes of the groups become unbalanced. Notably, DEXUS$_{exon}$ predicts with low sensitivity in both the second and third configurations. The poor performance of DEXUS$_{exon}$ could be due to the limited number of expression features in the gene models used here. In other words, the number of expression features is not sufficient for estimating the parametric model of DEXUS$_{exon}$ accurately. In contrast, SDEAP consistently provides high sensitivity and zero false positives in all three experimental settings. This good performance may be attributed to its robust clustering algorithm and feature selection method.

3.3.2 Experiments on Real Data

SDEAP detects different cancer subtypes

Some critical diseases, *e.g.*, breast cancer (BC), are known as heterogeneous diseases with a variety of transcriptomic alterations that severely affect the diagnosis and prognosis of the diseases. Identifying DTE genes as the transcriptomic biomarkers of the subtypes of such diseases could be important for the design of clinical trials to investigate targeted treatments. In this experiment, SDEAP and DEXUS$_{exon}$ are applied to a recently published RNA-Seq dataset including 17 individual human tissues belonging to three subtypes of breast cancers: TNBC, Non-TNBC and HER2-positive. The RNA-Seq reads of the BC samples are aligned against the Ensembl GRCh37.62 B (hg19) reference genome using TopHat [121] and the mapped reads are

used as the input data for the two methods.

SDEAP predicts 1366 DTE genes with the FDR under 0.1. These DTE genes are compared with the top 1366 genes in the ranked list of DTE genes predicted by DEXUS$_{exon}$ in the following analysis. In the BC dataset, 6 differentially expressed genes are validated experimentally by qPCR with fold change rates greater than 5.0. Three of the six validated DE genes are predicted by SDEAP while two are among the predicted DTE genes by DEXUS$_{exon}$. The dendrograms of hierarchical clustering using the DTE genes predicted by SDEAP and DEXUS$_{exon}$ are illustrated in Fig. 3.2. The 17 samples in each dendrogram are partitioned into three clusters and compared with the three subtypes of BC. In the clustering by SDEAP, only one of the 17 samples is misclassified while there are three misclassified samples in the clustering by DEXUS$_{exon}$. The Jaccard index of SDEAP's clustering is 0.760, which is significantly higher than that of DEXUS$_{exon}$ (0.481). The better clustering of the BC samples achieved by SDEAP is an evidence suggesting that SDEAP might have predicted more accurate DTE genes specific to the BC subtypes, while DEXUS$_{exon}$ might be more sensitive to random outliers.

SDEAP identifies more validated marker genes specific to cell types

Understanding the development and functions of a tissue or an organ requires the identification of all of its cell types [120]. The expression patterns of transcripts in

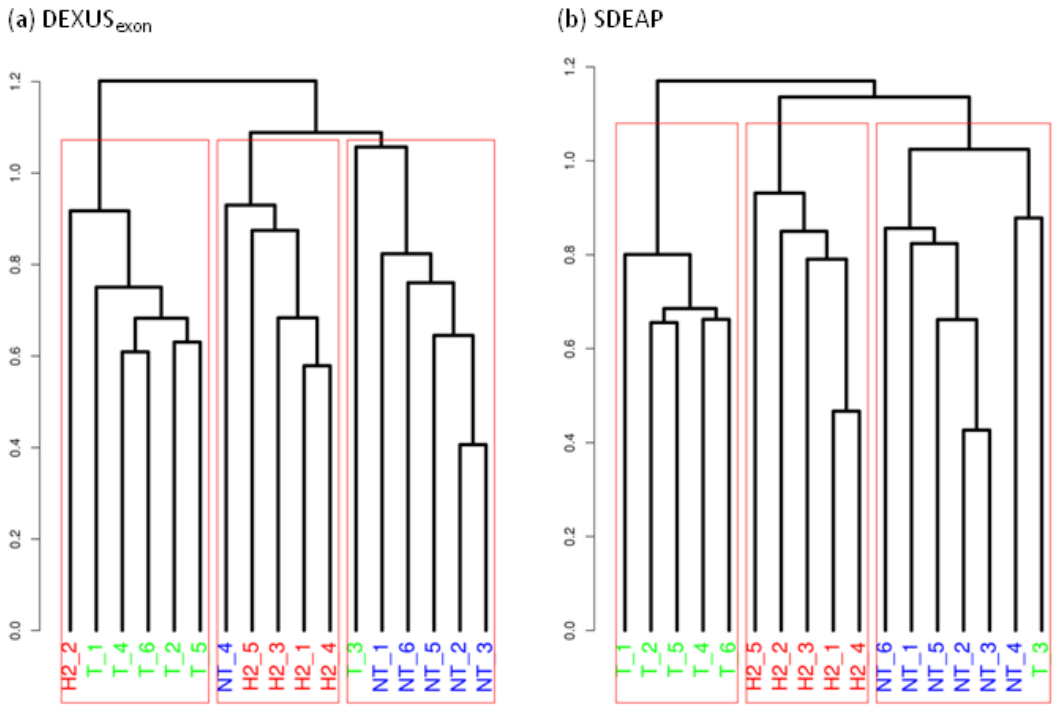

Figure 3.2: The dendrograms of the hierarchical clustering for the breast cancer dataset. Plots (a) and (b) depict the clustering by SDEAP and DEXUS$_{exon}$. The Y-axis is the measurement of similarity between the samples and X-axis are the labels of each sample. The HER2 samples are colored red, the TNBC samples green and the non-TNBC samples blue. The three red boxes in each dendrogram illustrates three clusters obtained by the corresponding method.

individual cells of various cell types can be revealed by the SC RNA-Seq technology and the DE transcripts have been used as biomarkers to separate different cell types and to analyze alternative cellular functions of the cell types [105]. In this experiment, a SC RNA-Seq dataset of two cell types, 12 mouse ES cells and 12 primitive endoderm (PrE) cells, is downloaded from the NCBI GEO database with accession code GSE42268. The RNA-Seq reads are mapped to the mouse reference genome (mm9) by TopHat and used as the input data for SDEAP and DEXUS$_{exon}$. On this dataset,

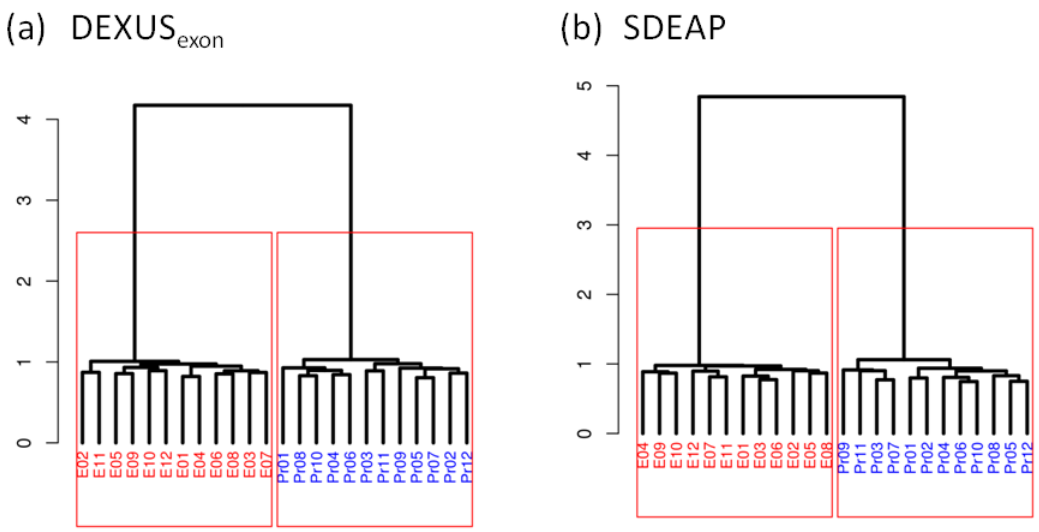

Figure 3.3: Hierarchical clustering of 12 mECS and 12 Pr cells based on the DTE genes predicted by SDEAP and DEXUS$_{exon}$. Plots (a) and (b) depict the dendrograms obtained by DEXUS$_{exon}$ and SDEAP, respectively. The Y-axis is the measurement of similarity between the samples and the X-axis shows the labels of the mESC and Pr cells. The mESC cells are colored red and the Pr cells blue. The two red boxs illustrate two clusters obtained from each clustering consistent with the cell types.

SDEAP predicts 1614 genes with the FDR less than 0.1, and the top 1614 DTE genes predicted by DEXUS$_{exon}$ are used in the following analysis. The predicted DTE genes by SDEAP and DEXUS$_{exon}$ are compared with manually selected biomarkers [105] and the expression features of the predicted DTE genes are used to cluster the 24 SC samples.

More specifically, 17 DE genes associated with critical cellular functions during cell differentiation are manually selected as the biomarkers [105]. The 17 biomarkers are all predicted by SDEAP while four of them are missed by DEXUS$_{exon}$. Among the 17 DE genes, 8 were further validated by qPCR in [105]. Although all of these 8 validated

biomarkers are among the DTE genes predicted by SDEAP, 2 of them are missed by DEXUS$_{exon}$. The detailed prediction results are summarized in Table 3.5. The better coverage of the biologically meaningful biomarkers by SDEAP suggests that it can provide a more comprehensive picture of transcript expression and is thus valuable for understanding the development and functions of cells. Although the comparison of the DTE prediction results by SDEAP and DEXUS$_{exon}$ to the manually selected biomarkers shows significant difference, the clustering of the 24 samples using their predicted DTE genes are equally well. The samples are perfectly separated by their cell types as shown in Figure 3.3. This can perhaps be explained by the fact that the two cell types have redundant biomarkers in the sense even though some of them were missed by DEXUS$_{exon}$, the rest are still able to separate the cell types.

Table 3.5: Comparison of the numbers of the manually selected and qPCR validated maker genes for the mESC and Pr cells in the DTE genes predicted by SDEAP and DEXUS$_{exon}$. The second column indicates the total numbers of manually selected or qPCR validated marker genes. The numbers of manually selected or validated maker genes that appear in the DTE genes predicted by the two methods are given in the third and fourth columns.

Type	All	SDEAP	DEXUS$_{exon}$
manually selected	17	17	13
aPCR validated	8	8	6

SDEAP is better at separating cell-cycle phases

Heterogeneity of transcript expression is not only found across different cell types but also observed between different cell-cycle phases of the same cell type [14]. To investigate the performance of the two DTE analysis methods in detecting DTE

between different cell-cycle phases, 35 SC RNA-Seq samples of mESC cells, where the cell-cycle phases of each cell is known *a priori*, are used as the benchmark data. Among the 35 samples, there are 20 cells in the Growth 1 phase (G1), 8 in the pre-mitotic/mitotic (G2/M) phase and 7 in the synthesis (S) phase. The SC RNA-Seq dataset is also downloaded from the NCBI GEO database with the accession code GSE42268. All sequenced reads of each RNA-Seq sample are aligned against the Ensembl GRCh37.62 B (mm9) reference genome using TopHat and the mapped reads are used as the input for SDEAP and DEXUS$_{exon}$.

Since this dataset does not offer any qPCR validated DTE genes, we use the clustering results of the 35 cells to indirectly examine whether cell-cycle dependent heterogeneity features among the mESC cells can be identified by SDEAP and DEXUS$_{exon}$. The same FDR threshold of 0.1 is used to call DTE genes in SDEAP. 532 genes are predicted as DTE genes by SDEAP. They are then compared with the top-ranked 532 DTE genes predicted by DEXUS$_{exon}$. The reason that SDEAP predicted fewer DTE genes than in the previous study concerning cell types can be explained by the subtle difference of transcript expression between cell-cycle phases. The similarity of the 35 SC samples encoded by the expression features of the predicted DTE genes is visualized in the 3D space by principal component analysis (PCA), as shown in Fig. 3.4(a) and 3.4(b). In the PCA transformation using the DTE genes predicted by SDEAP, although some S cells are mixed with G1 and G2/M cells, the cells of the three cell-cycle phases are still visually separable. However, in the PCA transforma-

tion using the DTE prediction of DEXUS$_{exon}$, all cells of the three cell-cycle phases are mixed together such that the separation between the cell-cycle phases becomes more subtle. The hierarchical clustering using the DTE features identified by SDEAP and DEXUS$_{exon}$ are illustrated in Fig 3.4(c) and 3.4(d), respectively. To assess the quality of the clustering, all cells in the dendrograms are partitioned into three clusters, as shown by the red boxes. The clusters are then compared with the three cell-cycle phases. In the clustering by SDEAP, some S cells are clustered together with G1 cells while the other S cells are with G2/M cells. This makes some sense because the S cell-cycle phase is between the G1 and G2/M phases in the cell-cyle and hence the expression profiles of some S cells are closer to those of G1 cells while the other S cells might be closer to G2/M cells. In general, the G1 cells and G2/M cells are well separated by SDEAP. However, the clustering by DEXUS$_{exon}$ fails to provide a reasonable partition consistent with the cell-cycle phases. As a result, the Jaccard index of the DEXUS$_{exon}$ clustering (0.261) is much lower than that of the SDEAP clustering (0.391).

In our simulation experiments, we concluded that SDEAP is less sensitive to outliers in SC RNA-Seq data than DEXUS$_{exon}$ and is able to discover more true DTE genes that characterize the biological conditions in a population. The above clustering results on real SC RNA-Seq data support these claims.

3.4 Conclusion

We have introduced SDEAP, an algorithm to identify DTE genes for a population of RNA-Seq samples with unknown conditions based on the splice graph data structure. SDEAP takes advantages of an accurate graph modular decomposition algorithm for discovering ASMs, efficient feature extraction for reducing the impact of technical noise, and a robust Dirichlet mixture model for inferring the groups in a population without assuming the number of biological conditions. These features make SDEAP more suitable for many practical applications. As shown in our simulation and real data experiments, the DTE features identified by SDEAP suffice to separate the subtypes of cancer, detect cell types and classify cell-cycle phases. We expect that SDEAP will serve as a useful differential expression/splicing analysis tool for RNA-Seq data in population studies with unknown biological conditions.

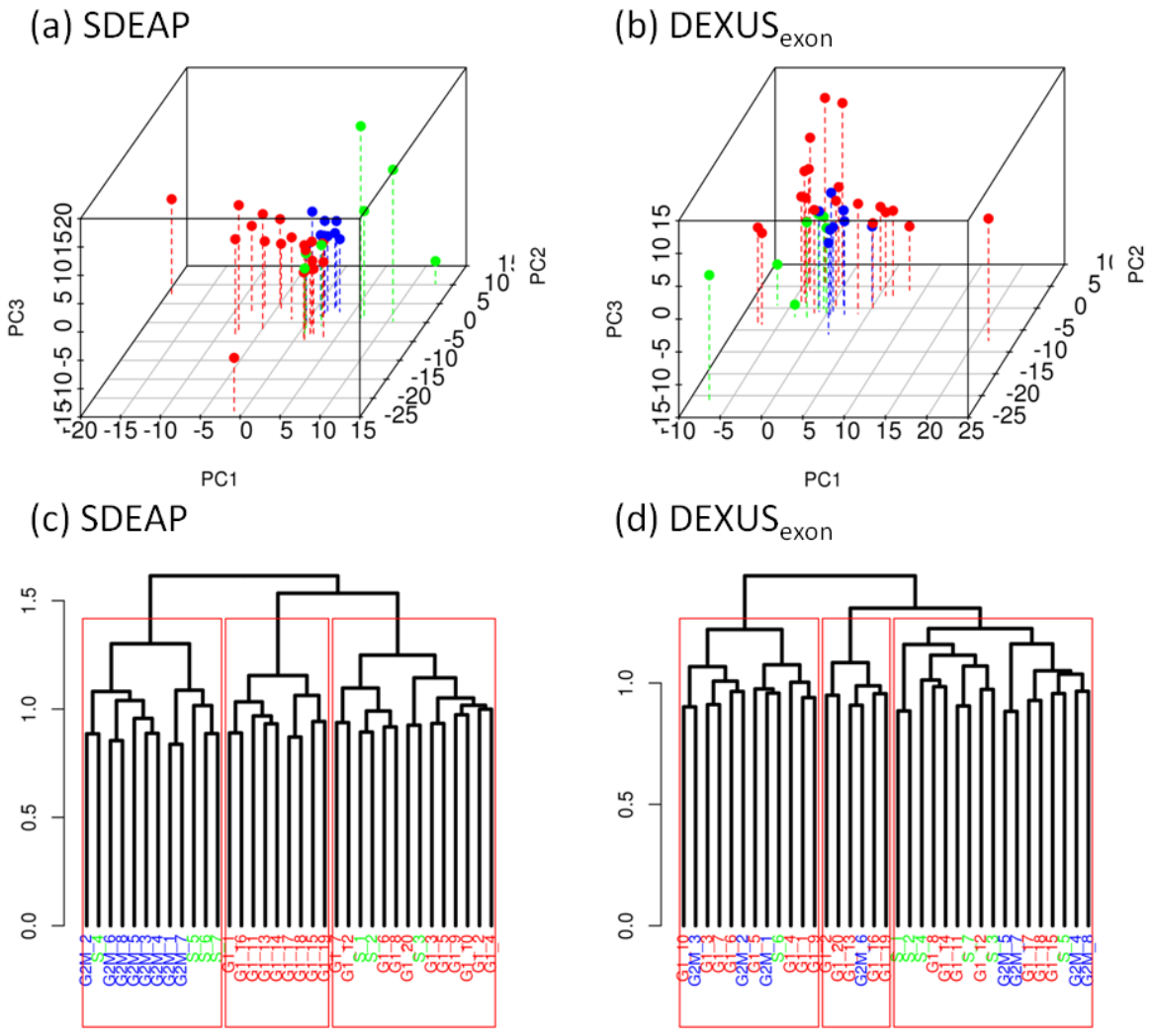

Figure 3.4: The PCA transformation of expression features and the hierarchical clstering of the mESC cells using the DTE features identified by SDEAP and DEXUS$_{exon}$. Plots (a) and (b) are the projections of predicted DTE features by SDEAP and DEXUS$_{exon}$. Every red dot is a cell in the G1 cell-cycle phase and every blue dot a cell in the G2/M phase. Cells in the S phase are represented by green dots. Plots (c) and (d) depict the dendrograms made from the DTE features predicted by SDEAP and DEXUS$_{exon}$. The Y-axis is the measurement of similarity between samples and the X-axis shows the labels of the mECS cells in the three cell-cycle phases. The labels are colored in the same away as in plots (a) and (b).

Chapter 4

GDNorm: An Improved Poisson Regression Model for Reducing Biases in Hi-C Data

4.1 Introduction

Three dimensional (3D) conformation of chromosomes in nuclei plays an important role in many chromosomal mechanisms such as gene regulation, DNA replication, maintenance of genome stability, and epigenetic modification [26]. Alterations of chromatin 3D conformations are also found to be related to many diseases including cancers [49]. Because of its importance, the spatial organization of chromosomes has been studied for decades using methods of varying scale and resolution. However, owing to the high complexity of chromosomal structures, understanding the spatial organization of chromosomes and its relation to transcriptional regulation is still coarse and fragmented [74].

An important approach for studying the spatial organization of chromosomes is

florescent in situ hybridization (FISH) [33]. In FISH-based methods, florescent probes are hybridized to genomic regions of interests and then the inter-probe distance values on two dimensional fluorescence microscope images are used as the measurement for spatial proximity of the genomic regions. Because FISH-based methods rely on image analysis involving a few hundred cells under the microscope, they are generally considered to be of low throughput and resolution [49]. Recently, the limitation of throughput and resolution was alleviated by the introduction of the 3C technology that is able to capture the chromatin interaction of two given genomic regions in a population of cells by using PCR [25]. Combining this with microarray and next generation sequencing technologies has yielded more powerful variants of the 3C methods. For example, 4C methods [108, 142] can simultaneously capture all possible interacting regions of a given genomic locus in 3D space while 5C methods can further identify complete pairwise interactions between two sets of genomic loci in a large genomic region of interests [30]. However, when it comes to genome-wide studies of chromatin interactions, 5C methods require a very large number of oligonucleotides to evaluate chromatin interactions for an entire genome. The cost of oligonucleotide synthesis makes the 5C methods unsuitable for genome-wide studies [49]. To overcome this issue, another NGS-based variant of the 3C technology, called Hi-C, was proposed to quantify the spatial proximity of the conformations of all the chromosomes [74]. By taking advantages of the NGS technology, Hi-C can quantify the spatial proximity between all pairs of chromosomal regions at an unprecedentedly high resolution. As a

revolutionary tool, the introduction of Hi-C facilitates many downstream applications of chromosome spatial organization studies such as the discovery of the consensus conformation in mammalian genomes [28], the estimation of conformational variations of chromosomes within a cell population [49], and the discovery of a deeper relationship between genome spatial structures and functions [79].

The Hi-C technology involves the generation of DNA fragments spanning genomic regions that are close to each other in 3D space in a series of experimental steps, such as formaldehyde cross-linking in solution, restriction enzyme digestion, biotinylated junctions pull-down, and high throughput paired-end sequencing [74]. The number of DNA fragments spanning two regions is called the *contact frequency* of the two regions. The physical (spatial) distance between a pair of genomic regions is generally assumed to be inversely proportional to the contact frequency of the two regions and hence the chromosome structure can in principle be recovered from the contact frequencies between genomic regions [49, 141]. However, during the experimental steps of Hi-C, systematic biases from different sources are often introduced into contact frequencies. Several systematic biases were shown to be related to genomic features such as number of restriction enzyme cutting sites, GC content and sequence uniqueness in the work of Yaffe and Tanay [135]. Without being carefully detected and eliminated, these systematic biases may distort many down-stream analyses of chromosome spatial organization studies. To remove such systematic biases, several bias reduction methods have been proposed recently. These bias reduction methods can be divided

into two categories, the normalization methods and bias correction methods according to [49]. The normalization methods, such as ICE [52] and the method in [21], aims at reducing the joint effect of systematic biases without making any specific assumption on the relationships between systematic biases and related genomic features. Their applications are limited to the study of equal sized genomic loci [49]. In contrast, the bias correction methods, such as HiCNorm [48] and the method of Yaffe and Tanay (YT) [135], build explicit computational models to capture the relationships between systematic biases and related genomic features that can be used to eliminate the joint effect of the biases.

Although it is well known that observed contact frequencies are determined by both systematics biases and spatial distance between genomic segments, the existing bias correction methods do not take spatial distance into account explicitly. This incomplete characterization of causal relationships for contact frequencies is known to cause problems such as poor goodness-of-fitting to the observed contact frequency data [48]. In this paper, we build on the work in [48] and propose an improved Poisson regression model that corrects systematic biases while taking spatial distance (between genomic regions) into consideration. We also present an efficient algorithm for solving the model based on gradient descent. This new bias correction method, called GDNorm, provides more accurate normalized contact frequencies and can be combined with a distance-based chromosome structure determination method such as ChromSDE [141] to obtain more accurate spatial structures of chromosomes,

as demonstrated in our simulation study. Moreover, two recently published Hi-C datasets from human lymphoblastoid and mouse embryonic stem cell lines are used to compare the performance of GDNorm with the other state-of-the-art bias reduction methods including HiCNorm, YT and ICE at 40kb and 1M resolutions. Our experiments on the real data show that GDNorm outperforms the existing bias reduction methods in terms of the reproducibility of normalized contact frequencies between biological replicates. The normalized contact frequencies by GDNorm are also found to be highly correlated to the corresponding FISH distance values in the literature. With regard to time efficiency, GDNorm achieves the shortest running time on the two real datasets and the running time of GDNorm increases linearly with the resolution of data. Since more and more high resolution (*e.g.*, 5 to 10kb) data are being used in the studies of chromosome structures [54], the time efficiency of GDNorm makes it a valuable bias reduction tool, especially for studies involving high resolution data.

4.2 Methods

4.2.1 Genomic Features

A chromosome g can be binned into several disjoint and consecutive genomic segments. Given an ordering to concatenate the chromosomes, let $S = \{s_1, s_2, ..., s_n\}$ be a linked list representing all n genomic segments of interest such that the linear

order of the segments in S is consistent with the sequential order in the concatenation of the chromosomes. For each genomic segment s_i, the number of restriction enzyme cutting sites (RECSs) within s_i is represented as R_i. The GC content G_i of segment s_i is the average GC content within the 200 bps region upstream of each RECS in the segment. The sequence uniqueness U_i of segment s_i is the average sequence uniqueness of 500 bps region upstream or downstream of each RECS. To calculate the sequence uniqueness for a 500 bps region, we use a sliding window of 36bps to synthesize 55 reads of 35 bps by taking steps of 10bps from $5'$ to $3'$ as done in [48]. After using the BWA algorithm [71] to align the 55 reads back to the genome, the percentage of the reads that is still uniquely mapped in the 500 bps region is considered as the sequence uniqueness for the 500 bps region. These three major genomic features have been shown to be either positively or negatively correlated to contact frequencies in the literature [135]. In the following, we will present a new bias correction method based on gradient search to eliminate the joint effect of the systematic biases correlated to the three genomic features, building on the Poisson regression model introduced in [48].

4.2.2 A Bias Correction Method Based on Gradient Descent

Let $F = \{f_{i,j} | 1 \leq i \leq n, 1 \leq j \leq n\}$ be the contact frequency matrix for the genomic segments in S such that each $f_{i,j}$ denotes the observed contact frequency between two segments s_i and s_j. HiCNorm [48] assumes that the observed contact

frequency $f_{i,j}$ follows a Poisson distribution with rate determined by the joint effect of systematic biases and represents the joint effect as a log-linear model of the three genomic features mentioned above (*i.e.*, the number of RECSs, GC content and sequence uniqueness). In other words, if the Poisson distribution rate of $f_{i,j}$ is $\theta_{i,j}$, then

$$\log(\theta_{i,j}) = \beta_0 + \beta_{recs}\log(R_iR_j) + \beta_{gcc}\log(G_iG_j) + \beta_{seq}\log(U_iU_j), \tag{4.1}$$

where β_0 is a global constant, β_{recs}, β_{gcc} and β_{seq} are coefficients for the systematic biases correlated to RECS, GC content and sequence uniqueness, and R_i, G_i and U_i are the number of RECSs, GC content and sequence uniqueness in segment s_i, respectively. The coefficient β_{seq} was fixed at 1 in [48] so the term $\log(U_iU_j)$ acts as the Poisson regression offset when estimating $\theta_{i,j}$.

However, this log-linear model does not capture all known causal relationships that affect the observed contact frequency $f_{i,j}$, because the spatial distance $d_{i,j}$ is not included in the model. To characterize more comprehensive causal relationships for observed contact frequencies, in a recently published chromosome structure determination method BACH [49], the spatial distance was modeled explicitly such that

$$\log(\theta_{i,j}) = \beta_0 + \beta_{dist}\log(d_{i,j}) + \beta_{recs}\log(R_iR_j) + \beta_{gcc}\log(G_iG_j) + \beta_{seq}\log(U_iU_j), \tag{4.2}$$

where $\beta = \{\beta_{recs}, \beta_{gcc}, \beta_{seq}\}$ again represents the systematic biases, β_{dist} represents

the *conversion factor* and $D = \{d_{i,j} | 1 \leq i \leq n, i < j\}$ are variables representing the spatial distance values to be estimated. However, without any constraint or assumption on spatial distance, the model represented by Eq. 4.2 is non-identifiable, because for any constant k, $\beta_{dist}\log(d_{i,j}) = k \times \beta_{dist}\log({d_{i,j}}^{1/k})$. BACH solved this issue by introducing some spatial constraints from previously predicted chromosome structures. (Eq. 4.2 was used by BACH to iteratively refine the predicted chromosome structure.) Hence, Eq. 4.2 is infeasible for bias correction methods that do not rely on any spatial constraint. To get around this, we introduce a new variable $z_{i,j} = \beta_0 + \beta_{dist}\log(d_{i,j})$ and rewrite Eq. 4.2 as follows:

$$\log(\theta_{i,j}) = z_{i,j} + \beta_{recs}\log(R_i R_j) + \beta_{gcc}\log(G_i G_j) + \beta_{seq}\log(U_i U_j), \tag{4.3}$$

where the systematic biases β and $Z = \{z_{i,j} | 1 \leq i \leq n, i < j\}$ are the variables to be estimated. Note that applying a Poisson distribution on read count data sometimes leads to the overdispersion problem, *i.e.*, underestimation of the variance [75], which is generally solved by using a negative binomial distribution instead. However, the results in [48] suggest that there is usually no significant difference in the performance of bias correction methods when a negative binomial distribution or a Poisson distribution is applied to Hi-C data. For the mathematical simplicity of our model, we use Poisson distributions.

Let θ denote the set of $\theta_{i,j}, 1 \leq i \leq n, 1 \leq j \leq n$. Given the observed contact

frequency matrix F and genomic features of S, the log-likelihood function of the observed contact frequencies over the Poisson distribution rates can be written as:

$$\begin{aligned}\log(Pr(F|\beta,Z)) &= \log(Pr(F|\theta)) = \log(\prod_{i=1,i<j}^{n} Pr(f_{i,j}|\theta_{i,j})) = \log(\prod_{i=1,i<j}^{n} \frac{e^{-\theta_{i,j}}\theta^{f_{i,j}}}{f_{i,j}!}) \\ &= \sum_{i=1,i<j}^{n} -\theta_{i,j} + f_{i,j}\log(\theta_{i,j}) - \log(f_{i,j}!). \end{aligned} \tag{4.4}$$

We can estimate the variables Z and systematic biases β by finding parameters $x^* = \{\beta^*, Z^*\}$ to maximize the log-likelihood function in Eq. (4.4), which is equivalent to solving the following multivariate optimization problem:

$$\begin{aligned} x^* &= \arg\min_x -\log(Pr(F|\beta,Z)) = \arg\min_x -\log(Pr(F|\theta)) \\ &= \arg\min_x \sum_{i=1,i<j}^{n} \theta_{i,j} - f_{i,j}\log(\theta_{i,j}) \end{aligned} \tag{4.5}$$

However, without any constraint on the variables Z, the above model is still generally non-identifiable since for any β, we can always choose a $z_{i,j}$ such that $f_{i,j} = \theta_{i,j}$ and the likelihood function is maximized. Therefore, we require that for any i, j, $|z_{i,i+1} - z_{j,j+1}| \leq \epsilon$ for some threshold ϵ, since we expect that the distance between neighboring segments is roughly the same across a chromosome.

Observe that Eq. 4.5 cannot be solved by using the same Poisson regression fitting method as in HiCNorm, because Eq. 4.5 is no longer a standard log-linear model like Eq. 4.1. A popular technique for solving multivariate optimization problems is

gradient descent. Gradient descent searches the optimum of a minimization problem with an objective function $g(x)$ from a given initial point x_1 at the first iteration and then iteratively moves toward a local minimum by following the negative of the gradient function $-\nabla g(x)$. In other words, at every iteration i, we compute $x_i \leftarrow x_{i-1} - \alpha \nabla g(x)$, where α is a constant. In our case, the objective function to be minimized is the negative of the above log-likelihood function $g(x) = g(\beta, Z) = -\log(Pr(F|\beta, Z))$. By taking partial derivatives of the objective function with respect to the variables β and Z, we have the gradient function $-\nabla g(x) = \{\frac{\partial g(x)}{\partial \beta}, \frac{\partial g(x)}{\partial Z}\}$ as

$$\begin{aligned}
\frac{\partial g(\beta, Z)}{\partial z_{i,j}} &= \theta_{i,j} - f_{i,j} \\
\frac{\partial g(\beta, Z)}{\partial \beta_{recs}} &= \sum_{i=1, i<j}^{n} \log(R_i R_j)(\theta_{i,j} - f_{i,j}) \\
\frac{\partial g(\beta, D)}{\partial \beta_{gcc}} &= \sum_{i=1, i<j}^{n} \log(G_i G_j)(\theta_{i,j} - f_{i,j}) \\
\frac{\partial g(\beta, D)}{\partial \beta_{seq}} &= \sum_{i=1, i<j}^{n} \log(U_i U_j)(\theta_{i,j} - f_{i,j})
\end{aligned}$$

To initialize $x_1 = \{\beta^1, Z^1\}$, we first set the variable $z_{i,i+1}$ as a uniform constant z for every two neighboring segments, s_i and s_{i+1}, because we assume that the distance between every pair of neighboring segments is similar. The systematic biases are then initialized as β^1 by solving Eq. 4.1, with $z = \beta_0$, on neighboring segments only. To obtain initial variables $z_{i,j}$, where $j - i > 1$, $\theta_{i,j}$ is sampled from the conjugate prior of Poisson distribution $\Gamma(1, f_{i,j} + 1)$ and then $z_{i,j}$ is cal-

culated by using Eq. 4.3 with the fixed parameter β^1. After the convergence of the gradient descent search, the normalized contact frequency $\hat{f}_{i,j}$ is computed by $\hat{f}_{i,j} = f_{i,j}/\{(R_iR_j)^{\beta_{recs}}(G_iG_j)^{\beta_{gcc}}(U_iU_j)^{\beta_{seq}}\}$. Our complete algorithm for GDNorm is summarized in Algorithm 1. Here, N_{max} denotes the maximum number of iterations allowed and its default is set to be 10 based on our empirical observation that the gradient descent search usually converges in no more than 10 iterations.

We assess the performance of GDNorm in terms of (i) the accuracy of its normalized contact frequencies and (ii) the accuracy of structure determination using the normalized contact frequencies. The latter will be done by simulating biased Hi-C read count data from some simple reference chromosome structures and then trying to recover the reference structures from normalized contact frequencies in combination with the most recent chromosome structure determination algorithm, ChromSDE [141]. In other words, we will consider the impact of normalized contact frequencies on the chromosome structures predicted by ChromSDE. To measure the quality of bias correction, we consider the reproducibility of normalized contact frequencies between biological replicates of an mESC line [28] and the correlation between normalized contact frequencies and FISH distance values in the literature. The performance of GDNorm will be compared with the state-of-the-art bias reduction algorithms HiC-Norm [48], YT [135] and ICE [52].

Algorithm 1 *Bias Reduction Based on Gradient Descent*

procedure BIAS REDUCTION

Input: *Contact frequency matrix F and genomic features R, G and U*

Spatial Distance and Systematic Bias Estimation:

Initialize $x_1 = \{\beta^1, Z^1\}$;

for *i from 2 to* N_{max} ***do***

$x_i \leftarrow x_{i-1} - \alpha \nabla g(x)$;

if $g(x_i) > g(x_{i-1})$ ***then***

Go to Contact Frequency Normalization;

end if

end for

Contact Frequency Normalization:

for $i < j$ ***do***

$\hat{f}_{i,j} = f_{i,j} / \{(R_i R_j)^{\beta_{recs}} (G_i G_j)^{\beta_{gcc}} (U_i U_j)^{\beta_{seq}}\}$

end for

return $\hat{F} = \{\hat{f}_{i,j} | 1 < i, j < n\}$

end procedure

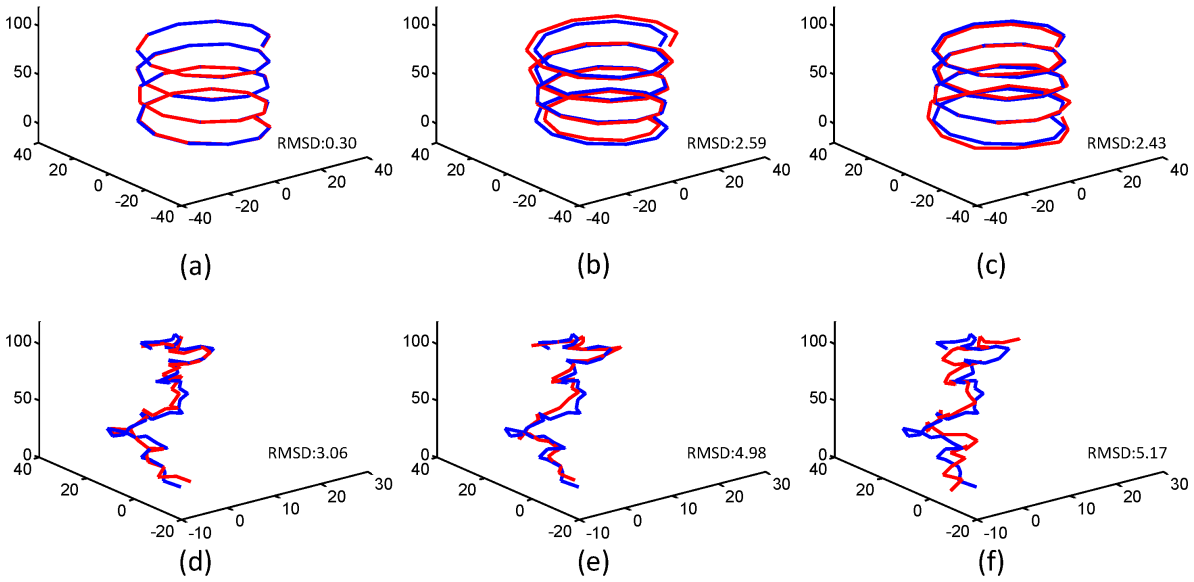

Figure 4.1: Alignment between the reference chromosome 3D structures and structures predicted by GDNorm$_{sde}$, HiCNorm$_{sde}$ and BACH on simulated data. The red curves indicate the predicted structures and blue curves the reference structures. The results of GDNorm$_{sde}$, HiCNorm$_{sde}$ and BACH are shown from left to right. The top row is for the helix and bottom for the random walk. The quality of each structural alignment is evaluated by an RMSD value.

4.3 Experimental Results

4.3.1 Simulation Studies

To evaluate the accuracy of chromosome structure prediction, two reference 3D structures, a helix and an arbitrary random walk, are constructed as shown in Fig. 4.1. In order to be close to the real chromosome structure prediction practice, each of the reference 3D structures consists of 44 segments, where the number 44 was determined by the average size of the chromosomal structure units studied in [49] (*i.e.*,

conserved domains). Let B_i denote the systematic bias in a segment s_i and $T_{i,j}$ the true (unbiased) contact frequency between segments s_i and s_j. To synthesize observed contact frequencies $f_{i,j}$, we follow the assumption $f_{i,j} = T_{i,j}B_iB_j$ as in [52]. Here, $T_{i,j}$ is assumed to be inversely proportional to the spatial distance $d_{i,j}$. That is, $T_{i,j} = d_{i,j}^{\rho}$, where $\rho < 0$ is called the conversion factor between the unbiased contact frequency and its corresponding spatial distance. The value of B_iB_j is estimated by using the log-linear function $\log(B_iB_j) = \beta_0 + \beta_{recs}\log(R_iR_j) + \beta_{gcc}\log(G_iG_j) + \beta_{seq}\log(U_iU_j)$ introduced in [48]. The coefficient β_{seq} is set to 1 as in [48] while ρ is set to -1.2 as estimated from a mouse cell line by ChromSDE [141]. To determine the coefficients β_0, β_{recs} and β_{gcc}, HiCNorm is run on the mm9 mESC data to form a pool of coefficients. A set of coefficients β_0, β_{recs} and β_{gcc} are then randomly drawn from the pool and used throughout the simulation study.

Because currently there is no tool to synthesize Hi-C reads reasonably from a given 3D structure and the methods YT and ICE require actual Hi-C reads as input, they are excluded from this simulation study but will be discussed in the real data experiments in the section 3.2. The method GDNorm and HiCNorm are run on the simulated contact frequencies and their normalized contact frequencies are then used to predict chromosome 3D structures. Two structure prediction software, MCMC5C and ChromSDE, in the literature use normalized contact frequencies to predict chromosome 3D structures [104, 141]. Here, we choose ChromSDE, instead of MCMC5C, as the structure prediction method because MCMC5C is not specific to

Hi-C data and ChromSDE significantly outperformed MCMC5C in the most recent study [141]. The combination of HiCNorm and ChromSDE is denoted as HiCNorm$_{sde}$ while the combination of GDNorm and ChromSDE is called as GDNorm$_{sde}$ in the following discussion. To further study the performance of GDNorm$_{sde}$ and HiCNorm$_{sde}$ as chromosome structure prediction tools on biased Hi-C data, another independent prediction method, BACH [49], is also included in our comparisons. Note that BACH always normalizes the size of its predicted structure by fixing the distance between the first and the last segments to be 1 while ChromSDE does not perform this normalization. To obtain a fair comparison, we calibrate the predicted structure sizes in GDNorm$_{sde}$ and HiCNorm$_{sde}$ such that the distance between the first and last segment is fixed at 100. Finally, the accuracy of structure prediction is assessed using the root mean square difference (RMSD) measure after optimally aligning a predicted structure to the reference structure by Kabsch's algorithm [55].

GDNorm provides the most accurate chromosome structure prediction on noise-free data

The optimal alignments of the predicted and reference chromosome structures are shown together with their RMSD values in Figure 1. In the structure predictions for both the helix and random walk, GDNorm$_{sde}$ predicted the chromosome structures with the minimum RMSDs. In the structure prediction for the helix, GDNorm$_{sde}$ obtained a structure that can be almost perfectly aligned with the reference structure

with a very small RMSD value of 0.3. This is because GDNorm was able to significantly reduce the effect of systematic bias and the semi-definite programming method employed by ChromSDE can guarantee perfect recovery of a chromosome structure when the given distance values between segments are noise-free.

GDNorm reduces systematic biases significantly in noise-free data

To examine how much the effect of systematic biases can be reduced by the selected bias reduction methods, we further analyze the predicted spatial distance values between neighboring segments in the structure prediction for the helix. Because the spatial distance between neighboring segments s_i and s_{i+1} in the reference structure of the helix is the same for all i, the difference in the observed contact frequency between s_i and s_{i+1}, for different i, is mainly a result of the systematic biases. If the systematic biases are correctly estimated and eliminated, the distance between any two consecutive segments in the predicted structure is expected to be the same. The spatial distance values between 10 pairs of consecutive segments with the greatest systematic biases are compared with the distance values between 10 pairs with the smallest systematic biases for each of the chromosome structures predicted by GDNorm$_{sde}$, HiCNorm$_{sde}$ and BACH. The box plots in Figure 4.2 summarizes the comparison results. The absolute differences between the means of the two sets of 10 distance values obtained by GDNorm$_{sde}$, HiCNorm$_{sde}$ and BACH are 0.045, 3.47 and 2.61, respectively. The statistical significance of the difference between two sets of

Table 4.1: RMSD values of the predicted structures on noisy data.

Reference Structure	Noise Level	GDNorm$_{sde}$	HiCNorm$_{sde}$	BACH
Helix	30%	2.65	3.33	14.9
	50%	4.19	4.26	20.0
Random Walk	30%	4.26	6.40	5.26
	50%	5.17	7.11	6.43

10 distance values obtained by each method is also examined by a two-tailed t-Test [42], which yielded a non-significant p-value of 0.42 for GDNorm$_{sde}$ and significant p-values of 1.3×10^{-12} and 1.56×10^{-6} for HiCNorm$_{sde}$ and BACH, respectively.

GDNorm provides the most accurate chromosome prediction on noisy data

We have demonstrated the superior performance of GDNorm$_{sde}$ on Hi-C data without noise (but with systematic biases). To test its performance on noisy data, a uniformly random noise $\delta_{i,j}$ is injected into every contact frequency $f_{i,j}$ such that the noisy frequency $\tilde{f}^{i_j} = f_{i,j}(1+\delta_{i,j})$. In this test, we consider two noise levels, 30% and 50%. Table 4.1 summarizes the RMSD values of the optimal alignments between the predicted structures and the reference structures. The results show GDNorm$_{sde}$ still outperforms the other two methods by achieving the overall smallest RMSD values at both noise levels. Note that BACH failed to predict the helix structure at both noise levels in this test, perhaps because its MCMC algorithm could sometimes be trapped in a local optimum when the input data contains a significant level of noise.

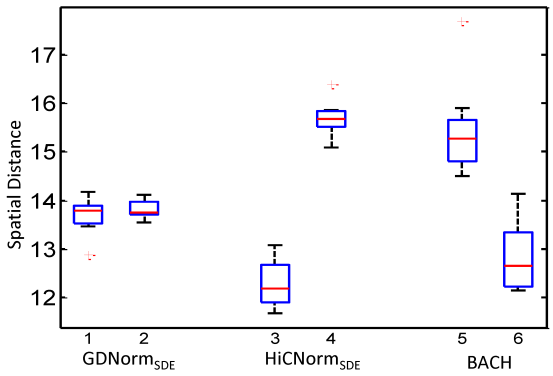

Figure 4.2: Comparison of the predicted spatial distance values with the 10 greatest and 10 smallest systematic biases. For each structure prediction method studied, two sets of 10 distance values form the two boxes in a comparison group. The left box depicts the distribution of the distance values for contacts with the greatest systematic biases while the right shows the distribution of the distance values for contacts with the smallest systematic biases. Clearly, GDNorm$_{sde}$ produced the most consistent distance values and HiCNorm$_{sde}$ the least.

4.3.2 Performance on Real Hi-C Data

In addition to the simulation study, several experiments on real Hi-C data are conducted to evaluate the bias reduction capability of GDNorm, in comparison with other state-of-the-art bias reduction methods, HiCNorm, YT and ICE. Unlike the assessment in the previous simulation study, the reference structures for real Hi-C datasets are hardly obtainable because of the complexity of chromosome structures. To compare the performance of the studied bias reduction methods on real Hi-C data, a commonly used evaluation criterion is the similarity (or reproducibility) between normalized contact frequency matrices from biological replicates using different enzymes. Since these replicates are derived from the same chromosomal structures in

the cell line, the contact frequencies normalized by a robust bias reduction algorithm using one enzyme are expected to be similar to those using another enzyme. However, a high reproducibility is a necessary but not sufficient condition for robust bias reduction algorithms. As suggested in [49], we further compare the correlation between normalized contact frequencies and the corresponding spatial distance values measured by FISH experiments. Both the similarity between the normalized contact frequency matrices and the correlation to FISH data will be measured in terms of Spearman's rank correlation coefficient that is independent to the conversion between normalized contact frequencies and spatial distance values.

To prepare benchmark datasets for the performance assessment, we use two recently published Hi-C data from human lymphoblastoid cells (GM06990) [74] and mouse stem cells (mESC) [28]. For the GM06990 dataset, the Hi-C raw reads, SRR027956 and SRR027960, of two biological replicates using restriction enzymes HindIII and NcoI, respectively, were downloaded from NCBI (GSE18199). Each of the chromosomes in the GM06990 cell line is binned into 1M bps segments and the pre-computed observed frequency matrices at 1M resolution were obtained from the publication website of [135]. For the mESC dataset, the mapped reads, uniquely aligned by the BWA algorithm [71], were downloaded from NCBI (GSE35156). Because of the enhanced sequencing depth in the mESC dataset, the Hi-C data can be analyzed at a higher resolution, *i.e.*, 40kb. In other words, the 20 chromosomes in the mESC cell line are binned into 40kb bps segments. To calculate observed contact

frequencies from the mapped reads, the preprocessing protocols used in the literature [74, 135] are followed. For every paired-end read, its total distance to the two closest RECSs is calculated. Any read with a total distance greater than 500 bps is defined as a non-specific ligation and thus removed to prevent reads from random ligation being used, as suggested in [135]. Reads from RECSs with low sequence uniqueness (smaller than 0.5) are also discarded. The remaining paired-end reads over the 20 chromosome, chr1 to chr20 (chrX), are used for calculating the observed contact frequencies.

The contact frequencies are derived from a cell population that may consist of several subpopulations of different chromosome structures. Without fully understanding the structural variations in a cell population, any structural inference from the Hi-C data can be distorted [49]. A recent single-cell sequencing study found that inter-chromosome (or trans) contacts have much higher variability among cells of the same cell line than intra-chromosome (or cis) contacts [85]. To avoid potential uncertainty that may be caused by significant variations in a cell line, we follow suggestions in the literature [28, 49] and focus on cis contacts within a chromosome.

To obtain normalized frequencies of the bias reduction methods, we run both GDNorm and HiCNorm on the contact frequencies and ICE on the raw Hi-C reads. The normalized frequencies by the YT method are downloaded from the publication websites of the literature [28, 135]. Note that although the primary objective of BACH is to predict chromosome structures, it also estimates systematic biases in the

prediction of chromosome structures, using the log-linear regression model given in Eq. 4.2.

Hence, BACH can be regarded as a bias reduction method if we divide each observed contact frequency by its estimated systematic biases and use the quotient as the normalized frequency. To study the accuracy of bias estimation by BACH, we also include BACH in the comparison of bias correction methods. The reproducibility between the two biological replicates and correlation to FISH data achieved by the compared methods are discussed below.

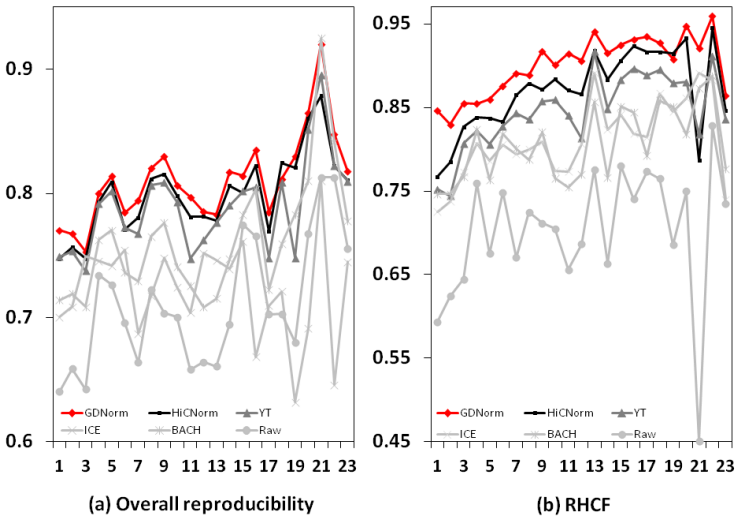

Figure 4.3: Comparison of the reproducibility between two biological replicates achieved by GDNorm, HiCNorm, YT, ICE, and BACH on the 23 chromosomes, chr1 to chr23 (chrX), in the GM06990 cell line at 1M resolution. The distribution of Spearman's correlation coefficients achieved by a bias reduction method is represented as a solid curve over the 23 chromosomes. Plot (a) illustrates the overall reproducibility and plot (b) shows the reproducibility of high contact frequencies (RHCF).

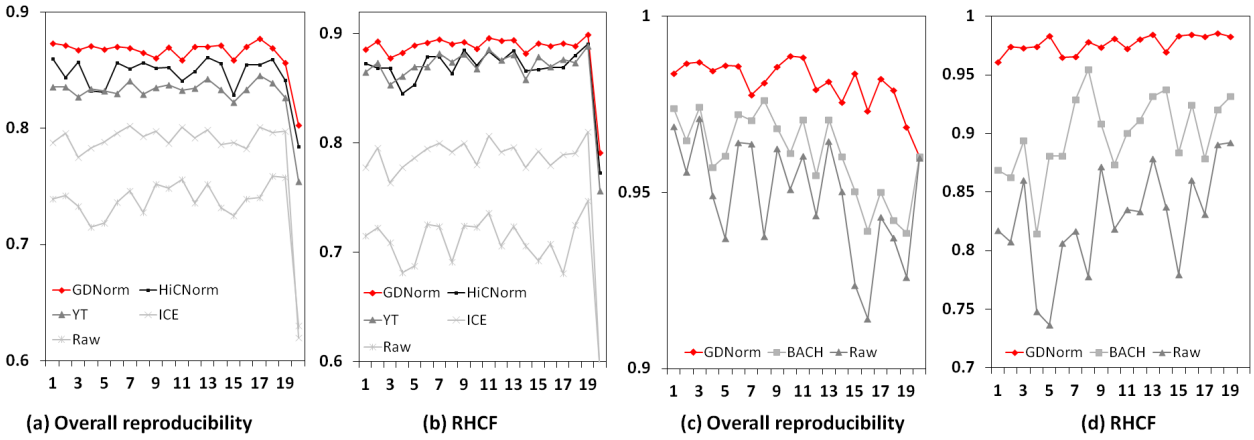

Figure 4.4: Comparison of the reproducibility in the mESC dataset. Plots (a) and (b) illustrate the overall reproducibility and RHCF of GDNorm, HiCNorm, YT, and ICE on the 20 chromosomes, chr1 to chr20 (chrX), in the mESC cell line at 40kb resolution, respectively. Here, the distribution of Spearman's correlation coefficients achieved by each bias reduction method is represented as a solid curve over the 20 chromosomes. Plots (c) and (d) show the overall reproducibility and RHCF of GDNorm and BACH at 1M resolution, respectively.

GDNorm achieves the best reproducibility on the two real datasets

The reproducibility between biological replicates is measured by Spearman's correlation coefficient. To prevent the assessment biased by background noise, when calculating Spearman's correlation coefficient, 2% of bins with lowest read counts in the matrices are deleted as done in [52]. The reproducibility over the remaining 98% of the bins is referred to as the overall reproducibility. Some recent studies in the literature using Hi-C data focused on high contact frequencies, *e.g.*, studies concerning gene promoter-enhancer contacts [54] and spatial gene-gene interaction networks [130]. To assess the capability of reducing systematic biases in high contact frequencies, we calculate another Spearman's correlation coefficient, called the repro-

ducibility of high contact frequencies (RHCF), by using only the top 20% of bins with the highest observed contact frequencies.

The Spearman's correlation coefficients over the 23 chromosomes in the GM06990 dataset are summarized in Figure 4.3. The average overall reproducibility of the observed (*i.e.*, raw) contact frequencies is 0.711 and GDNorm achieves the best overall reproducibility 0.811 on average while HiCNorm, YT, BACH, and ICE obtain 0.799, 0.789, 0,761, and 0.721, respectively. GDNorm improves the average overall reproducibility by up to 0.04 on an individual chromosome, over the second best method, HiCNorm. In terms of RHCF, the improvement by GDNorm over the second best method (HiCNorm) is more striking, 0.02 on average and up to 0.13 on an individual chromosome.

In the experiments on the mESC dataset, all the selected methods are run on the data at 40kb resolution except for BACH. The running time of BACH is prohibitive for performing chromosome-wide bias correction on the mESC dataset at the 40kb resolution, because it requires 5000 iterations to refine the predicted structure by default and each iteration takes about 30 minutes on average on our computer. So, we excluded BACH from the experiments at 40kb resolution, but will compare it with GDNorm at 1M resolution separately. The comparisons over the 20 chromosomes in the mESC dataset at 40kb resolution are summarized in Figure 4.4 (a) and (b). The average overall reproducibility of the observed (raw) contact frequencies is 0.734. The average overall reproducibility provided by GDNorm is 0.865, which is

about 0.02 higher than the average overall reproducibility (0.846) obtained by HiCNorm and 0.03 higher than the third best (0.83) obtained by YT. Although ICE can eliminate systematic biases without assuming their specific sources, it achieves the lowest average overall reproducibility, 0.783, which is significantly lower than the average reproducibilities obtained by the other three methods. GDNorm achieves similar improvements in terms of RHCF, which is also 0.02 higher than the second best by HiCNorm on average and up to 0.04 on an individual chromosome. The comparisons between BACH and GDNorm at 1M resolution are shown in Figure 4.4 (c) and (d). GDNorm significantly outperforms BACH on both average overall reproducibility (0.02) and average RHCF (0.07). In the tests on individual chromosomes, the maximum improvement on RHCF by GDNorm is up to 0.15. This result shows that, although GDNorm and BACH both include spatial distance explicitly in their models, the gradient descent method of GDNorm can estimate the systematic biases more accurately than the MCMC based optimization procedure of BACH. These experimental results demonstrate that GDNorm is able to consistently improve on the reproducibility between biological replicates at both high (40kb) and low (1M) resolutions.

The normalized contact frequencies obtained by GDNorm are well correlated to the FISH data

To further validate the quality of normalized contact frequencies, we use an mESC 2d-FISH dataset that contains distance measurement for six pairs of genomic loci as our benchmark data. The six pairs of genomic loci are distributed on chromosomes 2 and 11 of the mESC genome, with three pairs on chromosome 2 and the other three on chromosome 11. The distance between each pair of the genomic loci is measured by inter-probe distance on 100 cell images from 2d-FISH experiments and normalized by the size of cell nucleus such that any change in the distance measurement is attributed solely to altered nucleus size on the images as described in the literature [33]. The average of the 100 normalized distance values for each pair of the genomic segments is used to correlate with the normalized contact frequency corresponding to the pair. The normalized frequencies are expected to be inversely correlated to the corresponding spatial distance values. Table 4.2 compares Spearman's correlation coefficients obtained by all four methods. The correlation coefficient between the 2d-FISH distance values and observed contact frequencies is low, -0.45 and -0.25 in the HindIII and NcoI replicates, respectively. YT and GDNorm are able to improve both correlation coefficients and achieve a strong correlation (smaller than -0.6) in the HindIII replicate while HiCNorm and ICE fail to deliver strongly correlated normalized frequencies in either replicate.

Table 4.2: Correlation between normalized contact frequencies at 40kb resolution and spatial distance measured by FISH experiments in the two biological replicates of the mESC data.

Replicates	Raw	GDNorm	HiCNorm	YT	ICE
HindIII	-0.49	-0.66	-0.60	-0.66	-0.25
NcoI	-0.25	-0.37	-0.14	-0.37	0.31

The time efficiency of GDNorm

We evaluate the time efficiency of the selected methods by comparing their running time on the two real datasets. Our computing platform is a high-end compute server with eight 2.6GHz CPUs and 256GB of memory, but a single thread is used for each method. Because the normalized frequencies of YT were downloaded from the publication website, we did not run YT (in fact, we were unable to make YT run on our server) and will exclude YT from the comparison. The running time of the other four methods is summarized in Table 4.3. Due to the intensive computation requirement of the MCMC algorithm for refining chromosome structures, BACH is more than 10 time slower than HiCNorm and GDNorm on the 1M dataset (*i.e.*, GM06990). As mentioned before, the running time of BACH increases drastically with the number of genomic segments and becomes prohibitive when BACH is applied to the 40kb dataset (*i.e.*, mESC). ICE is significantly slower HiCNorm and GDNorm because it starts from raw Hi-C reads (instead of read counts) and requires additional time for iteratively mapping and processing the raw reads. Note that YT also uses raw Hi-C reads as its input and was found to be more than 1000 times slower than HiCNorm on the 1M dataset [48]. On both real datasets, GDNorm runs faster than

Table 4.3: The running time on the GM06990 and mESC datasets.

Datasets	GDNorm	HiCNorm	BACH	ICE
GM06990	0.8 s	2.0 s	2 hr 17 m	5 hr 45 m
mESC	37 s	15 m 58 s	-	8 hr 36 m

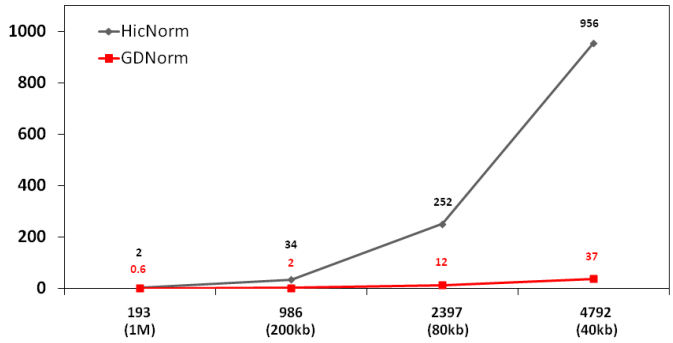

Figure 4.5: The running time of GDNorm and HiCNorm on the mESC data at four different resolutions. The Y-axis shows the running time in seconds and the X-axis indicates the number of genomic segments at each resolution.

HiCNorm. The standard iteratively reweighted least squares (IRIS) algorithm [29] was implemented in the software of HiCNorm to solve its log-linear regression model. In every iteration, the running time of the IRIS algorithm is quadratic in the number of genomic segment pairs. However, in our gradient descent method, the execution time of each iteration is only linear in the number of segment pairs, which makes GDNorm faster than HiCNorm. As illustrated in Figure 4.5, a simple experiment on the mESC data with resolutions at 40kb, 80kb, 200kb, and 1M shows that, when the number of genomic segments increases, the running time of HiCNorm grows much faster than that of GDNorm.

4.4 Conclusion

The reduction of systematic biases in Hi-C data is a challenging computational biology problem. In this paper, we proposed an accurate bias reduction method that takes advantage of a more comprehensive model of causal relationships among observed contact frequency, systematic biases and spatial distance. In our simulation study, GDNorm was able to provide more accurate normalized contact frequencies that resulted in improved chromosome structure prediction. Our experiments on two real Hi-C datasets demonstrated that GDNorm achieved a better reproducibility between biological replicates consistently at both high and low resolutions than the other state-of-the-art bias reduction methods and provided stronger correlation to published 2d-FISH data. The experiments also showed GDNorm's high time efficiency. With the rapid accumulation of high throughput genome-wide chromatin interaction data, the method could become a valuable tool for understanding the higher order architecture of chromosome structures.

Bibliography

[1] T. Äijö et al. Methods for time series analysis of rna-seq data with application to human th17 cell differentiation. *Bioinformatics*, 30(12):i113–20, 2014.

[2] D. Altshuler et al. A map of human genome variation from population-scale sequencing. *Nature*, 467(7319):1061–1073, 2010.

[3] S. Anders et al. Detecting differential usage of exons from rna-seq data. *Genome Research*, 22:2008–2017, 2012.

[4] S. Anders and W. Huber. Differential expression analysis for sequence count data. *Genome Biology*, 11(10):R106, 2010.

[5] X. Argout et al. The genome of theobroma cacao. *Nature Genetics*, 43(2):101–108, 2011.

[6] Monya Baker. Next-generation sequencing: adjusting to data overload. *Nature Methods*, 7:495 – 499, 2010.

[7] D. Beard and T. Schlick. Computational modeling predicts the structure and dynamics of chromatin fiber. *Structure*, 9(01):105–114, 2001.

[8] Y. Benjamini and Y. Hochberg. Controlling the false discovery rate: a practical and powerful approach to multiple testing. *Journal of the Royal Statistical Society*, B 57:289–300, 1995.

[9] J. Besag. Spatial interaction and the statistical analysis of lattice systems. *Journal of the Royal Statistical Society*, B 36(2):192–236, 1974.

[10] J. Besag. On the statistical analysis of dirty pictures. *Journal of the Royal Statistical Society*, B 48:259–302, 1986.

[11] Y. Boykov. Fast approximate energy minimization via graph cuts. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 23(11):1222–1239, 2001.

[12] Y. Boykov et al. Markov random fields with efficient approximations. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, page 648, 1998.

[13] P. Brennecke et al. Accounting for technical noise in single-cell rna-seq experiments. *Nature Methods*, 10(11):1093–1095, 2013.

[14] F. Buettner et al. Computational analysis of cell-to-cell heterogeneity in single-cell RNA-sequencing data reveals hidden subpopulations of cells. *Nature Biotechnology*, 33(2), 2015.

[15] J. H. Bullard et al. Evaluation of statistical methods for normalization and differential expression in mrna-seq experiments. *BMC Bioinformatics*, 11:94, 2010.

[16] R. H. Byrd et al. A limited memory algorithm for bound constrained optimization. *SIAM Journal on Scientific Computing*, 16(5):1190–1208, September 1995.

[17] M. R. Carlson et al. Gene connectivity, function, and sequence conservation: predictions from modular yeast co-expression networks. *BMC Genomics*, 7:40, 2006.

[18] P. Carninci et al. The transcriptional landscape of the mammalian genome. *Science*, 309(5740):1559–63, 2005.

[19] N. Cloonan et al. Stem cell transcriptome profiling via massive-scale mRNA sequencing. *Nature Methods*, 5(7):613–619, 2008.

[20] T. H. Cormen et al. *Introduction to Algorithms*. McGraw-Hill Higher Education, 2nd edition, 2001.

[21] A. Cournac et al. Normalization of a chromosomal contact map. *BMC Genomics*, 13:436, 2012.

[22] A. Culhane et al. Made4: an r package for multivariate analysis of gene expression data. *Bioinformatics*, 21(11):2789–2790, 2005.

[23] M. Danan-Gotthold et al. Identification of recurrent regulated alternative splicing events across human solid tumors. *Nucleic Acids Research*, 43(10):5130–5144, 2015.

[24] Carolyn de Graaf et al. Chromatin organization: form to function. *Current Opinion in Genetics & Development*, 23(2):185–90, 2013.

[25] J. Dekker et al. Capturing chromosome conformation. *Science*, 295(5558):1306–11, 2002.

[26] J. Dekker et al. Exploring the three-dimensional organization of genomes: interpreting chromatin interaction data. *Nature Reviews. Genetics*, 14(6):390–403, 2013.

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

[28] J. R. Dixon et al. Topological domains in mammalian genomes identified by analysis of chromatin interactions. *Nature*, 485(7398):376–380, 2012.

[29] A. J. Dobson. *An Introduction to Generalized Linear Models*. Chapman and Hall, London, 1990.

[30] J. Dostie et al. Chromosome Conformation Capture Carbon Copy (5C): a massively parallel solution for mapping interactions between genomic elements. *Genome Research*, 16(10):1299–309, 2006.

[31] Z. Duan et al. A three-dimensional model of the yeast genome. *Nature*, 465(7296):363–7, 2010.

[32] J. Edmonds and R. M Karp. Theoretical improvements in algorithmic efficiency for network flow problems. *Journal of the ACM*, 19(2):248–264, 1972.

[33] R. Eskeland et al. Ring1B compacts chromatin structure and represses gene expression independent of histone ubiquitination. *Molecular Cell*, 38(3):452–64, 2010.

[34] J. Eswaran et al. Transcriptomic landscape of breast cancers through mRNA sequencing. *Scientific Reports*, 2:264, 2012.

[35] J. Eswaran et al. RNA sequencing of cancer reveals novel splicing alterations. *Scientific Reports*, 3:1689, January 2013.

[36] R. A Fisher. On the interpretation of χ^2 from contingency tables, and the calculation of p. *Journal of the Royal Statistical Society*, 85(1):87–94, 1922.

[37] Ronald Aylmer Fisher. *Statistical methods for research workers; 13th ed.* Oliver and Boyd, Edinburgh, 1958.

[38] P. Flicek et al. Ensembl 2011. *Nucleic Acids Res*, 39(Database issue):D800–6, 2011.

[39] A. C. Frazee et al. Recount: a multi-experiment resource of analysis-ready rna-seq gene count datasets. *BMC Bioinformatics*, 12:449, 2011.

[40] M. Gierlinski et al. Statistical models for RNA-seq data derived from a two-condition 48-replicate experiment. *Bioinformatics*, pages 1–6, 2015.

[41] P. Glaus et al. Identifying differentially expressed transcripts from rna-seq data with biological variation. *Bioinformatics*, 28:1721–1728, 2012.

[42] C. H. Goulden. *Methods of Statistical Analysis, 2nd ed.* New York, Wiley, 1956.

[43] J. Graul et al. Prroc: computing and visualizing precision-recall and receiver operating characteristic curves in r. *Bioinformatics*, 31(15):2595–2597, 2015.

[44] B. R. Graveley et al. The developmental transcriptome of drosophila melanogaster. *Nature*, 471(7339):473–9, 2011.

[45] M. Griffith et al. Alternative expression analysis by rna sequencing. *Nat Methods*, 7(10):843–7, 2010.

[46] T. J. Hardcastle and K. A. Kelly. bayseq: empirical bayesian methods for identifying differential expression in sequence count data. *BMC Bioinformatics*, 11:422, 2010.

[47] G. Hon et al. ChromaSig: a probabilistic approach to finding common chromatin signatures in the human genome. *PLoS Computational Biology*, 4(10):e1000201, 2008.

[48] M. Hu et al. HiCNorm: removing biases in Hi-C data via Poisson regression. *Bioinformatics*, 28(23):3131–3, 2012.

[49] M. Hu et al. Bayesian inference of spatial organizations of chromosomes. *PLoS Computational Biology*, 9(1):e1002893, 2013.

[50] Y. Hu et al. Diffsplice: the genome-wide detection of differential splicing events with rna-seq. *Nucleic Acids Research*, 41(2):e39, 2013.

[51] Hübner et al. Chromatin organization and transcriptional regulation. *Current Opinion in Genetics Development*, 23(2):89–95, 2013.

[52] Imakaev et al. Iterative correction of Hi-C data reveals hallmarks of chromosome organization. *Nature Methods*, 2012.

[53] L. Jacob et al. More power via graph-structured tests for differential expression of gene networks. *The Annals of Applied Statistics*, 6(2):561–600, 2012.

[54] F. Jin et al. A high-resolution map of the three-dimensional chromatin interactome in human cells. *Nature*, 503(7475):290–4, 2013.

[55] W. Kabsch. A solution for the best rotation to relate two sets of vectors. *Acta Crystallographica Section A*, 32(5):922–923, 1976.

[56] M. Kanehisa and S. Goto. Kegg: kyoto encyclopedia of genes and genomes. *Nucleic Acids Research*, 28(1):27–30, 2000.

[57] Y. Katz et al. Analysis and design of rna sequencing experiments for identifying isoform regulation. *Nature Methods*, 7(12):1009–15, 2010.

[58] C. M. Kendziorski et al. On parametric empirical bayes methods for comparing multiple groups using replicated gene expression profiles. *Stat. Med.*, 22(24):3899–914, 2003.

[59] Kharchenko et al. Bayesian approach to single-cell differential expression analysis. *Nature Methods*, 11:740742, 2014.

[60] Khrameeva et al. Spatial proximity and similarity of the epigenetic state of genome domains. *PLOS ONE*, 7(4):e33947, 2012.

[61] P. K. Kimes et al. SigFuge: single gene clustering of RNA-seq reveals differential isoform usage among cancer samples. *Nucleic Acids Rresearch*, 42(14):e113, 2014.

[62] R. Kindermann and J. L. Snell. *Markov Random Fields and Their Applications*. American Mathematical Society, 1980.

[63] G. Klambauer et al. Dexus: identifying differential expression in rna-seq studies with unknown conditions. *Nucleic Acids Research*, 41(21):e198, 2013.

[64] V. Kolmogorov and R. Zabih. What energy functions can be minimized via graph cuts? *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(2):147–159, 2004.

[65] X. Lan et al. Integration of Hi-C and ChIP-seq data reveals distinct types of chromatin linkages. *Nucleic Acids Research*, 40(16):7690–704, 2012.

[66] B. D. Lehmann et al. Identification of human triple-negative breast cancer subtypes and preclinical models for selection of targeted therapies. *Journal of Clinical Investigation*, 121(7):2750, 2011.

[67] R. Leinonen et al. The sequence read archive. *Nucleic Acids Research*, 39(Database issue):D19–21, 2011.

[68] N. Leng et al. Ebseq: an empirical bayes hierarchical model for inference in rna-seq experiments. *Bioinformatics*, 29:1035–1043, 2013.

[69] B. Lewin. *genes VII*. OXFORD, 2000.

[70] B. Li and C. N. Dewey. Rsem: accurate transcript quantification from rna-seq data with or without a reference genome. *BMC Bioinformatics*, 12:323, 2011.

[71] H. Li and R. Durbin. Fast and accurate short read alignment with Burrows-Wheeler transform. *Bioinformatics (Oxford, England)*, 25(14):1754–60, 2009.

[72] N. Li et al. Whole genome dna methylation analysis based on high throughput sequencing technology. *Methods*, 52(3):203–12, 2010.

[73] W. Li and T. Jiang. Transcriptome assembly and isoform expression level estimation from biased rna-seq reads. *Bioinformatics*, 28(22):2914–2921, 2012.

[74] Lieberman-Aiden et al. Comprehensive mapping of long-range interactions reveals folding principles of the human genome. *Science*, 326(5950):289–93, 2009.

[75] J. K. Lindsey and P. M. E. Altham. Analysis of the human sex ratio by using overdispersion models. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 47(1):pp. 149–157, 1998.

[76] D. P. Locke et al. Comparative and demographic analysis of orang-utan genomes. *Nature*, 469(7331):529–533, 2011.

[77] I. W. Manfield et al. Arabidopsis co-expression tool (act): web server tools for microarray-based gene expression analysis. *Nucleic Acids Res*, 34(Web Server issue):W504–9, 2006.

[78] J. C. Marioni et al. Rna-seq: an assessment of technical reproducibility and comparison with gene expression arrays. *Genome Research*, 18(9):1509–17, 2008.

[79] M. A. Marti-Renom and L. A. Mirny. Bridging the resolution gap in structural modeling of 3D genome organization. *PLoS Computational Biology*, 7(7):e1002125, 2011.

[80] L. M. McIntyre et al. RNA-seq: technical variability and sampling. *BMC Genomics*, 12(1):293, January 2011.

[81] L. R. Meyer et al. The ucsc genome browser database: extensions and updates 2013. *Nucleic Acids Research*, 41(Database issue):D64–9, 2013.

[82] Kappen H. Mooij, J. Sufficient conditions for convergence of the sum-product algorithm. *IEEE Transactions on Information Theory*, 53(12), 2007.

[83] A. Mortazavi et al. Mapping and quantifying mammalian transcriptomes by RNA-Seq. 5(7):1–8, 2008.

[84] U. Nagalakshmi et al. The transcriptional landscape of the yeast genome defined by rna sequencing. *Science*, 320(5881):1344–9, 2008.

[85] T. Nagano et al. Single-cell Hi-C reveals cell-to-cell variability in chromosome structure. *Nature*, 502(7469):59–64, 2013.

[86] R. M. Neal. Markov Chain Sampling Methods for Dirichlet Process Mixture Models. *Journal of Computational and Graphical Statistics*, 9(2):249–265, 2007.

[87] M. A. Newton et al. On differential variability of expression ratios: improving statistical inference about gene expression changes from microarray data. *Journal of Computational Biology*, 8(1):37–52, 2001.

[88] T. Obayashi et al. Atted-ii: a database of co-expressed genes and cis elements for identifying co-regulated gene groups in arabidopsis. *Nucleic Acids Research*, 35(Database issue):D863–9, 2007.

[89] T. Obayashi and K. Kinoshita. Coxpresdb: a database to compare gene coexpression in seven model animals. *Nucleic Acids Research*, 39(Database issue):D1016–22, 2011.

[90] A. L. Oberg et al. Technical and biological variance structure in mrna-seq data: life in the real world. *BMC Genomics*, 13(1):304, 2012.

[91] Y. Ogata et al. Cop: a database for characterizing co-expressed gene modules with biological information in plants. *Bioinformatics*, 26(9):1267–8, 2010.

[92] A. Oshlack et al. From rna-seq reads to differential expression results. *Genome Biology*, 11(12):220, 2010.

[93] A. Oshlack and M. J. Wakefield. Transcript length bias in rna-seq data confounds systems biology. *Biol. Direct.*, 4:14, 2009.

[94] P. J. Park. Chip-seq: advantages and challenges of a maturing technology. *Nature Review in Genetics*, 10(10):669–80, 2009.

[95] A. Patil et al. Assessing the utility of gene co-expression stability in combination with correlation in the analysis of protein-protein interaction networks. *BMC Genomics*, 12 Suppl 3:S19, 2011.

[96] J. Paulsen et al. Handling realistic assumptions in hypothesis testing of 3D co-localization of genomic elements. *Nucleic Acids Research*, 41(10):5164–74, 2013.

[97] J. Pearl. Reverend bayes on inference engines: A distributed hierarchical approach. *the Second National Conference on Artificial Intelligence (AAAI)*, 1982.

[98] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[99] J. Rahnenfuhrer et al. Calculating the statistical significance of changes in pathway activity from gene expression data. *Stat. Appl. Genet. Mol. Biol.*, 3:Article16, 2004.

[100] F. Rapaport et al. Classification of microarray data using gene networks. *BMC Bioinformatics*, 8:35, 2007.

[101] M. D. Robinson et al. edger: a bioconductor package for differential expression analysis of digital gene expression data. *Bioinformatics*, 26(1):139–40, 2010.

[102] M. D. Robinson and G. K. Smyth. Moderated statistical tests for assessing differences in tag abundance. *Bioinformatics*, 23(21):2881–7, 2007.

[103] M. D. Robinson and G. K. Smyth. Small-sample estimation of negative binomial dispersion, with applications to sage data. *Biostatistics*, 9(2):321–32, 2008.

[104] M. Rousseau et al. Three-dimensional modeling of chromatin structure from interaction frequency data using Markov chain Monte Carlo sampling. *BMC Bioinformatics*, 12(1):414, 2011.

[105] Y. Sasagawa et al. Quartz-Seq: a highly reproducible and sensitive single-cell RNA sequencing method, reveals non-genetic gene-expression heterogeneity. *Genome Biology*, 14(4):R31, 2013.

[106] S. Shen et al. Mats: a bayesian framework for flexible detection of differential alternative splicing from rna-seq data. *Nucleic Acids Research*, 40(8):e61, 2012.

[107] L. Shi et al. The microarray quality control (maqc) project shows inter- and intraplatform reproducibility of gene expression measurements. *Nature Biotechnology*, 24(9):1151–61, 2006.

[108] Simonis et al. Nuclear organization of active and inactive chromatin domains uncovered by chromosome conformation capture-on-chip (4C). *Nature Genetics*, 38(11):1348–54, 2006.

[109] D. Singh et al. Fdm: a graph-based statistical method to detect differential transcription using rna-seq data. *Bioinformatics*, 27(19):2633–2640, 2011.

[110] A. Sivachenko et al. Identifying local gene expression patterns in biomolecular networks. *IEEE Computational Systems Bioinformatics Conference*, 2005.

[111] P. Sneath. Some thoughts on bacterial classification. *Journal of General Microbiology*, 18:184–200, 1957.

[112] S. Srivastava and L. Chen. A two-parameter generalized poisson model to improve the analysis of rna-seq data. *Nucleic Acids Research*, 38(17):e170, 2010.

[113] D. Steinhauser et al. Csb.db: a comprehensive systems-biology database. *Bioinformatics*, 20(18):3647–51, 2004.

[114] J.D. Storey. A direct approach to false discovery rates. *Journal of the Royal Statistical Society*, B 64:479–498, 2002.

[115] J. M. Stuart et al. A gene-coexpression network for global discovery of conserved genetic modules. *Science*, 302(5643):249–55, 2003.

[116] A. I. Su et al. Large-scale analysis of the human and mouse transcriptomes. *Proc. Natl. Acad. Sci. U S A*, 99(7):4465–70, 2002.

[117] A. Subramanian et al. Gene set enrichment analysis: a knowledge-based approach for interpreting genome-wide expression profiles. *Proc. Natl. Acad. Sci. U S A*, 102(43):15545–50, 2005.

[118] S. Tarazona et al. Differential expression in rna-seq: a matter of depth. *Genome Research*, 21(12):2213–23, 2011.

[119] P. Tong et al. Siber: systematic identification of bimodally expressed genes using rnaseq data. *Bioinformatics*, 29(5):605–613, March 2013.

[120] C. Trapnell. Defining cell types and states with single-cell genomics. *Genome Research*, pages 1491–1498, 2015.

[121] C. Trapnell et al. Tophat: discovering splice junctions with rna-seq. *Bioinformatics*, 25(9):1105–11, 2009.

[122] C. Trapnell et al. Transcript assembly and quantification by rna-seq reveals unannotated transcripts and isoform switching during cell differentiation. *Nature Biotechnology*, 28(5):511–515, 2010.

[123] C. Trapnell et al. Differential analysis of gene regulation at transcript resolution with rna-seq. *Nature Biotechnology*, 31:4653, 2013.

[124] C. J van Rijsbergen. *Information Retrieval (2nd ed.)*. Butterworth, 1979.

[125] B. van Steensel and J. Dekker. Genomics tools for unraveling chromosome architecture. *Nature Biotechnology*, 28(10):1089–1095, 2010.

[126] R. Velasco et al. The genome of the domesticated apple (malus x domestica borkh.). *Nature Genetics*, 42(10):833–+, 2010.

[127] J. P. Venables et al. Identification of alternative splicing markers for breast cancer. *Cancer Research*, 68(22):9525–9531, November 2008.

[128] L. Wang et al. Degseq: an r package for identifying differentially expressed genes from rna-seq data. *Bioinformatics*, 26(1):136–8, 2010.

[129] Z. Wang et al. Rna-seq: a revolutionary tool for transcriptomics. *Nature Review in Genetics*, 10(1):57–63, 2009.

[130] Z. Wang et al. The properties of genome conformation and spatial gene interaction and regulation networks of normal and malignant human cell types. *PLoS ONE*, 8(3):e58793, 2013.

[131] M. Watson. Coxpress: differential co-expression in gene expression data. *BMC Bioinformatics*, 7:509, 2006.

[132] Z. Wei et al. The Biological Implications and Regulatory Mechanisms of Long-Range Chromosomal Interactions. *The Journal of Biological Chemistry*, 288(31):22369–77, 2013.

[133] Z. Wei and H. Li. A markov random field model for network-based analysis of genomic data. *Bioinformatics*, 23(12):1537–44, 2007.

[134] Yair. Weiss. Correctness of local probability propagation in graphical models with loops. *Neural Computation*, 12(1):1–41, 2000.

[135] E. Yaffe and A. Tanay. Probabilistic modeling of Hi-C contact maps eliminates systematic biases to characterize global chromosomal architecture. *Nature Genetics*, 43(11):1059–65, 2011.

[136] E.W. Yang et al. Differential gene expression analysis using coexpression and rna-seq data. *Bioinformatics*, 29 (17):2153–2161, 2013.

[137] M. D. Young et al. Gene ontology analysis for rna-seq: accounting for selection bias. *Genome Biology*, 11(2):R14, 2010.

[138] J. Zeng et al. A markov random field framework for protein side-chain resonance assignment. *The Annual International Conference on Research in Computational Molecular (RECOMB)*, 2010.

[139] F. Zhang et al. Novel alternative splicing isoform biomarkers identification from high-throughput plasma proteomics profiling of breast cancer. *BMC Systems Biology*, 7 Suppl 5(Suppl 5):S8, 2013.

[140] J. Zhang et al. Biomart: a data federation framework for large collaborative projects. *Database*, 2011:bar038, 2011.

[141] Z. Zhang et al. Inference of spatial organizations of chromosomes using semi-definite embedding approach and hi-c data. In *RECOMB 2013*, pages 317–332, 2013.

[142] Z. Zhao et al. Circular chromosome conformation capture (4C) uncovers extensive networks of epigenetically regulated intra- and interchromosomal interactions. *Nature Genetics*, 38(11):1341–7, 2006.

[143] S. Zheng and L. Chen. A hierarchical bayesian model for comparing transcriptomes at the individual transcript isoform level. *Nucleic Acids Research*, 37(10):e75, 2009.

[144] X. Zhou et al. Robustly detecting differential expression in RNA sequencing data using observation weights. *Nucleic Acids Research*, 42(11):e91, June 2014.

[145] P. Zimmermann et al. Gene-expression analysis and network discovery using genevestigator. *Trends. Plant Sci.*, 10(9):407–9, 2005.