Probabilistic Models for Collecting, Analyzing, and Modeling Expression Data

Hai-Son Phuoc Le

May 2013
CMU-ML-13-101

Carnegie Mellon

Probabilistic Models for Collecting, Analyzing, and Modeling Expression Data

Hai-Son Phuoc Le

May 2013
CMU-ML-13-101

Machine Learning Department
School of Computer Science
Carnegie Mellon University

Thesis Committee
Ziv Bar-Joseph, Chair
Christopher Langmead
Roni Rosenfeld
Quaid Morris

Submitted in partial fulfillment of the requirements for the Degree of Doctor of Philosophy.

This research was sponsored by the National Institutes of Health under grant numbers 5U01HL108642 and 1R01GM085022, the National Science Foundation under grant numbers DBI0448453 and DBI0965316, and the Pittsburgh Life Sciences Greenhouse. The views and conclusions contained in this document are those of the author and should not be interpreted as representing the official policies, either expressed or implied, of any sponsoring institution, the U.S. government or any other entity.

Keywords: genomics, gene expression, gene regulation, microarray, RNA-Seq, transcriptomics, error correction, comparative genomics, regulatory networks, cross-species, expression database, Gene Expression Omnibus, GEO, orthologs, microRNA, target prediction, Dirichlet Process, Indian Buffet Process, hidden Markov model, immune response, cancer.

To Mom and Dad.

Abstract

Advances in genomics allow researchers to measure the complete set of transcripts in cells. These transcripts include messenger RNAs (which encode for proteins) and microRNAs, short RNAs that play an important regulatory role in cellular networks. While this data is a great resource for reconstructing the activity of networks in cells, it also presents several computational challenges. These challenges include the data collection stage which often results in incomplete and noisy measurement, developing methods to integrate several experiments within and across species, and designing methods that can use this data to map the interactions and networks that are activated in specific conditions. Novel and efficient algorithms are required to successfully address these challenges.

In this thesis, we present probabilistic models to address the set of challenges associated with expression data. First, we present a novel probabilistic error correction method for RNA-Seq reads. RNA-Seq generates large and comprehensive datasets that have revolutionized our ability to accurately recover the set of transcripts in cells. However, sequencing reads inevitably contain errors, which affect all downstream analyses. To address these problems, we develop an efficient hidden Markov model-based error correction method for RNA-Seq data . Second, for the analysis of expression data across species, we develop clustering and distance function learning methods for querying large expression databases. The methods use a Dirichlet Process Mixture Model with latent matchings and infer soft assignments between genes in two species to allow comparison and clustering across species. Third, we introduce new probabilistic models to integrate expression and interaction data in order to predict targets and networks regulated by microRNAs.

Combined, the methods developed in this thesis provide a solution to the pipeline of expression analysis used by experimentalists when performing expression experiments.

Acknowledgements

A Ph.D. may be the highest personal academic reward which many wish to achieve, but the road leading to a Ph.D. is certainly not a work of a single person. I would like to express my deepest gratitude to the multitude of people who have taught, helped, and supported me during the joyful but also adventurous and challenging time at Carnegie Mellon University. Certainly for me, writing this acknowledgements is one of the most wonderful exercises in graduate school.

I am indebted to the generous support of my advisor, Ziv Bar-Joseph, who is truly a source of inspiration and ideas. Not too long after I started school, it immediately became clear to me that his academic success is a product of a remarkable balance of work, family, and life-long passions. He is not only an academic father but also a life role model. Ziv is always persistent and patient with answering a myriad of my questions. Every week, I look forward to our meeting with a list of questions and always leave with more ideas to work on. His instinct and fast thinking ability cut through conceptual layers of many problems so quickly and lead to questions, for which usually take me weeks to find good answers. Not only did I learn the technical and research methodology, but I also developed an appreciation for high-impact research, which must be well-motivated and driven by deliberate applications and substantial findings. He is so dedicated to the research and detail-oriented to the results. On one occasion, Ziv showed up at my office late in the evening to my surprise. It turned out that he went home earlier forgetting to send materials needed for our paper submission due at midnight. He walked back to school and gave it to me in person.

I appreciate the committee, Roni Rosenfeld, Chris Langmead, and Quaid Morris for their advice, comments, and suggestions to improve the work in this thesis and my oral presentation. In particular, Quaid meticulously read the draft and suggested ways to make the draft more readable. Roni insisted on making the presentation more accessible to the audience.

I want to thank past and current members of the Systems Biology group at CMU. Marcel Schulz is remarkable at selling new ideas and dedicated to new research collaboration. His openness to share knowledge led to the first part of this thesis. Saket Navlakha is instrumental in helping me improve my presentation skills. Our discussion about research, philosophical aspects of life, and religion is always entertaining and makes lunch more enjoyable. I enjoy small chats with Anthony Gitter, Shan Zhong, Aaron Wise, and Guy Zinman, with whom I shared room and explored new cities during conferences. I am grateful for administrative help of Diane Stidle and Michelle Martin in scheduling meetings, talks and paperwork.

I would like to thank my parents, Hai Le and Lanh Chau, for their unconditional love and support. My dad, who taught me maths in first grade, introduced me to the world of logical thinking. My humble and warm-hearted mom taught me how to listen and treat people with care and respect. Although both were not physically with me during my undergraduate and graduate study, their presence was always in my heart. My sister,

Tram Le, and her family is a source of comfort and encouragement in difficult time.

I cherish my time spending with many new friends in Pittsburgh. Thao Pham, Hoang Tran and Ha Nguyen cook delicious food and always welcome me to share their culinary delights. I enjoy listening to Hang Nguyen discussing, debating and ranting about politics, history or horoscope. Hoan Ho always reminds me of a determined and strong-willed person when I face a difficult task. Phuong Pham and Thang Ho motivated me to run and trained with me. I am thankful for Suze Ninh's diligent care and effort in revising my writing and listening to my practice talks. Her sincere love comforts me in stressful time. Tuan Nguyen's sense of humor puts away worries and troubles. They are always available to listen to my problems and cheerfully enjoys sips of whiskey or a bottle of beer.

I also thank other friends that I have exchanged ideas and interacted with: Lucia Castellanos, Rob Hall, Tzu-Kuo Huang, Wooyoung Lee, Ankur Parikh, Liang Xiong, Min Xu and Yang Xu. I will miss playing tennis and going to the gym with Hoan, Marcel, Saket, Yang, Chao Shen, and Hua Shan.

Pittsburgh, PA
May 2013

Contents

Contents vii

List of Figures xi

List of Tables xiii

List of Notations xv

1 Introduction 1
- 1.1 Growing amount of genomics data . . . 1
- 1.2 Review of probabilistic models . . . 7
- 1.3 Overview of this thesis . . . 8
- 1.4 Organization of this thesis . . . 9

I Collecting and preprocessing gene expression data 11

2 SEECER: a probabilistic method for error correction of RNA-Seq 13
- 2.1 Introduction . . . 13
- 2.2 Methods . . . 14
- 2.3 Experimental setup . . . 20
- 2.4 Robustness and comparison with other methods . . . 20
- 2.5 Assembly of error corrected RNA-Seq sea cucumber data . . . 30
- 2.6 Discussion . . . 33

II Cross-species analysis of functional genomics pathways 35

3 Querying large cross-species databases of expression experiments 37
- 3.1 Introduction . . . 37
- 3.2 Methods . . . 39
- 3.3 Results: Testing distance metrics on data from human and mouse tissues . 44
- 3.4 Results: Identifying similar experiments in GEO . . . 49
- 3.5 Conclusions and future work . . . 53

4 Cross-species Expression Analysis with Latent Matching of Genes 55
- 4.1 Introduction . . . 55
- 4.2 Problem definition . . . 56
- 4.3 Model . . . 57
- 4.4 Experiments and Results . . . 62
- 4.5 Conclusions . . . 66

III Using expression data to infer condition-specific miRNA targets 67

5 GroupMiR: Inferring Interaction Networks using the Indian Buffet Process 69
5.1 Introduction . . . 69
5.2 Interaction model . . . 70
5.3 Regression model for mRNA expression . . . 75
5.4 Inference by MCMC . . . 76
5.5 Results . . . 78
5.6 Conclusions . . . 82

6 PIMiM: Protein Interaction based MicroRNA Modules 83
6.1 Introduction . . . 83
6.2 Methods . . . 84
6.3 Constraint module learning for multiple condition analysis . . . 88
6.4 Results . . . 89
6.5 Conclusions . . . 99

7 Conclusions and Future Work 101
7.1 Conclusions . . . 101
7.2 Themes shared by the methods in this thesis . . . 102
7.3 Future work . . . 103

A Supplementary materials for Chapter 2 107
A.1 Detailed analysis of false positive and false negatives after TopHat alignment with the human data . . . 107
A.2 De novo assembly results by expression . . . 107
A.3 Detailed analysis of types of corrections made by SEECER . . . 107
A.4 Factors affecting running time of SEECER . . . 110

B Supplementary materials for Chapter 3 117
B.1 Metric properties . . . 117
B.2 Proof of Asymptotic Normality . . . 117
B.3 Pseudometric properties of the relational weighted rank matrix . . . 117
B.4 Matrix and Vector Weight metrics . . . 118
B.5 Normality of the null distribution . . . 118
B.6 Robustness of the methods . . . 118
B.7 Human and mouse tissue list . . . 120
B.8 Identifying similar experiments in GEO . . . 120

C Supplementary materials for Chapter 5 129
C.1 Taking the infinite limit . . . 129
C.2 The generative process . . . 130
C.3 GO results for clusters in Figure 5.5 . . . 132
C.4 Comparison with GenMiR++, K-means, and IBP . . . 132
C.5 Networks at 60% posterior probability. . . . 132

D Supplementary materials for Chapter 6 139

D.1 Solving the optimization problem (6.4) 139
D.2 Distribution of module sizes 141
D.3 Choosing the parameters K and α 141
D.4 Enrichment results of several modules from TCGA dataset 141

Bibliography **147**

List of Figures

1.1 An overview of the information flow in gene expression . . . 2
1.2 The role of MicroRNAs . . . 3
1.3 Hybridization to an Affymetrix array . . . 4
1.4 Improvements of sequencing technology . . . 5
1.5 Growth of microarray databases . . . 6
1.6 Typical steps in analyzing genomics data . . . 9

2.1 An overview of SEECER . . . 15
2.2 An example set of reads with genuine sequencing errors and intrinsic differences 17
2.3 Performance of spliced alignment with TopHat after SEECER error correction with different k values . . . 22
2.4 Performance of Oases *de novo* transcriptome assembly after SEECER error correction with different k values . . . 22
2.5 Performance of spliced alignment with TopHat after SEECER error correction with different values for the maximum entropy value . . . 23
2.6 Performance of spliced alignment and de-novo assembly after SEECER error correction with different values for α . . . 24
2.7 Error Distribution of the human data after TopHat alignment . . . 25
2.8 The distribution of mismatches to the reference . . . 26
2.9 SNP calling from TopHat alignments . . . 27
2.10 An illustrating example how Oases benefits from SEECER error correction . . . 30
2.11 *De novo* assembly of sea cucumber data . . . 32

3.1 PR curves of Matrix Weight metrics with different rank values . . . 46
3.2 PR curves of DiffExpr with different values of x . . . 46
3.3 Comparison of different metrics using human-mouse tissues . . . 47
3.4 The penalty matrix . . . 48
3.5 The learned weight vector . . . 48
3.6 PR curves of Spearman's rank correlation, TBR, NMF, Vector and Matrix Weight metrics . . . 48
3.7 PR curves of DiffExpr with different values of x (Novartis dataset) . . . 49
3.8 The penalty matrix between ranks learned from the Novartis dataset . . . 49
3.9 Correlation of orthologs . . . 50

4.1 Graphical model of DPMMLM . . . 58
4.2 Evaluation of the result on simulated data . . . 63
4.3 The heatmap for clusters inferred for the immune response dataset . . . 64

5.1 The data sources used by GroupMiR . . . 71
5.2 The posterior distribution of K . . . 79
5.3 An example synthetic dataset . . . 79
5.4 Performance of GroupMiR versus GenMiR++ . . . 80

5.5 Interaction network recovered by GroupMiR 81

6.1 Data used as input for PIMiM 84
6.2 Interactions between genes of the modules 90
6.3 MSigDB enrichment analysis 92
6.4 Gene Ontology enrichment analysis 93
6.5 The effect of protein interaction data to the result 94
6.6 Inferred miRNA modules of the three cancer types 96
6.7 MiRNAs and mRNAs assigned to Module 11 in all three cancer types 97
6.8 Network of miRNAs and mRNAs of Module 23 98
6.9 Network of miRNAs and mRNAs of Module 48 99

A.1 Analysis of transcript reconstruction accuracy according to expression level . 108
A.2 The number of corrections that SEECER made to the 55M paired-end 45bps reads of human T cells 109
A.3 The number of different types of *mismatch* corrections that SEECER made to the 55M paired-end 45bps reads of human T cells 109
A.4 The number of different types of *insertion* corrections 110
A.5 The number of different types of *deletion* corrections that SEECER made to the 55M paired-end 45bps reads of human T cells 111

B.1 The histogram of the Spearman correlation of 2000 random pairs of microarrays 119
B.2 PR curves for the Matrix Weight metric when starting with fewer orthologs . 119
B.3 PR curves of the metrics using 1000 most variant genes 120
B.4 PR curves of the metrics on a randomized dataset 120
B.5 Correlation of 500 orthologs 120
B.6 The similarity between 3416 human and 2991 mouse microarrays 123

C.1 Result of GenMiR++ 134
C.2 Network inferred by GroupMiR with 60% posterior probability 135
C.3 Network inferred by GenMiR++ with threshold of 0.6 136

D.1 Projection procedure to solve the optimization problem (6.4) 140
D.2 The histogram of the size of modules of SNMNMF and PIMiM 141
D.3 Performance of PIMiM with different values of K 142
D.4 PIMiM of performance with different values of α 142

List of Tables

2.1 Error rates for several sequencing platforms . . . 13
2.2 Evaluation using a RNA-Seq dataset of 55M paired-end 45bps reads of human T cells . . . 24
2.3 Evaluation using a RNA-Seq dataset of 64M paired-end 76bps reads of HeLa cell lines . . . 29
2.4 Evaluation using a RNA-Seq dataset of 145M paired-end 101bps reads . . . 29

3.1 Top 14 words identified in titles of pairs determined to be similar . . . 52
3.2 GO enrichment analysis for mouse genes using STEM . . . 52

4.1 The GO enrichment result for cluster 1 identified by DPMMLM . . . 65

6.1 Evaluation of all methods on the ovarian cancer dataset . . . 91
6.2 MiRNAs specifically identified for a cancer type . . . 95

A.1 Analysis of false positives and false negatives on the 5 lane human data . . . 111
A.2 Analysis of false positives and false negatives on the 64M paired-end 76bps reads of HeLa cell lines . . . 112
A.3 Analysis of false positives and false negatives on the dataset of 145M paired-end 101bps reads . . . 113
A.4 Analysis of reconstructed alternative isoforms in T-cell data . . . 113
A.5 Running time . . . 114
A.6 Analysis of SNP calls on the T-cell dataset after TopHat alignments . . . 114
A.7 GO table for Sea urchin peptides matched in both time points . . . 115
A.8 GO table for Sea urchin peptides only matched in the first time point . . . 115
A.9 GO table for Sea urchin peptides only matched in the second time point . . . 116
A.10 Analysis of blastx alignment matches to sea urchin peptides . . . 116

B.1 The one-one similarity list of human and mouse tissues . . . 121
B.2 Top 14 words identified in titles of pairs determined to be similar . . . 122
B.3 The result of human assessment of identified matched dataset pairs . . . 123
B.3 The result of human assessment of identified matched dataset pairs . . . 124
B.3 The result of human assessment of identified matched dataset pairs . . . 125
B.3 The result of human assessment of identified matched dataset pairs . . . 126
B.3 The result of human assessment of identified matched dataset pairs . . . 127
B.3 The result of human assessment of identified matched dataset pairs . . . 128

C.1 GO enrichment analysis of clusters in Figure 5.5 . . . 133
C.2 GO results for genes in Figure C.3 . . . 137

D.1 Enrichment analysis of the set of genes in Module 11 . . . 143
D.2 Enrichment analysis of the set of genes in Module 23 . . . 144
D.3 Enrichment analysis of the set of genes in Module 48 . . . 145

List of Notations

Symbol	Description
Bold uppercase letters	Matrices, e.g. $\mathbf{X}$, $\mathbf{Y}$
Bold lowercase letters	Column vectors, e.g. $\mathbf{u}$, $\mathbf{v}$
$\mathbf{z}_k$	kth row of a matrix $\mathbf{Z}$
$\mathbf{z}_{,k}$	kth column of a matrix $\mathbf{Z}$
x_i	ith element of a vector $\mathbf{x}$
x_{ij}	Entry (i, j) of a matrix $\mathbf{X}$
$\mathbf{x}^{\mathrm{T}}$	Transpose of $\mathbf{x}$
$\mathbf{A}^{\mathrm{T}}$	Transpose of $\mathbf{A}$
$\mathbf{I}$	Identify matrix
$\mathbb{1}[\cdot]$	Indicator function
$\mathbb{1}_{\mathbf{\Phi}}$	Binary matrix indicating whether an entry of $\mathbf{\Phi}$ is non-zero
$\mathrm{tr}(\mathbf{M})$	Trace of a matrix $\mathbf{M}$
$\mathrm{diag}(\mathbf{x}^{\mathrm{T}})$	Diagonal matrix whose diagonal is $\mathbf{x}$
$\mathrm{E}[X]$	Expected value of a random variable X

Introduction

The last couple decades have seen an explosion of biological data generated using advanced high-throughput methods such as microarray or deep sequencing technology. The emergence of large datasets leads to a new era of data-driven biology which requires new methods in biology, computer science, and machine learning. Many studies utilize and integrate different datasets to uncover the complex dynamics underlying biological systems. The new technologies and studies raise the need for computational methods that are robust against noise and can handle specific data characteristics arising from different technological limitations, experimental design, and measurement errors to support these types of studies [2]. Moreover, methods integrating information from different experiments or different sources of data are one of the keys to overcome the low signal to noise ratio and for insights into difficult biological problems [3]. Machine learning methods, specifically probabilistic models, which are the focus of this thesis, promise to help analyze these large biological datasets and can lead to testable hypotheses improving our understanding of biological systems of interest.

This thesis proposes new computational methods that address challenges arising from the study of gene expression. These challenges include preprocessing data, querying large databases of experiments to facilitate cross-species analysis, and mapping expression data onto regulatory interaction networks. Before elaborating on these challenges and providing an overview of our contributions, we briefly review the technologies that are used to measure gene expression in cells and concepts that are important for the discussion in this thesis.

1.1 Growing amount of genomics data

1.1.1 Gene expression and regulation process

Gene expression is the process, in which the genetic code stored in DNA is used to synthesize functional gene products (Figure 1.1). This highly regulated process, linking the static DNA code to the dynamics of living cells, underlines many biological processes including cell differentiation, cell cycle, development, metabolism, apoptosis, and signaling [4]. Recent literature has reported the link between the dysregulation of gene expression and complex human diseases such as cancer [5] or activity of viruses that specifically target pathways in the regulatory network to weaken the host immune response [6]. While gene expression is a complex system, we focus our efforts on transcription, the first step of expression, during which DNA is copied to generate transcripts including messenger RNAs (mRNAs) as well as other short RNAs (such as microRNAs). These transcripts convey genetic information from DNA and are transported from the nucleus to the cytoplasm, where ribosomes carry out the instructions to assemble proteins.

The transcriptome includes the total set of transcripts, RNA molecules, present in a particular sample or tissue at a given time. Messenger RNAs, which composes a large part of the transcriptome, carrying the coding information out of the nucleous to the sites of protein synthesis, reflects the amount of genetic code transcribed or the gene

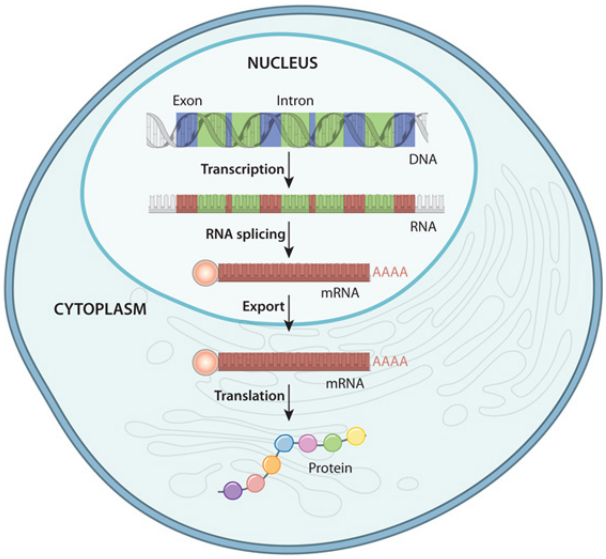


Figure 1.1: An overview of the information flow in gene expression. @2010 Nature Education All rights reserved.

activity. Several important processes concerning mRNAs include splicing, where certain non-coding sequences (introns) are removed from the pre-mRNA; editing where certain nucleotide positions are changed after transcription; and mRNA denadenylation and decay. mRNAs can also be post-transcriptionally regulated by proteins that bind to specific mRNA targets and affect their translational rates. One of the key post-transcriptional regulation processes is driven by microRNAs.

MicroRNAs (miRNAs) MicroRNAs are a family of small, non-coding RNAs that regulate gene expression at the post-transcriptional level. Since the initial discovery of the two miRNAs in *Caenorhabditis elegan*, hundreds of microRNAs have been found in many eukaryotes, including mammals, worms, flies, and plants [7]. MicroRNAs are single-stranded RNAs of 19-25 nucleotides long, initially transcribed by RNA polymerase II (RNAPII) either from miRNA genes or from introns of protein-encoding genes. These primary precursor RNAs (pri-miRNAs) contain one or more stem-loops, each containing mature miRNA sequences. Pri-miRNAs are processed through two main steps catalysed by two members of the RNase III family of enzymes, Drosha and Dicer, operating in complexes with dsRNA-binding proteins (dsRBPs). The first nuclear step produces pre-miRNAs, which are transported from the nucleus into the cytoplasm. In the following step, the pre-miRNA hairpin is cleaved by the enzyme Dicer yielding a miRNA-miRNA* duplex about 22 nucleotides in length. One strand of this duplex is incorporated into an miRNA-induced silencing complex (miRISC), while the other strand is released and degraded [8].

MiRNAs are regulators of post-transcriptional gene expression in a diverse range of

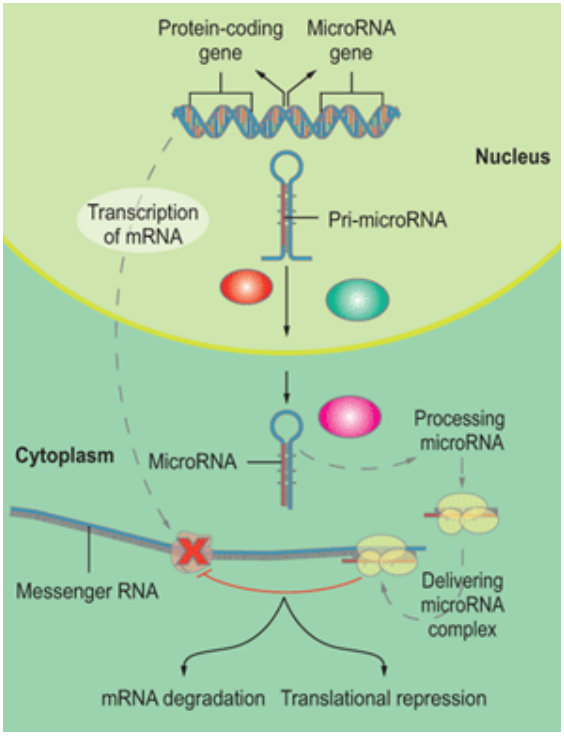


Figure 1.2: The role of MicroRNAs. @2009 National Cancer Insitute All rights reserved.

biological functions such as cell differentiation, division, and apoptosis. MicroRNAs were recently discovered as a class of regulatory RNA molecules that regulate the levels of messenger RNAs (mRNAs) (which are later translated to proteins) by binding and inhibiting their specific targets [9]. Most miRNAs imperfectly bind to sequences in the 3'-UTR of target mRNAs based on Watson-Crick complementary, down-regulate the expression of the targets, and inhibit protein synthesis by either repressing translation or promoting mRNA deadenylation and decay (Figure 1.2). It has been found that miRNA regulation is very ubiquitous as one microRNA can target thousands of genes. Different combinations of miRNAs are expressed in different cell types and coordinately regulate cell-specific targets [7]. Expression profiles of miRNAs have also been used to predict cancer survival, and miRNA dysregulation has also been linked to many inherited diseases and cancers [10]. These findings suggest that miRNAs are important regulators of a wide range of cellular processes.

Mapping interactions between genes and RNA transcripts to create a complex regulatory network is one particular subject of this thesis.

1.1.2 Transcriptome analysis and expression data

Unlike the genome, which is static and relatively fixed for a particular cell type, the transcriptome is dynamic and reflects how gene activity varies across cells. These variations are important because they underline a wide range of cellular activities, developmental processes as well as differences between healthy and disease tissues. Transcriptome analysis can explain the bridge between the genetic code and the functional gene product and phenotype by using genomics data to explore gene transcription, key regulators, and interaction between RNA molecules. Therefore, it is an important tool to study the complex dynamics of cells, human diseases, and for developing new drugs.

Below we discuss the technologies used to determine expression levels and highlight features that lead to challenges when performing downstream analysis.

Microarray technology

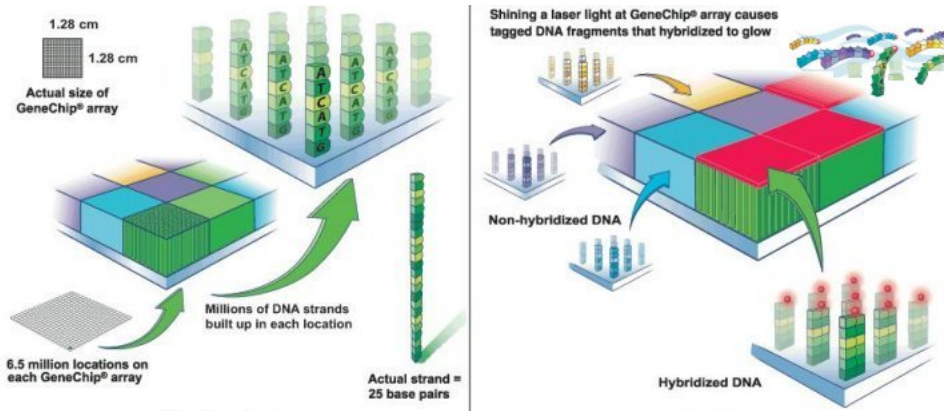


Figure 1.3: Hybridization to an Affymetrix array. Other brands of microarray work similarly. @2007 Affymetrix All rights reserved.

Over the last decade, microarrays have became a de facto tool for scientists to measure genome-wide transcription levels. This high-throughput technology allows the activity of thousand genes to be quantified in one pre-manufactured chip. The technology relies on a sequence-based design and hybridization to quantify expression levels of a set of known transcripts (Figure 1.3). Since hybridization requires high abundance of biological materials, amplification is needed, making microarrays less sensitive to lowly expressed genes.

Each chip contains thousands of DNA probes which are short sequences of genes for profiling. RNA material is extracted from samples or tissues and RNA molecules are broken into small pieces, purified and amplified for detection. These small pieces are used as substrates for reverse transcription in the presence of fluor-derivatized nucleotides (most commonly Cy3 and Cy5 dyes). The samples containing dye-labeled cDNAs are hybridized onto a microarray chip. The arrays are scanned in a specialized machine to visualize the

fluorescence for quantifying the hybridization intensity of each probe. Computational tools analyze the scanned images, subtract background noise using statistical models and eventually output the detected gene expression levels for each gene.

Unfortunately, although microarrays provide a very cost-effective means for assessing the transcriptome, the technology suffers from some technical limitations. First, the inter and intra-platform reproducibility of microarray measurements has been questioned [11], mostly due to the hybridization noise. Second, measuring hybridization intensity for individual probes to infer the transcript abundance level is difficult, requiring careful design of probes with sufficient sensitivity and specificity to avoid cross-hybridization. This raises a third limitation where probe design requires existing prior sequence information of genes or transcripts. In many cases, this knowledge is not available such as in detection of new transcripts and isoforms, or post-transcriptional modifications, or genetic variants in complex diseases such as cancers.

Sequencing technology

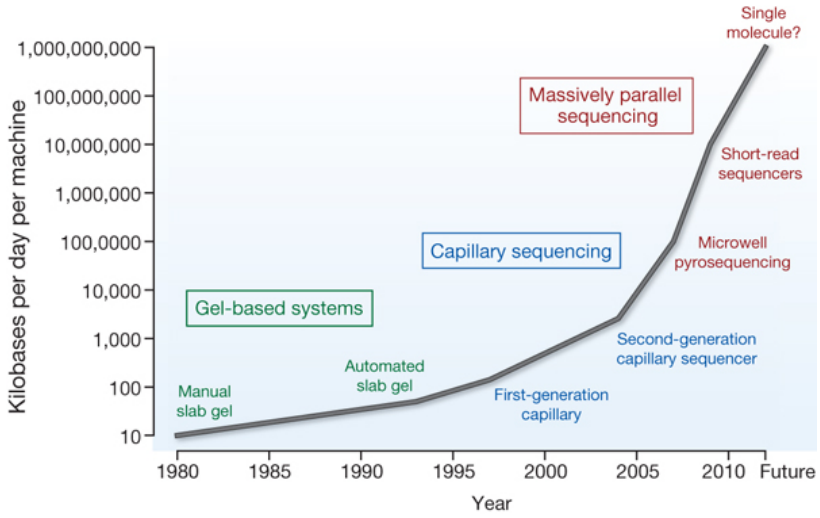


Figure 1.4: Improvements of sequencing technology. Source: [12].

The sequencing technology dates back to 1977, when Fred Sanger and Alan R. Coulson introduced methods to determine DNA sequences [13]. On the wake of the Human Genome Project, this sequencing method has been refined through parallelization and automation into a much more cost-effective and reliable tool. In 2005, 454 Life Sciences launched the first next-generation DNA sequencer that could read one gigabase of DNA sequence in a couple of days. Subsequently, Solexa (later bought by Illumina) has introduced new sequencers that improved on both speed and cost. In recent years, there has been a remarkable improvement in the rate of sequencing (Figure 1.4).

RNA sequencing (RNA-Seq) RNA Sequencing (RNA-Seq) [14] is a recently developed approach to transcriptome measurement that employs next generation sequencing machinery for a complete assessment of RNAs in a sample. Compared to hybridization approaches, e.g. microarrays, RNA-Seq provides several key advantages. It does not require knowledge of the sequences necessary to design probes, hence it allows detection and quantification of novel transcripts, new RNA molecules, genetic variants, and complex transcriptional events. It also does not suffer from high background noise due to cross-hybridization as in microarray technology. RNA-Seq can provide precise locations of transcription boundaries facilitating discovery of new isoforms, alternative splicing events, fusion genes or trans-splicings.

Each sequencing run could produce a few hundred million reads of 50-200 bases in length[1]. For instance, Illumina's HiSeq2000 outputs up to 35Gb per day for a 2×100 bp run[2]. RNA-Seq data is massive and without computational tools, it is impossible to analyze this data. Problems in storing, searching, assembling RNA-Seq reads have been actively studied recently.

Expression databases

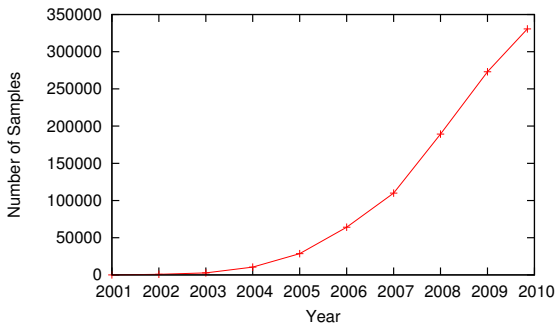


Figure 1.5: Growth of microarray databases. Growth in microarray datasets deposited in GEO in the last decade. The growth resembles the impressive growth of sequence databases in the 90's.

The lower cost, and the increased speed in generating gene expression data, has led to a tremendous increase in the number of datasets produced over the years (Figure 1.5). Collaborative efforts have created many public repositories for genomics experiments such as the Gene Expression Omnibus (GEO), the ArrayExpress Archive, and the Sequence Read Archive. These databases, which archive and freely distribute microarray, next-generation sequencing, and other forms of high-throughput data, provide data storage, encourage data sharing among researchers, and in some cases deliver curated data for follow-up analyses. In addition, many journals require authors to public deposit their data before

[1] as of May 2013

[2] `http://www.illumina.com/Documents/systems/hiseq/datasheet_hiseq_systems.pdf`, May 2013

publication. This creates challenges to store, archive and analyze data, especially when analysis requires integrating many different data sources across different experiments.

1.2 Review of probabilistic models

Throughout this thesis, we employ probabilistic models to help analyze noisy data, recognize patterns, and make inference and learning about the generative process of data. Probabilistic models have been known to perform well when dealing with noisy data, and provide confidence values. We discuss the detailed computational models in the specific chapters. Here we provide a brief overview of the general classes of probabilistic models that are used in this thesis.

Probabilistic graphical models Graphical models provide a framework to represent complex distributions over variables using a graph-based representation. Variables are nodes in the graph and dependency between variables are directed or undirected edges. This representation is a natural way to describe the model and compactly describes the dependencies. It allows inference about some variables given observations of the other variables, learning of parameters, making decision, and finding the most appropriate dependency structure of model variables for the observed data.

1.2.1 Nonparametric Bayes: infinite models

Fitting a probabilistic model to data is hard and requires choosing the right model complexity to balance between bias and variance, or solving the famous model section problem. Model selection is an important problem when analyzing real world data. Many clustering algorithms, including Gaussian mixture models, require as an input the number of clusters or in other models, the number of features is not known. In addition to domain knowledge, this model selection question can be addressed using cross validation. Bayesian nonparametric methods provide an alternative solution allowing the complexity of the model to grow based on the amount of available data. Under-fitting is addressed by the fact that the model allows for unbounded complexity while over-fitting is mitigated by the Bayesian assumption. The model which we proposed in this thesis use two popular infinite models: the Dirichlet Process Mixture Model and the Indian Buffet Process.

Dirichlet Process Mixture Model

Dirichlet Process The Dirichlet process is a nonparametric prior distribution for partitions over a set of objects. We could describe the Dirichlet process by the Chinese restaurant process, a discrete-time process. Consider N customers going to a chinese restaurant with an infinite number of tables. The first customer enters the restaurant and sits at a random table. The following customers enter one after the others and choose tables as follows: the nth customer either sits at an empty table with probability $\frac{\alpha}{n-1+\alpha}$ or an occupied table with probability $\frac{c}{n-1+\alpha}$, where c is the number of customers sitting at the table.

Dirichlet Process Mixture Model (DPMM) Dirichlet process has been used as a non-parametric prior on the parameters of a mixture model. This model is referred to as Dirichlet Process Mixture Model. In this model, the mixture membership variables are given a Dirichlet process prior. The number of clusters is inferred from the data.

Indian Buffet Process

We also use a binary matrix Z to represent interactions between miRNAs and mRNAs in our model. Griffiths and Ghahramani [15] proposed the Indian Buffet Process (IBP) as a nonparametric prior distribution on sparse binary matrices $\mathbf{Z}$. The IBP can be derived from a simple stochastic process, described by a culinary metaphor. In this metaphor, there are N customers (entities) entering a restaurant and choosing from an infinite array of dishes (groups). The first customer tries Poisson(α) dishes, where α is a parameter. The remaining customers enter one after the others. The ith customer tries a previously sampled dish k with probability $\frac{m_k}{i}$, where m_k is the number of previous customers who have sampled this dish. He then samples a Poisson($\frac{\alpha}{i}$) number of new dishes. This process defines an exchangeable distribution on the equivalence classes of $\mathbf{Z}$, which are the set of binary matrices that map to the same left-ordered binary matrices [15]. Exchangeability means that the order of the customers does not affect the distribution and that permutation of the data does not change the resulting likelihood.

1.3 Overview of this thesis

Advances in genomics allow researchers to quantify the set of transcripts in cells at a low cost and much higher efficiency than ever before. While this expression data is a great resource for reconstructing the activity of networks in the cells, it also presents several challenges. These challenges begin with the data collection stage since the technology used to generate the data is not perfect, leading to incomplete and noisy measurement. The first part of this thesis discusses SEECER, a general method for preprocessing RNA-Seq data, which improves many downstream analyses. Successful analysis of expression data requires researchers to integrate experiments from multiple conditions and studies. One particular type of analysis, cross-species study, compares and contrasts high throughput data including gene expression across species to reveal an overall role of common genes and processes underlying biological systems, and to study the differences between species driving speciation and adaptation. The second part of this thesis develops methods to facilitate cross-species analysis, namely querying of large expression databases and inferring orthologs using expression data. The dynamics of expression data allows researchers to construct regulatory networks and identify key regulators of gene expression. The last part proposes two new models to infer condition-specific targets of miRNAs, an important class of regulators.

Combined, the methods developed in this thesis provide an improvement to the pipeline of expression analysis used by experimentalists when performing expression experiments as summarized in Figure 1.6. These methods highlight the importance of data preprocessing, modeling of data characteristics, and encapsulation of structure to

model the underlying biology. Probabilistic models and efficient inference algorithms allow us to scale these methods to handle large expression datasets.

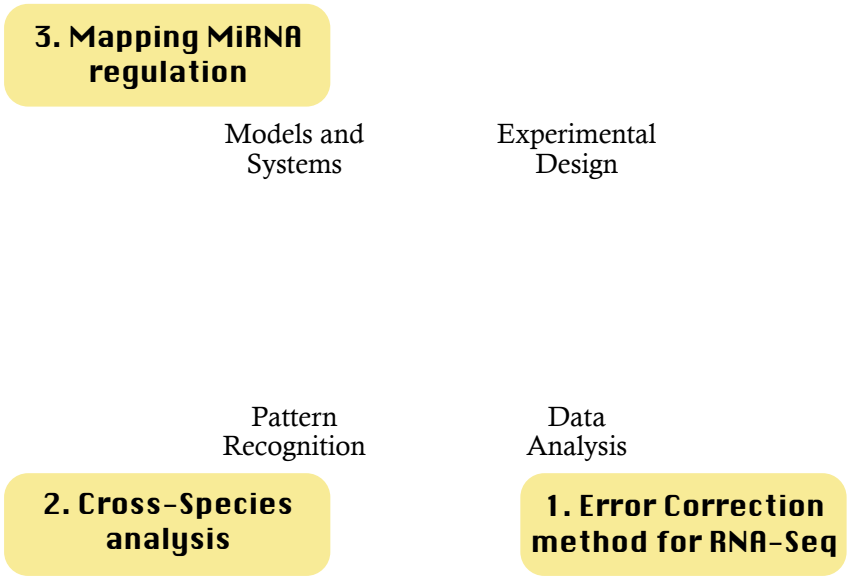


Figure 1.6: Typical steps in analyzing genomics data.

1.4 Organization of this thesis

The thesis is organized as follows. Chapter 2 presents SEECER, a method for error correction in RNA-Seq data and examines its performance through a series of analyses. Chapter 3and 4 discuss our treatment of cross-species analysis of expression data. We argue for the importance of querying large expression experiments and provide one method for performing such queries. For integrating experiments across species, we present the DPMMLM method, which allows discovery of "core" and "divergent" sets of genes in cross-species with probabilistic assignments of genes. Chapter 5 and 6 introduce two probabilistic models for inferring cooperative groups of regulatory miRNAs and their gene targets. These models incorporate other data sources such as sequence-based prediction databases and protein-protein interaction data. Finally, we conclude the thesis in Chapter 7 with some discussions and several directions for future work.

Part I

Collecting and preprocessing gene expression data

2

SEECER: a probabilistic method for error correction of RNA-Seq[1]

Transcriptome analysis has been revolutionized by next-generation sequencing technologies [17]. The sequencing of polyadenylated RNAs (RNA-Seq) is rapidly becoming standard practice in the research community due to its ability to accurately measure RNA levels [18, 19], detect alternative splicing [20], and RNA editing [21], determine allele [22] and isoform specific expression [23, 24], and perform *de novo* transcriptome assembly [25, 26, 27].

2.1 Introduction

Although RNA-Seq experiments are often more accurate than their microarray predecessors [18, 23], they still exhibit a high error rate. These errors can have a large impact on the downstream bioinformatics analysis and lead to wrong conclusions regarding the set of transcribed mRNAs. One class of errors concerns biases in the abundance of read sequences due to RNA priming preferences [28, 29], fragment size selection [30, 31], and GC-content [32]. Sequencing errors, that are a result of mistakes in base calling of the sequencer (*mismatch*), or the insertion or deletion of a base (*indel*), are another important source of errors for which no general solution for RNA-Seq is currently available. For example, error rates of up to 3.8% were observed when using Illumina's GenomeAnalyzer [33]. Table 2.1 summarizes common errors and error rates for commercially available platforms.

Instrument	Primary Errors	Single-pass Error Rate (%)	Final Error Rate (%)
3730xl (capillary)	Substitution	$0.1 - 1$	$0.1 - 1$
454, all models	Indel	1	1
Illumina, all models	Substitution	~ 0.1	~ 0.1
Ion Torrent - all chips	Indel	~ 1	~ 1
SOLiD - 5500xl	A-T bias	~ 5	≤ 0.1
Oxford Nanopore	deletions	≥ 4	4
PacBio RS	CG deletions	~ 15	≤ 15

Table 2.1: Error rates for several sequencing platforms. Source: [1]

A common approach to sequencing error removal is *read trimming* of bad quality bases from the read end to improve downstream analysis [20, 34]. Such an approach reduces the absolute amount of errors in the data, but can also lead to significant loss of data which affects our ability to identify lowly expressed transcripts.

[1] This work is published in [16].

A number of approaches were primarily proposed for the correction of *DNA sequencing data* [35]. These methods use suffix trees [36, 37], k-mer indices [38, 39], and multiple alignments [40]. While successful, these approaches are not always suited for RNA-Seq data. Unlike genome sequencing which often results in uniform coverage, transcripts exhibit non uniform expression levels. The only error correction method that we are aware of that explicitly targets non uniform coverage data is Hammer [41]. Unfortunately, Hammer cannot be used to correct reads as it only outputs corrected k-mers of much shorter length. Even after contacting the authors of Hammer and using their implementation, we could not use it with standard methods for read alignment or assembly and we are not aware of other papers that had. Finally, all the above methods often fail at the border of *alternatively spliced exons* which may lead to false positive corrections.

Other sequencing error correction methods have been designed for tag-based sequencing or microRNA sequencing where the read spans the complete tag or transcript region under investigation [42, 43, 44]. These methods, including SEED [44], are based on clustering similar read sequences, but do not consider partially overlapping read sequences, alternative splicing, and the correction of indel errors.

Here we present the first general method for SEquencing Error CorrEction in Rna-seq data (SEECER) that specifically addresses the shortcomings of previous approaches. SEECER is based on a probabilistic framework using hidden Markov models (HMMs). SEECER can handle different coverage levels of transcripts, joins partially overlapping reads into contigs to improve error correction, avoids the association of reads at exon borders of alternative splicing events, and supports the correction of mismatch and indel errors. Because SEECER does not rely on a reference genome, it is applicable to *de novo* RNA-Seq. We tested SEECER using diverse human RNA-Seq datasets and show that the error correction greatly improves performance of the downstream assembly and that it significantly outperforms previous approaches. We next used SEECER to correct RNA-Seq data for the *de novo* transcriptome assembly of the sea cucumber. The ability to accurately analyze *de novo* RNA-Seq data allowed us to identify both conserved and novel transcripts, and provided important insights into sea cucumber development.

2.2 Methods

Figure 2.1 presents a high level overview of SEECER's read error correction. The overall goal is to model each contig with a HMM allowing us to model substitutions, insertions, and deletions. We start by selecting a random read from the set of reads that have not yet been assigned to any HMM contig. Next, we extract (using a fast hashing of k-mers method) all reads that overlap with the selected read in at least k nucleotides. Because the subset of overlapping reads can be derived from alternatively spliced or repeated segments, we next perform clustering of these reads selecting the most coherent subset for forming the initial set of our HMM contig. Using this set we learn an initial HMM using the alignment specified by the k-mer matches. This learning step can either directly rely on the multiple alignment of reads or use standard HMM learning (Expectation Maximization) but with a limited number of indels in order to keep the run time of the Forward-Backward algorithm linear. Next, we use the consensus sequence defined by the HMM to extract more reads from our unassigned set by looking for those that overlap

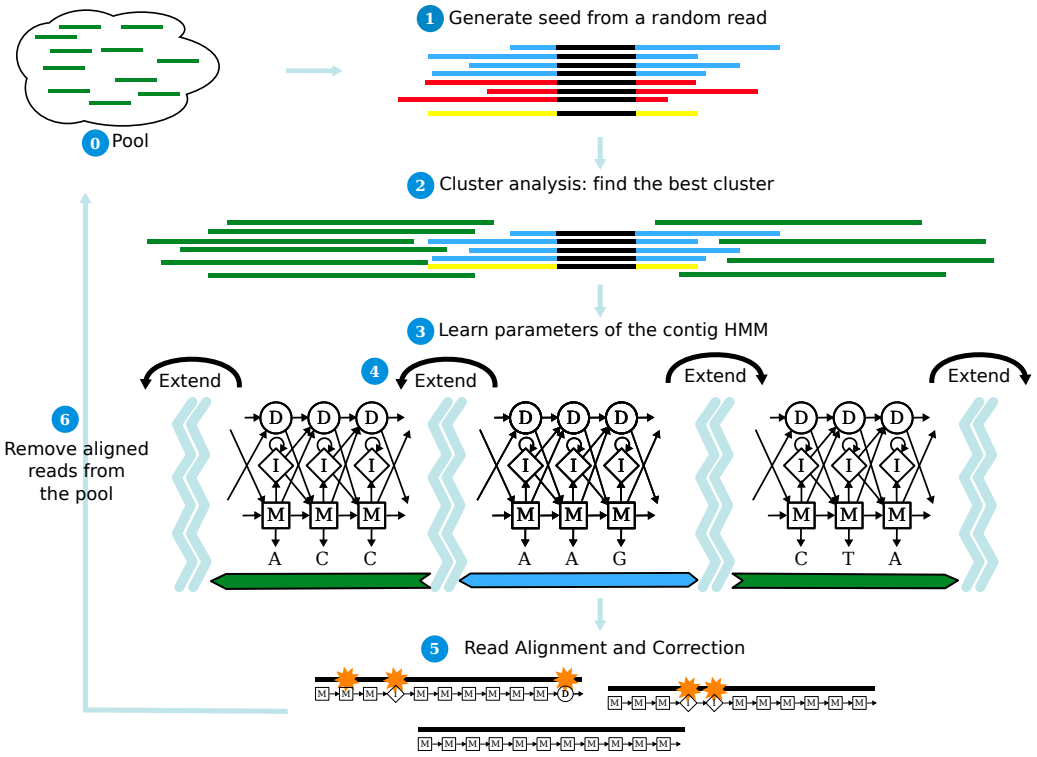


Figure 2.1: An overview of SEECER. *Step 1*: We select a random read that has not yet been assigned to any contig HMM. Next, we extract all reads with at least k consecutive nucleotides that overlap with the selected read. *Step 2*: We cluster all reads and then select the most coherent subset as the initial set of the contig HMM. *Step 3*: We learn an initial HMM using the alignment specified by the k-mer matches of selected reads. *Step 4*: We use the consensus sequence defined by the contig HMM to extract additional reads from our unassigned set. These additional reads are used to extend the HMM in both directions. *Step 5*: When no more reads can be found to extend the HMM we determine for each of the reads that were used to construct the HMM the likelihood of being generated by this contig HMM. For those with a likelihood above a certain threshold, we use the HMM consensus to correct errors. *Step 6*: We remove the reads that are assigned or corrected from the unassigned pool.

the current consensus in k or more nucleotides. These additional reads likely overlap the edges of the HMM (because those overlapping the center have been previously retrieved) and so they can be used to extend the HMM in both directions in a similar manner to the method used to construct the initial HMM. This process (learning HMM, retrieving new overlapping reads, etc.) repeats until no more reads overlap the current HMM or the

entropy at the edges of the HMM exceeds a predefined threshold.

When the algorithm terminates for a HMM, we determine for each of the reads that were used to construct the HMM how likely it is that they have been generated by this contig HMM. For those reads where this likelihood is above a certain threshold, we use the HMM consensus to correct errors in places where the read sequence disagrees with the HMM. We use several filtering steps to avoid false positive corrections including testing for the number of similar errors at the same position, the entropy of a position in the HMM and the number of corrections made to a single read.

The rest of this section describes these steps in more details.

2.2.1 Overview of SEECER

Error correction of a read is done by trying to determine its context (overlapping reads from the same transcript) and using these to identify and correct errors. SEECER builds a set of contigs from reads where each contig is theoretically a subsequence of a transcript. Ideally, we would like each contig to be exactly one transcript. However, in several cases transcripts may share common subsequences due to sequence repeats or alternative splicings. In such cases, each contig in our model represents an unbranched subsequence of some transcript.

We use a profile hidden Markov model (HMM) to represent contigs. Such models are appropriate for handling the various types of read errors we anticipate (including substitutions and insertion / deletion). Due to several restrictions imposed by the read data, even though we may need to handle a large number of contigs, learning these HMMs can be done efficiently (linearly in the size of the reads assigned to the contig).

2.2.2 Contig Hidden Markov Model (HMM)

Profile HMM is a HMM that was originally developed to model protein families in order to allow multiple sequence alignment with gaps in the protein sequences. The set of states in profile HMMs: $Q = \{\mathsf{I}, \mathsf{D}, \mathsf{M}\}$, are respectively the insertion, deletion (gaps) or match state. Emission probabilities in the match and insertion states corresponds to a distribution of possible nucleotides for a particular position in the alignment. The transition probabilities between all pairs of hidden states except for D*arrow*I are non-zero. More details of profile HMMs can be found in [45].

Profile HMM provides a theoretical framework for aligning sequences from the same family. Here, we extend profile HMMs to model the sequencing of reads from a contig. We thus call this a *contig HMM*. Each contig HMM includes a consensus sequence based on the set of reads assigned to this contig. The consensus is constructed from the most probable output nucleotides of the match states. Using this consensus sequence we can make correction to the reads assigned to this contig HMM.

In order to determine if the HMM parameters converged during learning, we use a convergence criterion that is commonly used in the Machine learning community. We stop the learning procedure for a contig HMM whenever the total absolute change in the parameters of the models (emission probabilities) is within ϵ (in our case, $\epsilon = 1e^{-6}$).

The core functionality of SEECER is constructing the contig HMM from sequencing reads. We now outline the details of each step in the following sections.

Pool of reads We maintain a global pool $\mathcal{P}$ (Figure 2.1, step 0) of reads during the execution of our method. SEECER creates many threads, each independently builds a separate contig HMM. For each such HMM we start with a random read as the seed and iteratively extend it using overlapping reads. To avoid collision between two HMMs (i.e. prevent two threads from reconstructing the same transcript) we do the following. First, we randomize the seeds so that threads running in parallel would likely use seeds from different transcripts. In addition, we keep track of whether a read has been assigned to a contig. When a thread tries to assign a read that has been assigned to another contig HMM, we detect this as a collision and stop the construction of the new contig.

2.2.3 Selecting an initial set of reads for a contig HMM

Using the seed read we obtain an initial set of reads to use for constructing the HMM contig (Figure 2.1, step 1). We build a k-mer hash dictionary, where the keys are k-mers and the values are the indices of the reads and the position of the k-mers within them. This hash table could be large, hence we discard k-mers appearing in less than c reads (here we use $c = 3$). Counting of k-mers is efficiently done using Jellyfish [46], a parallel k-mer counter. After counting, only k-mers that appear at least c times are stored in a hash table that also records the positions of the k-mer within a read, and as a result, we keep memory requirements as small as possible. Read sequences are saved in the ReadStore from the SeqAn library [47].

SEECER starts the contig construction by selecting (without replacement) a random read (or seed) s from the pool $\mathcal{P}$ of reads. We use the dictionary to retrieve a set $\mathcal{S}$ of reads ($\mathcal{S} \subseteq \mathcal{P}$) such that each read in $\mathcal{S}$ shares at least one k-mer with the seed s. At the same time, we record the locations of the shared k-mers among the reads to construct a multiple sequence alignment $\mathbb{A}_{\mathcal{S}}$. For each column i ($1 \leq i \leq n$) of $\mathbb{A}_{\mathcal{S}}$, let T_i be the nucleotide that is the most frequent in that column. Let $T = \{T_1, \ldots, T_n\}$ be set of such nucleotides from all columns. Using our current alignment we define $m_i = \{x \in \mathcal{S} : \mathbb{A}_{\mathcal{S}}(x, i) \neq T_i\}$, that is, m_i are the set of reads that have a mismatch with T_i. For each read x, we also define $m(x) = \{i : \mathbb{A}_{\mathcal{S}}(x, i) \neq T_i\}$. In other words, $m(x)$ are the set of columns for which x has a mismatch with T.

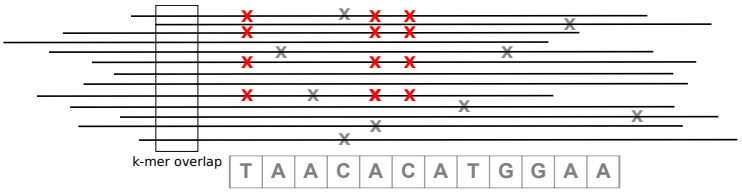


Figure 2.2: An example of a multiple alignment of RNA-Seq reads with genuine sequencing errors (gray crosses) and intrinsic differences (red crosses). Cluster analysis on the alignment columns marked with crosses is used to separate both sets of reads, see text.

2.2.4 Cluster analysis of reads initially retrieved by k-mer overlaps

Because it is only based on k-mer matches, our initial set $\mathcal{S}$ is most likely from a mixture of different transcripts. This situation arises from genomic repeats and alternative splices. To build a homogenous contig, we use cluster analysis to identify the largest subset $\mathcal{S}^*$ of $\mathcal{S}$ which satisfies a quality measure.

In order to identify the largest subset, the main challenge is in distinguishing genuine sequencing errors from other intrinsic differences such as polymorphisms in repeats. Note that real biological differences should be supported by a set of reads with similar mismatches to the consensus. This means that we could identify a set of reads associated with intrinsic differences by looking at the intersections of m_i's. For example in Figure 2.2, there are 4 reads with mismatches at red marked locations which means that most likely these 4 reads are from a different transcript.

Based on this intuition we use the following steps (Figure 2.1, step 2) to identify $\mathcal{S}^*$. We consider only columns i such that $|m_i| > \alpha$ since columns with smaller number of mismatches are more likely due to errors. The value of α is empirically set to 3 as discussed in Section 2.4.2. Let M be the set of these columns. For a pair of columns i and j, their similarity score is defined as:

$$w_{ij} = \begin{cases} \frac{1}{1+\exp(-(|m_i \cap m_j| - 3))}, & i \neq j \\ 1, & i = j. \end{cases} \tag{2.1}$$

Using this similarity score, we use spectral clustering [48] and a spectral relaxation of k-means [49] to find clusters of columns in M. The number of clusters is determined by spectral clustering [50]. For each cluster C, we remove all reads having at least five or half of the mismatches at the columns in the cluster from $\mathcal{S}$. The remaining reads constitute $\mathcal{S}^*$:

$$\mathcal{S}^* = \{x \in \mathcal{S} : \forall C, |m(x) \cap C| < \min(5, \frac{|C|}{2})\}. \tag{2.2}$$

Spectral clustering of columns in M

Spectral clustering is a well studied clustering algorithm method, which has been shown to perform well in practice. This clustering method is particularly suitable for our purpose since it is robust against noise [50] and is implemented by matrix decompositions, which are numerically stable and we can take advantage of existing optimized implementation. The normalized Laplacian matrix is defined as:

$$\mathbf{L} = \mathbf{I} - \mathbf{D}^{-1/2}\mathbf{W}\mathbf{D}^{-1/2} \tag{2.3}$$

where $\mathbf{D}$ is the diagonal degree matrix: $d_i = \sum_j w_{ij}$.

Spectral clustering compute the first k eigenvectors $\mathbf{u}_1, \ldots, \mathbf{u}_k$ of $\mathbf{L}$ and let $\mathbf{X} = [\mathbf{u}_1, \ldots, \mathbf{u}_k]$. Instead of running k-means on the rows of $\mathbf{X}$ to assign cluster membership, we use a spectral relaxation of k-means approach by a pivoted QR decomposition of $\mathbf{X}$ [49]. Given the QR decomposition with a permutation $\mathbf{P}$:

$$\mathbf{X}^T\mathbf{P} = \mathbf{Q}\mathbf{R} = \mathbf{Q}[\mathbf{R}_{11}, \mathbf{R}_{12}] \tag{2.4}$$

where $\mathbf{Q}$ is a k-by-k orthogonal matrix, and $\mathbf{R}_{11}$ is a k-by-k upper triangular matrix. The cluster membership of each column is determined by the row index of the largest element in absolute value of the corresponding column of $\hat{\mathbf{R}}$ defined by:

$$\hat{\mathbf{R}} = \mathbf{R}_{11}^{-1}[\mathbf{R}_{11}, \mathbf{R}_{12}]\mathbf{P}^{\mathrm{T}} = [\mathbf{I}, \mathbf{R}_{11}^{-1}\mathbf{R}_{12}]\mathbf{P}^{\mathrm{T}} \tag{2.5}$$

This approach yields a global optimal solution, hence is more stable and faster. The number of clusters k is determined by the largest decrease in values of eigenvalues of the normalized Laplacian matrix [50].

2.2.5 Learning the parameters of the contig HMM

SEECER has two learning options (Figure 2.1, step 3). In the first one, we implemented online EM algorithm [51] in which we restricted the alignment to have at most v indels to speed up the Forward-Backward algorithm. In the second one, we estimate the parameters based on the alignment of reads using k-mer positions. The first option is much slower than the second because we have to run Forward-Backward algorithm until the the EM converges. The second option is faster because we only need to do one pass over all reads. Our experiments show that the second option is good enough for correction and keeps the runtime tractable, because often the set of reads is consistent and the amount of errors is low, therefore yielding a good read alignment.

Implementation of the cluster analysis of reads retrieved by k-mer overlaps This clustering step can be implemented efficiently as follows. It takes $O(nL)$ time complexity, where n and L are the number of reads and the read length respectively, to find the set M, the columns with errors to the consensus. We then use Spectral Clustering (see above) and compute the normalized Laplacian matrix between columns in M. This matrix is of size $|M|^2$ so this clustering step takes at most $O(|M|^3)$ additional time complexity. Note that in total, this step only adds a linear computational cost in the number of reads and $|M|$ is upper bounded by L.

2.2.6 Consensus extension using Entropy

We discard positions in the contig HMM with high entropy of the emission probabilities in the match states. Entropy is a probabilistic statistic which captures the uncertainty in the discrete distribution of emissions. Positions with high entropy (default max entropy=0.6) indicate that the initial alignment estimation is not reliable because the set of reads is not consistent. For example, at splitting positions in alternative splicing events, reads from different isoforms may be retrieved, which will lead to high entropy. By discarding these ambiguities, we improve the contig quality and reduce false positive corrections.

Contig Extension Before contig extension (Figure 2.1, step 4) all parameters learned for the HMM thus far are fixed. We iteratively extend the contig HMM by repeatedly retrieving more reads sharing k-mers with the new consensus using the dictionary. Each additional read is *partially* aligned to the HMM and read bases that are not overlapping the HMM are used to learn the newly extended columns of the HMM, repeating cluster analysis, and entropy computation. This iterative process stops when we cannot retrieve any new reads or extend the consensus further.

Probabilistic assignment and correction of reads After the construction of the contig HMM, each read that was used in the construction, is aligned to the HMM using Viterbi's algorithm. Reads whose log-likelihood of being generated by the contig HMM exceeds a threshold of -1 are considered 'assigned' to that HMM. We also restrict the number of corrections for a single read to 5 to avoid making false positive corrections. Finally, assigned reads are removed from the pool of reads (Figure 2.1, step 6).

Handling of ambiguous bases and poly-A tails We remove ambiguous bases (Ns) from the read sequences before running SEECER by randomly substituting an N with one of the nucleotides (A,T,G,C). However, if there are regions with many Ns in a read, we discard the whole read unless these regions occur at the end, in which case, we truncate and keep the read if the new truncated length is at least half of the original. Reads that have more than 70% of their bases all As or all Ts are also discarded, as they likely originate from sequenced poly-A tails.

2.3 Experimental setup

The spliced alignment of reads was performed using TopHat version 1.3.3 and Bowtie version 0.12.5 [52]. Number of aligned reads is reported for uniquely mapped reads as described in [19]. Quake version 0.3 [38] was run as suggested in the manual for RNA-Seq data, the k-mer size was set to 18 and the automatic cutoff mode was disabled, instead all k-mers with count 1 were classified as erroneous. The other programs were run as follows: Coral version 1.4 [40] with the `-illumina` option, HiTEC 64bit version 1.0.2 [37] with options `57000000 4`, and Echo version 1.12 [39] with options `--ncpu 8 -nh 1024 -b 2000000`.

***De novo* RNA-Seq assembly** While the ability to align individual reads is important, another important goal of *de novo* RNA-Seq experiments is transcriptome assembly. To test the impact of error correction on downstream assembly we used the Oases (version 0.2.5) for the *de novo* RNA-Seq assembly for the human and sea cucumber datasets. Similar to [27] we conducted a merged assembly for $k = 21, \ldots, 35$ using default parameters. SEED (version 1.5.1) was run with default parameters, and the resulting cluster sequences were used as input to Oases as described in [44].

Computational infrastructures SEECER and other error correction methods were run with a 8 core Intel Xeon CPU with 2.40GHz and 128GB RAM. The *de novo* assembly with Oases was run on a 48 core AMD Opteron machine with 265GB RAM.

2.4 Robustness and comparison with other methods

We first tested SEECER on human data in order to compare it with other approaches that are widely used for other sequencing data (primarily DNA sequencing as mentioned above).

Human datasets Three human paired-end RNA-seq datasets were downloaded for the comparisons: 55M reads of length 45 bps (ID SRX011546, `http://www.ncbi.nlm.nih.gov/sra/`) [22], 64M reads of length 76bps [53] were downloaded from the GEO database [54] (Accession: GSM759888) and 145 M reads of length 101bps from the ENCODE consortium (`http://genome.ucsc.edu/cgi-bin/hgFileUi?g=wgEncodeCshlLongRnaSeq`) .

2.4.1 Evaluation metrics

Read alignment with TopHat Unlike *de novo* RNA-Seq data, when analyzing human data we can utilize a reference genome to determine the accuracy of the resulting corrections and assembly. An established metric to measure the success of error correction after read alignment is the *gain* metric [35], which is defined as the ratio of newly created versus correctly removed errors.

To compute the gain metrics, we used Tophat to align original and corrected reads to the human reference sequence. Using the reference sequence as ground truth we used the following definitions [55]: a *false positive* was a base that was changed (corrected) although it was correct in the original read. A *true positive* was a base that was corrected to the nucleotide in the reference. A *false negative* was a base that was not corrected even though it is wrong while a *true negative* was a base that was left uncorrected and aligned with the reference. The gain metric was computed as explained in [55]. See Appendix A.1 for more details.

***De novo* RNA-Seq assembly** The evaluation of the human assemblies was conducted by aligning assembled transfrags to the human genome with Blat version 34 [56] and comparing to Ensembl 65 transcript annotation to derive 80% and full length covered transcripts, as previously described [27]. The evaluation metrics were computed using custom scripts.

2.4.2 Influence of parameters

Prior to testing SEECER on the human data we used a subset of $\sim$34 Million reads to assess the influence of the two main parameters for SEECER, the length of k-mers k for the initial hashing phase and the value for the maximum entropy at a position.

Influence of value k on error correction

In order to assess the performance of the SEECER algorithm for different parameters we have benchmarked the influence of the value k on the performance of alignments and *de novo* assembly using a subsample of the complete human dataset, using only 3 of the 5 lanes resulting in $\sim$ 34.7 of the 55 M reads. The performance difference after SEECER error correction (the number of alignments reported by TopHat) with $k = 11, 13, 15, 17, 19, 21,$ and 23 for spliced alignment of reads is presented in Figure 2.3. The same parameters have been tested for assembly with Oases in Figure 2.4. The experiments show that small k-values of 11-15 lead to fewer corrections, most likely because no homogeneous contigs for the HMM can be formed for parts of the read population, due to random overlaps and repeats. $k = 17$ performs best in both alignment and assembly (for the 3 lane data) and is used in this analysis and as default value in the software. For k-values larger than

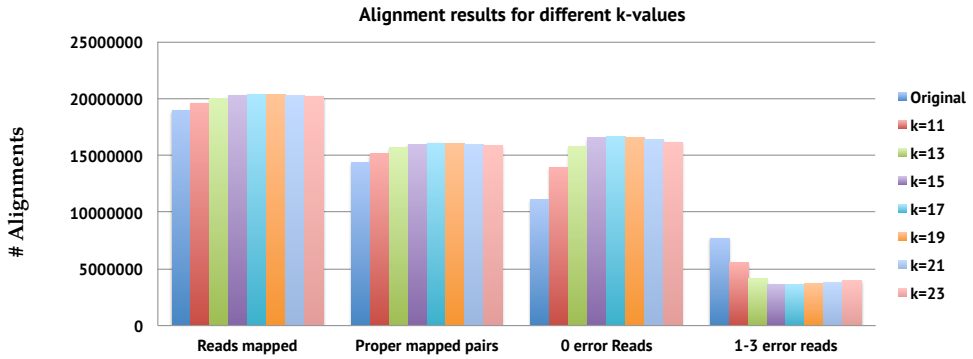


Figure 2.3: Performance of spliced alignment with TopHat after SEECER error correction with different values for the hash length k on 3 lane human data.

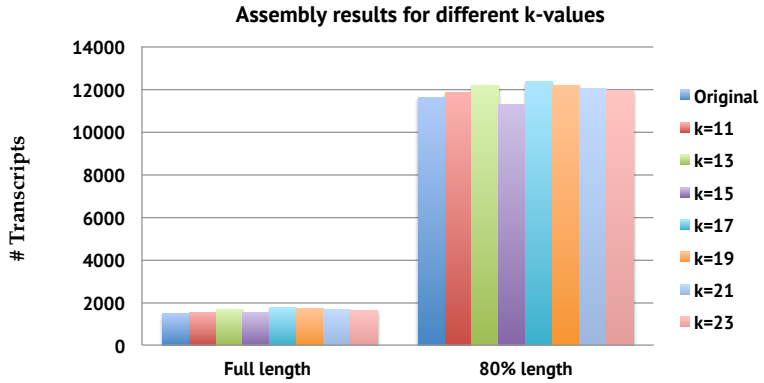


Figure 2.4: Performance of Oases *de novo* transcriptome assembly after SEECER error correction with different values for the hash length k on 3 lane human data. We show the number of alignments reported by TopHat. We show the number of transfrags reported by Oases that are reconstructed from a known human transcript to full and 80% length.) with

17 the number of corrections starts to deteriorate again, because many read-read overlaps are lost due to higher influence of sequencing errors for larger k.

Influence of Entropy value for border extension

We also analyzed the influence of the maximum entropy value allowed for contig extension. The performance difference after SEECER error correction, with varied entropy value

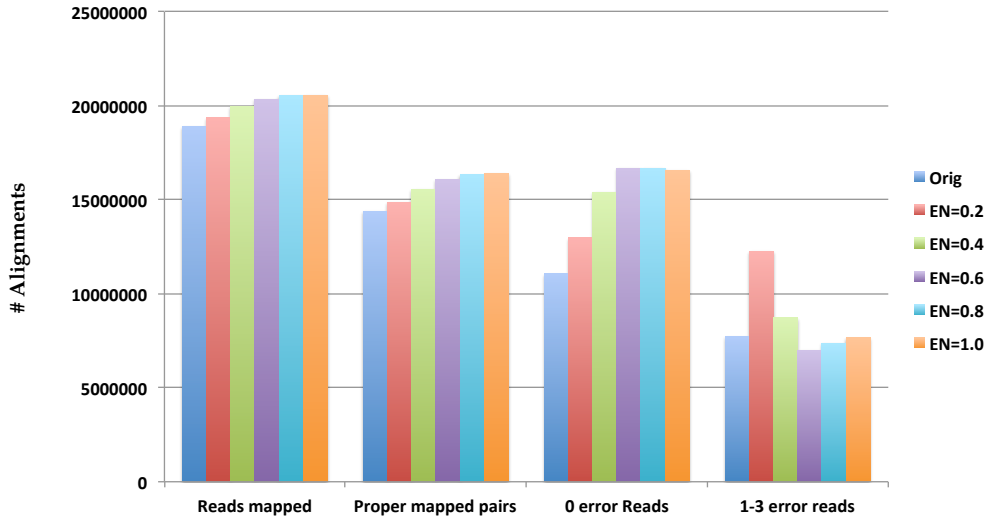


Figure 2.5: Performance of spliced alignment with TopHat after SEECER error correction with different values for the maximum entropy value for contig extension on 3 lane human data. We show the number of alignments reported by TopHat.

from 0.2 to 1 in steps of size 0.2, for spliced alignment of reads is presented in Figure 2.5. The result is that entropy of 0.6 (default value for SEECER) is the best value to achieve the largest number of error free reads. If the entropy threshold is low, it means that contigs get rarely extended and as such many reads are not being corrected.

Influence of α used in the cluster analysis of reads

Cluster analysis of reads is an important step which allows SEECER to handle overlapping effects of alternative splices and genomic repeats. As discussed in Section 2.2.4, we only analyze columns that contain at least α mismatches. Figure 2.6 depicts the performance of alignment and de-novo assembly using SEECER-corrected data with different values of α. Cluster analysis is less comprehensive with large values of α. As a result, more corrections are false positives and may lead to loss of transcripts and other genomic variants. Indeed, the number of transcripts assembled by Oases to full length drops after the value of 3. In contrast, the performance of spliced alignment is improved with large values of α because false positive corrections reduce differences among the reads, so more reads are mappable although many of them may be aligned to wrong locations.

2.4.3 SEECER outperforms other methods

We next have used these parameters ($k = 17$ and entropy was set to 0.6) to compare SEECER to 4 other methods for correcting the reads by initially testing their ability to

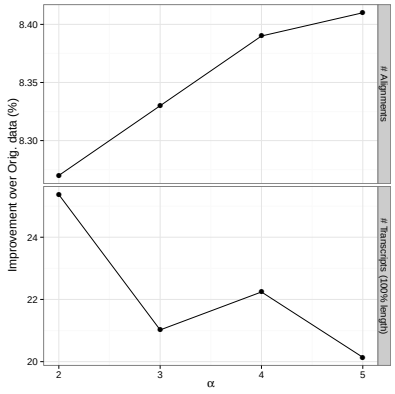


Figure 2.6: Performance of spliced alignment and de-novo assembly after SEECER error correction with different values for α on 3 lane human data. We show the percentage of improvement over the original data.

Method	orig	SEECER	Quake	SEED	Coral	HiTEC	Echo
aligned reads (M)	31.2	**33.8 (+8.4%)**	32.3 (+3.6%)	-	32.6 (+4.5%)	31.2 (+0.0%)	31.6 (+1.3%)
proper pairs (M)	22.1	**25.5 (+15.1%)**	23.4 (+5.8%)	-	24.0 (+8.7%)	22.1 (-0.0%)	22.7 (+2.5%)
0 error reads (M)	18.3	**27.3 (+49.6%)**	22 (+20.4%)	-	23.9 (+30.7%)	18.3(0.1%)	19.6 (+7.2%)
gain	-	**0.56**	0.25	-	0.38	0.00	0.024
full length	1749	**2120 (+21%)**	1979 (+13%)	1358 (-22%)	2092 (+19.6%)	1713(-2.7%)	1916 (+9.6%)
80% length	13852	**14833 (+7%)**	14267 (+3%)	9686 (-30%)	14643 (+5.7%)	13450 (-2.9%)	14273 (+3.0%)
memory (GB)	-	27	32	-	34.3	49	72
time (hours)	-	12.25	7.25	-	2.42	6.33	13.7

Table 2.2: Evaluation using a RNA-Seq dataset of 55M paired-end 45bps reads of human T cells. Percentages in brackets denote performance compared to original data. - means not applicable. The evaluation is based on Ensembl v.65 annotation.

improve the unique alignment of reads to the human genome after correction. We used three diverse datasets to compare SEECER with the k-mer based methods: Quake [38] and ECHO [39], Coral [40] which relies on multiple alignments of reads for correction, as well as with HiTEC [37] which builds a suffix tree and automatically estimates parameters for correction.

The first dataset we used was derived from human T-cell RNA sequencing resulting in 55 million paired-end reads of length 45 bps [22]. In Table 2.2 we list important statistics regarding the success of the error correction methods. Using SEECER, the number of aligned reads increased by 8.4% when compared to the uncorrected reads, much higher

than Quake (3.6%), Coral (4.5%) and ECHO (1.3%). Unlike the other methods, error correction with HiTEC did not result in a higher number of reads mapped. Similarly, the number of reads that align without mismatch errors to the reference sequence using SEECER increased by 50%, which was by far the biggest improvement for all methods tested (Figure 2.7). None of the error correction methods uses paired-end information, therefore the number of properly aligned read pairs can serve as a good indicator for the accuracy of the error correction. Again SEECER error corrected reads showed the highest improvement with 15% more pairs properly aligned. The gain metric shows the normalized difference between true positive and false positive corrections (Table A.1 and Appendix A.1) and again SEECER outperforms the other methods.

Distribution of errors in aligned reads

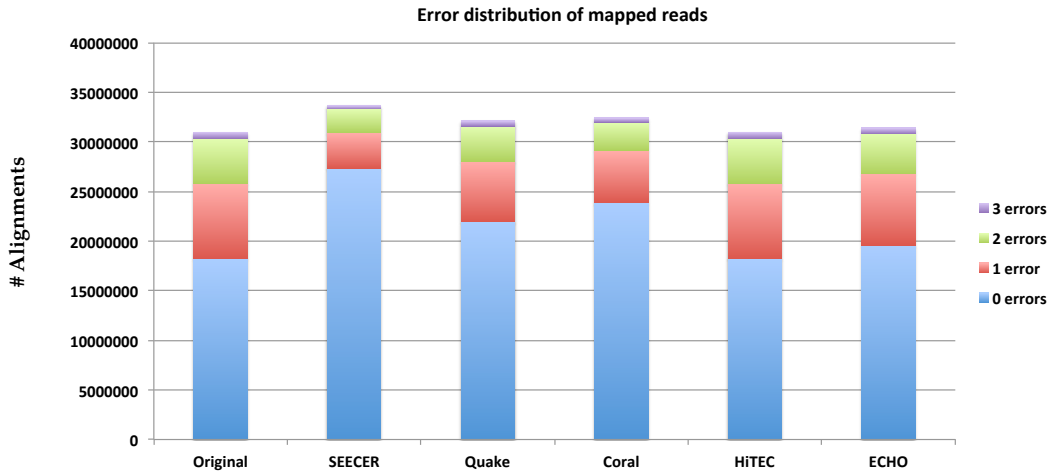
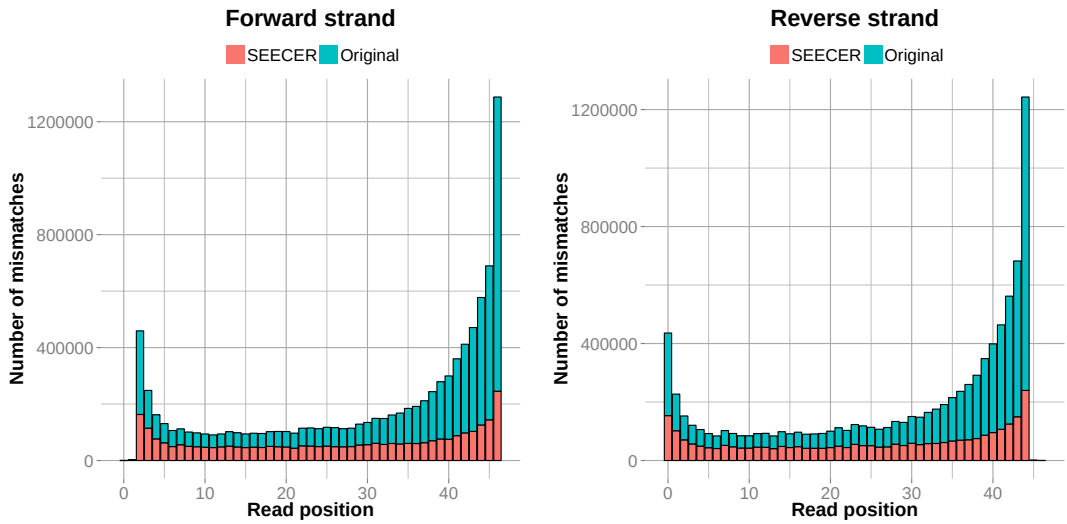


Figure 2.7: Distribution of inconsistent bases with the reference, errors, after TopHat alignment. For each program, reads are partitioned into one of four groups: (i) perfect alignment,(ii) one error, (iii) two errors, and (iv) three errors.

We provide a fine grained analysis of the number of errors per aligned TopHat read from the results in Table 2.2. As can be seen in Figure 2.7, after error correction with SEECER, more reads are aligned and the number of reads with 0 errors is increased. Roughly, the total number of reads that align with ≤ 1 error after SEECER error correction is similar to the total number of original reads aligned with up to three errors. This explains the improvement for the *de novo* assembly results (see below), because exact k-mer overlaps between reads are important for de Bruijn graph based assemblers, like Oases. The reduced error rate should simplify other downstream analyses, including the detection of RNA editing events.

Error bias before and after error correction



(a) (b)

Figure 2.8: The distribution of mismatches to the reference of pair-mapped reads (using TopHat alignment) of the 55M paired-end 45bps reads of human T cells dataset: only reads that are aligned both before and after error correction are shown.

In addition, we investigated the error bias in terms of read positions and forward/reverse read strands. Figure 2.8 presents the distribution of mismatches following TopHat alignments relative to the read positions before, and after error correction by SEECER. As can be seen, the previously reported bias that higher error rates are found at read ends for Illumina data [33], is observed in our data as well. However, after SEECER error correction much of this bias is removed and the corrected reads have a more uniform distribution of mismatches along the read positions. See Figs. A.2-A.5 and Appendix A.3 for details on other types of corrections made by SEECER.

SNP analysis in error corrected RNA-Seq data

In order to further test the influence of error correction on downstream analysis we investigated the ability to identify homozygous SNPs before and after error correction. This analysis demonstrates the usefulness of error correction for such downstream SNP studies and in particular shows that using SEECER corrected reads leads to the identification of the highest number of SNPs.

We downloaded the table _loc_snp_summary.txt from dbSNP build 132 [57]. All variants classified as "trueSNP" were retrieved for the analysis. We used the SnpStore program ([58], `http://www.seqan.de/projects/snpstore/`) to call SNPs from the TopHat

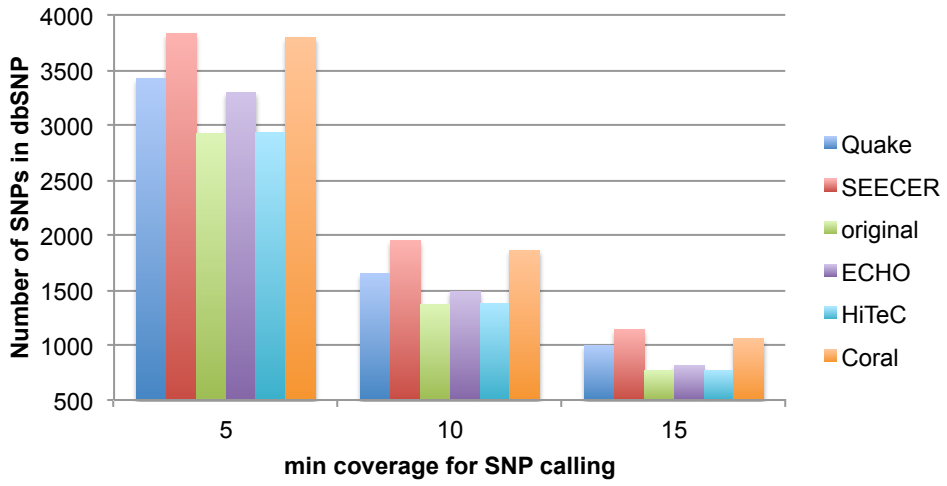


Figure 2.9: SNP calling from TopHat alignments using SnpStore on the T-cell data. Predicted SNPs are compared to annotated SNPs in the dbSNP database (y-axis). The minimum read coverage c for which SNP calls are produced was varied (x-axis).

read alignments before and after correction. A non-reference base b was called a SNP at a genomic position if (i) read coverage on the position $\geq c$, (ii) and the relative frequency of b was ≥ 0.8 to investigate homozygous SNPs. All non-reference SNP calls were compared to non-reference SNPs annotated in dbSNP. We denote as Precision the percentage of SNP calls that are annotated (with the correct base) in dbSNP, i.e., Precision=|annotated in dbSNP|/|total calls|.

In Table A.6 we show the number of SNP calls, their overlap with dbSNP, and the precision for each method. We compute the SNP calls for varying read coverage cutoff ($c = 5, 10, 15$) to investigate different levels of confidence in SNP calling. Figure 2.9 examines the number of annotated SNPs in dbSNP that were called by all methods. SEECER corrected data leads to the highest number of SNP calls and the largest number of SNP calls that are annotated in dbSNP, albeit having a higher precision compared to the other methods. All methods improve upon using the original data in the number of annotated SNPs that were called, although HiTEC and ECHO show only a minor improvement.

2.4.4 *De novo* RNA-Seq assembly

While the ability to align individual reads is important, another important goal of *de novo* RNA-Seq experiments is transcriptome assembly. To test the impact of error correction on downstream assembly we used the Oases *de novo* assembler [27]. In addition to the

read based error correction methods we compared to above, we have compared to SEED read clustering and subsequent Oases assembly as previously suggested [44]. In Table 2.2 the results for the human T-cell data are shown. An important metric for assembly comparisons is the number of full length assembled transcripts. Compared to the original reads, after SEECER error correction 21% more transcripts are reconstructed to full length. SEECER also leads to a 46% increase of detected alternative isoforms (Table A.4). Quake, Echo and Coral led to a lower improvement of assembled full length transcripts with 13%, 9.6% and 19.6% respectively, whereas SEED and HiTEC resulted in a reduction of full length reconstructed transcripts of -22% and -2.7% respectively. The clustering approach used by SEED discards some of the data which leads to loss of lowly to mid-level expressed transcripts (Figure A.1).

Example figure from IGV for improvement after error correction

To illustrate how the correction made by SEECER improves both the alignment of reads and the *de novo* assembly with Oases, we show in Figure 2.10 assembled transfrags from both the original and corrected data in the genomic region containing the transcript ENST00000380876 (EIF3CL). We aligned all transfrags to the human genome and display two longest transfrags which are the best hits to the transcript: Locus_621_Transcript_11 (from SEECER corrected data) shown in red, and Locus_9156_Transcript_20 (from original data) shown in blue. As shown in the bottom box of the figure, with error correction, the transcript ENST00000380876 (EIF3CL) was assembled 95% in length in Locus_621_Transcript_11 as opposed to only 45% in length in Locus_9156_Transcript_20. This improvement in the assembly clearly comes from the removal of errors in the reads, as shown in the top box of the figure. Here, we show the alignment of reads of both data to the region containing exons 9-13 using TopHat. Mismatches of the reads with the reference are denoted as red/blue/green/orange dots. Most of the mismatches were removed from SEECER corrected reads. As a result, Oases using these corrected reads was able to assemble all exons of the transcript ENST00000380876 (EIF3CL).

2.4.5 Additional comparisons using larger datasets with longer reads

To test the scalability of SEECER when using datasets with more reads and longer read length we further tested SEECER on two additional human datatsets: a HeLa cell line dataset of 64M reads of length 76bps (GEO Accession: GSM759888) [53] and 145 M reads of length 101bps from the ENCODE consortium. Due to the time requirements of the assembly step, we have only focused here on the top three performing methods in our original analysis (SEECER, Quake and Coral). SEECER scales well and for both datasets it achieves the best performance for the number of aligned reads, read pairs, full length assembly and gain (Tables 2.3 and 2.4). Additional information about the number of true positive and false positive corrections can be found in Tables A.2 and A.3. While SEECER memory requirements scaled more or less linearly with the size of the dataset, Coral's requirements did not scale in a similar manner. Specifically, we could not run Coral on the largest dataset (Table 2.4) because its memory requirements were beyond the available memory on the machine we used to test all methods.

Method	original	SEECER	Quake	Coral
aligned reads (M.)	28.9	**30.9 (+6.9%)**	30.6 (+5.9%)	29.5 (+2.1%)
proper pairs (M.)	19.4	**21.4 (+10.4%)**	20.8 (+7.2%)	20.0 (+2.8%)
0 error reads (M.)	13.7	**16.9 (+23.4%)**	15.5 (+12.7%)	14.9 (+8.7%)
gain	-	**0.21**	0.11	0.07
assembly full length	4067	**4422 (+8.7%)**	4113 (+1.1%)	4378 (+7.65%)
assembly 80% length	25647	**26507 (+3.4%)**	25644 (-0.0%)	26414 (+2.99%)
memory (GB)	-	52	32	37.3
time (hours)	-	20.33	1	3.5

Table 2.3: Evaluation using a RNA-Seq dataset of 64M paired-end 76bps reads of HeLa cell lines. Percentages in brackets denote performance compared to original data. - means not applicable. The evaluation is based on Ensembl v.65 annotation.

Method	original	SEECER	Quake	Coral
aligned reads (M.)	119.0	**123.1 (+3.47%)**	121.9 (+2.46%)	-
proper pairs (M.)	81.1	**85.4 (+5.4%)**	83.5 (+2.9%)	-
0 error reads (M.)	76.2	**105.3 (+38.2%)**	92.4 (+21.3%)	-
gain	-	**0.58**	0.32	-
assembly full length	13148	**18851 (+43.4%)**	14968 (+13.84%)	-
assembly 80% length	61522	61178 (-0.6%)	**62231 (+1.2%)**	-
memory (GB)	-	113	60	>130
time (hours)	-	40.25	3	-

Table 2.4: Evaluation using a RNA-Seq dataset of 145M paired-end 101bps reads from the Long RNA-seq of IMR90 cell lines from ENCODE Consortium. Percentages in brackets denote performance compared to original data. - means not applicable. The evaluation is based on Ensembl v.65 annotation.

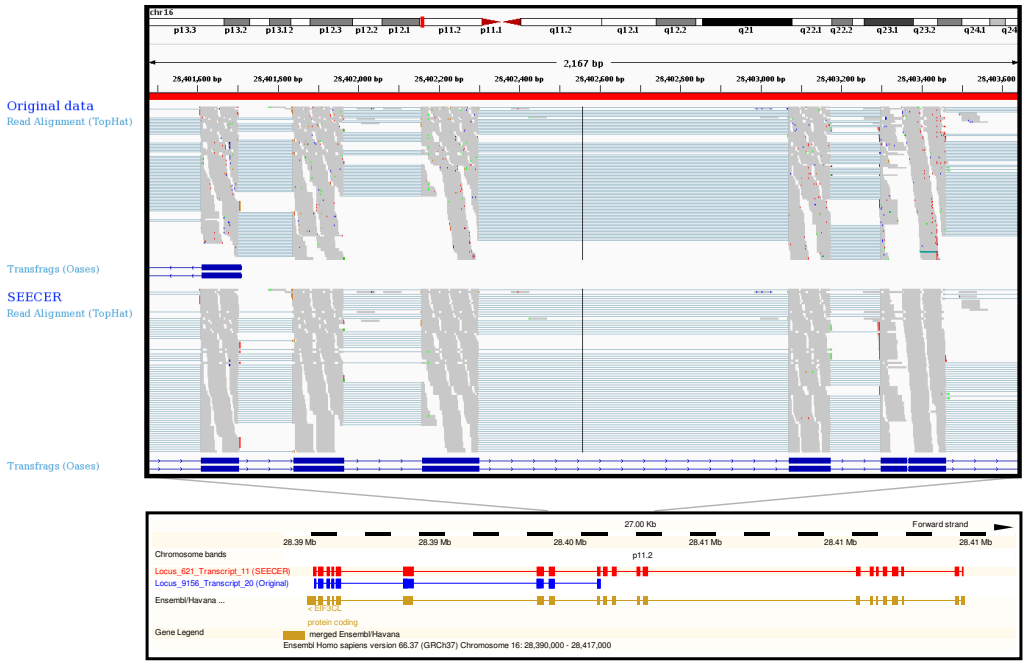


Figure 2.10: An illustrating example how Oases benefits from SEECER error correction. Top: Tophat read alignments in the *EIF3CL* gene for exons 9-13 before (1st track) and after (2nd track) SEECER correction with human data. Colored dots highlight positions with deviations to the reference sequence in the gray read alignments. Bottom: Summary view of the whole region displaying the longest transfrag assembled. Oases assembled the transcript ENST00000380876 (*EIF3CL*) to 95% of its length with SEECER corrected data (red transfrag) whereas it was only assembled to 45% of its length when using the original data (blue transfrag).

2.5 Assembly of error corrected RNA-Seq sea cucumber data

The sea urchin *Strongylocentrotus purpuratus* is a model system for understanding the genetic mechanisms of embryonic development, e.g., [59]. Other species of echinoderms, including the Californian warty sea cucumber *Parastichopus parvimensis* (Figure 2.11A), are being developed as comparative developmental model systems, e.g., [60]. This work however is limited by the absence of a sequenced genome for the sea cucumber. It is thus critical for comparative studies that methods are developed to inexpensively obtain highly accurate transcriptome for organisms for which no sequenced genome exists.

2.5.1 Sea cucumber sequencing and validation

Gravid *P. parvimensis* adults were spawned by heat shock and embryos grown in artificial sea water at 15 degrees Celsius. Total RNA was extracted from 2 day old gastrula and 6 day old larvae using the Total Mammalian RNA Miniprep kit (Sigma). RNA was sent to the Wistar Institute for library preparation with Illumina adaptors and 72bp paired-end sequencing was performed on a Solexa Genome Analyzer II. First strand cDNA synthesis was performed with the iScript Select cDNA Synthesis Kit (BioRad).

From the top 100 expressed transfrags that were expressed in both time points 14 were randomly selected, 7 with a match to either RefSeq or Swissprot and 7 without a match. For the validation, PCR primers were designed with Primer3Plus [61] to amplify approximately 300bp to 500bp products corresponding to the 14 selected transfrags. The PCR was performed using GoTaq (Promega) standard protocols on RNA samples from the first time point.

2.5.2 Sea cucumber transcriptome analysis

Experimental setup For peptide searches we used Blastx [62] with an E-value cutoff of 10^{-5} to avoid spurious alignments in Swissprot [63] and the Sea Urchin known proteome (`SPU_peptide.fasta` at `http://www.spbase.org/SpBase/download/`) . Similarly for the search in Refseq [64] we used Blastn with the same cutoff. The expression of all assembled transfrags was quantified using RSEM with default parameters [65] after read alignment of the reads to the transfrags with Bowtie [66]. The Gene Ontology annotation for the known and predicted Sea Urchin proteome was downloaded from SpBase (`annotation.build6.tar` at `http://www.spbase.org/SpBase/download/`) . Gene Ontology enrichment analysis was done using FuncAssociate 2.0 [67] with a multiple-testing corrected P-value cutoff of 0.05.

To test how SEECER can help in this direction we have produced two new datasets for the transcriptome of *P. parvimensis*. These datasets allow us to determine the expressed mRNAs at the embryonic gastrula (time point 1) and feeding larval (time point 2) stages, which provides insights into the development of this species. Illumina paired-end 72nt sequencing was conducted and resulted in 88,641,446 and 85,575,446 reads for time points 1 and 2, respectively. We have next used SEECER to correct errors in these datasets resulting in 28,655,078 and 25,546,050 corrections for 19,465,515 and 17,305,905 reads, respectively. Each corrected read set was then used to produce a *de novo* RNA-Seq assembly. Error correction took $\sim$ 4.7 and $\sim$ 4.6 hours, whereas *de novo* assembly took $\sim$ 11.3 and $\sim$ 13 hours for time points 1 and 2. 850,056 transcript fragments (transfrags) were assembled for the embryonic stages (time 1) and 682,913 transfrags for the larval (time 2) stage using Oases (Methods).

The only other echinoderm with a sequenced genome is the sea urchin *S. purpuratus* which last shared a common ancestor with sea cucumbers almost 350 million years ago [68]. Thus, we initially analyzed the similarity between the transfrags we obtained and sea urchin proteins. For the embryonic and larva stages 261405 and 189101 transfrags mapped to fragments of 13330 and 11793 distinct known peptides in sea urchin (min length 50 amino acids for each match). Although we only sequenced RNAs from two developmental stages, thereby not sampling much of the long developmental process and

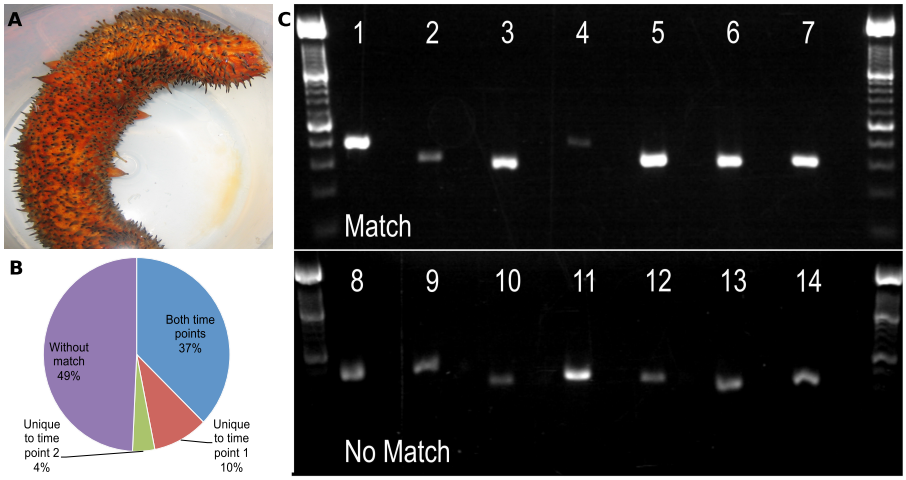


Figure 2.11: *De novo* assembly of sea cucumber data. A) A living sea cucumber *P. parvimensis*. B) Distribution of BlastX matches of sea cucumber transfrags to known sea urchin peptides. The percentages represent the subset of sea urchin peptides that we have significantly matched to at least one transfrag in time point 1 and / or time point 2 and those that were not matched to any transfrag. C) Ethidium bromide stained image of PCR products amplified from sea cucumber cDNA. Primer pairs were designed against 14 assembled transfrags, 7 of which matched to known peptides of RNAs (top row) and 7 other which had no match in the data base (bottom row). 100bp size standard ladders are in the first and last lanes. Each lane is followed by the appropriate no template control to demonstrate that amplification was not due to non specific contamination.

many adult tissues of these organisms, the assembled transfrags from both time points nonetheless matched to more than 50% of known sea urchin peptides. This suggests both that we have achieved a high sequence coverage in the assembly, and that many of the sea cucumber genes are already being expressed during early development. In addition, the fact that 14% of these matches were restricted to only one of the two time points suggests that we are able to detect stage specific developmentally regulated genes, an important requirement for developmental studies (see Figure 2.11B). To illustrate the usefulness of *de novo* sequencing, we next performed a Gene Ontology (GO) enrichment analysis for sea urchins peptides matched to both time points, and those matched only to time point 1 or time point 2. The results are presented in Tables A.7-A.9.

Time point 1 embryos are undergoing active development including cell movements involved with gastrulation. Larval stages (time point 2) meanwhile are actively swimming and feeding in the water column. As can be seen in the GO analysis, many differences in expression between these stages are of mRNAs that encode for proteins involved in energy metabolism which is likely due to a switch in how sessile non feeding embryos and motile feeding larvae utilize energy resources. We also find an enrichment of expression of genes involved in RNA splicing and translation control in time point 1 (embryos) which may be

related to the active transcriptional processing requirements of early embryogenesis. This analysis thus provides an entry point into understanding these important processes.

Even though 62-65% of transfrags matched known sea urchin peptides, 297,173 and 255,672 sea cucumber transfrags for time points 1 and 2 did not significantly match any sea urchin peptide (Methods). We computed the expression levels of the assembled transfrags and investigated the top 100 expressed transfrags that we could not match to sea urchin peptides from both time points in more detail. In the top 100, 28 and 9 transfrags matched to the RefSeq and Swissprot data bases, respectively. Still, we were unable to match 64 transfrags expressed in both time points to any known entry in these data bases. To further test the accuracy of our correction and assembly and whether the non matched transfrags are indeed novel expressed RNAs we have performed additional follow up experiments. We selected 14 transfrags that were highly expressed in both time points and performed RT-PCR analysis on these to confirm that the predicted products could be amplified from sea cucumber derived embryonic cDNA (Figure 2.11C). Of the 14, 7 were derived from transfrags that matched known peptides and another 7 were derived from transfrags with no match to any of the databases we looked at. As can be seen in Figure 2.11C, all 14 transfrags were successfully validated indicating that these are indeed expressed mRNAs and lending support to our correction and assembly procedure.

2.6 Discussion

We have developed and tested SEECER, a new method based on profile HMMs to perform error correction in RNA-Seq data. Our method does not require a reference genome. We first learn a contig HMM using a subset of reads and use the HMM to correct errors in reads that are very likely associated with the HMM. Our method can handle non uniform coverage and alternative splicing, both key challenges when performing RNA-Seq. We tested SEECER using complex human RNA-Seq data and have shown that it outperforms several other error correction methods that have been used for RNA-Seq data, in some cases leading to a large improvement in our ability to correctly identify full length transcripts. We next applied it to perform *de novo* transcriptome correction and assembly of sea cucumber expression data providing new insights regarding the development of this species and identifying novel transcripts that cannot be matched to proteins in other species. We note that although a recent report of a 454 sequencing analysis of mixed embryo, larval and adult tissues provides some coverage of an unrelated species, the Japanese sea cucumber *Apostichopus japonicas* [69], to the best of our knowledge this is the first published transcriptome of *Parastichopus parvimensis*.

Our analysis of the sea cucumber data indicates that we were able to obtain good transcriptome coverage. The expressed genes from the two developmental stages matched 50% of the protein coding regions of sea urchin. In addition, *de novo* correction and assembly was able to accurately detect taxon specific transcripts. This is critical for comparative development studies which, in the absence of a genome sequence, often rely on gene discovery from homology searches in related model species. Full appreciation of the role of species specific genes is essential in order to understand the developmental origins of animal diversity.

Even though one of the main motivations for developing SEECER are applications of *de novo* RNA-Seq, the human data is useful because alignments allow us to explore the accuracy of the methods and it is thus a common practice for testing sequencing error correction approaches [35]. However, we would like to point out that the classification into false and true positives/negatives is based on the human reference sequence, which may miss haplotype alleles. Thus, the false positive rates reported in the tables may be slightly higher than the real false positive rates. Nevertheless, we doubt that this approach favors any of the methods, because none of them use the reference sequence for performing corrections.

The genome read error correction methods Quake and Coral were able to correct many reads but resulted in a large number of false negatives, as indicated by their lower rates of aligned reads and the drop in the gain statistic compared to SEECER. Coral was the closest to SEECER in terms of the resulting number of full length assembled transcripts for two of the three datasets. However, Coral seems to suffer from lack of scalability which may be problematic as dataset size increase. Indeed, its memory requirements for the largest dataset we analyzed were larger than the capacity of our machine cluster.

Our experiments have shown that read clustering leads to a loss of assembled full length transcripts especially for low-to-mid level expressed transcripts, because parts of the data are discarded. Due to non-uniform expression levels in RNA-Seq data, error correction sensitivity critically depends on a methods' ability to detect errors. The performance drop for HiTEC and ECHO, compared to the other methods tested, may be explained by their uniform coverage assumption leading to missing higher frequency errors in highly expressed genes. In contrast, Quake and Coral do not have these strong assumptions and perform much better. However, unlike SEECER they do not employ a probabilistic HMM model and read clustering. These steps allowed SEECER to outperform all other methods in the number of alignable reads, full length assemblies, and false negative rate with only linear increase in memory requirements for larger datasets.

While we have focused here on the improvement to RNA assembly following error correction, it has been shown that *de novo* assemblies allow reliable detection of genes that are differentially expressed between two conditions [70]. Thus, by improving the resulting assembly SEECER is likely to improve downstream differential expression analyses as well.

There are many directions to improve SEECER further by utilizing base call quality scores to improve performance on lowly expressed transcripts or using the paired-end information to improve construction of contigs. Currently, SEECER was designed to work without an available reference sequence (*de novo* RNA-Seq) but an available reference sequence could help with correction of repetitive regions and lowly expressed transcripts.

Finally, while we have primarily developed SEECER for RNA-Seq data, it may also prove useful for single cell and single molecule sequencing. In other studies, including metagenomics and ribosome profiling experiments, researchers encounter sequencing data where the coverage is non-uniform and as such SEECER, which does not assume uniformity, can improve the analysis of these data as well.

Part II

Cross-species analysis of functional genomics pathways

3 Querying large cross-species databases of expression experiments [1]

In the previous chapter, we discuss preprocessing of gene expression data, specifically RNA-Seq data. To support the use of expression data in cross-species analysis, we aim to facilitate the retrieval of similar experiments in large databases of expression studies. Querying cross-species sequence databases have been successfully used before to identify and characterize coding and functional non coding regions in multiple species [56]. Since most drugs are initially tested on model organisms, the ability to compare expression experiments across species may help identify pathways that are activated in a similar way in human and other organisms. However, while several methods exist for finding co-expressed genes in the same species as a query gene, looking at co-expression of homologs or arbitrary genes in other species is challenging. Unlike sequence, which is static, expression is dynamic and changes between tissues, conditions and time. Thus, to carry out cross species analysis using these databases we need methods that can match experiments in one species with experiments in another species.

3.1 Introduction

Advances in sequencing technology have led to a remarkable growth in the size of sequence databases over the last two decades. This has allowed researchers to study newly sequenced genes by utilizing knowledge about their homologs in other species [72]. Alignment and search methods, most notably BLAST [73], have become standard tools and are extensively used by molecular biologists. Cross species analysis of sequence data is now a standard practice. However, similar usage of expression databases has not materialized. Expression databases, including Gene Expression Omnibus[2] (GEO) and ArrayExpress[3] hold hundreds of thousands of arrays from multiple species (see Figure 1.5). Co-expression is a powerful method for assigning new function to genes within a single species [74]. If we are able to identify a large set of matched expression experiments across species, this method can be extended and used in a cross-species analysis setting as well. Consider a human gene with unknown function that is co-expressed (across many different conditions) with a mouse gene with known function. This information can provide useful clues about the function of the human gene. This information is also useful for identifying orthologs. If a gene has multiple homologs in another species then the homolog with the highest co-expression similarity in several conditions is likely its orthologs since they are involved in the same processes in both species.

While promising, querying expression datasets to identify co-expressed genes in other species is challenging. Unlike sequence, which is static, expression is dynamic and changes between tissues, conditions and time. Thus, a key challenge is to match

[1] This work is published in [71].

[2] `www.ncbi.nih.gov/geo/`

[3] `www.ebi.ac.uk/Databases/microarray.html`

experiments in one species with experiments in another species. Almost all studies that have analyzed expression datasets in multiple species relied on one of two methods. They have either carried out experiments under the same condition in multiple species or have looked at co-expression within a species and tested whether these relationships are retained across species. Examples of the former set of methods include comparison of cell cycle experiments across species [75], comparing response programs [76] and comparing tissue expression between human and mouse citegeneatlas. Examples of the latter strategy include the metaGene analysis [77] and cross-species clustering methods citeoscar. See [78] for a recent review of these methods.

While successful, the approaches discussed above are not appropriate for querying large databases. In almost all cases it is impossible to find a perfect match for a specific condition in the database. Even in the rare cases when such matches occur, it is not clear if the same pathways are activated in the different species. For example, many drugs that work well on animal models fail when applied to humans, at least in part because of differences in the pathways involved [79]. Looking at relationships within and between species would also not answer the questions we mentioned above since these require knowledge of orthologs assignment to begin with. These methods are also less appropriate for identifying one-to-one gene matchings because they are focusing on clusters instead.

The only previous attempt we are aware of to facilitate cross species queries of expression data is the nonnegative matrix factorization (NMF) approach presented by Tamayo et al. [80]. This unsupervised approach discovers a small number of metagenes (similar to principal component analysis) that capture the invariant biological features of the dataset. The orthologs of the genes included in the metagenes are then combined in a similar way in the query species to identify related expression datasets. While the approach was successfully used to compare two specific experiments in humans and mouse, as we show in Results, the fact that the approach is unsupervised makes it less appropriate for large scale queries of expression databases.

In this chapter, we present a new method for identifying similar experiments in different species. Instead of relying on the description of the experiments we develop a method to determine the similarity of expression profiles by introducing a new distance function and utilizing a group of known orthologs. Our method uses a training dataset of known similar pairs to learn the parameters for distance functions between pairs of experiments based on the rank of orthologous genes, thus it overcomes problems related to difference in noise and platforms between species. We show that the function we learn outperforms simpler rank comparison methods that have been used in the past [81, 82]. We next use our method to compare millions of array pairs from mouse and human experiments. The resulting matches highlight conditions and diseases that are activating similar pathways in both species and can also hint at diseases were these pathways seem to differ. Given the large number of arrays in current databases our methods can also be used to aid manual annotations of cross species similarity by focusing on a small subset of the millions of possible matches.

We note that while the discussion below focuses on microarray data and we have only tested our methods on such data, our methods are appropriate for deep sequencing expression data as well. As long as a partial orthologs list can be obtained the methods we present below can be used to compare any expression datasets across species.

3.2 Methods

3.2.1 Using ranking for comparing microarrays across species

Our goal is to obtain a distance function that given two microarray datasets outputs a small distance between experiments that are very similar and a large distance for those pairs that study different processes or in which different pathways are activated in the two species being compared. Since we are comparing experiments from different platforms and species, the first decision we made was to compare the ranking of the genes in each array rather than their expression levels (previous methods for comparing experiments in the same species have relied on ranking as well citecellmontage). There are a number of other properties that we seek for such scoring functions. First, they should of course be able to separate similar pairs from non similar pairs. In addition, it would be useful if the function is a metric or a pseudometric (a pseudometric satisfies all properties of a metric except for the identity, that is $d(x, y)$ could be 0 even if $x \neq y$). This will guarantee useful distance properties including symmetry and triangle inequality (See Appendix B). Finally, we would like to be able to determine some statistical properties for these scoring methods in order to determine a p-value for the similarity / difference between the experiments being compared (Section 3.2.3).

Notations

We first provide notations that are used in the rest of this chapter. As mentioned above our function would be constructed from metrics on permutations (ordering) of ranks. Each microarray experiment is a vector in R^n, where each dimension is the expression value for a specific gene. We consider the problem of comparing a microarray $\mathcal{X}$ of a species A with n_A genes and a microarray $\mathcal{Y}$ of a species B with n_B genes. There are m orthologs between the two species. In other words, there is a one-to-one mapping $\mathcal{O}$ from m species A genes to m species B genes. $1, \ldots, m$ are the orthologs, $\mathbf{x} = \{x_i : 1 \leq i \leq m\}$ and $\mathbf{y} = \{y_i : 1 \leq i \leq m\}$ are the expression values of the orthologs in $\mathcal{X}$ and $\mathcal{Y}$, respectively. Let π and σ be the rank orderings of the expression values of the orthologs in $\mathcal{X}$ and $\mathcal{Y}$. For simplicity, we assume that there are no ties in rankings. Therefore, π and σ are two elements of the permutation group $\mathcal{G}_m$. Recall that $\pi, \sigma : \{1, \ldots, m\} \rightarrow \{1, \ldots, m\}$ are bijections: π_i and σ_i are the ranks given to the ortholog i, with lowered numbered ranks given to higher expression values. Also let $\mathbf{I}_m$ be the identity permutation in $\mathcal{G}_m$.

Assume we have a metric d on $\mathcal{G}_m$. For our significance analysis we test the null hypothesis H_0 that π and σ are not associated versus the alternate hypothesis that they are. One way is to ask how large $d(\pi, \sigma)$ would be if σ were chosen uniformly at random. More formally, let D_d be the distribution of $d(\pi, \sigma)$ when σ is drawn uniformly from $\mathcal{G}_m$. We reject the null hypothesis H_0 if $d(\pi, \sigma)$ is significantly smaller than $E(D_d)$. This setting is a standard approach in literature [83].

3.2.2 Fixed distance function: Spearman's rank correlation

Below we discuss distance functions that satisfy the requirements mentioned above for cross-species analysis. We first discuss a method that does not require any parameter tuning. Such methods have been extensively used for comparing permutations. However,

as we show in Section 3.3 they are less appropriate for gene expression data due to the unique properties of such data. In the next section we discuss modification of these methods that are more appropriate for the expression data we are working with.

The Spearman's rank correlation R metric is defined as:

$$R(\pi, \sigma) = \sqrt{\sum_{i=1}^{m} (\pi_i - \sigma_i)^2} \tag{3.1}$$

In other words it is the L_2 distance between π and σ. Hence, it is a metric. Moreover, using Hoeffding's central limit theorem it can be proved that R^2 has a limiting normal distribution [83]. Note that frequently, R is standardized to have values in $[-1, 1]$. This yields the widely used Spearman's rank correlation ρ.

$$\rho = 1 - \frac{6R^2(\pi, \sigma)}{(m^3 - m)} \tag{3.2}$$

3.2.3 Adaptive Metrics

While fixed methods that do not require parameter tuning have proven useful for many cases they are less appropriate for expression data. In such data the importance of the ranking is not uniform. In other words genes that are expressed at very high or very low levels compared to baseline may be very informative whereas the exact ranking of genes that are expressed at baseline levels may be much less important. Thus, rank differences for genes in the middle of the rankings are more likely due to noise. An appropriate way to weight the differences between the rankings may lead to a better distance function between arrays. The key challenge is to determine what are the important ranks and how they should be weighted. Below we present a number of adaptive methods that can address this issue. The methods we present differ in the number of parameters that needs to be learned and thus each may be appropriate for different cases depending on the amount of training data that exists.

Weighted Rank Metric

Using a weight vector $\mathbf{w}$ of length m, we can modify the Spearman's rank correlation and define the following metric:

$$d(\pi, \sigma) = \sqrt{\sum_{i=1}^{m} (w_{\pi_i} - w_{\sigma_i})^2} \tag{3.3}$$

The vector $\mathbf{w}$ defines the weight of each rank and thus captures the significance of each rank in measuring the association of two microarrays. Consider two arrays $(1, 2, 3, 4)$ and $(1, 3, 2, 4)$. Their Spearman R distance is $\sqrt{2}$ while for a weight vector $\mathbf{w} = (1, 0, 0, 1)$, their distance would be 0. Such a weight vector places the weight on the top and bottom matches and disregards middle orderings. This vector $\mathbf{w}$ defines a mapping of the ranking vectors in $\mathcal{G}_m$ onto $\mathcal{R}^m$.

The resulting function is no longer a metric, but rather a pseudo-metric in the original π, σ space ($d(\pi, \sigma) = 0$ does not imply $\pi = \sigma$). However, it is easy to see that it is a metric

in the transformed $\mathbf{w}$-space because it is a L_2 distance between the vectors $\mathbf{w}_\pi$ and $\mathbf{w}_\sigma$, where $\mathbf{w}_\pi = (w_{\pi_1}, \ldots, w_{\pi_m})$ and similarly for $\mathbf{w}_\sigma$. In other words the $\mathbf{w}$-transformation makes some of the permutations indistinguishable indicating that the changes made are not significant and so the two permutations result in the same weighted vector. However, for those permutations that are still distinguishable following the w-transformation the metric properties are preserved. The distribution D_d of $d(\pi, \sigma)$ when σ is drawn uniformly from $\mathcal{G}_m$ is asymptotically normal. See Appendix B for proof. We can calculate the mean and variance of D_d through exact calculation or random sampling. P-value can then be calculated based on this normal distribution.

A specific assignment of weights which is in line with our assumptions regarding the importance of genes expression ranks is the following modified Spearman's rank correlation.

Top-Bottom R (TBR)

For any $0 < k < 1$ and $r > 0$ we can define w as following:

$$w_i = \begin{cases} r(i - km) & \text{if } 1 \leq i < km, \\ r\big(i - (1-k)m\big) & \text{if } (1-k)m < i \leq m, \\ 0 & \text{otherwise.} \end{cases} \tag{3.4}$$

Note that genes expressed at a high level will have negative weights and those with low levels positive weights allowing the method to penalize experiments in which genes move from one extreme to the other. All middle ranks $[km, (1-k)m]$ are assigned the same weight so genes that have ranks changed within this interval do not affect the distance at all. At the same time, it scales the high and low ranks r times to a wider range to increase the granularity of rank difference. Choosing the value of k and r can either be done using cross validation or it could be manually specified.

Learning a complete weight vector w

While the above method leads to different weights for different rankings it specifies a very strict cutoff which may not accurately represent the importance of the differences in ranking. An alternative approach is to assign weights that are continuously changing based on the ranking by learning a weight vector from training data. Here we assume that we have access to such training data which is indeed the case for a number of pairs of species (most notably tissue data for human and mouse as we use in Section 3.3). Assume we have M microarrays of species A and N microarrays of species B and for each microarray, let $\mathcal{S}$ be the set of pairs of similar arrays and $\mathcal{D}$ is the set of pairs of dissimilar arrays. If the dissimilar arrays are not known, we can select $\mathcal{D}$ as the set of all pairs that are not in $\mathcal{S}$.

Each permutation π can be represented as a binary m-by-m matrix $\mathbf{M}_\pi$ such as:

$$M_\pi(i, j) = \begin{cases} 1 & \text{if } \pi_i = j, \\ 0 & \text{otherwise.} \end{cases} \tag{3.5}$$

Using this notation we can define an L_2 metric d as:

$$d(\pi, \sigma) = \|\mathbf{M}_\pi \mathbf{w} - \mathbf{M}_\sigma \mathbf{w}\|_2 \tag{3.6}$$

$$= \sqrt{\mathbf{w}^{\mathrm{T}}(\mathbf{M}_\pi - \mathbf{M}_\sigma)^{\mathrm{T}}(\mathbf{M}_\pi - \mathbf{M}_\sigma)\mathbf{w}} \tag{3.7}$$

Our goal is to learn a vector w such that this distance be small for the positive set and large for the negative set. This leads to the following optimization problem:

$$\min \quad \sum_{(x,y)\in\mathcal{S}} \mathbf{w}^{\mathrm{T}}(\mathbf{M}_{\pi_x} - \mathbf{M}_{\pi_y})^{\mathrm{T}}(\mathbf{M}_{\pi_x} - \mathbf{M}_{\pi_y})\mathbf{w} \tag{3.8}$$

$$\text{s.t} \quad \sum_{(x,y)\in\mathcal{D}} \mathbf{w}^{\mathrm{T}}(\mathbf{M}_{\pi_x} - \mathbf{M}_{\pi_y})^{T}(\mathbf{M}_{\pi_x} - \mathbf{M}_{\pi_y})\mathbf{w} = 1 \tag{3.9}$$

Note that the summation is on different groups. The optimization (top) is summed over the similar pairs whereas the constraint (bottom) is summed over the dissimilar pair. The choice of the constant 1 on the right hand side of (3.9) is arbitrary. However, replacing it with any constant $c > 0$ results only in $\mathbf{w}$ being multiplied by $\sqrt{c}$ which leads to the same order of scores for microarray pairs and so does not change our results. We can further simplify the problem to

$$\min \quad \mathbf{w}^{\mathrm{T}}\mathbf{Z}_{\mathcal{S}}\mathbf{w} \tag{3.10}$$

$$\text{s.t} \quad \mathbf{w}^{\mathrm{T}}\mathbf{Z}_{\mathcal{D}}\mathbf{w} = 1 \tag{3.11}$$

with

$$\mathbf{Z}_{\mathcal{S}} = \sum_{(x,y)\in\mathcal{S}} \mathbf{w}^{\mathrm{T}}(\mathbf{M}_{\pi_x} - \mathbf{M}_{\pi_y})^{\mathrm{T}}(\mathbf{M}_{\pi_x} - \mathbf{M}_{\pi_y})\mathbf{w}$$

$$\mathbf{Z}_{\mathcal{D}} = \sum_{(x,y)\in\mathcal{D}} \mathbf{w}^{\mathrm{T}}(\mathbf{M}_{\pi_x} - \mathbf{M}_{\pi_y})^{T}(\mathbf{M}_{\pi_x} - \mathbf{M}_{\pi_y})\mathbf{w}$$

The matrices $\mathbf{Z}_{\mathcal{S}}$ and $\mathbf{Z}_{\mathcal{D}}$ are positive semidefinite since they are sums of positive semidefinite matrices $(\mathbf{M}_{\pi_x} - \mathbf{M}_{\pi_y})^{\mathrm{T}}(\mathbf{M}_{\pi_x} - \mathbf{M}_{\pi_y})$. Although this optimization is not convex, there exists global minima based on the reformulation of this problem to finding eigenvalues of the Rayleigh quotient. The derivation is similar to Fisher's Linear Discriminant Analysis [84].

Relational Weighted Rank Metric

A drawback of the weight vector distance metric discussed above is that it assigns weights to ranks in each microarray independent of the ranks in the other microarray. To overcome this problem we extend the vector weight $\mathbf{w}$ into a full matrix $\mathbf{W}$ to incorporate the dependence between ranks in two microarrays. For a pair of microarrays with ortholog rankings π and σ, define a symmetric m-by-m matrix $\mathbf{M}^F_{\pi,\sigma}$, whose entries (i, j) are nonzeros if and only if there exists a gene g such that g is ranked i and j in the microarrays, respectively. Formally,

$$M^F_{\pi,\sigma}(i, j) = \mathbb{1}\left[\pi^{-1}(i) = \sigma^{-1}(j)\right] + \mathbb{1}\left[\pi^{-1}(j) = \sigma^{-1}(i)\right] \tag{3.12}$$

In other words, $\mathbf{M}^F_{\pi,\sigma}$ is a matrix where an entry of 1 in location (i,j) indicates that the gene in location i in the first experiment is the same as the gene in location j in the second or vice versa. By definition, $\mathbf{M}^F_{\pi,\sigma}$ is a symmetric matrix. Note that this definition implies that if a gene g is ranked ith in both π and σ then $M^F(i,i) = 2$ and when $\pi = \sigma$, $\mathbf{M}^F = 2\mathbf{I}$. Let $\mathbf{W}$ be a positive semidefinite m-by-m matrix, with each entry w_{ij} being the weight assigned to a gene having rank i and j in the two microarrays. The larger the entries are, the more dependent the two ranks are.

Given these notations we define the distance between the two microarrays as:

$$d(\pi,\sigma) = \sqrt{\sum_{i=1}^{m}\sum_{j=1}^{m}\left((2\mathbf{I} - \mathbf{M}^F_{\pi,\sigma}) \circ \mathbf{W}\right)_{i,j}} \tag{3.13}$$

$$= \sqrt{\sum_{\substack{i,j:\pi^{-1}(i)=\sigma^{-1}(j) \\ \text{or } \pi^{-1}(j)=\sigma^{-1}(i)}} \left(\frac{w_{ii}+w_{jj}}{2} - w_{ij}\right)} \tag{3.14}$$

$$= \sqrt{\text{tr}\left((2\mathbf{I} - \mathbf{M}^F_{\pi,\sigma})\mathbf{W}\right)} \tag{3.15}$$

As mentioned above, if the two permutations are identical then $\mathbf{M}^F = 2\mathbf{I}$ and the distance is 0. Otherwise, the penalty for a disagreement of a pair (i,j) between the rankings is $(w_{ii}+w_{jj})/2 - w_{ij}$. This captures both the importance of the individual ranks (very high or very low ranking genes maybe more important than middle genes) as well as the penalty for the disagreement between the pair. Equation (3.14) also shows that the entity under the square root is non-negative since for a positive semidefinite matrix $\mathbf{W}$, $(w_{ii}+w_{jj})/2 \geq w_{ij}, \forall i,j$. Equation (3.15) follows from Equation (3.13) since $\mathbf{M}^F$ has only one entry in each column or row. This distance function is a pseudometric in the original permutation space and a metric in the $\mathbf{W}$-transformed space (see Appendix B).

Learning algorithm To determine the values of $\mathbf{W}$ using the training data we solve the following optimization problem:

$$\min \quad \sum_{(x,y)\in\mathcal{S}} \text{tr}\left((2\mathbf{I} - \mathbf{M}^F_{\pi_x,\pi_y})\mathbf{W}\right) \tag{3.16}$$

$$\text{s. t.} \quad \sum_{(x,y)\in\mathcal{D}} \text{tr}\left((2\mathbf{I} - \mathbf{M}^F_{\pi_x,\pi_y})\mathbf{W}\right) = 1 \tag{3.17}$$

$$\mathbf{W} \succeq 0 \tag{3.18}$$

Like for the weight vector the constraint (equality to 1) is arbitrary and guarantees that dissimilar arrays are distant from each other. This optimization is a semidefinite program (SDP) [85]. The objective function is a summation of traces of semi-definite matrices and so this is a convex optimization problem and there exists a global minimum solution. However, the matrix $\mathbf{W}$ is very large (m-by-m) and would require large amounts of training data for learning. Since such data is limited using a full rank matrix will likely lead to overfitting. Instead we seek a low-rank approximation of $\mathbf{W}$. Let $\mathbf{Z}$ be the rank-k approximation of $\mathbf{W}$: $\mathbf{W} \approx \mathbf{Z} = \mathbf{U}\mathbf{U}^{\mathrm{T}}$, where $\mathbf{U} \in \mathcal{R}^{n\times k}$. Given these changes the

optimization problem is:

$$\min \quad \mathrm{tr}(\mathbf{U}^{\mathrm{T}}\mathbf{U}_{\mathcal{S}}\mathbf{Y}) \tag{3.19}$$

$$\text{s.t.} \quad \mathrm{tr}(\mathbf{U}^{\mathrm{T}}\mathbf{Z}_{\mathcal{D}}\mathbf{U}) = 1 \tag{3.20}$$

with

$$\mathbf{Z}_{\mathcal{S}} = \sum_{(x,y)\in\mathcal{S}} (\mathbf{M}_{\pi_x} - \mathbf{M}_{\pi_y})^{\mathrm{T}}(\mathbf{M}_{\pi_x} - \mathbf{M}_{\pi_y})$$

$$\mathbf{Z}_{\mathcal{D}} = \sum_{(x,y)\in\mathcal{D}} (\mathbf{M}_{\pi_x} - \mathbf{M}_{\pi_y})^{\mathrm{T}}(\mathbf{M}_{\pi_x} - \mathbf{M}_{\pi_y})$$

Regularization An additional constraint that is useful for controlling overfitting is to regularize the solution. In our case, since nearby locations can be affected by small amounts of noise a reasonable regularization policy is to require that the $\mathbf{W}$ matrix is smooth. To achieve this we add linear inequality constraints to enforce that column-adjacent entries in $\mathbf{U}$ differ by at most $\delta > 0$:

$$\left|u_{ij} - u_{(i+1)j}\right| \leq \delta, \qquad \forall 1 \leq i < m, 1 \leq j \leq k$$

We solve this optimization by using the augmented Lagrangian approach. Similarly, we can incorporate the smoothness constraints to the Lagrangian. See [85] for a detailed discussion on the augmented Lagrangian method.

3.3 Results: Testing distance metrics on data from human and mouse tissues

We first used a training dataset from human and mouse tissues to learn parameters for our distance functions and to test the different methods on a dataset for which the correct answer is known. We next downloaded a large number of microarray expression datasets from GEO and applied our distance function to select pairs of experiments that are similar. For this section we consider the cross-species analysis between human (*Homo sapiens*) and mouse (*Mus musculus*) biological samples. We obtained the list of 16,376 human and mouse orthologs from Inparanoid[4]. For evaluation and comparisons of all metrics discussed in this chapter, we used an expression dataset, which we call 'Toronto dataset', consisting of expression profiles for 26 human tissues and their corresponding tissues in mice [86]. These 26 tissues pairs were profiled using species specific custom arrays. For each tissue, we had one human and one mouse arrays, which were processed and normalized by the authors of [86]. See Table B.1 for the list of tissues. We computed the log2 fold changes by using the means of expression values in all tissues as the controls.

3.3.1 Additional methods

In addition to the NMF method [80] and the distance metrics discussed in Section 3.2, we tested the Pearson correlation, which differs from the Spearman's rank correlation

[4] `http://inparanoid.sbc.su.se`

by using the expression values instead of the ranking of genes. We also examine the performance of a distance function DiffExpr, which was computed as follows. Let $de(x_i)$ denote a function that assigns the value 1 to the top x% expressed orthologs, -1 to the bottom x% expressed orthologs, and 0 otherwise. We define DiffExpr as follows:

$$\text{DiffExpr}(\mathbf{x}, \mathbf{y}) = \sum_i |de(x_i) - de(y_i)| \tag{3.21}$$

In essence, DiffExpr computes the difference between the sets of (non)-differentially expressed orthologs in two different microarrays.

3.3.2 Experimental setup

Gene Variance While the methods described in Section 3.2 can work for any number of orthologs, the larger the number the more data we would need to fit the weight vector and matrix methods. Since all our expression levels were log ratios to a reference data (see below) we have excluded from the analysis genes that did not vary much *within* each species. We selected the top 500 most varying orthologs for further analysis. We note two things. First, methods that are not affected by over-fitting (in our case Spearman's correlation and TBR) were also tested using all orthologs with results very similar to the results obtained from the 500 gene list. Second, while such a selection favors genes with high variance across a large number of experiments, at no stage in the selection have we considered the agreement between the actual levels of orthologous genes in specific experiments.

We used 2 fold cross-validation with 10 random permutations of tissues to compare the performance of the NMF method [80] and the five different distance metrics discussed above. For Pearson correlation, we select the varying 500 genes based on their expression values. For NMF we used the R code provided by the authors which also performs model selection to limit the number of metagenes [87]. The human samples were used to discover the metagenes and the mouse orthologs of these genes were used for the mouse metagenes. For training of the methods, we use the set of similar tissues as the positive set and all the remaining pairs as negative examples. Using parameters learned in the training phase we rank all test pairs by their distance and plot a Precision-Recall (PR) curve for all methods. Since the data set is highly skewed (i.e. there are many more negative than positive pairs), PR curves provide a more informative picture of the metrics' performance than the Receiver Operator Characteristic (ROC) curves [88].

3.3.3 Comparison of cross species comparison metrics

Different rank values of the weight matrix method We assessed the performance of the weight matrix method with the rank values of 2,3 and 4 in Figure 3.1. Both ranks 2 and 4 do not improve the overall success. We also have tested using a different number of negative examples for each array in the training set (since the number of positive examples is only 1 it is hard to change that number). For this test we used 5 negative examples (in the original analysis we used 12). As can be seen in Figure 3.1, this change did not affect the results much and the PR curve for such setting is very close to the original PR curve.

DiffExpr with different values of x We tried different values of x to determine differentially expressed orthologs in DiffExpr. Figure 3.2 depicts the performance of DiffExpr. $x = 1\%$ yields the best result and is used to compare with other methods.

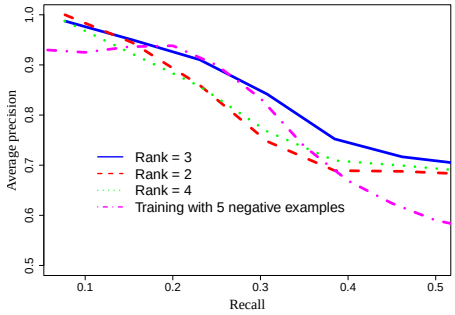


Figure 3.1: PR curves of Matrix Weight metrics with different rank values.

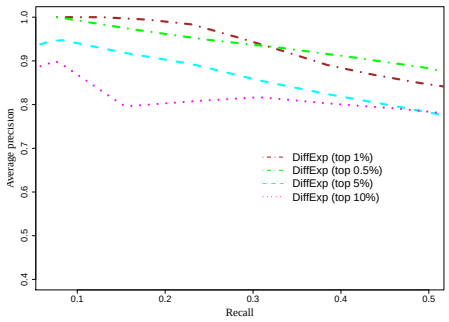


Figure 3.2: PR curves of DiffExpr with different values of x.

DiffExpr is the best method for this dataset. Since this dataset contains one-to-one corresponding tissues between human and mouse, the list of differentially expressed orthologs highly overlaps between human and mouse samples for each tissue. DiffExpr achieve high recall and precision by exploiting this particular structure in the data. As can be seen in Figure 3.3 other methods (except for Spearman's rank correlation) achieved a very high precision to begin with (80% and higher). However, this precision level drops and when reaching 20% recall only the weight matrix and DiffExpr method achieve a precision that is higher than 90%. As for the other methods we believe that Spearman's rank correlation performs worse than Pearson correlation because the test dataset is well normalized so nonparametric methods lose statistical power. For NMF, the fact that it is unsupervised and does not use information from the query species to construct the components likely led to its weaker performance. Figure 3.4 presents the residual weights $(w_{ii} + w_{jj})/2 - w_{ij}$ which are the penalties for differences in a ranked pair as shown in (3.14). High (red) values indicate bigger penalty while lower (blue) values indicate that the penalty is smaller. Interestingly the method seems to focus more on the repressed genes and puts a higher weight on genes that move from being unexpressed to being expressed at a high or medium level. We also observed similar trend in the learned weight vector $\mathbf{w}$ of the Vector method (Figure 3.5).

3.3.4 Novartis dataset

We have repeated the above analysis (comparison of methods) using another, independent, human-mouse tissue dataset, which we term the 'Novartis dataset', from [89].

For an additional evaluation of all metrics discussed in this paper, we used a second human-mouse expression dataset consisting of 79 human and 61 mouse tissues from [89] (note that some are repeats). In cases where the cell types differed between human

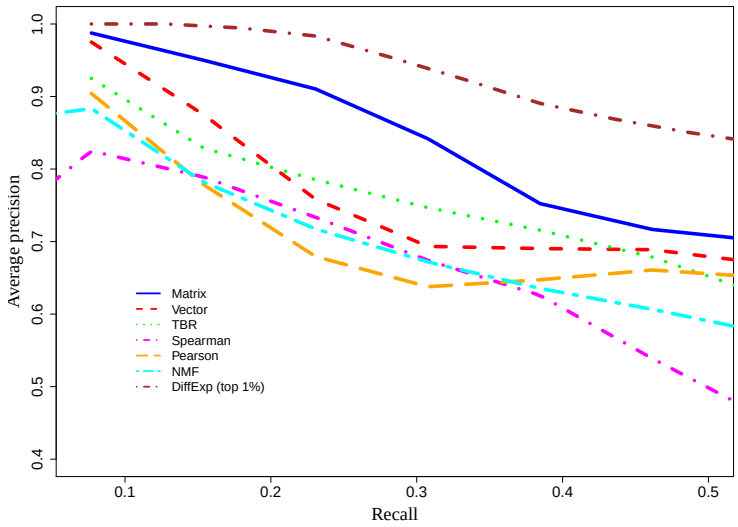


Figure 3.3: Comparison of different metrics using human-mouse tissues. PR curves of Spearman's rank correlation, TBR, NMF, Vector and Matrix Weight metrics.

and mouse we have assigned each human tissue sample to at most three mouse samples based on a mapping by a pathologist (Oltvai). The assignment of human tissues to mouse tissues are based on the following criteria (see Website for complete assignments):

1. Same organs, cell types, and developmental stages.

2. Spatially closer structures within an organ.

3. Insights that are not necessarily evident from anatomy, e.g, the ontogenic similarity of brown adipose tissue and muscle.

We next used 4 fold cross-validation with 4 random permutations of the tissues to compare the performance of the NMF method [80] and the four different distance matrices discussed above. The results presented used an approximation matrix with rank 3.

The overall success for this dataset is lower than for the Toronto dataset. This agrees with the initial analysis of this data that indicated a large deviation between human and mouse expression data for some of the tissues [89]. Due to this main reason, DiffExpr does very poorly and has the lowest recall and precision. We tried different x values for DiffExpr (Figure 3.7).

As can be seen in Figure 3.6, the weight matrix method achieves a high precision (65%) for a much larger recall (10%). As discussed previously, the reason NMF does not perform well on this dataset is likely related to the fact that it is unsupervised and does not use information from the query species to construct the components. Figure 3.8 presents the residual weights, which are the penalties for differences in a ranked pair as shown

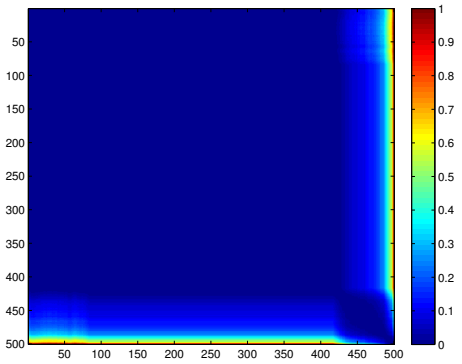


Figure 3.4: The penalty matrix between ranks as shown in (3.14), learned from the human and mouse tissues data.

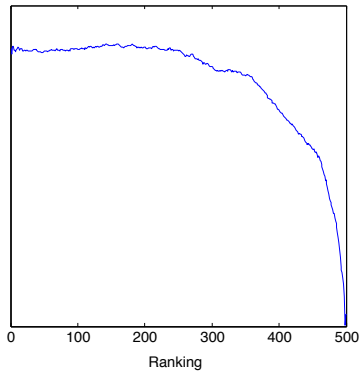


Figure 3.5: The weight vector **w** learned from the human and mouse tissues data.

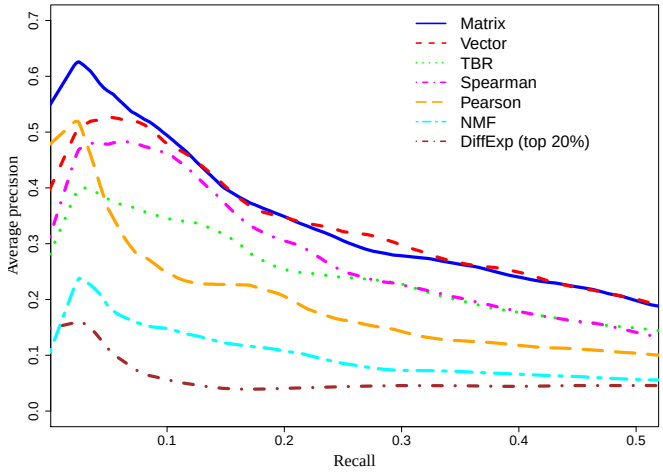


Figure 3.6: PR curves of Spearman's rank correlation, TBR, NMF, Vector and Matrix Weight metrics.

in (3.14). We note the similarity with the learnt matrix in Figure 3.4 in putting a higher weight on genes that move from being repressed although the penalty is smaller. Thus, the overall weighting seems to be dataset and platform independent.

Weigh matrix method is the best overall method In application to large, heterogenous, datasets the assumption of normalization across the datasets is less likely. We need to use methods that are robust against difference in normalization techniques, thus we need a method that works well in both the "Toronto" and "Novartis" datasets. Since

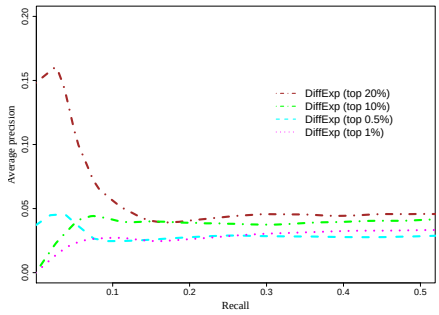


Figure 3.7: PR curves of DiffExpr with different values of x (Novartis dataset).

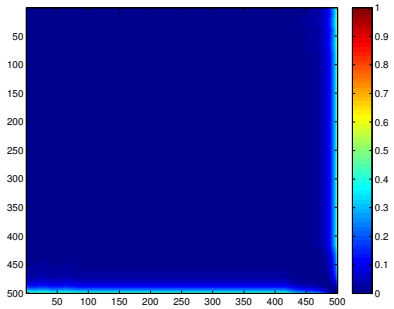


Figure 3.8: The penalty matrix between ranks learned from the Novartis dataset.

there are hundreds of thousands of expression experiments in GEO, precision is more important than recall for our goals. At these high precision rates the weight matrix method dominates the other methods we have considered. The weigh matrix method is most appropriate for querying large cross-species gene expression databases and thus we used it in all subsequent analyses.

3.4 Results: Identifying similar experiments in GEO

The previous section shows that our weight matrix performs better than standard metrics on the Toronto and Novartis datasets and moreover can get a very high precision for the recall value of 20%. Our goal is to apply this new metric for retrieving cross-species similar pairs of microarray experiments in a large dataset.

Data Collection We downloaded 715 human and 769 mouse datasets from GEO and used GDS data and metadata to identify control samples for each dataset (Website). Such samples are important for properly normalizing and transforming the data so that all data used is log2 ratio of the response sample to its control. We excluded from the analysis all datasets for which we could not positively identify the control sample leaving us with 3416 human and 2991 mouse microarrays from 535 human and 641 mouse datasets.

Identification of associated pairs of microarrays

We used the weight matrix trained using the full set of human-mouse tissue pairs. We used the results of Figure 3.3 to select a similarity cutoff corresponding to the cutoff that led to 95% precision and 10% recall. Using this cutoff we ended up with 301,453 pairs of microarrays whose distances are smaller than the cutoff which is roughly 3% of all pairs tested. These pairs are from 14493 dataset pairs (many array pairs are from the same pair of human and mouse datasets).

We also looked at the distribution of scores under the null hypothesis (since more than 95% of microarray pairs are not similar, this can be done by selecting random human-

mouse array pairs) and determined that the p-value for the null hypothesis is uniformly distributed, as expected. As a sanity check for our results we also computed the Pearson correlation across the pairs determined to be significant by our method for all human and mouse orthologs that were not part of the 500 genes we used for learning the parameters. Figure 3.9 shows the histogram of this correlation and the histogram of the correlation for the same set of genes in a randomly selected set of 301,453 microarray pairs. As can be seen the selected experiments are indeed more similar for many of the orthologs when compared to random selected pairs indicating that our method can identify correlated array pairs without using the experiment description.

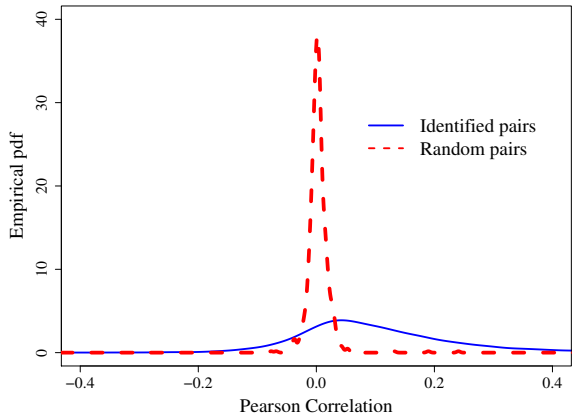


Figure 3.9: Blue curve: Correlation of orthologs not used for training in a random sample of 301,453 microarray pairs from human and mouse. Red curve: Correlation of orthologs not used for training in the set of microarray pairs selected by our method.

Description and dataset analysis

The list of pairs derived by our method allows us to address many questions. We first asked what conditions / organs / tissues are the most similar between human and mouse in terms of expression. We used the titles provided in the metadata section of the GDS to identify common words that are significantly over-represented in the microarray pairs we extracted. For each pair of similar experiments, a word that appears in both titles could provide information about the relationship between the pair. For each word we have also computed the number of times it appeared in a title for all microarrays used from each species and the expected number of times it should have appeared in the pairs we selected. Using the hypergeometric distribution we computed the overrepresentation P-value for each word. Table 3.1 presents the results of the analysis of over represented words in matched titles. As can be seen some organs and tissue types are much more represented than others. For example, brain, muscles and blood appear to have similar expression patterns between the two species. Certain conditions are also

overrepresented, most notably immune response. Several words are associated with experiments related to such response including different types of cells participating in the response (macrophages, dendritic, cd8). In contrast, cancer, one of the most common words in the human studies (roughly 10% of human datasets contained cancer in the title) was not overrepresented supporting recent results that most mice are not an ideal model system for at least some types of cancer [90, 91]. We repeated this analysis using the abstracts provided instead of the titles leading to similar results (see Website for full results). We have also looked beyond pairwise similarities and identified entire datasets (GDS files) that contained several similar pairs of arrays between human and mouse. An expert pathologist (Oltvai) manually inspected the top 100 matched datasets and determined that over 80% of them make biological sense (see Table B.3). Many of the datasets identified as similar contained experiments for the same tissue (most notably muscle, but also blood and brain). However, some of the matches were less obvious. Fibrosis is a chronic progressive and often lethal lung disease. One of the top 50 matches in our results was between a human dataset titled non-diseased lung tissue (GDS1673) and the mouse dataset titled Pulmonary fibrosis(GDS251). However, upon a closer inspection of the mouse dataset it can be seen that it compares two mouse strains treated with bleomycin. One is determined to be susceptible to fibrosis (C57BL6/J) whereas the other is determined to be resistant (BALB/c). When looking at the similarities computed by our method it can be seen that the vast majority of the top 100 matches are for the BALB/c strains. Thus, our cross-species comparisons can be used to identify cases in which similar pathways are activated even though the conditions may be different.

Quarrying GEO to identify cycling mouse genes

To demonstrate the utility of our method for quarrying large cross-species databases like GEO we used a set of 50 known human cycling genes extracted from [92]. For each of these genes we used all 301,453 microarray pairs determined to be similar to identify the set of similarly expressed mouse genes using Spearman correlations (regardless of their sequence similarity). We retrieved the top 10 most similar mouse genes for each query human gene resulting in a set of 206 genes. Note that the database we used contained a diverse set of experiments and, while a few may have been focused on cell cycle studies the vast majority were not. Importantly, our analysis here did not rely on any specific cell cycle time series dataset.

We used STEM [93] to determine significant GO categories associated with this list of mouse genes. As can be seen in Table 3.2, all top categories that are enriched for this set are related to cell cycle (including cell cycle itself). The set of mouse genes contains orthologs of the original set of human genes including CDC2A, a cell division control protein and CCNB1, an essential component of the cell cycle regulatory machinery. The list also contains many known mouse cell cycle genes with no homologs on the human list. These include members of a highly conserved complex which is essential for the initiation of DNA replication (ORC1L and ORC6L) and PRIM1 and PRIM2 which are involved in chromosomal replication during cell cycle. See Website for complete list. These results highlight the potential use of our method for identifying functionally related genes across species.

Rank	P-value	Word	#Pairs	
			Identified	Expected
1	7.14429e-13	MUSCLE	121	28.46752
2	7.39409e-13	DENDRITIC	24	2.13506
3	1.76946e-11	SKELETAL	42	12.12506
4	3.12418e-11	MACROPHAGE	18	2.21414
5	1.89634e-08	ERYTHROID	6	0.15815
6	2.52933e-08	OBESITY	9	0.63261
7	8.35063e-08	HEMATOPOIETIC	13	1.84512
8	2.36749e-07	BRAIN	19	4.42828
9	1.52768e-06	CD8+	5	0.18451
10	1.67619e-06	CARDIAC	6	0.34266
11	1.45374e-05	STEM	43	20.87618
12	2.02795e-05	HAIR	5	0.31631
13	9.19217e-05	FIBROBLASTS	12	3.08398
14	2.04560e-04	AIRWAY	7	1.15979

Table 3.1: Top 14 words identified in titles of pairs determined to be similar. #Pairs Identified is the number of time this pair was observed. #Pairs Expected is the number of time expected based on single species occurrences. The P-value is computed using the hypergeometric distribution.

		# Genes			
Rank	Category Name	Assigned	Expected	P	P adj
1	cell cycle	39.0	9.1	8.5E-15	<0.001
2	cell division	26.0	4.5	5.5E-13	<0.001
3	cell cycle phase	26.0	4.7	1.6E-12	<0.001
4	M phase	24.0	4.2	4.8E-12	<0.001
5	cell cycle process	26.0	5.5	4.6E-11	<0.001
6	mitotic cell cycle	21.0	3.8	2.4E-10	<0.001
7	mitosis	17.0	2.9	6.7E-9	<0.001
8	nuclear division	17.0	2.9	5.8E-9	<0.001
9	M phase of mitotic cycle	17.0	3.0	6.7E-9	<0.001

Table 3.2: GO enrichment analysis for mouse genes using STEM.

3.5 Conclusions and future work

The growth of microarray databases opens the door to applications that can simultaneously query sequence and expression databases to identify both static and dynamic matches. However, these methods would require a set of matching expression datasets in the species being queried. Such matches are hard to come by. It is rare to find the exact same experiment (condition, time, tissues etc.) in multiple species. To allow the use of these databases we looked at several different distance metrics between expression experiments. We defined a new distance function which utilizes the ranking of orthologs in both species. Our method uses a training dataset to learn weights for differences in rankings between the species and these differences are then summed up to determine the similarity between the two experiments. Testing this method on a training dataset of known similar pairs showed that it indeed improves upon other distance measures and that it can achieve high precision.

We used our new distance function to retrieve similar experiment pairs from GEO. The set of experiments identified by our method allowed us to look at questions regarding the conditions and tissues that activate similar expression patterns in human and mouse and to find a set of cycling mouse genes based on a set of known human cycling genes. Many of these mouse genes are known to be cycling and the rest of the genes identified are candidates for further study into their role in the cell cycle.

Our method attempts to learn a new distance function for permutations based on training data. There has been recent work in Machine Learning on trying to learn new distance function for feature vectors [94], though we are not aware of any work so far that attempted to learn such methods for permutations. A number of the methods developed for feature vectors were later kernelized allowing for much faster computations. It would be interesting to see if the Matrix weight method discussed here can also be kernelized. We have primarily relied on one-to-one orthology matches for computing the distance between pairs of experiments. Since many orthology assignments are many to one or many-to-many, methods that can utilize such information may be able to improve upon the results suggested in this chapter. Our overall goal is to compile a large set of expression pairs that can be used for querying human and mouse genes. As we noted in the introduction our method can also help in distinguishing between orthologs and homologs by looking for genes with similar sequence that are also co-expressed in the set of similar experiments. We would also like to extend this work to other species and we are looking for training data for these species.

4 Cross-species Expression Analysis with Latent Matching of Genes[1]

While useful, cross-species analysis of expression data is challenging. In addition to the regular issues with expression data (noise, missing values, etc.) when comparing expression levels across species researchers need to match genes across species. For most genes the correct match in another species (known as ortholog) is not known. In developing the method in the previous chapter, we have primarily relied on one-to-one orthology matches for computing the distance between pairs of experiments. In this chapter, we relax the one-to-one orthology requirement by incorporating a latent matching component, which can determine probabilistically the matchings of genes in two species, into a unified model to assign genes into clusters.

4.1 Introduction

A number of methods have been suggested to solve the matching problem. The first set of methods is based on a one-to-one deterministic assignment by relying on top sequence matches. Such an assignment can be used to concatenate the expression vectors for matched genes across species and then cluster the resulting vectors. For example, Stuart et al. [96] constructed "metagenes" consisting of top sequence matches from four species. These were used to cluster the data from multiple species to identify conserved and divergent patterns. Bergmann et al. [97] defined one of the species (species A) as a reference and first clustered genes in A. They then used matched genes in the second species (B) as starting points for clustering genes in B. When the clustering algorithm converges in B, genes that remain in the cluster are considered "core" whereas genes that are removed are "divergent". Quon et al. [98] used a mixture of Gaussians model, which takes as input the expression data of orthologous genes and a phylogenetic tree connecting the species, to reconstruct the expression profiles as well as detecting divergent links in the phylogeny. The second set of methods allowed for soft matches but was either limited to analyzing binary or discrete data with very few labels. For example, Lu et al. combined experiments from multiple species by using Markov Random Fields [99] and Gaussian Random Fields [100] in which edges represent sequence similarity and potential functions constrain similar genes across species to have a similar expression pattern.

While both approaches led to successful applications, they suffer from drawbacks that limit their use in practice. In many cases the top sequence match is not the correct ortholog and a deterministic assignment may lead to wrong conclusions about the conservation of genes. Methods that have used soft assignments were limited to summarization of the data (up or down regulated) and could not utilize more complex profiles. Here we present a new method that uses soft assignments to allow comparison and clustering across species of arbitrary expression data without requiring prior knowledge on the phylogeny. Our method takes as input expression datasets in two species and a prior on

[1] This work is published in [95].

matches between homologous genes in these species (derived from sequence data). The method simultaneously clusters the expression values for both species while computing a posterior for the assignment of orthologs for genes. We use Dirichlet Process model to automatically detect the number of clusters.

We have tested our method on simulated and immune response data. In both cases the algorithm was able to find correct matches and to improve upon methods that used a deterministic assignment. While the method was developed for, and applied to, biological data, it is general and can be used to address other problems including matchings of captions to images (see Section 4.5).

4.2 Problem definition

In this section, we first describe in details the cross-species analysis problem for gene expression data. Next, we formalize this as a general clustering and matching problem for cases in which the matches are not known in advance.

Using microarrays or new sequencing techniques researchers can monitor the expression levels of genes under certain conditions or at specific time points. For each such measurement we obtain a vector whose elements are the expression values for all genes (there are usually thousands of entries in each vector). We assume that the input consists of microarray experiments from two species and each species has a different set of genes. While the exact matches between genes in both species are not known for most genes, we have a prior for gene pairs (one from each species) which is derived from sequence data [101]. Our goal is to simultaneously cluster the genes in both species. Such clustering can identify coherent and divergent responses between the species. In addition, we would like to infer for each gene in one species whether there exists a homolog that is similarly expressed in the other species and if so, who.

The problem can also be formalized more generally in the following way. Denote by $\mathbf{X} = (\mathbf{x}_1, \ldots, \mathbf{x}_{n_x})^{\mathrm{T}}$ and $\mathbf{Y} = (\mathbf{y}_1, \ldots, \mathbf{y}_{n_y})^{\mathrm{T}}$ the datasets of samples from two different experiment settings, where $\mathbf{x}_i \in \Re^{p_x}$ and $\mathbf{y}_j \in \Re^{p_y}$. In addition, let $\boldsymbol{\mathcal{M}}$ be a sparse non-negative n_x-by-n_y matrix that encodes prior information regarding the matching of samples in $\mathbf{X}$ and $\mathbf{Y}$. We define the match probability between $\mathbf{x}_i$ and $\mathbf{y}_j$ as follows:

$$p(\mathbf{x}_i \text{ and } \mathbf{y}_j \text{ are matched}) = \frac{\mathcal{M}_{ij}}{N_i} = \pi_{ij} \tag{4.1}$$

$$p(\mathbf{x}_i \text{ is not matched}) = \frac{1}{N_i} = \pi_{i0} \tag{4.2}$$

where $N_i = 1 + \sum_{j=1}^{n_y} \mathcal{M}_{ij}$.

π_{i0} is the prior probability that $\mathbf{x}_i$ is not matched to any element in $\mathbf{Y}$. We use $\boldsymbol{\pi}_i$ to denote the vector $(\pi_{i0}, \ldots, \pi_{in_y})^{\mathrm{T}}$. Finally, let $m_i \in \{0, 1, \ldots, n_y\}$ be the latent matching variable. If $m_i = 1$ we say that x_i is matched to y_{m_i}. If $m_i = 0$ for we say that x_i has no match in $\mathbf{y}$. Our goal is to infer both, the latent variables m_i's and cluster membership for pairs of samples $(\mathbf{x}_i, \mathbf{y}_{m_i})$'s.

4.3 Model

Model selection is an important problem when analyzing real world data. Many clustering algorithms, including Gaussian mixture models, require as an input the number of clusters. In addition to domain knowledge, this model selection question can be addressed using cross validation. Bayesian nonparametric methods provide an alternative solution allowing the complexity of the model to grow based on the amount of available data. Under-fitting is addressed by the fact that the model allows for unbounded complexity while over-fitting is mitigated by the Bayesian assumption. We use this approach to develop a nonparametric model for clustering and matching cross-species expression data. Our model, termed Dirichlet Process Mixture Model with Latent Matchings (DPMMLM) extends the popular Dirichlet Process Mixture Model to cases where priors are provided to matchings between vectors to be clustered.

4.3.1 Dirichlet Process

Let G_0 a probability measure on a measurable space. We write $G \sim DP(\alpha, G_0)$ if G is a random probability measure drawn from a Dirichlet process (DP). The existence of the Dirichlet process was first proven by [102]. Furthermore, measures of G are discrete with probability one. This property can be seen from the explicit stick-breaking construction due to Sethuraman [103] as follows.

Let $(V_i)_{i=1}^{\infty}$ and $(\eta_i)_{i=1}^{\infty}$ be independent sequences of i.i.d random variables: $V_i \sim \text{Beta}(1, \alpha)$ and $\eta_i \sim G_0$. Then a random measure G defined as

$$\theta_i = V_i \prod_{j=1}^{i-1} (1 - V_j) \tag{4.3}$$

$$G = \sum_{i=1}^{\infty} \theta_i \delta_{\eta_i} \tag{4.4}$$

where δ_η is a probability measure concentrated at η, is a random probability measure distributed according to $\text{DP}(\alpha, G_0)$ as shown in [103] .

4.3.2 Dirichlet Process Mixture Model (DPMM)

Dirichlet process has been used as a nonparametric prior on the parameters of a mixture model. This model is referred to as Dirichlet Process Mixture Model. Let $\mathbf{z}$ be the mixture membership indicator variables for data variables $\mathbf{X}$. Using the stick-breaking construction in (4.3), the Dirichlet process mixture model is given by

$$G \sim \text{DP}(\alpha, G_0) \tag{4.5}$$

$$\mathbf{z}_i, \eta_i \mid G \sim G \tag{4.6}$$

$$\mathbf{x}_i \mid z_i, \eta_i \sim F(\eta_i) \tag{4.7}$$

where $F(\eta_i)$ denotes the distribution of the observation $\mathbf{x}_i$ given parameter η_i.

4.3.3 Dirichlet Process Mixture Model with Latent Matchings (DPMMLM)

In this section, we describe the new mixture model based on DP with latent variables for data matching between $\mathbf{X}$ and $\mathbf{Y}$. We use $F_X(\eta), F_Y(\eta)$ to denote the marginal distribution of X and Y respectively; and $F_{X|Y}(y, \eta)$ to denote the conditional distribution of X given Y. The parameter η is a random variable of the prior distribution $G_0(\eta \mid \lambda_0)$ with hyperparameter λ_0. Also, let z_i be the mixture membership of the sample pair (x_i, y_{m_i}). That is $z_i = k$ if the datapoint belongs to the kth cluster.

Our model is given by:

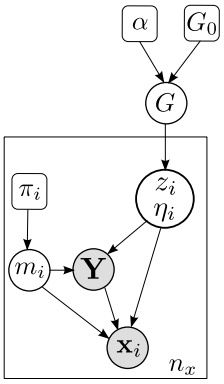


Figure 4.1: Graphical model of DPMMLM.

$$
\begin{aligned}
G &\sim \mathrm{DP}(\alpha, G_0) \\
z_i, \eta_i \mid G &\sim G \\
m_i \mid \boldsymbol{\pi}_i &\sim \mathrm{Discrete}(\boldsymbol{\pi}_i) \\
\mathbf{y}_{m_i} \mid m_i, z_i, \eta_i &\sim F_Y(\eta_i), \text{if } m_i > 0 \\
\mathbf{x}_i \mid m_i, z_i, \eta_i, \mathbf{Y} &\sim \begin{cases} F_{X|Y}(\mathbf{y}_{m_i}, \eta_i) & \text{if } m_i > 0 \\ F_X(\eta_i) & \text{otherwise} \end{cases}
\end{aligned}
\tag{4.8}
$$

The major difference between our model and a regular DPMM is the dependence of x_i on y if $m_i > 0$. In other words the assignment of x to a cluster depends on both, its own expression levels and the levels of the y component to which it is matched. If x is not matched to any y component then we resort to the marginal distribution F_X of the mixture.

4.3.4 Mean-field variational methods

For probabilistic models, mean-field variational methods [104, 105] provide a deterministic and bounded approximation to the intractable joint probability of observed and hidden variables. Briefly, given a model with observed variables x and hidden variables h, we would like to compute $\log p(x)$, which requires us to marginalize over all hidden variables h. Since $p(x, h)$ is often intractable, we can find a tractable probability $q(h)$ that gives the

best lower bound of $\log p(x)$ using Jensen 's inequality:

$$\begin{aligned}
\log p(x) &= \log \int_h p(x,h)\, dh && (4.9)\\
&= \log \int_h \frac{p(x,h)q(h)}{q(h)}\, dh && (4.10)\\
&\geq \int_h q(h) \log p(x,h) - q(h) \log q(h)\, dh && (4.11)\\
&= \mathrm{E}_q[\log p(x,h)] - \mathrm{E}_q[\log q(h)] && (4.12)
\end{aligned}$$

Maximizing this lower bound is equivalent to finding the distribution $q(h)$ that minimizes the KL divergence between $q(h)$ and $p(h \mid x)$. Hence, $q(h)$ is the best approximation model within the chosen parametric family.

4.3.5 Variational Inference for DPMMLM

Although the DP mixture model is an "infinite" mixture model, it is intractable to solve the optimization problem when allowing for infinitely many variables. We thus follow the truncation approach used in [106], and limit the number of cluster to K. When K is chosen to be large enough, the distribution is a drawn from the Dirichlet process [106]. To restrict the number of clusters to K, we set $V_K = 1$ and thus obtain $\theta_{i>K} = 0$ in (4.3).

For convenience, we use m_i^j to denote a binary variable indicating whether m_i equals j. That is $m_i^j = \mathbb{1}[m_i = j]$. Similarly, z_i^k indicates whether z_i equals k. The likelihood of the observed data is:

$$\begin{aligned}
p(\mathbf{X}, \mathbf{Y} \mid \alpha, \lambda_0) = \int_{\mathbf{m},\mathbf{z},\mathbf{v},\boldsymbol{\eta}} & p(\boldsymbol{\eta} \mid \lambda_0)\; p(\mathbf{v} \mid \alpha) \prod_{i=1}^{n_x} p(z_i \mid \mathbf{v}) \\
& \prod_{k=1}^{K} \Big\{ \left(\pi_{i0} f_X(\mathbf{x}_i \mid \eta_k)\right)^{m_i^0} \prod_{j=1}^{n_y} \left(\pi_{ij} f_{X|Y}(\mathbf{x}_i \mid \mathbf{y}_j, \eta_k) f_Y(\mathbf{y}_j \mid \eta_k)\right)^{m_i^j} \Big\}^{z_i^k} \qquad (4.13)
\end{aligned}$$

where $p(z_i \mid \mathbf{v}) = v_{z_i} \prod_{k=1}^{z_i - 1}(1 - v_k)$ and $\mathbf{v}$ is the stick breaking variables given in Section 4.3.1. The first part of (4.13) $p(\boldsymbol{\eta} \mid \lambda_0)\; p(\mathbf{v} \mid \alpha)$ is the likelihood of the model parameters and the second part is the likelihood of the assignments to clusters and matchings.

Following the variational inference framework for conjugate-exponential graphical models [107] we choose the distribution that factorizes over $\{m_i, z_i\}_{i=1,\ldots,n_x}$, $\{v_k\}_{k=1,\ldots,K}$ and $\{\eta_k\}_{k=1,\ldots,K-1}$ as follows:

$$q(\mathbf{m}, \mathbf{z}, \mathbf{v}, \boldsymbol{\eta}) = \prod_{i=1}^{n_x} \{ q_{\phi_i}(m_i) \prod_{j=0}^{n_y} q_{\theta_{ij}}(z_i)^{m_i^j} \} \prod_{k=1}^{K-1} q_{\gamma_k}(v_k) \prod_{k=1}^{K} q_{\lambda_k}(\eta_k) \qquad (4.14)$$

where $q_{\phi_i}(m_i)$ and $q_{\theta_{i,j}}(z_i)$ are multinomial distributions and $q_{\gamma_k}(v_k)$ are beta distributions. These distributions are conjugate distributions for the likelihood of the parameters in (4.13). $q_{\lambda_k}(\eta_k)$ requires special treatment due to the coupling of the marginal and conditional distributions in the likelihood. These issues are discussed in details in section 4.3.6.

Using this variational distribution we obtain a lower bound for the log likelihood:

$$\log p(\mathbf{X}, \mathbf{Y} \mid \alpha, \lambda_0) \;\geq\; \mathrm{E}[\log p(\boldsymbol{\eta} \mid \lambda_0)] + \mathrm{E}[\log p(\mathbf{v} \mid \alpha)] \\ + \sum_{i=1}^{n_x} \Big\{ \mathrm{E}[\log p(z_i \mid \mathbf{v})] + \sum_{j=0}^{n_y} \sum_{k=1}^{K} \mathrm{E}[m_i^j z_i^k](\log \pi_{ij} + \rho_{ijk}) \Big\} - \mathrm{E}[\log q(\mathbf{m}, \mathbf{z}, \mathbf{v}, \boldsymbol{\eta})] \quad (4.15)$$

where all expectations are with respect to the distribution $q(\mathbf{m}, \mathbf{z}, \mathbf{v}, \boldsymbol{\eta})$ and

$$\rho_{ijk} = \begin{cases} \mathrm{E}[\log f_{X|Y}(\mathbf{x}_i \mid \mathbf{y}_j, \eta_k)] + \mathrm{E}[\log f_Y(\mathbf{y}_j \mid \eta_k)] & \text{if } j > 0 \\ \mathrm{E}[\log f_X(\mathbf{x}_i \mid \eta_k)] & \text{if } j = 0 \end{cases}$$

To compute the terms in (4.15), we note that

$$\mathrm{E}[m_i^j z_{ik}] = \phi_{ij}\theta_{ijk} = \psi_{ijk}$$
$$\mathrm{E}[\log p(z_i \mid \mathbf{v})] = \sum_{k=1}^{K} q(z_i > k)\mathrm{E}[\log(1 - v_k)] + q(z_i = k)\mathrm{E}[\log v_k]$$

where $q(z_i > k) = \sum_{j=0}^{n_y} \sum_{t=k+1}^{K} \psi_{ijt}$ and $q(z_i = k) = \sum_{j=0}^{n_y} \psi_{ijk}$.

Coordinate ascent inference algorithm

The lower bound above can be optimized by a coordinate ascent algorithm. The update rules for all terms except for the $q_{\lambda_k}(\eta_k)$, are presented below. These are direct applications of the variational inference for conjugate-exponential graphical models [107]. We discuss the update rule for $q_{\lambda_k}(\eta_k)$ in section 4.3.6.

- Update for $q_{\gamma_k}(v_k)$:

$$\gamma_{k1} = 1 + \sum_{i=1}^{n_x} \sum_{j=0}^{n_y} \psi_{ijk} \quad (4.16)$$

$$\gamma_{k2} = \alpha + \sum_{i=1}^{n_x} \sum_{j=0}^{n_y} \sum_{t=k+1}^{K} \psi_{ijt} \quad (4.17)$$

- Update for $q_{\theta_{ij}}(z_i)$ and $q_{\phi_i}(m_i)$:

$$\theta_{ijk} \;\propto\; \exp\big(\rho_{ijk} + \sum_{k=1}^{k-1} \mathrm{E}[\log(1 - v_k)] + \mathrm{E}[\log v_k]\big) \quad (4.18)$$

$$\phi_{ij} \;\propto\; \exp\Big(\log \pi_{ij} + \sum_{k=1}^{K} \theta_{ijk}\big(\rho_{ijk} + \sum_{k=1}^{k-1} \mathrm{E}[\log(1 - v_k)] + \mathrm{E}[\log v_k]\big)\Big) \quad (4.19)$$

4.3.6 Application of the model to multivariate Gaussians

The previous sections described the model in a general terms. In the rest of this section, and in our experiments, we focus on data that is assumed to be distributed as a multivariate Gaussian with unknown mean and covariance matrix. The prior distribution G_0 is then given by the conjugate prior Gaussian-Wishart distribution. In a classical DP Gaussian Mixture Model with Gaussian-Wishart prior, the posterior distribution of the parameters could be computed analytically. Unfortunately, in our model, the coupling of the conditional and marginal distribution in the likelihood makes it difficult to derive analytical formulas for the posterior distribution. Note that if $(X, Y) \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ with $\boldsymbol{\mu} = (\boldsymbol{\mu}_X, \boldsymbol{\mu}_Y)$ and $\boldsymbol{\Sigma} = \begin{pmatrix} \boldsymbol{\Sigma}_X & \boldsymbol{\Sigma}_{XY} \\ \boldsymbol{\Sigma}_{YX} & \boldsymbol{\Sigma}_Y \end{pmatrix}$ then $X \sim \mathcal{N}(\boldsymbol{\mu}_X, \boldsymbol{\Sigma}_X)$, $Y \sim \mathcal{N}(\boldsymbol{\mu}_Y, \boldsymbol{\Sigma}_Y)$ and

$$X|Y = \mathbf{y} \sim \mathcal{N}(\boldsymbol{\mu}_X + \boldsymbol{\Sigma}_{XY}\boldsymbol{\Sigma}_Y^{-1}(\mathbf{y} - \boldsymbol{\mu}_Y), \boldsymbol{\Sigma}_X - \boldsymbol{\Sigma}_{XY}\boldsymbol{\Sigma}_Y^{-1}\boldsymbol{\Sigma}_{YX}). \tag{4.20}$$

Therefore, we introduce an approximation distribution for the datasets which decouples the marginal and conditional distributions as follows:

$$f_X(\mathbf{x} \mid \boldsymbol{\mu}_X, \boldsymbol{\Lambda}_X) = \mathcal{N}(\boldsymbol{\mu}_X, \boldsymbol{\Sigma} = \boldsymbol{\Lambda}_X^{-1}) \tag{4.21}$$

$$f_Y(\mathbf{y} \mid \boldsymbol{\mu}_Y, \boldsymbol{\Lambda}_Y) = \mathcal{N}(\boldsymbol{\mu}_Y, \boldsymbol{\Sigma} = \boldsymbol{\Lambda}_Y^{-1}) \tag{4.22}$$

$$f_{X|Y}(\mathbf{x} \mid \mathbf{y}, \mathbf{W}, \mathbf{b}, \boldsymbol{\mu}_X, \boldsymbol{\Lambda}_X) = \mathcal{N}(\boldsymbol{\mu}_X + \mathbf{b} - \mathbf{W}y, \boldsymbol{\Sigma} = \boldsymbol{\Lambda}_X^{-1}) \tag{4.23}$$

where $\mathbf{W}$ is a p_x-by-p_y projection matrix and $\boldsymbol{\Lambda}$ is the precision matrix. In this approximation, we assume that the covariance matrices of X and $X|Y$ are the same. In other words, the covariance of X is independent of Y. The matrix $\mathbf{W}$ models the linear correlation of X on Y, similar to $-\boldsymbol{\Sigma}_{XY}\boldsymbol{\Sigma}_Y^{-1}$ in (4.20).

The priors for $\boldsymbol{\mu}_X, \boldsymbol{\Lambda}_X$ and $\boldsymbol{\mu}_Y, \boldsymbol{\Lambda}_Y$ are given by Gaussian-Wishart(GW) distributions with hyper-parameters $\{\kappa_{X0}, \mathbf{m}_{X0}, \mathbf{S}_{X0}, \nu_{X0}\}$ and $\{\kappa_{Y0}, \mathbf{m}_{Y0}, \mathbf{S}_{Y0}, \nu_{Y0}\}$. A flat improper prior is given to $\mathbf{W}$ and b, $p_0(\mathbf{W}) = 1, p_0(\mathbf{b}) = 1$ for all $\mathbf{W}, \mathbf{b}$. These assumptions lead to decoupling of the marginal and conditional distributions. Therefore, the distribution $q_{\lambda_k}(\eta_k)$ can now be factorized into two GW distributions and a distribution of $\mathbf{W}$. To avoid over-cluttering symbols, we omit the subscript k of the specific cluster k.

$$q^*_{\lambda_k}(\eta_k) = \mathrm{GW}(\boldsymbol{\mu}_X, \boldsymbol{\Lambda}_X \mid \kappa_X, \mathbf{m}_X, \mathbf{S}_X, \nu_X)\ \mathrm{GW}(\boldsymbol{\mu}_Y, \boldsymbol{\Lambda}_Y \mid \kappa_Y, \mathbf{m}_Y, \mathbf{S}_Y, \nu_Y)\ g(\mathbf{W})\ g(\mathbf{b})$$

Posterior distribution of $\boldsymbol{\mu}_Y, \boldsymbol{\Lambda}_Y$ The update rules follow the standard posterior distribution of Gaussian-Wishart conjugate priors.

Posterior distribution of $\boldsymbol{\mu}_X, \boldsymbol{\Lambda}_X$ and $\mathbf{W}, \mathbf{b}$

Due to the coupling of $\boldsymbol{\mu}_X, \boldsymbol{\Lambda}_X$ with $\mathbf{W}$, we do a coordinate ascent procedure to find the optimal posterior distribution. We do a point estimation of $\mathbf{W}$ and $\mathbf{b}$. (The posterior distribution of $\mathbf{W}, b$ is a singleton discrete distribution g such that $g(\mathbf{W}^*) = 1, g(\mathbf{b}^*) = 1$.)

Update for posterior distribution of μ_X, Λ_X

$$\kappa_X = \kappa_{X0} + n_X \tag{4.24}$$

$$\mathbf{m}_X = \frac{1}{\kappa_X}(\kappa_{X0}\mathbf{m}_{X0} + n_X\bar{\mathbf{x}}) \tag{4.25}$$

$$\mathbf{S}_X^{-1} = \mathbf{S}_{X0}^{-1} + \mathbf{V}_X + \frac{\kappa_{X0}n_X}{\kappa_{X0}+n_X}(\bar{\mathbf{x}} - \mathbf{m}_{X0})(\bar{\mathbf{x}} - \mathbf{m}_{X0})^{\mathrm{T}} \tag{4.26}$$

$$\nu_X = \nu_{X0} + n_X \tag{4.27}$$

where

$$n_X = \sum_{n=1}^{n_x}\sum_{j=0}^{n_y}\psi_{ijk} \tag{4.28}$$

$$\bar{\mathbf{x}} = \frac{1}{n_X}\sum_{i=1}^{n_x}\left(\psi_{i0k}\mathbf{x}_i + \sum_{j=1}^{n_y}\psi_{ijk}(\mathbf{x}_i - \mathbf{b} + \mathbf{W}^*\mathbf{y}_j)\right) \tag{4.29}$$

$$\mathbf{V}_X = \sum_{i=1}^{n_x}\left\{\psi_{i0k}(\mathbf{x}_i - \bar{\mathbf{x}})(\mathbf{x}_i - \bar{\mathbf{x}})^{\mathrm{T}} + \sum_{j=1}^{n_y}\psi_{ijk}(\mathbf{x}_i - \mathbf{b} + \mathbf{W}^*\mathbf{y}_j - \bar{\mathbf{x}})(\mathbf{x}_i - \mathbf{b} + \mathbf{W}^*\mathbf{y}_j - \bar{\mathbf{x}})^{\mathrm{T}}\right\} \tag{4.30}$$

Update for $\mathbf{W}^*, \mathbf{b}^*$ We find $\mathbf{W}^*, \mathbf{b}^*$ that maximizes the log likelihood. Taking the derivative with respect to $\mathbf{W}^*$ and solving for $\mathbf{W}^*$, we get

$$\mathbf{W}^* = \Big(\sum_{i=1}^{n_x}\sum_{j=1}^{n_y}\psi_{ijk}(\mathbf{x}_i - \mathbf{m}_X - \mathbf{b})\mathbf{y}_j^{\mathrm{T}}\Big)\Big(\sum_{i=1}^{n_x}\sum_{j=1}^{n_y}\psi_{ijk}\mathbf{y}_j\mathbf{y}_j^{\mathrm{T}}\Big)^{-1}$$

$$\mathbf{b}^* = -\Big(\sum_{i=1}^{n_x}\sum_{j=1}^{n_y}\psi_{ijk}(\mathbf{x}_i - \mathbf{m}_X + \mathbf{W}^*\mathbf{y}_j)\Big) / \sum_{i=1}^{n_x}\sum_{j=1}^{n_y}\psi_{ijk}$$

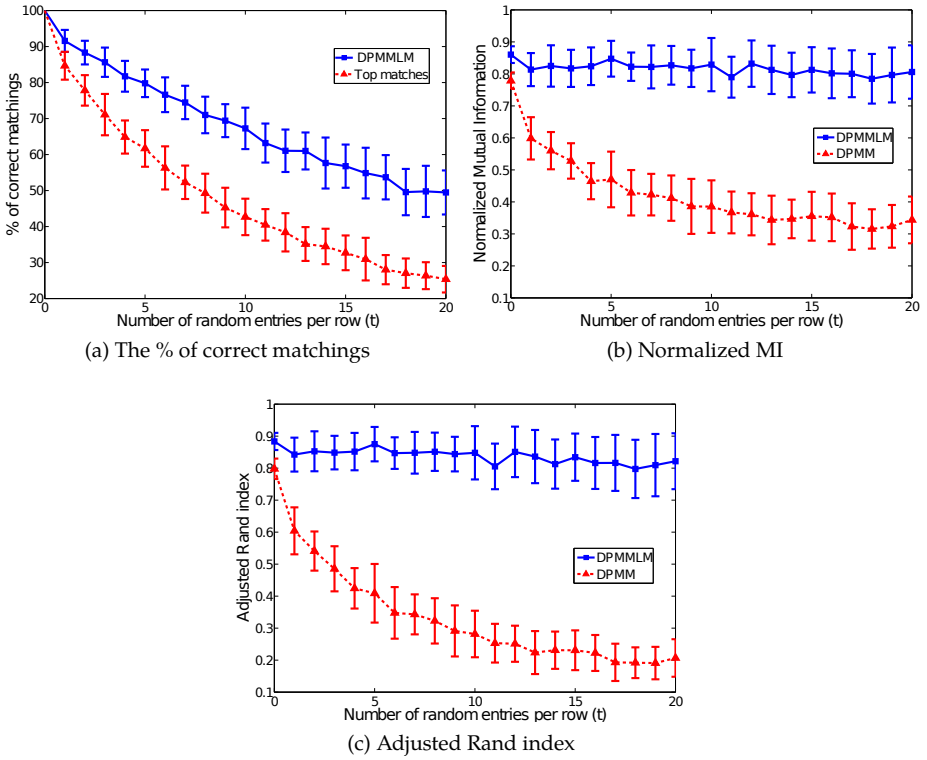
4.4 Experiments and Results

4.4.1 Simulated data

We demonstrate the performance of the model in identifying data matchings as well as cluster membership of datapoints using simulated data. To generate a simulated dataset, we sample 120 datapoints from a mixture of three 5-dimensional Gaussians with separation coefficient = 2 leading to well separated mixtures[2]. The covariance matrix was derived from the autocorrelation matrix for a first-order autoregressive process leading to highly dependent components ($\rho = 0.9$). From these samples, we use the first 3 dimensions to create 120 datapoints $\mathbf{x} = [x_1, \ldots, x_{120}]$. The last two dimensions of the first 100 datapoints are used to create $\mathbf{y} = [y_1, \ldots, y_{100}]$ (note that there are no matches for 20 points in x). Hence, the ground truth $\mathcal{M}$ matrix is a diagonal 120-by-100 matrix. We

[2]Following [108], a Gaussian mixture is c-separated if for each pair (i, j) of components, $\|m_i - m_j\|^2 \geq c^2 D \max(\lambda_i^{\max}, \lambda_j^{\max})$, where $\lambda^{\max}$ denotes the maximum eigenvalue of their covariance.

selected a large value for the diagonal entries ($\tau = 1000$) in order to place a strong prior for the correct matchings. Next, for $t = 0, \ldots, 20$, we randomly select t entries on each row of $\mathcal{M}$ and set them to $\frac{\tau}{2}r$, where $r \sim \chi^2_1$. We repeat the process 20 times for each t to compute the mean and standard deviation shown in Figure 4.2a and Figure 4.2b. We compare the performance of our model(DPMMLM) with a standard Dirichlet Process Mixture Model where each component in $\mathbf{x}$ is matched based on the highest prior: $\{(x_i, y_{j^*}) \mid i = 1, \ldots, 100 \text{ and } j^* = \text{argmax}_j \mathcal{M}(i, j)\}$ (DPMM). For all models, the truncation level (K) is set to 20 and α is 1. Figure 4.2a presents the percentage of correct matchings inferred by



(a) The % of correct matchings

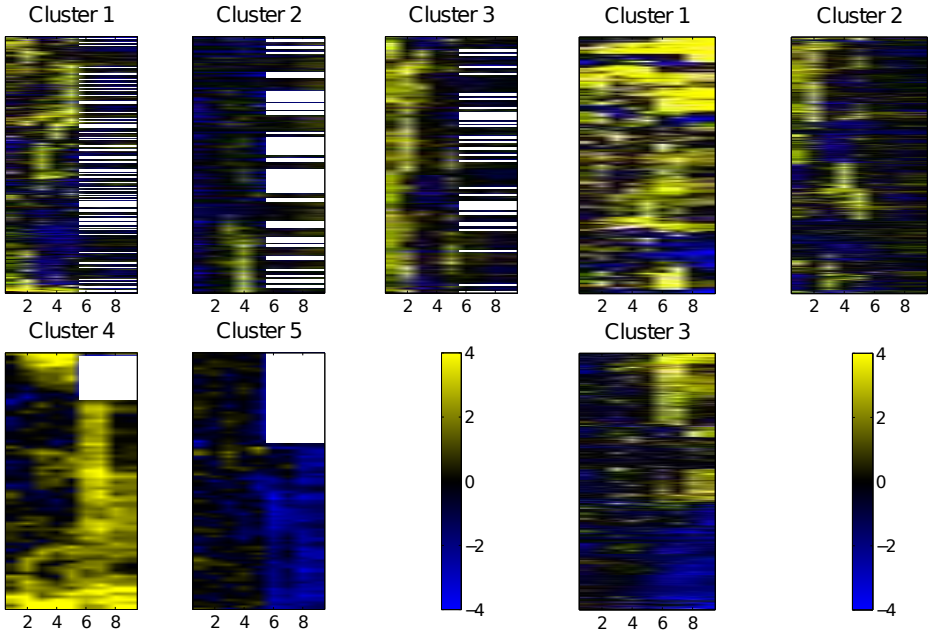
(b) Normalized MI

(c) Adjusted Rand index

Figure 4.2: Evaluation of the result on simulated data.

DPMMLM and the highest prior matching. For DPMMLM, a datapoint x_i is matched to the datapoint y_j with the largest posterior probability $\phi_{i,j}$. With the added noise, DPMMLM can still achieve an accuracy of 50% when the highest prior matching leads to only 25% accuracy. Figure 4.2b and 4.2c show the Normalized Mutual Information (NMI) and Adjusted Rand index [109] for the clusters inferred by the two models compared to the true clusters. As can be seen, while the percentage of correct matchings decreased with the added noise, DPMMLM still achieves high NMI of 0.8 and Adjusted Rand index of 0.92. In conclusion, by relying on matchings of points DPMMLM can still performs very well in terms of its ability to identify correct clusters even with the high noise levels.

4.4.2 Immune response dataset



(a) DPMMLM (b) DPMM

Figure 4.3: The heatmap for clusters inferred for the immune response dataset.

We compared human and mouse immune response datasets to identify similar and divergent genes. We selected two experiments that studied immune response to gram negative bacteria. The first was a time series of human response to Salmonella [110]. Cells were infected with Salmonella and were profiled at: 0.5h, 1h, 2h, 3h and 4h. The second looked at mouse response to Yersinia enterocolitica with and without treatment by IFN-γ [111]. We used BLASTN to compute the sequence similarity (bit-score) between all human and mouse genes. For each species we selected the most varying 500 genes and expanded the gene list to include all matched genes in the other species with a bit score greater than 75. This led to a set of 1476 human and 1967 mouse genes which we compared using our model. The $\mathcal{M}$ matrix is the bit scores between human and mouse genes thresholded at 75.

The resulting clusters are presented in Figure 4.3a. In that figure, the first five dimensions are human expression values and each gene in human is matched to the mouse gene with the highest posterior. Human genes which are not matched to any mouse gene in the cluster have a blank line on the mouse side of the figure. The algorithm identified five different clusters. Clusters 1, 4 and 5 display a similar expression pattern in human and mouse with genes either up or down regulated in response to the infection. Genes in cluster 2 differ between the two species being mostly down regulated in humans while

slightly upregulated in mouse. Human genes in cluster 3 also differ from their mouse orthologs. While they are strongly upregulated in humans, the corresponding mouse genes do not change much.

P value	Adj P	GO term description
2.86216e-10	<0.001	regulation of apoptosis
4.97408e-10	<0.001	regulation of cell death
7.82427e-10	<0.001	protein binding
4.14320e-10	<0.001	regulation of programmed cell death
4.49332e-09	<0.001	positive regulation of cellular process
4.77653e-09	<0.001	positive regulation of biological process
8.27313e-09	<0.001	response to chemical stimulus
1.17013e-07	0.001	cytoplasm
1.28299e-07	0.001	response to stress
2.20104e-07	0.001	cell proliferation
5.06685e-07	0.001	response to stimulus
6.15795e-07	0.001	negative regulation of biological process
7.70651e-07	0.001	cellular process
7.78266e-07	0.002	regulation of localization
1.09778e-06	0.002	response to organic substance
1.42704e-06	0.002	collagen metabolic process
1.91735e-06	0.003	negative regulation of cellular process
3.23244e-06	0.005	multicellular organismal macromolecule metabolic process
3.39901e-06	0.005	interspecies interaction
3.66178e-06	0.005	negative regulation of apoptosis

Table 4.1: The GO enrichment result for cluster 1 identified by DPMMLM.

We used the Gene Ontology (GO, www.geneontology.org) to calculate the enrichment of functional categories in each cluster based on the hypergeometric distribution. Genes in cluster 1 (Table 4.1) are associated with immune and stress responses. Interestingly the most significant category for this cluster is "regulation of apoptosis" (corrected p-value <0.001). Indeed, both Salmonella and Yersinia are known to induce apoptosis in host cells [112]. When clustering the two datasets independently the p-value for this category is greatly reduced indicating that accurate matchings can lead to better identification of core pathways (see Appendix). Cluster 4 contains the most coherent set of upregulated genes across the two species. One of top GO categories for this cluster is 'response to molecule of bacterial origin' (corrected p-value < 0.001) which is the most accurate description of the condition tested. See Appendix for complete GO tables of all clusters. In contrast to clusters in which mouse and human genes are similarly expressed, cluster 3 genes are strongly upregulated in human cells while not changing in mouse. This cluster is

enriched for ribosomal proteins (corrected p-value <0.001). This may indicate different strategies utilized by the bacteria in the two experiments. There are studies that show that pathogens can upregulate the synthesis of ribosomal genes (which are required for translation) [113] whereas other studies indicate that ribosomal genes may not change much, or may even be reduced, following infection [114]. The results of our analysis indicate that while following Salmonella infection in human cells ribosomal genes are upregulated, they are not activated following Yarsinia infection in mouse.

We have also analyzed the matchings obtained using sequence data alone (prior) and by combining sequence and expression data (posterior) using our method. The top posterior gene is the same as the top prior gene in most cases (905 of the 1476 human genes). However, there are several cases in which the prior and posterior differ. 293 human genes are not matched to any mouse gene in the cluster they are assigned to indicating that they are expressed in a species dependent manner. Additionally, for 278 human genes the top posterior and prior mouse gene differ. To test whether these differences inferred by the algorithm are biologically meaningful we compared our Dirichlet method to a method that uses deterministic assignments, as was done in the past. Using such assignments the algorithm identified only three clusters as shown in Figure 4.3b. Neither of these clusters looked homogenous across species.

4.5 Conclusions

We have developed a new model for simultaneously clustering and matching genes across species. The model uses a Dirichlet Process to infer the number of clusters. We developed an efficient variational inference method that scales to large datasets with almost 2000 datapoints. We have also demonstrated the power of our method on simulated data and immune response dataset. While the method was presented in the context of expression data it is general and can be used for other matching tasks in which a prior can be obtained. For example, when trying to determine a caption for images extracted from webpages a prior can be obtained by relying on the distance between the image and the text on the page. Next, clustering can be employed to utilize the abundance of images that are extracted and improve the matching outcome.

Part III

Using expression data to infer condition-specific miRNA targets

5

GroupMiR: Inferring Interaction Networks using the Indian Buffet Process [1]

MicroRNAs (miRNAs) are a family of small non-coding RNA molecules that regulate gene expression post-transcriptionally. These single-stranded RNAs, 19-25 nucleotides long, are initially transcribed as longer independent genes, or together with host genes (and then processed out of their introns). MiRNAs are now known to play a major role in development [116], various brain functions [117], and diseases [118]. Since their discovery, several hundred miRNAs were identified in each of several different species including mammals, worms, flies, and plants [7]. Most miRNAs target the genes they regulate by binding to the 3'-UTR of the target mRNAs (using complementary base-pairing) and recruiting additional machinery to either degrade these mRNAs or prevent them from being translated. The miRNA regulation is ubiquitous and a single miRNA can target hundreds and even thousands of genes. Since the effect of each miRNA on any single target is often limited, they often work cooperatively with multiple miRNAs targeting the same mRNA in a specific condition [119, 8]. They were shown to play an important role in a number of diseases including cancer, and determining the set of genes that are targeted by each miRNA is an important question when studying these diseases.

5.1 Introduction

Initial discovery of large sets of miRNAs relied heavily on sequence and conservation analysis [116], though recent advances in sequencing capacity are now allowing researchers to validate and identify additional miRNAs experimentally [120]. While these predictions are useful, due to the short length of miRNAs, they lead to many false positives and some false negatives [121, 122]. In addition to sequence information, it is now possible to obtain the expression levels of miRNAs and their predicted mRNA targets using microarrays. Since miRNAs inhibit their direct targets, integrating sequence and expression data can improve predictions regarding the interactions between miRNAs and their targets [122, 123, 124] This has led to several studies that isolated the miRNA target prediction task by integrating sequence, mRNA and miRNA expression data [123, 122, 124, 125]. Unlike sequence data, expression data is dynamic and condition-specific and thus provides useful clues about the set of active miRNAs and mRNAs. A number of methods, mostly based on (anti) correlation or regression analysis using the expression levels of miRNAs and predicted mRNA targets were suggested for this task [126, 127]. A representative example for this group is GenMiR++ [122], one of the first methods to integrate miRNA and mRNA expression profiles in a unified probabilistic model.

Expression data Consider making predictions using the expression profiles of M messenger RNA (mRNA) transcripts and R miRNA transcript across P samples. Let $\mathbf{X} =$

[1] This work is published in [115].

$(\mathbf{x}_1, \ldots, \mathbf{x}_M)^{\mathrm{T}}$, where each row vector x_i is the expression profile of miRNA i in all samples. Similarly, let $\mathbf{Y} = (\mathbf{y}_1, \ldots, \mathbf{y}_R)^{\mathrm{T}}$ represent the expression profiles of R mRNAs.

GenMiR++ Given a set of putative miRNA-mRNA interactions $\mathbf{C}$ ($c_{kg} = 1$ if miRNA k is predicted to target mRNA g), GenMIR++ employs a generative model in which each miRNA expression profile is used to explain the down-regulation of the expression of its mRNA targets. The model depends on nuisance variables $\{\mathbf{\Lambda}, \mathbf{\Gamma}\}$ and parameters $\Theta = \{\boldsymbol{\mu}, \mathbf{\Sigma}, \pi, \alpha\}$ and σ^2. The variables $\mathbf{\Lambda} = \{\lambda_1, \lambda_2, \ldots, \lambda_M\} > 0$ is a vector of down regulation effect of miRNAs and $\mathbf{\Gamma} = \mathrm{diag}(\gamma_1, \ldots, \gamma_P)$ is a diagonal matrix of positive sample-scaling parameters that considers the normalization difference between different sample. The most important variable $\mathbf{S}$ is a M-by-R binary matrix where $s_{kg} = 1$ means mRNA g is a target of miRNA k. The generative model specifies the relationship between expression profiles $\mathbf{X}$ and $\mathbf{Y}$ as follows [128]:

$$p(\mathbf{y}_g|\mathbf{Y}, \mathbf{S}, \mathbf{\Gamma}, \mathbf{\Lambda}, \Theta) = \mathcal{N}(\mathbf{x}_g; \boldsymbol{\mu} - \sum_k \lambda_k s_{kg} \mathbf{\Gamma} \mathbf{x}_k, \mathbf{\Sigma}) \tag{5.1}$$

$$p(\mathbf{S}|\mathbf{C}, \Theta) = \prod_{(k,g)} p(s_{kg}|\mathbf{C}, \Theta) = \prod_{(k,g)|c_{kg}=0} (1 - s_{gk}) \prod_{(k,g)|c_{kg}=1} \pi^{s_{kg}} (1 - \pi)^{(1-s_{kg})} \tag{5.2}$$

The learning and inference was done by a Variational Bayesian technique approximating the posterior distribution of $\mathbf{S}$ and the optimal values of Θ. The target prediction is made based on the posterior over $\mathbf{S}$ for each putative target in the set $\mathbf{C}$.

While methods utilizing expression data improved upon methods that only used sequence data, they often treated each target mRNA in isolation. In contrast, it has now been shown that each miRNA often targets hundreds of genes, and that miRNAs often work in groups to achieve a larger impact [129]. Thus, rather than trying to infer a separate regression model for each mRNA, we proposed GroupMiR, a probabilistic model to infer a joint regression model for a cluster of mRNAs and the set of miRNAs that regulate them. Such a model would provide statistical confidence (since it combines several observations) while adhering more closely to the underlying biology. In addition to inferring the interactions in the dataset, such a model would also provide a grouping for genes and miRNAs which can be used to improve function prediction.

We present GroupMiR in the following sections as follows. First, we derive a distribution on infinite binary matrices starting with a finite model and taking the limit as the number of features goes to infinity. Second, we apply this distribution to the miRNA target prediction problem using a Gaussian additive model, completing the description of GroupMiR.

5.2 Interaction model

Let z_{ik} denote the (i, k) entry of a matrix $\mathbf{Z}$ and let $\mathbf{z}_{,k}$ denote the kth column of $\mathbf{Z}$. The group membership of N entities is defined by a (latent) binary matrix $\mathbf{Z}$ where $z_{ik} = 1$ if entity i belongs to group k. Given $\mathbf{Z}$, we say that entity i interacts with entity j if $z_{ik} z_{jk} = 1$ for some k. Note that two entities can interact through many groups where each group represents one type of interaction. In many cases, a prior on such interactions can be

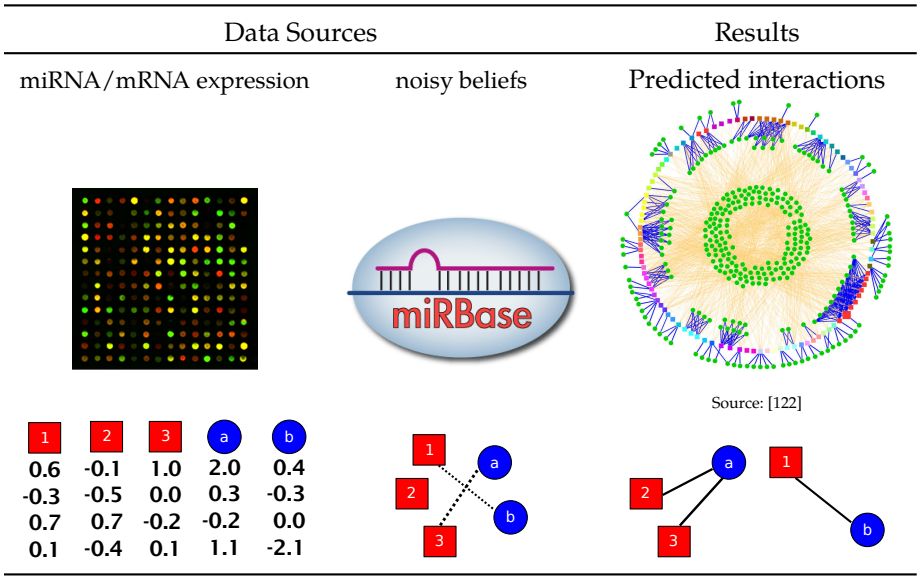


Figure 5.1: The data sources used by GroupMiR.

obtained. Assume we have a N-by-N symmetric matrix $\mathbf{W}$, where w_{ij} indicates the degree that we believe that entity i and j interact: $w_{ij} > 0$ if entities i and j are more likely to interact and $w_{ij} < 0$ if they are less likely to do so.

Nonparametric prior for Z Griffiths and Ghahramani [15] proposed the Indian Buffet Process (IBP) as a nonparametric prior distribution on sparse binary matrices $\mathbf{Z}$. The IBP can be derived from a simple stochastic process, described by a culinary metaphor. In this metaphor, there are N customers (entities) entering a restaurant and choosing from an infinite array of dishes (groups). The first customer tries Poisson(α) dishes, where α is a parameter. The remaining customers enter one after the others. The ith customer tries a previously sampled dish k with probability $\frac{m_k}{i}$, where m_k is the number of previous customers who have sampled this dish. He then samples a Poisson$(\frac{\alpha}{i})$ number of new dishes. This process defines an exchangeable distribution on the equivalence classes of $\mathbf{Z}$, which are the set of binary matrices that map to the same left-ordered binary matrices. [15]. Exchangeability means that the order of the customers does not affect the distribution and that permutation of the data does not change the resulting likelihood.

The prior knowledge on interactions discussed above (encoded by $\mathbf{W}$) violates the exchangeability of the IBP since the group membership probability depends on the identities of the entities whereas exchangeability means that permutation of entities does not change the probability. In [130], Miller et al. presented the phylogenetic Indian Buffet Process (pIBP), where they used a tree representation to express non-exchangeability. In their model, the relationships among customers are encoded as a tree allowing them to exploit the sum-product algorithm in defining the updates for an MCMC sampler, without significantly increasing the computational burden when performing inference.

We combine the IBP with pairwise potentials using $\mathbf{W}$, constraining the dish selection of customers. Similar to the pIBP, the entries in z_k are not chosen independently given π_k but rather depend on the particular assignment of the remaining entries. In the following sections, we start with a model with a finite number of groups and consider the limit as the number of groups grows to derive the nonparametric prior. Note that in our model, as in the original IBP [15], while the number of rows are finite, the number of columns (features) could be infinite. We can thus define a prior on interactions between entities (since their number is known in advance) while still allowing for an infinite number of groups. This flexibility allows the group parameters to be drawn from an infinite mixtures of priors which may lead to identical groups of entities each with a different set of parameters.

5.2.1 Prior on finite matrices Z

We have an N-by-K binary matrix $\mathbf{Z}$ where N is the number of entities and K is a fixed, finite number of groups. In the IBP, each group/column k is associated with a parameter π_k, chosen from a Beta$(\alpha/K, 1)$ prior distribution where α is a hyperparameter:

$$\pi_k|\alpha \sim \text{Beta}\big(\frac{\alpha}{K}, 1\big)$$
$$P(\mathbf{z}_{,k}|\pi_k) = \exp\Big(\sum_i \big((1-z_{ik})\log(1-\pi_k) + z_{ik}\log\pi_k\big)\Big)$$

The joint probability of a column k and π_k in the IBP is:

$$P(\mathbf{z}_{,k}, \pi_k|\alpha) = \frac{1}{B(\frac{\alpha}{K}, 1)} \exp\Big(\sum_i \big((1-z_{ik})\log(1-\pi_k) + z_{ik}\log\pi_k\big) + \big(\frac{\alpha}{K} - 1\big)\log\pi_k\Big) \tag{5.3}$$

where $B(\cdot)$ is the Beta function.

For our model, we add the new pairwise potentials on memberships of entities. Defining $\Phi_{z_{,k}} = \exp\big(\sum_{i<j} w_{ij} z_{ik} z_{jk}\big)$, the joint probability of a column k and π_k is:

$$P(\mathbf{z}_{,k}, \pi_k|\alpha) = \frac{1}{Z'}\Phi_{z_{,k}} \exp\Big(\sum_i \big((1-z_{ik})\log(1-\pi_k) + z_{ik}\log\pi_k\big) + \big(\frac{\alpha}{K} - 1\big)\log\pi_k\Big) \tag{5.4}$$

where Z' is the partition function. Note that IBP is a special case of our model when all w's are zeros ($\mathbf{W} = 0$).

Following [15], we define the lof-equivalence classes $[\mathbf{Z}]$ as the sets of binary matrices mapped to the same left-ordered binary matrices. The history h_i of a feature k at an entity i is defined as $(z_{1k}, \ldots, z_{(i-1)k})$. When no object is specified, h refers to the full history. m_k and m_h denote the number of non-zero entries of a feature k and a history h respectively. K_h is the number of features possessing the history h while K_0 is the number of features having $m_k = 0$. $K_+ = \sum_{h=1}^{2^N-1} K_h$ is the number of features for which $m_k > 0$.

By integrating over all values of π_k, we get the marginal probability of a binary

matrix $\mathbf{Z}$:

$$P(\mathbf{Z}) = \prod_{k=1}^{K} \int_0^1 P(\mathbf{z}_{,k}, \pi_k | \alpha)\, d\pi_k \tag{5.5}$$

$$= \prod_{k=1}^{K} \frac{1}{Z'} \Phi_{z_{,k}} \int_0^1 \exp\Big((\frac{\alpha}{K} + m_k - 1) \log \pi_k + (N - m_k) \log(1 - \pi_k) \Big)\, d\pi_k \tag{5.6}$$

$$= \prod_{k=1}^{K} \frac{1}{Z'} \Phi_{z_{,k}} B\big(\frac{\alpha}{K} + m_k, N - m_k + 1\big) \tag{5.7}$$

The partition function Z' could be written as: $Z' = \sum_{h=0}^{2^N-1} \Phi_h B\big(\frac{\alpha}{K} + m_h, N - m_h + 1\big)$.

Taking the infinite limit

The probability of a particular lof-equivalence class of binary matrices, $[\mathbf{Z}]$, is:

$$P([\mathbf{Z}]) = \sum_{\mathbf{Z}} P(\mathbf{Z}) = \frac{K!}{\prod_{h=0}^{2^N-1} K_h!} \prod_{k=1}^{K} \frac{1}{Z'} \Phi_{z_{,k}} B\big(m_k + \frac{\alpha}{K}, N - m_k + 1\big) \tag{5.8}$$

Taking the limit when $K \to \infty$, we can show that with $\Psi = \sum_{h=1}^{2^N-1} \Phi_h \frac{(N-m_h)!(m_h-1)!}{N!}$:

$$\lim_{K\to\infty} P([\mathbf{Z}]) = \lim_{K\to\infty} \frac{K!}{\prod_{h=0}^{2^N-1} K_h!} \prod_{k=1}^{K_+} \Phi_{z_{,k}} \frac{B(m_k + \frac{\alpha}{K}, N - m_k + 1)}{B(\frac{\alpha}{K}, N + 1)} \prod_{k=1}^{K} \frac{1}{Z'} B(\frac{\alpha}{K}, N + 1) \tag{5.9}$$

$$= \frac{\alpha^{K+}}{\prod_{h=1}^{2^N-1} K_h!} \prod_{k=1}^{K_+} \Phi_{z_{,k}} \frac{(N - m_k)!(m_k - 1)!}{N!} \exp\big(-\alpha\Psi\big) \tag{5.10}$$

The detailed derivations are shown in C.1.

5.2.2 The generative process

We now describe a generative stochastic process for $\mathbf{Z}$. It can be understood by a culinary metaphor, where each row of $\mathbf{Z}$ corresponds to a customer and each column corresponds to a dish. We denote by $h(i)$ the value of z_{ik} in the complete history h. With $\overline{\Phi}_h = \Phi_h \frac{(N-m_h)!(m_h-1)!}{N!}$, we define $\Psi_i = \sum_{h:\, h_i=0 \text{ and } h(i)=1} \overline{\Phi}_h$ so that $\Psi = \sum_{i=1}^{N} \Psi_i$. Finally, let $z_{<ik}$ be entries $1, \ldots, (i-1)$ of z_k.

Assume that we are provided with a compatibility score between pairs of customers. That is, we have a value w_{ij} for the food preference similarity between customer i and customer j. Higher values of w_{ij} indicate similar preferences and customers with such values are more likely to select the same dish. Therefore, the dishes a customer selects may depend on the choices of previous customers. The first customer tries Poisson$(\alpha\Psi_1)$ dishes. The remaining customers enter one after the others. The ith customer selects dishes with a probability that partially depends on the selection of the previous customers. The probability that a dish would be selected is $\sum_{h:\, h_i=z_{<ik} \text{ and } h(i)=1} \overline{\Phi}_h / \sum_{h:\, h_i=z_{<ik}} \overline{\Phi}_h$. He then

samples a Poisson($\alpha\Psi_i$) number of new dishes. This process repeats until all customers have made their selections. Although this process is not exchangeable, the sequential order of customers is not important. This means that we get the same marginal distribution for any particular order of customers. Let $K_1^{(i)}$ denote the number of new dishes sampled by customer i, the probability of a particular matrix generated by this process is:

$$P(\mathbf{Z}) = \frac{\alpha^{K+}}{\prod_{i=1}^{N} K_1^{(i)}} \prod_{k=1}^{K_+} \overline{\Phi}_{z_{,k}} \exp\left(-\alpha\Psi\right) \tag{5.11}$$

If we only pay attention to the lof-equivalence classes $[\mathbf{Z}]$, since there are $\frac{\prod_{i=1}^{N} K_1^{(i)}}{\prod_{h=1}^{2^N-1} K_h!}$ matrices generated by this process mapped to the same equivalence classes, multiplying $P(\mathbf{Z})$ by this quantity recovers Equation (5.10). We show in Appendix C that in the case of the IBP where $\Phi_h = 1$ for all histories h (when $\mathbf{W} = 0$), this generative process simplifies to the Indian Buffet Process.

5.2.3 Related work in Machine Learning

Determining interactions between entities based on observations is a major challenge when analyzing biological and social network data [131, 132, 9]. In most cases we can obtain information regarding each of the entities (individuals in social networks and proteins in biological networks) and some information about possible relationships between them (friendships or conversation data for social networks and motif or experimental data for biology). The goal is then to integrate these datasets to recover the interaction network between the entities being studied. To simplify the analysis of the data it is also beneficial to identify groups, or clusters, within these interaction networks. Such groups can then be mapped to specific demographics or interests in the case of social networks [131] or to modules and pathways in biological networks [133].

A large number of generative models were developed to represent entities as members of a number of classes. Many of these models are based on the stochastic blockmodel introduced in [134]. While the number of classes in such models could be fixed, or provided by the user, nonparametric Bayesian methods have been applied to allow this number to be inferred based on the observed data [135]. The stochastic blockmodel was also further extended in [131] to allow mixed membership of entities within these classes. An alternate approach is to use latent features to describe entities. [136] proposed a nonparametric Bayesian matrix factorization method to learn the latent factors in relational data whereas [132] presented a nonparametric model to study binary link data. All of these methods rely on the pairwise link and interaction data and in most cases do not utilize properties of the individual entities when determining interactions.

Here we present a model that extends the Indian Buffet Process (IBP) [15], a nonparametric Bayesian prior over infinite binary matrices, to learn the interactions between entities with an unbounded number of groups. Specifically, we represent each group as a latent feature and define interactions between entities within each group. Such latent feature representation has been used in the past to describe entities [15, 136, 132] and IBP is an appropriate nonparametric prior to infer the number of latent features.

However, unlike IBP our model utilizes interaction scores as priors and so the model is not exchangeable anymore. We thus extend IBP by integrating it with Markov random field (MRF) constraints, specifically pairwise potentials as in Ising model. MRF priors has been combined with Dirichlet Process mixture models for image segmentation in a related work of Orbanz and Buhmann [137]. Pairwise information is also used in the distance dependent Chinese restaurant process [138] to encourage similar objects to be clustered. Zhou et al. [139, 140] present a dependent hierarchical beta process using covariate-dependent features to impose that objects with similar covariates are likely to be clustered. The relationship between objects are summarized by a matrix $\mathbf{A}$ using a kernel $\mathcal{K}$. One way to apply this prior to our biological application would require converting the prior likelihood $\mathbf{C}$ matrix to the summary matrix, by defining a kernel over covariates. In contrast, our model avoids this requirement since all samples are drawn from a single process that encapsulates the dependencies.

Our model is well suited for cases in which we are provided with information on both link structure and the outcome of the underlying interactions. In social networks such data can come from observations of conversation between individuals followed by actions of the specific individuals (for example, travel), whereas in biology it is suited for regulatory networks.

5.3 Regression model for mRNA expression

In this section, we describe the application using the nonparametric prior to the miRNA target prediction problem. However, the method is applicable in general settings where there is a way to model properties of one entity from properties of its interacting entities. Recall that our input data are expression profiles of R messenger RNA (mRNA) transcripts and M miRNA transcript across P samples: $\mathbf{X} = (\mathbf{x}_1, \ldots, \mathbf{x}_M)^{\mathrm{T}}$, and $\mathbf{Y} = (\mathbf{y}_1, \ldots, \mathbf{y}_R)^{\mathrm{T}}$. Furthermore, suppose we are given a M-by-R matrix $\mathbf{C}$ where c_{ij} is the prior likelihood score for the interaction of miRNA i and mRNA j. Such matrix $\mathbf{C}$ could be obtained from sequence-based miRNA target predictions as discussed above. Applying our interaction model to this problem, the set of $N = M + R$ entities are divided into two disjoint sets of mRNAs and miRNAs. Let $\mathbf{Z} = (\mathbf{U}^{\mathrm{T}}, \mathbf{V}^{\mathrm{T}})^{\mathrm{T}}$ where $\mathbf{U}$ and $\mathbf{V}$ are the group membership matrices for miRNAs and mRNAs respectively, $\mathbf{W}$ is given by $\begin{pmatrix} \mathbf{0} & \mathbf{C} \\ \mathbf{C}^{\mathrm{T}} & \mathbf{0} \end{pmatrix}$. Therefore, miRNA i and mRNA j interact through all groups k such that $u_{ik}v_{jk} = 1$.

5.3.1 Gaussian additive model

In the interaction model suggested by GenMiR++ [122], each miRNA expression profile is used to explain the downregulation of the expression of its targeted mRNAs. Our model uses a group specific and miRNA specific coefficients ($\mathbf{s} = (s_1, \ldots, s_\infty)^{\mathrm{T}}$, with $s_k > 0$ for groups and $\mathbf{r} = (r_1, \ldots, r_M)^{\mathrm{T}}$ for all miRNAs) to model the down-regulation effect. These coefficients represent the baseline effect of group members and the strength of specific miRNAs, respectively. Using these parameters the expression level of a specific mRNA could be explained by summing over expression profiles of all miRNAs targeting the

mRNA:

$$\mathbf{y}_j \sim \mathcal{N}\big(\boldsymbol{\mu} - \sum_{i:\, \mathbf{u}_i^{\mathrm{T}}\mathbf{v}_j \neq 0} (r_i + \sum_{k:\, u_{ik}v_{jk}=1} s_k)\, \mathbf{x}_i,\ \sigma^2\mathbf{I}\big) \tag{5.12}$$

where $\boldsymbol{\mu}$ represents baseline expression for this mRNA and σ is used to represent measurement noise. Thus, under this model, the expression of a mRNA are reduced from their baseline values by a linear combination of expression values of the miRNAs that target them. The probability of the observed data given $\mathbf{Z}$ is:

$$P(\mathbf{X}, \mathbf{Y}|\mathbf{Z}, \Theta) \propto \exp\Big(-\frac{1}{2\sigma^2}\sum_i (\mathbf{y}_j - \overline{\mathbf{y}}_j)^{\mathrm{T}}(\mathbf{y}_j - \overline{\mathbf{y}}_j)\Big) \tag{5.13}$$

with $\Theta = \{\boldsymbol{\mu}, \sigma^2, \mathbf{s}, \mathbf{r}\}$ and $\overline{\mathbf{y}}_j = \mu - \sum_{i:\, \mathbf{u}_i^{\mathrm{T}}\mathbf{v}_j \neq 0}(r_i + \sum_{k:\, u_{ik}v_{jk}=1} s_k)\, \mathbf{x}_i$.

5.3.2 Priors for model variables

We use the following as prior distributions for the variables in our model:

$$\begin{aligned} s_k &\sim \text{Gamma}(\alpha_s, \beta_s) && (5.14)\\ \mathbf{r} &\sim \mathcal{N}(0, \sigma_r^2 I) && (5.15)\\ \boldsymbol{\mu} &\sim \mathcal{N}(0, \sigma_\mu^2 I) && (5.16)\\ 1/\sigma^2 &\sim \text{Gamma}(\alpha_v, \beta_v) \end{aligned}$$

where the α and β are the shape and scale parameters. The parameters are given hyperpriors: $1/\sigma_r^2 \sim \text{Gamma}(a_r, b_r)$ and $1/\sigma_\mu^2 \sim \text{Gamma}(a_\mu, b_\mu)$. $\alpha_s, \beta_s, \alpha_v, \beta_v$ are also given Gamma hyperpriors.

5.4 Inference by MCMC

As with many nonparametric Bayesian models, exact inference is intractable. Instead we use a Markov Chain Monte Carlo (MCMC) method to sample from the posterior distribution of $\mathbf{Z}$ and Θ. Although, our model allows $\mathbf{Z}$ to have infinite number of columns, we only need to keep track of non-zero columns of $\mathbf{Z}$, an important aspect which is exploited by several nonparametric Bayesian models [15]. Our sampling algorithm involves a mix of Gibbs and Metropolis-Hasting steps which are used to generate the new sample.

5.4.1 Sampling from populated columns of Z

Let m_{-ik} is the number of one entries not including z_{ik} in z_k. Also let $\mathbf{z}_{-ik}$ denote the entries of $\mathbf{z}_{,k}$ except z_{ik} and let $\mathbf{Z}_{-(ik)}$ be the entire matrix $\mathbf{Z}$ except z_{ik}. The probability of an entry given the remaining entries in a column can be derived by considering an ordering of customers such that customer i is the last person in line and using the generative process

in Section 5.2.2:

$$
\begin{aligned}
P(z_{ik}=1|\mathbf{z}_{-ik}) &= \frac{\overline{\Phi}_{z_{<ik},z_{ik}=1}}{\overline{\Phi}_{z_{<ik},z_{ik}=1}+\overline{\Phi}_{z_{<ik},z_{ik}=0}} \\
&= \frac{\exp\big(\sum_{j\neq i} w_{ij}z_{jk}\big)(N-m_{-ik}-1)!m_{-ik}!}{\exp\big(\sum_{j\neq i} w_{ij}z_{jk}\big)(N-m_{-ik}-1)!m_{-ik}!+(N-m_{-ik})!(m_{-ik}-1)!} \\
&= \frac{\exp\big(\sum_{j\neq i} w_{ij}z_{jk}\big)m_{-ik}}{\exp\big(\sum_{j\neq i} w_{ij}z_{jk}\big)m_{-ik}+(N-m_{-ik})}
\end{aligned}
$$

We could also get the result using the limiting probability in Equation (5.10). The probability of each $\mathbf{z}_{ik}$ given all other variables is: $P(z_{ik}|\mathbf{X},\mathbf{Y},\mathbf{Z}_{-(ik)}) \propto P(\mathbf{X},\mathbf{Y}|\mathbf{Z}_{-(ik)},z_{ik})P(z_{ik}|\mathbf{z}_{-ik})$. We need only to condition on z_{-ik} since columns of $\mathbf{Z}$ are generated independently.

5.4.2 Sampling other variables

Sampling a new column of $\mathbf{Z}$: New columns are columns that do not yet have any entries equal to 1 (empty groups). When sampling for an entity i, we assume this is the last customer in line. Therefore, based on the generative process described in Section 5.2.2, the number of new features are Poisson$(\frac{\alpha}{N})$. For each new column, we need to sample a new group specific coefficient variable s_k. We can simply sample from the prior distribution given in Equation (5.14) since the probability $P(\mathbf{X},\mathbf{Y}|\mathbf{Z},\Theta)$ is not affected by these new columns since no interactions are currently represented by these columns.

Sampling s_k for populated columns: Since we do not have a conjugate prior on $\mathbf{s}$, we cannot compute the conditional likelihood directly. We turn to Metropolis-Hasting to sample $\mathbf{s}$. The proposed distribution of a new value s_k^* given the old value s_k is $q(s_k^*|s_k) =$ Gamma$(h, \frac{s_k}{h})$ where h is the shape parameter. The mean of this distribution is the old value s_k. The acceptance ratio is

$$
\mathcal{A}(s_k \to s_k^*) = \min\Big[1, \frac{P(\mathbf{X},\mathbf{Y}|\mathbf{Z},\Theta\setminus\{s_k\},s_k^*)\ p(s_k^*|\alpha_s,\beta_s)\ q(s_k|s_k^*)}{P(\mathbf{X},\mathbf{Y}|\mathbf{Z},\Theta)\ p(s_k|\alpha_s,\beta_s)\ q(s_k^*|s_k)}\Big]
$$

In our experiments, h is selected so that the average acceptance rate is around 0.25 [141].

Sampling $\mathbf{r}, \mu, \sigma^2$ and prior parameters: Closed-form formulas for the posterior distributions of $\mathbf{r}$,μ and σ^2 can be derived due to their conjugacy. For example, the posterior distribution of $1/\sigma^2$ given the other variables is:

$$
p\big(\frac{1}{\sigma^2} \mid \alpha_v,\beta_v,\mathbf{X},\mathbf{Y},\mathbf{Z},\Theta\setminus\{\sigma^2\}\big) \ \propto\ p\big(\frac{1}{\sigma^2}|\alpha_v,\beta_v\big)P(\mathbf{X},\mathbf{Y}|\mathbf{Z},\Theta)
$$

Hence,

$$
\frac{1}{\sigma^2} \mid \alpha_v,\beta_v,\mathbf{X},\mathbf{Y},\mathbf{Z},\Theta\setminus\{\sigma^2\} \ \sim \text{Gamma}\big(\alpha_v+\frac{MT}{2}, \big(\frac{1}{\beta_v}+\frac{\sum_i (x_i-\overline{x}_i)^{\mathrm{T}}(x_i-\overline{x}_i)}{2}\big)^{-1}\big) \tag{5.17}
$$

Equations for updates of $\mathbf{r}$ and μ are omitted due to lack of space. Gibbs sampling steps are used for σ_r^2 and σ_μ^2 since we can compute the posterior distribution with conjugate priors. For prior parameters $\{\alpha_s,\beta_s,\alpha_v,\beta_v\}$, we use Metropolis-Hasting steps discussed previously.

5.5 Results

In this section we compare the performance of GroupMiR with GenMiR++ [122], which is one of the popular methods for predicting miRNA-mRNA interactions. However, unlike our method it does not use grouping of mRNAs and attempts to predict each one separately. Besides, there are two other important differences of GenMiR++ from our method: *1*) GenMiR++ only consider interactions in the candidate set while our method consider all possible interactions. *2*) GenMiR++ accepts a binary matrix as a candidate set while our method allows continuous valued scores. To our best knowledge, GenMiR++, which uses the regression model for interaction between entities, is the only appropriate method[2] for comparison.

5.5.1 Synthetic data

We generated 9 synthetic datasets. Each dataset contains 20 miRNAs and 200 mRNAs. We set the number of groups to $K = 5$ and $T = 10$ for all datasets. The miRNA membership $\mathbf{U}$ is a random matrix with at most 5 ones in each column. The mRNA membership $\mathbf{V}$ is a random matrix with density of 0.1. The expression of mRNAs are generated from the model in Equation (5.12) with $\sigma^2 = 1$. The remaining random variables are sampled as follows: $x \sim \mathcal{N}(0, 1)$, $s \sim \mathcal{N}(1, 0.1)$ and $r \sim \mathcal{N}(0, 0.1)$. Since the sequence based predictions of miRNA-mRNA interactions are based on short complementary regions they often result in many more false positives than false negatives. We thus introduce noise to the true binary interaction matrix $\mathbf{C}'$ by probabilistically changing each zero value in that matrix to 1. We tested different noise probabilities: $0.1, 0.2, 0.4$ and 0.8. We use $\mathbf{C} = 2\mathbf{C}' - 1.8$, $\alpha = 1$ and the hyperprior parameters are set to generic values. Our sampler is ran for 2000 iterations and 1000 iterations are discarded as burn-in.

Figure 5.2 plots the estimated posterior distribution of K from the samples of the 9 datasets for all noise levels. As can be seen, when the noise level is small (0.1), the distributions are correctly centered around $K = 5$. With increasing noise levels, the number of groups is overestimated. However, GroupMiR still does very well at a noise level of 0.4 and estimates for the higher noise level are also within a reasonable range.

We estimated a posterior mean for the interaction matrix $\mathbf{Z}$ by first ordering the columns of each sampled $\mathbf{Z}$ and then selecting the mode from the set of $\mathbf{Z}$ matrices. GenMiR++ returns a score value in $[0, 1]$ for each potential interaction. To convert these to binary interactions we tested a number of different threshold cutoffs: $0.5, 0.7$ and 0.9. Figure 5.4 presents a number of quality measures for the recovered interactions by the two methods. GroupMiR achieves the best F1 score across all noise levels greatly improving upon GenMiR++ when high noise levels are considered (a reasonable biological scenario). In general, while the precision is very high for all noise levels, recall drops to a lower rate. From a biological point of view, precision is probably more important than recall since each of the predictions needs to be experimentally tested, a process that is often time consuming and expensive.

In addition to accurately recovering interactions between miRNAs and mRNAs, GroupMiR also correctly recovers the groupings of mRNA and miRNAs. Figure 5.3 presents a

[2] We also tested with the original IBP (by setting $\mathbf{W} = 0$). The results for both the synthetic and real data were too weak to be comparable with GenMIR++. See Lemma C.2.1.

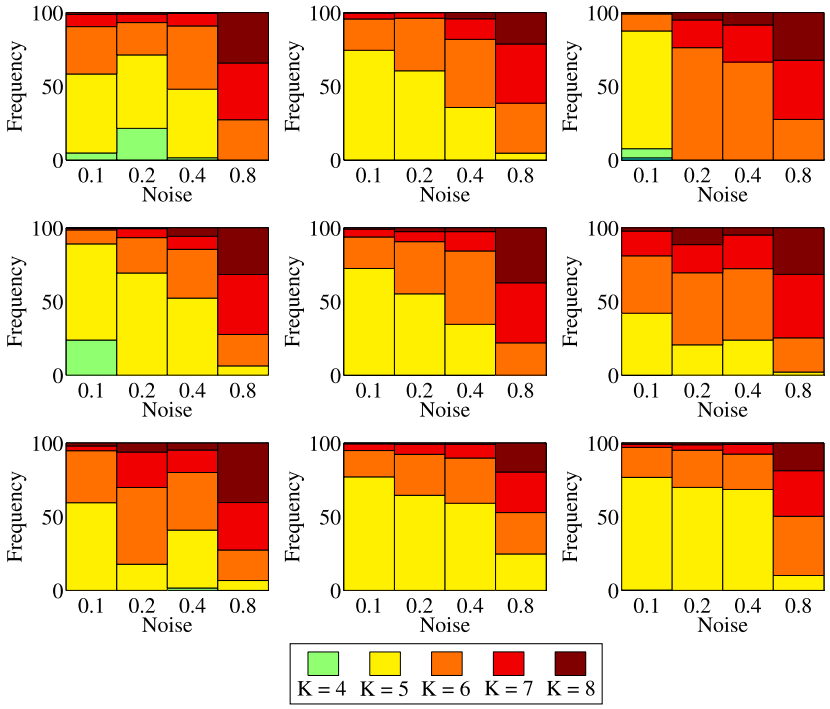


Figure 5.2: The posterior distribution of K.

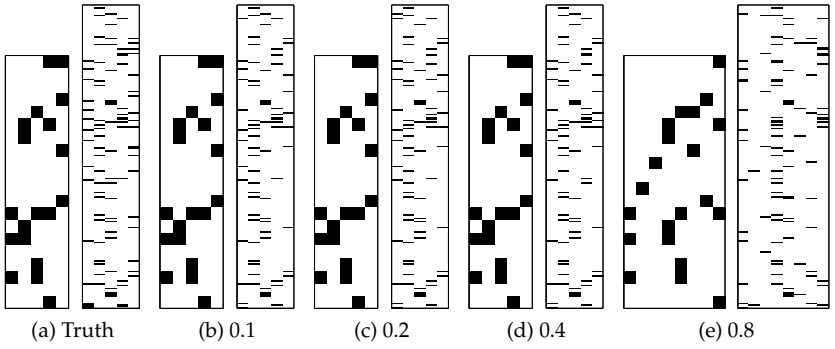


Figure 5.3: An example synthetic dataset.

graphical view of the group membership in both the true model and the model recovered by GroupMiR for one of the synthetic datasets. As can be seen, our method is able to accurately recover the groupings of both miRNAs and mRNAs with moderate noise levels (up to 0.4). For the higher noise level (0.8) the method assigns more groups than in the underlying model. However, most interactions are still correctly recovered. These results hold for all datasets we tested (not shown due to lack of space).

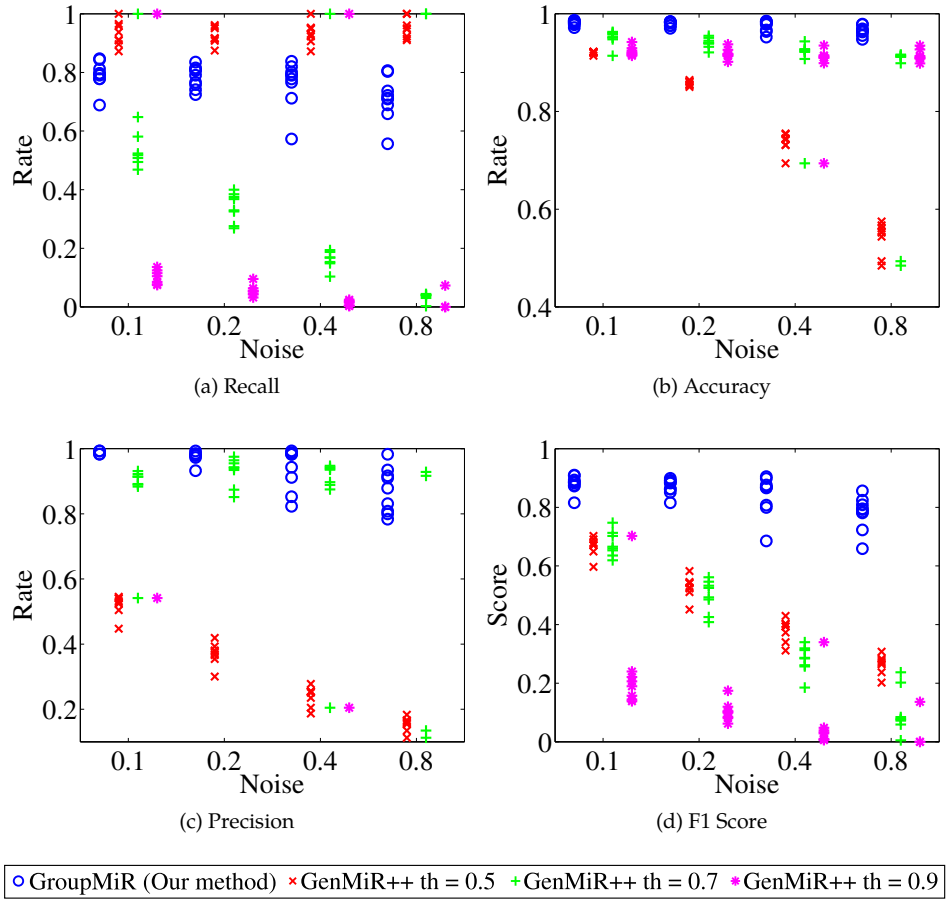


Figure 5.4: Performance of GroupMiR versus GenMiR++. Each data point is a synthetic dataset.

5.5.2 Application to mouse lung development

To test GroupMiR on real biological data, we used a mouse lung developmental dataset [142]. In this study, the authors used microarrays to profile both miRNAs and mRNAs at 7 time points, which include all recognized stages of lung development. We downloaded

the log ratio normalized data collected in this study. Duplicate samples were averaged and median values of all probes were assigned to genes. As suggested in the paper, we used ratios to the last time point resulting in 6 values for each mRNA and miRNA. Priors for interaction between miRNA and mRNA were downloaded from the MicroCosm Target[3] database. The prior score was computed by taking $-\log 10(\text{p-value})$ thresholded at the maximum value of 5. Selecting genes with variance in the top 10%, led to 219 miRNAs and 1498 mRNAs which were used for further analysis.

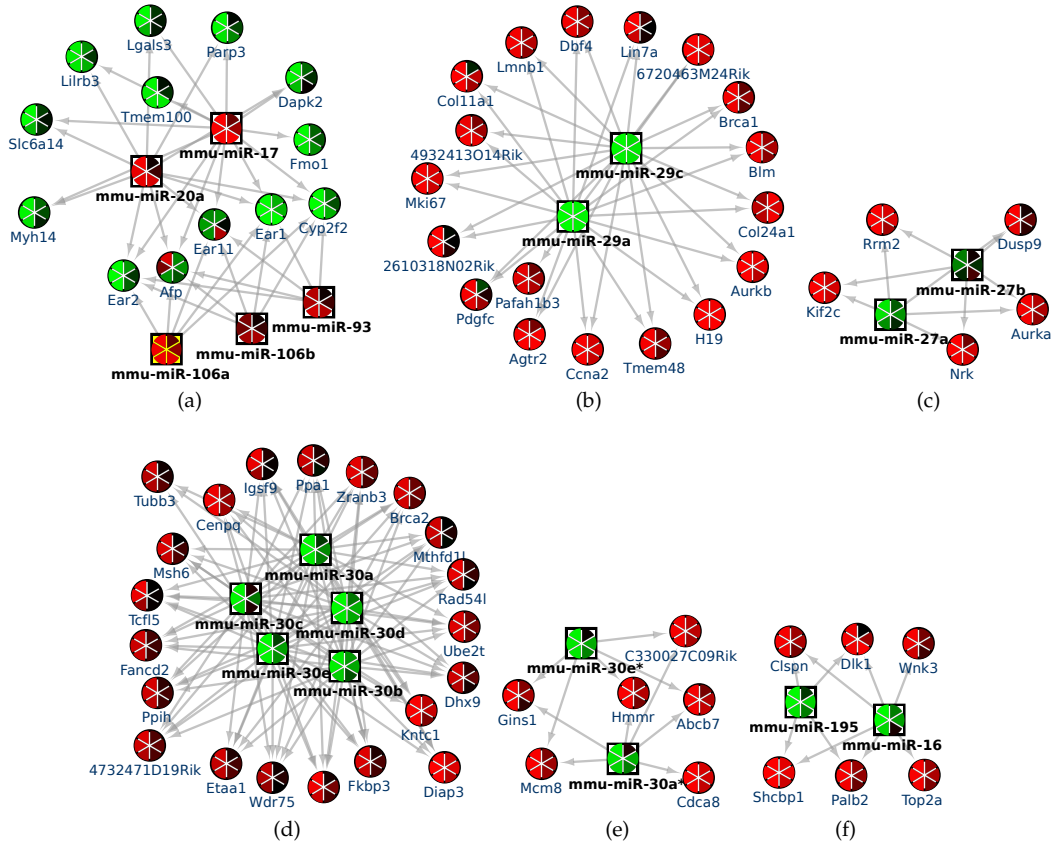


Figure 5.5: Interaction network recovered by GroupMiR. Each node is a pie chart corresponding to its expression values in the 6 time points (red: up-regulation, green: down-regulation).

We collected 5000 samples of the interaction matrix $\mathbf{Z}$ following a 5000 iteration burn-in period. We only kept every 10th sample to get a set of 500 samples. Convergence of the MCMC chain is determined by monitoring trace plots of K in multiple chains. Since there

[3]http://www.ebi.ac.uk/enright-srv/microcosm/

are many more entries for real data compared to synthetic data we computed a consensus for $\mathbf{Z}$ by reordering columns in each sample and averaging the entries across all matrices.

We further analyzed the network constructed from groups with at least 90% posterior probability. The network recovered by GroupMiR is more connected (89 nodes and 208 edges) when compared to the network recovered by GenMiR++ (using equivalent 0.9 threshold) with 37 nodes and 34 edges (Figure C.1). We used Cytoscape [143] to visualize the 6 groups of interactions in Figure 5.5. The network contains several groups of co-expressed miRNAs controlling sets of mRNA, in agreement with previous biological studies [144].

To test the function of the clusters identified, we performed Gene Ontology (GO) enrichment analysis for the mRNAs using GOstat [145]. The full results (Bonferroni corrected) are presented in Appendix C. As can be seen, several cell division categories are enriched in cluster (b) which is expected when dealing with a developing organ (which undergoes several rounds of cell division). Other significant functions include organelle organization and apoptosis which also are associated with development (cluster (c)). We performed similar GO enrichment analysis for the GenMiR++ results and for K-means when using the same set of mRNAs (setting $k = 6$ as in our model). In both cases we did not find any significant enrichment indicating that only by integrating sets of miRNAs with the mRNAs for this data we can find functional biological groupings. We also tried running GenMiR++ with threshold 0.6 (Figure C.1). The network has no clear modular structure. The miRNAs and mRNAs seem to be divided into two connected components, one with down-regulated genes and one with up-regulated genes. See Appendix C.4 for details.

We have also looked at the miRNAs controlling the different clusters and found that in a number of cases these agreed with prior knowledge. Cluster (a) includes 2 members of the miR 17-92 cluster, which is known to be critical to lung organogenesis [146]. MiRNA families miR-30, miR-29, miR-20 and miR-16, all identified by our method, were also reported to play roles in the early stages of lung organogenesis [142]. It is important to point out that we did not filter miRNAs explicitly based on expression but these miRNAs came in the results based on their strong effect on mRNA expression.

5.6 Conclusions

We have described an extension to IBP that allows us to integrate priors on interactions between entities with measured properties for individual entities when constructing interaction networks. The method was successfully applied to predict miRNA-mRNA interactions and we have shown that it works well on both synthetic and real data. While our focus in this chapter was on a biological problem, several other datasets provide similar information including social networking data. Our method is appropriate for such datasets and can help when attempting to construct interaction networks based on observations.

6

PIMiM: Protein Interaction based MicroRNA Modules

The previous chapter discusses GroupMiR, an integrated model to infer cooperative regulation of clusters of miRNAs and sets of target mRNAs. The model associates each miRNA with a feature corresponding to a group of cooperative miRNAs and target mRNAs. Although this representation allows discovering group structure in miRNA regulation, it may be too restrictive–it requires every miRNAs and mRNAs in a group to interact. Building on GroupMiR, we developed PIMiM, a new method which infers groups of target miRNAs participating in common pathways using protein-protein interaction data.

6.1 Introduction

An approach to link miRNAs with pathways is to project mRNA expression data on pathway databases and compute the correlation between miRNAs and average pathway expression levels to identify likely regulators of signaling pathways [125]. While this method does not identify specific targets, it can be used to infer the function of specific miRNAs based on the pathways they regulate. Recently, there is growing evidence that interacting proteins are more likely to be co-regulated by the same miRNAs [147, 148]. It has also been shown that some miRNAs coordinately target protein complexes [149]. While such complimentary information may be important, few prior works has taken advantage of it to predict condition-specific interactions. An exception is a recent work by Zhang et al. which developed SNMNMF [150] to integrate protein interactions with miRNA and mRNA expression data. The method is based on a non-negative matrix factorization analysis which factorizes the two expression data matrices such that the two share one common factor, which is assumed to be the module basis matrix $\mathbf{W}$. Note however that while this method was successfully applied to analyze Ovarian cancer data, it does not use a regression model to explain mRNA expression levels, or require that miRNAs and mRNAs in the same module be anti-correlated, and so the resulting modules do not fully utilize current knowledge regarding the inhibitory role of miRNAs which may lead to missing important interactions.

The methods discussed above successfully integrated expression and sequence data. However, a major point that is often ignored by these prediction methods is the combinatorial aspect of miRNA regulation. To allow the use of such group- or module-based regulatory model, we discussed GroupMiR in Chapter 5 which uses a nonparametric Bayesian prior based on the Indian Buffet Process (IBP [15]) to identify modules of co-regulated miRNAs and their target mRNAs. As we have shown, by using a module-based approach we can improve upon methods that treat miRNAs or mRNAs individually improving the set of correctly recovered miRNA-mRNA interactions [115].

Here, we present the Protein Interaction based MicroRNA Modules (PIMiM) method which extends the regression framework of GroupMiR by using an additional type of data: protein interactions. As we show, by defining a new target function that encourages

interacting proteins to belong to the same module we can utilize such data and integrate it with expression and sequence-based data in a probabilistic model. We develop an iterative learning procedure to learn the parameters of our model and show that it converges to a local minima. Comparison of PIMiM to previous methods indicates that by combining a module based approach with protein interaction data we can improve upon both methods that only rely on modules (GroupMiR) and methods that rely on protein interaction (SNMNMF). We used PIMiM to study miRNA in several types of cancer allowing us to identify novel regulators that either span multiple cancer types or are unique to specific cancers.

6.2 Methods

6.2.1 Overview

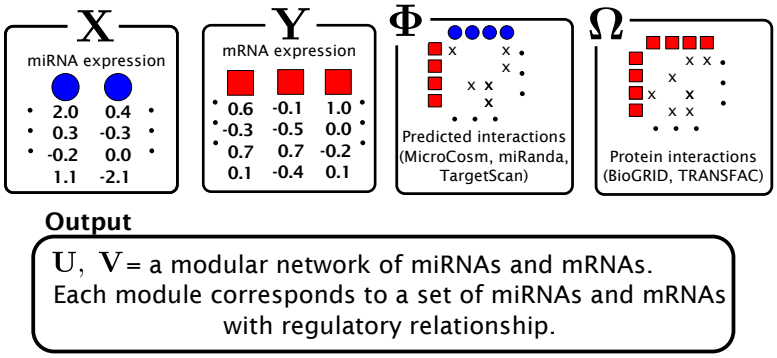


Figure 6.1: Data used as input for PIMiM. In addition the miRNA and mRNA expression data, PIMiM uses sequence based predictions of miRNA-mRNA interactions and protein-protein interactions.

We developed PIMiM, a module-based method which predicts targets for miRNAs by assigning them, together with the mRNAs they regulate, to one of K modules. Modules may contain several miRNAs and many mRNAs, and both miRNA and mRNAs can be assigned to 0, 1 or multiple modules and thus modules may overlap.

The input to PIMiM is condition specific miRNA and mRNA expression data (usually multiple measurements from patients or different time points). In addition, we use sequence-based predictions of miRNA-mRNA interactions (any probabilistic predictions can be used) and static protein interaction data. Using these datasets we learn a regularized probabilistic regression model in which mRNA data is regressed to the expression data of miRNAs assigned to modules regulating it. The down-regulation effect of a miRNA on the expression of its target mRNA is aggregated across all modules allowing information to be shared between modules in the learning process. Our probabilistic model rewards the assignments of predicted miRNA-mRNA pairs to the same module and also rewards

assignment of mRNAs of interacting proteins to the same module. Combined, the modules explain the observed mRNA expression data as a function of their regulating miRNAs and the set of proteins they interact with.

6.2.2 Notations

Following the same notation in the previous chapter, we assume that there are M miRNAs and R mRNAs in each sample. We denote expression profiles of miRNAs by $\mathbf{X} = (\mathbf{x}_1, \ldots, \mathbf{x}_M)^{\mathrm{T}}$ and of mRNAs by $\mathbf{Y} = (\mathbf{y}_1, \ldots, \mathbf{y}_R)^{\mathrm{T}}$, where $\mathbf{x}_i$ and $\mathbf{y}_j$ are vectors with the expression levels of miRNA i and mRNA j, respectively, in all samples. Both matrices have P columns corresponding to the P matched samples. In addition, let $\mathbf{\Omega}$ (sparse R-by-R matrix) be the weighted adjacency matrix of the protein interactions (obtained from databases such as BioGRID [151] or TRANSFAC [152]) and $\mathbf{\Phi}$ (sparse M-by-R matrix) be the list of predicted interactions of miRNAs and mRNAs from sequence data (obtained from prediction databases such as MicroCosm [153]). We also define $\mathbb{1}_{\mathbf{\Phi}}$ and $\mathbb{1}_{\mathbf{\Omega}}$ as binary matrices indicating whether an entry of $\mathbf{\Phi}$ and $\mathbf{\Omega}$ respectively is non-zero.

For learning K modules our goal is to determine (learn) the values of the membership parameters u_{ik} and v_{jk}, which represent the propensity that miRNA i or mRNA j belong to module k. Naturally, we restrict these parameters to be non-negative: $u_{ik} \geq 0$ and $v_{jk} \geq 0$, where we interpret that a miRNA or mRNA is not assigned to a module if the corresponding parameter is zero. We use matrices $\mathbf{U} = (\mathbf{u}_1, \ldots, \mathbf{u}_M)^{\mathrm{T}}$ and $\mathbf{V} = (\mathbf{v}_1, \ldots, \mathbf{v}_R)^{\mathrm{T}}$ to represent this complete set of membership parameters. Lastly, we use the following subscript such as $\mathbf{u}_{,k}$ or $\mathbf{v}_{,k}$ to denote the kth column of the matrices.

$\mathbf{U}, \mathbf{V}$	: miRNA and mRNA module membership
K	: number of modules
$\mathbf{u}_i, \mathbf{v}_j$	: ith or jth rows of the matrices
$\mathbf{u}_{,k}, \mathbf{v}_{,k}$	: kth columns of the matrices
$\mathbb{1}_{\mathbf{\Phi}}, \mathbb{1}_{\mathbf{\Omega}}$	: binary indicators of $\mathbf{\Phi}, \mathbf{\Omega}$

6.2.3 Probabilistic regression model

Following previous works [128, 115], we employ a regression-based method to link the expression profiles of miRNAs and mRNAs. Expression values of mRNAs are assumed to be down-regulated from a baseline expression level by a linear combination of expression profiles of all their predicted miRNA regulators. For example, mRNA j's expression values are distributed as: $\mathbf{y}_j \sim \mathcal{N}\left(\boldsymbol{\mu} - \sum_{i \in \mathcal{S}_j} w_{ij}\mathbf{x}_i,\ \mathbf{\Sigma}\right)$, where $\boldsymbol{\mu}$ is the baseline expression level, $\mathbf{w}_i$ are weights associated with miRNAs (which previous methods learn individually for each mRNA) and $\mathcal{S}_j$ is the set of predicted miRNA regulators of mRNA j.

We depart from these previous models in how we specify miRNA regulators and how we learn the weights $\mathbf{w}_i$. First, each mRNA is assumed to be a target of all miRNAs assigned to the modules it belongs to as long as they are predicted to regulate it ($\phi_{ij} \neq 0$). Formally, mRNA j is the target of the set of miRNAs $\mathcal{S}_j = \{i : \mathbf{u}_i^{\mathrm{T}}\mathbf{v}_j > 0 \text{ and } \phi_{ij} \neq 0\}$.

Secondly, the down-regulation weights are aggregated across all modules such as $w_{ij} = \mathbf{u}_i^{\mathrm{T}}\mathbf{v}_j$.

Given these assumptions, the likelihood of the observed expression values is:

$$\begin{aligned} p(\mathbf{Y} \mid \mathbf{U}, \mathbf{V}, \mathbf{X}, \boldsymbol{\mu}, \boldsymbol{\Sigma}) &= \prod_j \mathcal{N}(\mathbf{y}_j \mid \boldsymbol{\mu} - \sum_{i \in \mathcal{S}_j} \mathbf{u}_i^{\mathrm{T}}\mathbf{v}_j\mathbf{x}_i, \boldsymbol{\Sigma}) \\ &= \prod_j \mathcal{N}(\mathbf{y}_j \mid \boldsymbol{\mu} - \mathbf{X}^{\mathrm{T}}\big((\mathbb{1}_{\boldsymbol{\Phi}})_{,j} \circ (\mathbf{U}\mathbf{v}_j)\big), \boldsymbol{\Sigma}) \end{aligned} \tag{6.1}$$

where $\boldsymbol{\Sigma} = \mathrm{diag}(\sigma_1^2, \ldots, \sigma_P^2)$ are the per-sample variance terms.

6.2.4 Utilizing protein interactions

So far PIMiM only uses expression values in a regression setting (while we constrain the regulators to come from the sequence-based predicted set, the regression model itself does not directly encourage the assignment of miRNA and predicted mRNA targets to the same module). To incorporate the input interaction data (predicted miRNA-mRNA pairs $\boldsymbol{\Phi}$ and protein interactions $\boldsymbol{\Omega}$), we use a function that rewards assignments to the same module based on the strength of the predicted edge as follows:

$$\begin{aligned} p(\mathbb{1}_{\phi_{ij}} = 1 \mid \mathbf{U}, \mathbf{V}) &= \frac{1}{1 + \exp(-\alpha\ \phi_{ij}\mathbf{u}_i^{\mathrm{T}}\mathbf{v}_j)} = \sigma(\alpha\ \phi_{ij}\mathbf{u}_i^{\mathrm{T}}\mathbf{v}_j) \\ p(\mathbb{1}_{\phi_{ij}} = 0 \mid \mathbf{U}, \mathbf{V}) &= 1 - \sigma(\alpha\ \mathbf{u}_i^{\mathrm{T}}\mathbf{v}_j) \\ p(\mathbb{1}_{\omega_{jj'}} = 1 \mid \mathbf{V}) &= \sigma(\beta\ \omega_{jj'}\mathbf{v}_j^{\mathrm{T}}\mathbf{v}_{j'}) \end{aligned} \tag{6.2}$$

Where α and β are positive tuning parameters which are used to adjust the contributions of the two types of interaction data in our model and $\sigma(.)$ is the logistic-sigmoid function. If available (as is the case for the miRNA-mRNA interaction data) we use probabilities for $\boldsymbol{\Phi}$ and $\boldsymbol{\Omega}$ derived directly from the prediction or experimental databases (see Results). We deliberately do not include penalty terms for zero entries of $\boldsymbol{\Omega}$ because this interaction matrix is extremely sparse (the number of known protein-protein interactions is small compared to the total number of possible interactions). Penalizing zero entries when using such a sparse matrix would lead to very small modules and may be less biologically accurate since not all co-targets of a miRNA interact.

These terms indicates that the higher the probability of interaction (both miRNA-mRNA and protein-protein) the more likely it is that the interacting entities would be assigned to the same set of modules. This is done globally across all modules. For instance, if ϕ_{ij} is positive, we have a prior knowledge that miRNA i and mRNA j interact. In order to maximize the likelihood $p(\phi_{ij}|\mathbf{U}, \mathbf{V})$, we would need to learn parameters that lead to large values of $\mathbf{u}_i^{\mathrm{T}}\mathbf{v}_j$, which means that the method is more likely to place them in the same module.

6.2.5 Overall log-likelihood

To summarize, our target is to minimize the following negative log-likelihood:

$$\mathcal{L}(\mathbf{Y}, \mathbf{X}, \mathbf{\Phi}, \mathbf{\Omega}) = -\log p(\mathbf{Y}|\mathbf{U}, \mathbf{V}, \mathbf{X}, \boldsymbol{\mu}, \mathbf{\Sigma}) \\ - \sum_{i,j} \log p(\mathbb{1}_{\phi_{ij}}|\mathbf{U}, \mathbf{V}) - \sum_{j \neq j'} \log p(\mathbb{1}_{\omega_{jj'}} = 1|\mathbf{V}) \quad (6.3)$$

The first term evaluates how well the miRNA expression explains the observed mRNA expression whereas the second and third terms are rewards for assigning predicted miRNA-mRNA pairs and protein interaction pairs to the same module, respectively. This function is non-convex and thus can have multiple local minima solutions. To constrain the set of solutions we add a number of regularization terms. First, we add two sets of ℓ_1 norm constraints for the vectors $\{\mathbf{u}_i\}$ and $\{\mathbf{v}_j\}$. ℓ_1 norm contraints encourage sparsity leading to smaller and tighter modules. Since our goal is to reduce false positives, such constraints are very useful as they reduce the set of predicted miRNA-mRNA pairs. Specifically, we require that:

$$\|\mathbf{u}_i\|_1 \leq C_1, \quad i = 1, \ldots, M$$
$$\|\mathbf{v}_j\|_1 \leq C_2, \quad j = 1, \ldots, R$$

We are using two different regularization parameters C_1 and C_2. This is because the number miRNAs and mRNAs are very different so a single number does not yield good solutions. Moreover, we choose to use these constraints explicitly instead of adding them to the objective function (using Lagrangian multipliers) since this formulation is simpler to solve in our optimization procedure.

Together, our learning phase solves the following optimization:

$$\begin{aligned} \min_{\mathbf{U} \geq 0, \mathbf{V} \geq 0, \boldsymbol{\mu}, \mathbf{\Sigma}} \quad & \mathcal{F} = \mathcal{L}(\mathbf{Y}, \mathbf{X}, \mathbf{\Phi}, \mathbf{\Omega}) \\ \text{s.t.} \quad & \|\mathbf{u}_i\|_1 \leq C_1, i = 1, \ldots, M \\ & \|\mathbf{v}_j\|_1 \leq C_2, j = 1, \ldots, R \end{aligned} \quad (6.4)$$

6.2.6 Learning the parameters of our model

In this section, we discuss how to solve the optimization problem from (6.4) in order to determine values for the parameters of our model. As mentioned above, this problem is non-convex and we cannot analytically compute general solutions. However, we notice that by holding $\mathbf{U}$ and $\mathbf{V}$ fixed, we can solve for $\boldsymbol{\mu}$ and $\mathbf{\Sigma}$ in a closed form using standard linear regression:

$$\hat{\mu}_p = \frac{1}{N} \sum_{j=1}^{N} (z_{jp} + y_{jp}) \quad (6.5)$$

$$\hat{\sigma}_p^2 = \frac{1}{N} \sum_{j=1}^{N} (\hat{\mu}_p - y_{jp} - z_{jp})^2 \quad (6.6)$$

where $z_{jp} = \mathbf{x}_{,p}^{\mathrm{T}}\left((\mathbb{1}_{\mathbf{\Phi}})_{,j} \circ (\mathbf{U}\mathbf{v}_j)\right)$ for $j = 1, \ldots, R$ and $p = 1, \ldots, P$.

To solve for $\mathbf{U}$ and $\mathbf{V}$ for given values of $\boldsymbol{\mu}$ and $\mathbf{\Sigma}$, we use a projected quasi-Newton (PQN) method [154]. Quasi-Newton methods construct an approximation to the Hessian by using the observed gradients at successive iterations. We use the MATLAB implementation `min_PQN`[1]. There are several reasons why we chose this method instead of directly working with the Hessian. First, our set of constraints is convex and the projection on this set can be done analytically. Second, although we can compute both the gradients and Hessian of $\mathcal{F}$, the memory required to store the Hessian is often too large given the dimensions of the expression data ($O((M+R)^2K^2)$). Moreover, due to interactions between miRNAs and mRNAs, the Hessian is not necessary sparse even if both $\mathbf{\Phi}$ and $\mathbf{\Omega}$ are. During the projection step, to speed up the convergence of the algorithm, we set the entries of $\mathbf{U}$ which do not have predicted interactions to zero.

Using the updated values for $\mathbf{U}$ and $\mathbf{V}$ we once again solve for $\boldsymbol{\mu}$ and $\mathbf{\Sigma}$ and so on. These two steps lead to an iterative procedure to solve (6.4) along the lines of coordinate-descent methods. This procedure converges to the local minima due to the fact that the objective function is bounded below and the sequence of function values is monotonically decreasing and the gradients at the convergence are zeros. Since the problem is non-convex, we perform the learning process several times, randomly initializing the parameters each time. After repeating this process several times (10 iterations in our experiments), we select the parameters from the result that leads to the lowest value for our objective function.

Finally, the regularization and data type weighting parameters K, α, β, C_1 and C_2 are chosen based on an external evaluation discussed in Results.

6.3 Constraint module learning for multiple condition analysis

So far we have discussed our approach for identifying miRNA regulated modules using a condition-specific expression dataset. Although the optimization problem in Eq. (6.4) can be used with expression data from multiple conditions (e.g. different types of cancer), the output is one set of modules for all conditions. In some cases, directly identifying similar and divergent modules across conditions is an important goal. Consider for example joint analysis of multiple types of cancers. While some researchers may be interested in regulatory modules that are activated in all different cancer types, others may be interested in unique aspects, or modules, of a specific cancer type when compared to other types of cancer.

In our problem, we would like to learn a set of modules for T different conditions. The interaction input matrices $\mathbf{\Phi}$ and $\mathbf{\Omega}$ are fixed while for each condition t, we have a set of expression measurements $\mathbf{X}_t$ and $\mathbf{Y}_t$. Given this input we jointly learn T sets of modules $\{\mathbf{U}^t, \mathbf{V}^t\}_{t=1,\ldots,T}$. The number of modules is also fixed for all conditions.

This type of learning is called multi-task learning [155] in the machine learning community, where many related models are learned simultaneously using the same internal representation. Such learning allows different models (or cancer types) to share some

[1] `http://www.di.ens.fr/~mschmidt/Software/PQN.html`

parameters which improves learning while at the same time it can also identify unique parameters for specific types. In several cases such framework was shown to lead to better solutions [155]. Many existing methods proposed for multi-task learning focus on multi-output regression problems, where it is often desirable to obtain sparse solutions by performing covariate selection. They rely on regularization technique to jointly select a set of covariates that are relevant to many tasks. One can apply ℓ_1/ℓ_2 penalty of group lasso to select covariates relevant to all tasks [156].

Here we adopt the ℓ_1/ℓ_2 penalty of group lasso to regularize the modules over T conditions with the following penalty:

$$\lambda\left(\sum_{i,k}\sqrt{\sum_t (u_{ik}^t)^2}+\sum_{j,k}\sqrt{\sum_t (v_{jk}^t)^2}\right)$$

This penalty encourages entries $\{u_{ik}^t\}_{t=1,\ldots,T}$ and $\{v_{jk}^t\}_{t=1,\ldots,T}$ to be selected together which means that miRNAs and mRNAs are assigned to the same modules across conditions. Since the penalty is not differentiable at 0, we reformulate the optimization problem by moving the non-differentiable part to the constraints as suggested in [157]:

$$\begin{aligned}\min_{\mathbf{U}\geq 0,\mathbf{V}\geq 0,\boldsymbol{\mu},\boldsymbol{\Sigma},\{a_{ik},b_{jk}\}} \quad & \mathcal{F}+\lambda\Big(\sum_{i,k}a_{ik}+\sum_{j,k}b_{jk}\Big)\\ \text{s.t. } & \sqrt{\sum_t (u_{ik}^t)^2}\leq a_{ik};\quad \sqrt{\sum_t (v_{ik}^t)^2}\leq b_{ik}\\ & \|\mathbf{u}_i\|_1\leq C_1;\quad \|\mathbf{v}_j\|_1\leq C_2\\ & i=1,\ldots,M;\ j=1,\ldots,R;\ k=1,\ldots K\end{aligned} \tag{6.7}$$

Here we have introduced new variables $\{a_{ik}\}$ and $\{b_{jk}\}$ into the problem. We update the projection step in Section 6.2.6 with the projection on the new ℓ_2 norm balls in the constraint set as shown in [157] (Theorem 4).

6.4 Results

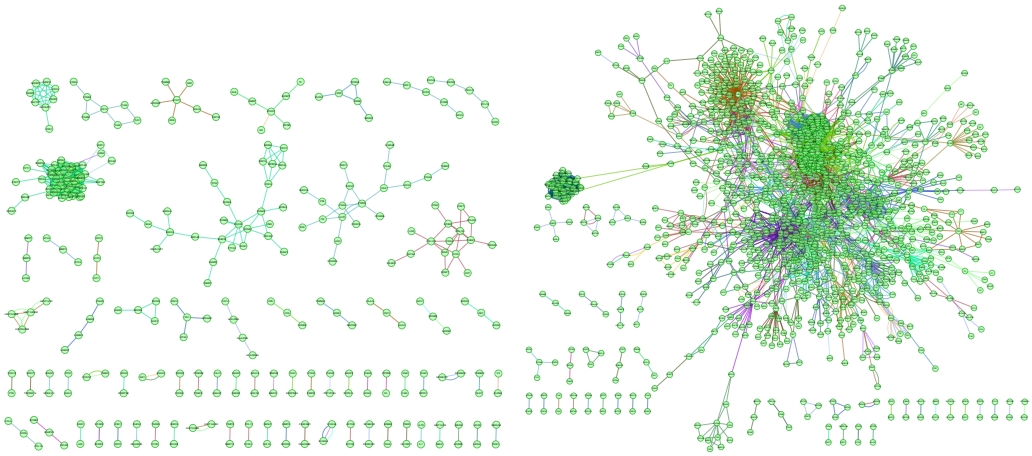
6.4.1 MiRNA regulation in ovarian cancer

To test PIMiM and to compare it with previous methods for determining condition-specific miRNA regulation (SNMNMF and GroupMiR) we use the ovarian cancer dataset from [150]. This dataset contains 385 samples from cancer patients, each measuring the expression of 559 miRNAs and 12456 mRNAs and was downloaded from the Cancer Genome Atlas data portal (TCGA)[2]. In addition to expression data, the sequence-based prediction of miRNA-mRNA interactions was downloaded from MicroCosm [153] and protein interaction data was downloaded from TRANSFAC [152]. We only use MicroCosm here to allow a fair comparison to SNMNMF which only uses this data. In subsequent analysis we use other sequence-based prediction methods as well. To evaluate the accuracy of each method, we used a set of 115 cancer miRNAs that were determined to participate

[2] `https://tcga-data.nci.nih.gov/tcga/`

in ovarian cancer in a recent review article ([158] Table 1 and 2). Using this set we compute the Precision, Recall and F1 score (the harmonic mean of Precision and Recall) of the set of miRNAs identified by each method.

The number of modules K was set to 50 for the non negative matrix factorization method (SNMNMF) as suggested in [150]. PIMiM also requires setting regularization and weight parameters α, β, C_1 and C_2. To set these we performed an iterative line search (holding 3 of the 4 parameters fixed and adjusting the value of the 4th until convergence) to determine the values of these parameters using the F1 score as the target function to optimize. Based on this analysis we selected $K = 40$ for PIMiM (See Figure D.3 for details). SNMNMF was also run with the optimized set of parameters and input data described in [150]. Unlike PIMiM and SNMNMF, GroupMiR uses a nonparametric Bayesian prior for the number of modules and so this number cannot be fixed in advance. Thus, for GroupMiR we report modules and interactions with posterior probability at least 0.3 to get a set of comparable size to other methods. Previously, GroupMiR was shown to outperform several other methods [115] including GenMiR++ [128] and so we omitted comparison to these methods here. Figure 6.2 presents a graphical view of the modules identified by PIMiM and SNMNMF. We color interaction edges between genes using different colors for each module. The modules identified by PIMiM are more dense and so are in better agreement with previous findings regarding the regulation of interacting proteins by miRNAs.



(a) SNMNMF (b) PIMiM

Figure 6.2: Interactions between genes of the modules. We show an edge between two genes if they are members of a module and their interaction exists in the database. Each color corresponds to one module. Genes with no edges are omitted to improve visualization.

Evaluation: identifying cancer miRNAs.

We first looked at the set of miRNAs identified by each method (those belonging to the modules returned by each of the methods). The results in Table 6.1 demonstrate that using the protein interaction data greatly increases precision, recall and the F1 score. Both methods that use this data (PIMiM and SNMMNF) clearly outperform GroupMiR on this set. In addition, using a regression model also helps as indicated by the increase in F1 score PIMiM obtains over SNMNMF.

Table 6.1: Evaluation of all methods on the ovarian cancer dataset. The expression correlation values and number of genes are averaged across modules. Expression correlation: the correlation of expression values of miRNAs and mRNAs.

	Cancer miRNAs			Expression Correlation	#genes / module
	F1	Precision	Recall		
PIMiM	**0.3768**	**0.3230**	**0.4522**	-0.0131	67.80
SNMNMF	0.3588	0.3197	0.4087	0.0745	79.26
GroupMiR	0.1227	0.2083	0.0870	**-0.0408**	54.82

Expression coherence.

In addition to analyzing the set of identified miRNAs we also computed the average anti-correlation between miRNAs and mRNAs in the modules identified by each of the methods (Table 6.1). In this analysis, GroupMiR achieves the highest anti-correlation between miRNAs and the mRNAs they regulate in a module. This is the result of a much smaller module size identified by GroupMiR. Since protein interactions are not used, mRNAs in these modules are selected because they are strongly anti-correlated with the miRNAs predicted to regulate the modules. This requirement leads to smaller modules and a better (anti) correlation between miRNAs and mRNAs. Still, PIMiM improves upon SNMNMF in identifying anti-correlated miRNA-mRNA pairs. SNMNMF's objective function does not explicitly include a component for expression anti-correlation between miRNAs and mRNAs, which may explain why it does not capture the inhibitory role of miRNAs. Thus, PIMiM provides a useful compromise between relying strongly on protein interactions which improves accuracy and using the observed expression values in a regression setting.

MSigDB and Gene Ontology (GO) enrichment analysis

To test the biological function of the modules we looked at the Gene Ontology enrichment analysis for mRNAs in the modules identified by the different modules using TopGO [160] (which uses the Fisher count statistics, reporting up to 100 enriched terms for each module). We also used 880 gene sets of canonical pathways (C2-CP, v.3.0) from MSigDB [159]. We

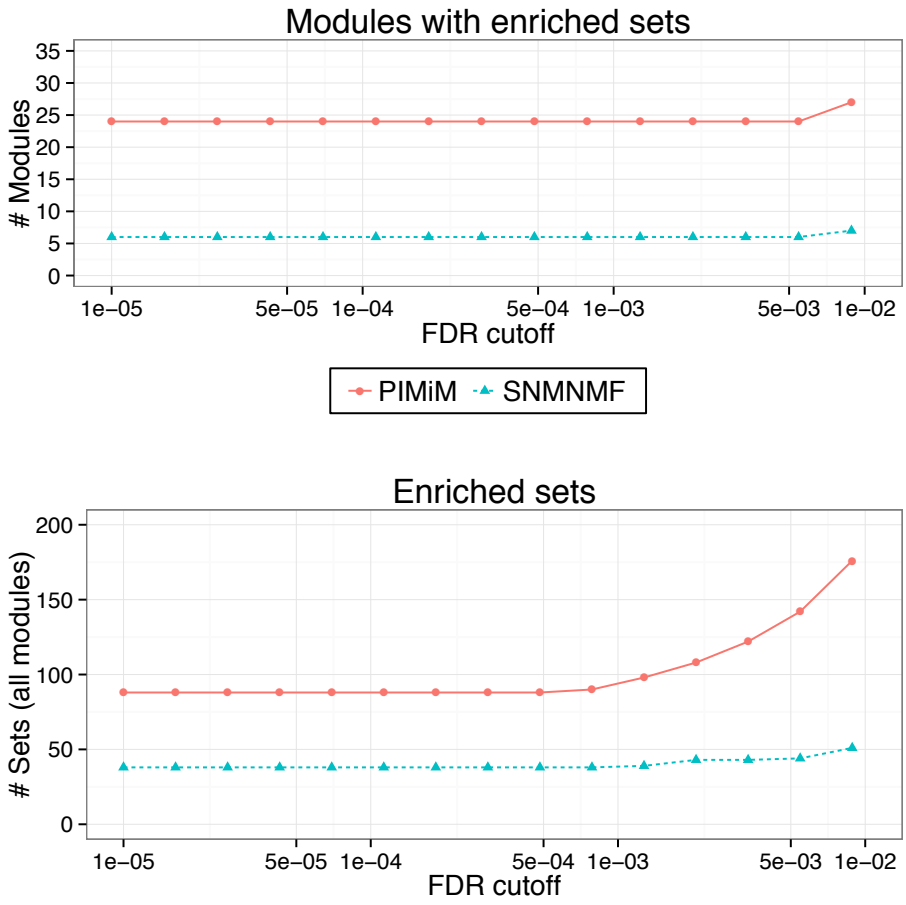


Figure 6.3: MSigDB enrichment analysis. Pathway enrichment analysis was done using 880 gene sets of canonical pathways (C2-CP) from MSigDB [159]. P-values were computed using hypergeometric test (with 10000 random permutations) on the intersection of the set of genes in each module with MSigDB gene sets. Benjamini-Hochberg procedure was used to control the FDR rate. Top: Number of modules significantly enriched for at least one MSigDB category for different significance cut-offs. Bottom - number of MSigDB categories identified as in enriched in at least one of the modules for different significance cut-off.

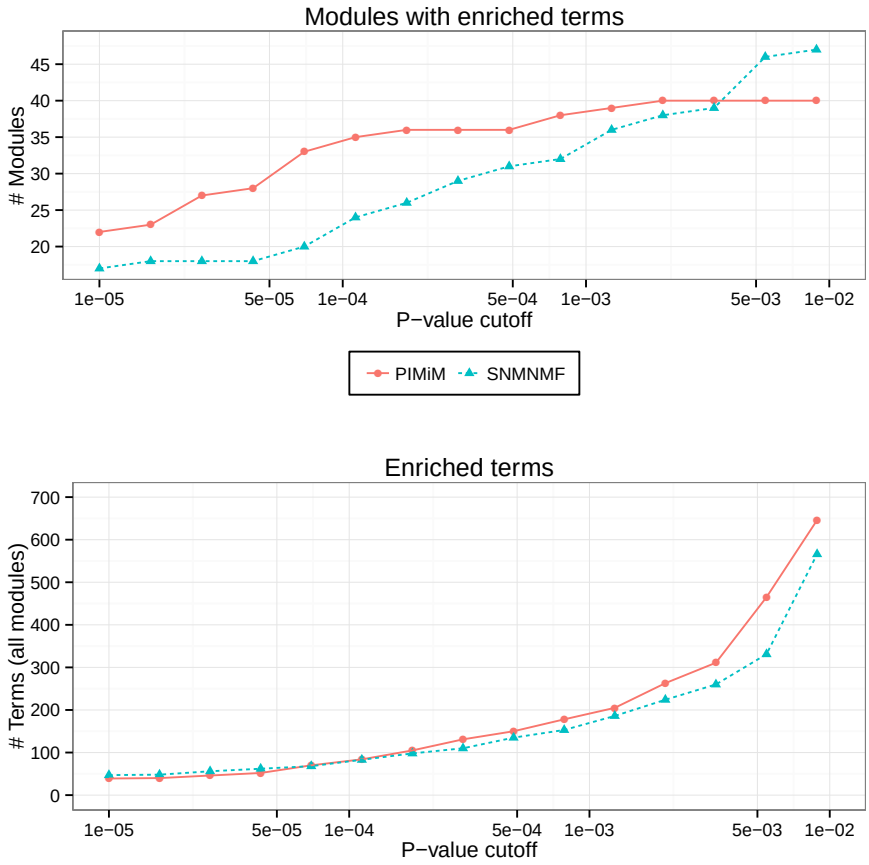


Figure 6.4: Gene Ontology (GO) enrichment analysis. We used topGO [160] with the Fisher count statistics to perform GO enrichment analysis.

used the hypergeometric distribution to compute enrichment p-values for each of the modules with each of the MSigDB gene sets. To correct for the multiple hypothesis testings we used the Benjamini-Hochberg procedure implemented in the R function `p.adjust` which computes a q-value for each intersection. The results are presented in Figure 6.3 and 6.4 which depicts the number of modules with at least one enriched set in the MSigDB or GO enrichment analysis and the total number of unique enriched GO terms or gene sets. PIMiM outperforms SNMNMF, achieving both better enrichment for individual modules and better coverage of different MSigDB sets. MSigDB pathways are biased towards cancer pathways and so may be more relevant for the data we are analyzing here than Gene Ontology analysis. In addition to cancer hits, top hits for MSigDB include signatures for Beta cells that have been linked to cancer [161] and several translation related categories.

The effect of β on the performance of PIMiM

To test the effects of using the protein interaction data in PIMiM, we re-run PIMiM with different β values. The results are presented in Figure 6.5 As the figure shows, when decreasing the value of β, the performance of PIMiM on all evaluation metrics decreases indicating the PPI data is useful for identifying coherent modules. On the other hand, increasing β too much leads to very high weight for PPI data at the expense of the expression information which also negatively affects the performance of PIMiM. Thus, balancing the two data types, which is done by setting an intermediate value for β is key to the success of PIMiM.

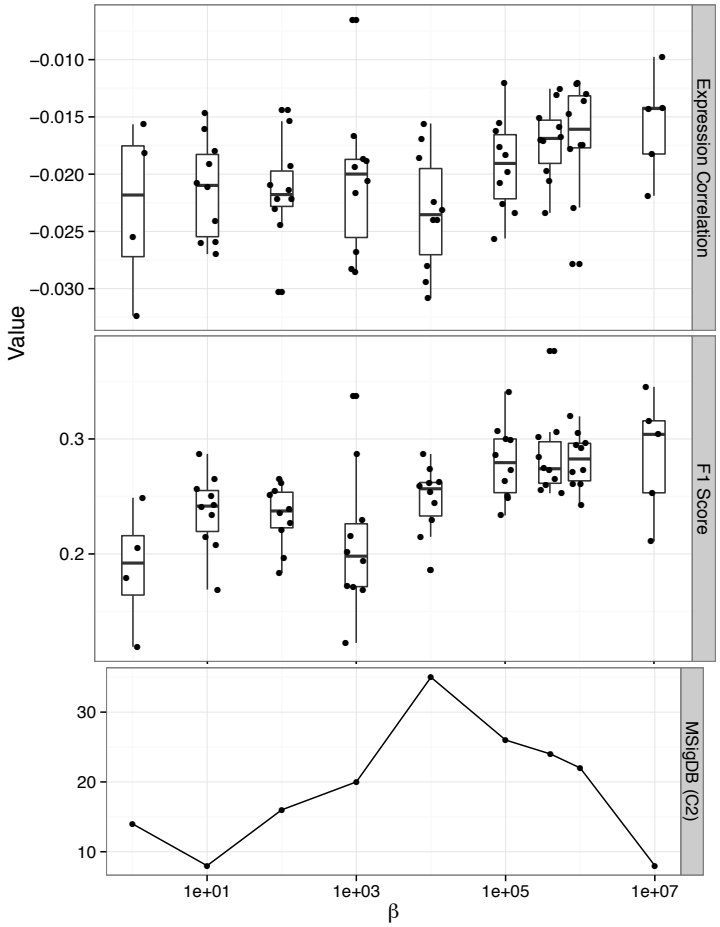


Figure 6.5: The effect of protein interaction data to the result. We varied the value of β and tested the different metrics: box plots are shown for different initializations, MSigDB result is only shown for the initialization leading to the best likelihood value. As can be seen, both high and low values lead to reduced performance.

6.4.2 Integrating data from multiple types of cancers

To further investigate miRNA control of different cancers, we applied PIMiM to a dataset of three cancer types using the multi-task learning framework described in Section. 6.3. We learn three sets of modules for three types of cancer: Breast invasive carcinoma (BRCA), Glioblastoma multiforme (GBM) and Acute Myeloid Leukemia (AML). The miRNA and gene expression profiles of 89 BRCA, 498 GBM and 173 AML patients were downloaded from the TCGA. This set has 285 miRNAs and 10922 mRNAs in common. Here we combine the miRNA-mRNA predicted interactions from three public databases (MicroCosm [153], miRanda [162] and TargetScan [163]) and protein interaction data from TRANSFAC [152]. For each cancer type, PIMiM learns one set of 50 modules. The parameters were set by optimizing for the F1 score of identifying miRNAs relevant to this dataset based on the set of cancer-related miRNAs from [158]. Figure 6.6 displays the miRNA regulating modules in all three cancer types.

Table 6.2: MiRNAs specifically identified for a cancer type.

MiRNAs	Predicted type	BRCA	GBM	AML
hsa-miR-663	BRCA	[164]	-	-
hsa-miR-433	GBM	-	[165]	-
hsa-miR-99b	AML	-	-	[166]

Analysis of identified miRNAs

Several of the modules identified by PIMiM are regulated by known cancer miRNAs. The overall F1 score for cancer miRNAs for the joint analysis was high for all three cancer types: BRCA (0.6167), GBM (0.5789) and AML (0.6111). Well known cancer miRNAs reported by PIMiM include the let-7b/c/d/e (active in BRCA: [167], GBM: [168] and AML: [169]), mirR-302a/b/c/d cluster (suppression of the CDK2 and CDK4/6 cell cycle pathways [170]) and miR-96 (active in BRCA:[171], AML:[172]), miR-34a (active in BRCA: [173], GBM: [174], AML: [175]) , miR-15a/b (active in AML: [176]). Some members of the miR-17-92 cluster (miR-18b, miR-19a, miR-20a/b, miR-93) are also identified by PIMiM (active in BRCA: [177], GBM: [178], AML: [179]). Note that some well known cancer miRNAs including miR-17 and miR-92 are missing from the modules because their expression is not available for enough of the samples. Several other subsets of miRNAs were assigned to cooperatively regulate modules in multiple types of cancer as shown in Figure 6.6.

Cancer specific miRNAs

In addition to finding common cancer regulators, PIMiM can be used to identify cancer type specific regulators. These can either be used as biomarkers for a sub-type or can be studied to determine the unique properties of each cancer type. While it is very hard to obtain negative information (i.e. a paper that mentions that a certain miRNA does

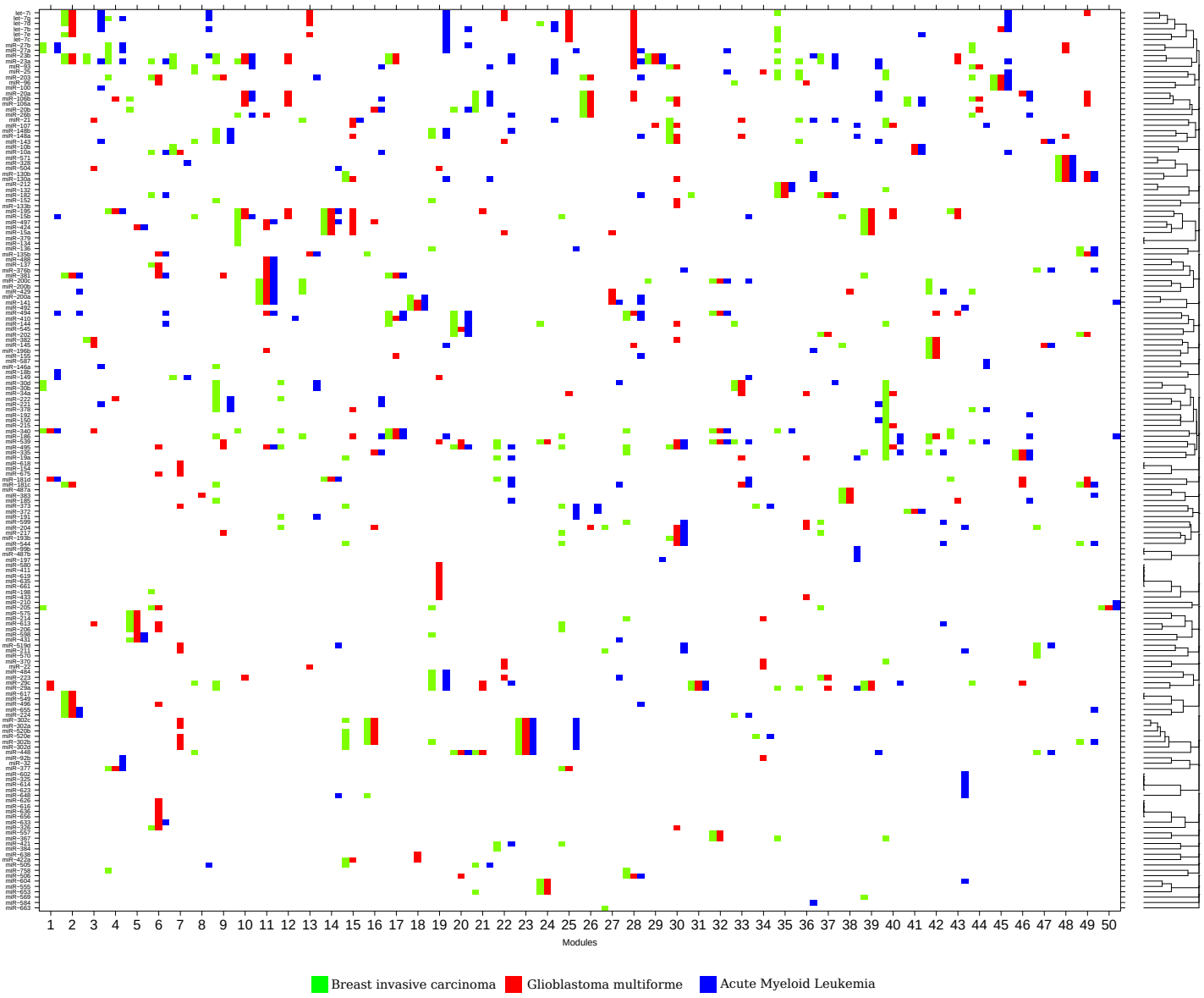


Figure 6.6: Inferred miRNA modules of the three cancer types (BRCA, GBM and AML). The x-axis shows the 50×3 modules learned for the three cancer types (each x-axis bar is subdivided into 3 with the color corresponding to the cancer type). The y-axis shows miRNAs ordered by hierarchical clustering of their module membership vector. In several cases the same miRNAs are predicted for all or two of the three cancer types.

not regulate a specific cancer type) several of the predictions made by PIMiM agree with current literature that, at least so far, only mentions their role in the cancer they were assigned to by PIMiM. Table 6.2 lists a few of these miRNAs and the cancer type they were predicted to regulate.

Analyzing the miRNAs and mRNAs in identified modules

In addition to identifying important miRNAs for this particular study, PIMiM returns a set of modules providing predictions of cooperative regulation of miRNAs and their mRNAs targets. To demonstrate the informative power of this modular structure, we analyze in more details one of these modules.

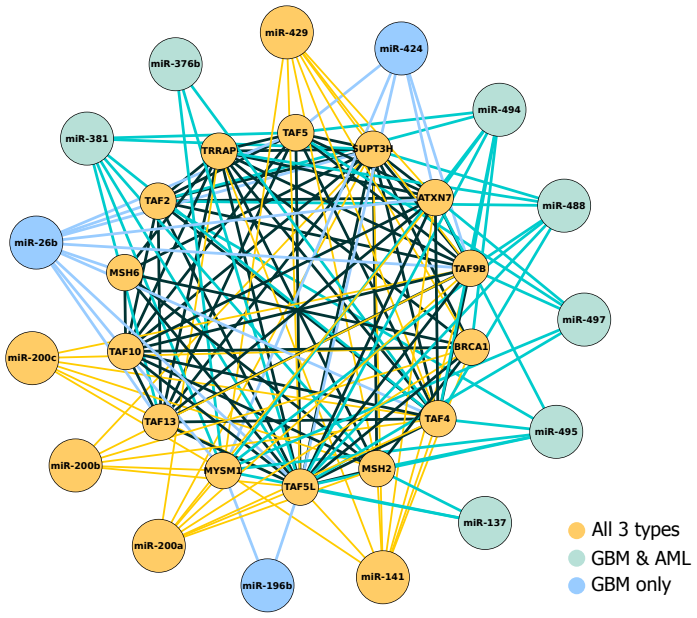


Figure 6.7: MiRNAs and mRNAs assigned to Module 11 in all three cancer types. Color indicate the specific cancer type for which the mRNA or miRNA was selected as part of the module.

Module 11 Figure 6.7 depicts a network of miRNAs and mRNAs identified as part of Module 11. Across all cancer types, PIMiM identified a set of 14 strongly connected proteins. MiR-200a/b/c, miR-141 and miR-429 are predicted to regulate this set of mRNAs in all types of cancer. These miRNAs have previously been reported to play a role in cancer and cell proliferation [180, 181]. Interestingly, the miR-200 family is located in two chromosomal regions on 1p36.33 (200b, 200a and 429) and 12p13.31 (200c and 141), respectively [182], which may support our prediction of their cooperative regulation. Applying Gene Ontology analysis (using FuncAssociate [183]) and MSigDB enrichment analysis to the set of 14 mRNAs in this module indicates that this set is enriched with

members of transcription factor TFTC/STAGA and TFFIID complexes. Recent findings support the link between between these complexes and cancer [184]. This module also includes a tumor suppressor gene MSH2 [185] and a famous breast-cancer susceptibility gene BRCA1 [186].

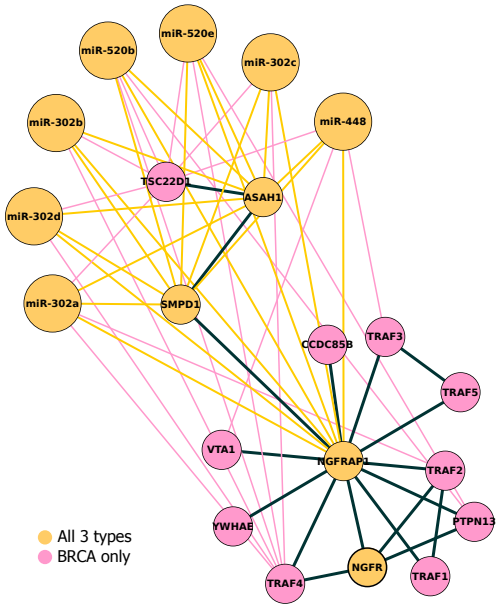


Figure 6.8: Network of miRNAs and mRNAs of Module 23.

Module 23 This module (Figure 6.8) includes the miR-302 and miR-520 clusters. These two clusters are shown to display similar expression pattern in the differentiation of human embryonic stem cells [187]. Specifically, the miR-302 family is known for coordinately suppressing genes in the CDK2 and CDK4/6 cell cycle pathways [170]. Indeed, miRNAs in the miR-302 family were assigned to the same module by PIMiM indicating that the module-based approach can help in recovering cooperative regulation of groups of miRNAs. Among the top terms and gene sets from Gene Ontology and MSigDB enrichment analysis are: cell death, CD40 receptor complex, regulation of apoptosis, B cell immune response, TNF receptor signaling pathway, … (Table D.2).

Module 48 (Figure 6.9 and Table D.3) All miRNAs in this module were previously reported as active in cancer: miR-130a/b [188], miR-328 [189] and mirR-504 which negatively regulates tumor suppressor p53 [190]. Mutation of the gene hub CEBPE is shown to increase the risk of acute leukemia[191].

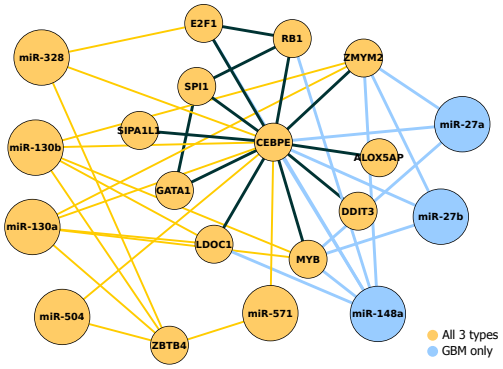


Figure 6.9: Network of miRNAs and mRNAs of Module 48.

6.5 Conclusions

We presented PIMiM, a new method for inferring condition-specific regulation of miRNAs and for identifying their targets. PIMiM combines sequence, expression and interaction data to discover miRNA regulated modules of mRNAs. We use a probabilistic model that combines regression with network information to discover these modules. We developed an iterative learning procedure to learn the parameters of our model and a multi-task learning method for combining data from multiple conditions.

We tested PIMiM on ovarian cancer expression data and have shown that it can identify miRNAs regulating this cancer type and that it is able to group relevant genes together. Comparison to other methods indicates that by using protein interaction data we can improve accuracy while at the same time PIMiM also maintains expression coherence among mRNAs and anti-correlation between miRNAs and the mRNAs they are predicted to regulate improving upon previous methods that have also used protein interaction data. Application of the method to compare and contrast three types of cancer identified both common and unique regulators, which can allow researchers to determine the core cancer regulatory network and the differences in regulation among the various cancers we studied.

While we believe PIMiM can already be of use to researchers that collect mRNA and miRNA expression data, there are a number of extensions that can further improve it. As mentioned above, we follow several other papers in isolating the miRNA target prediction task from the combinatorial analysis of miRNA-TF regulation. While such an approach leads to good results as discussed above, our longer term goal is to develop a method that can incorporate both types of regulation in a single modeling framework. For this, we would need to determine the role a specific TF plays (activator or repressor) and its activity level (either based on its expression levels or on the set of its targets [192]). With this information we can incorporate TFs into our regression model to account for their part in regulating expression which will hopefully lead to better results regarding the role played by specific miRNAs. In addition, we would like to incorporate additional types of high-throughput data, for example epigenetic data to our analysis framework.

7 Conclusions and Future Work

7.1 Conclusions

This thesis proposes solutions to challenges in analyzing gene expression data using probabilistic models. Computational tools introduced here enable researchers to collect and preprocess RNA-Seq data, to conduct cross-species analysis to find coherent patterns of expression of sets of genes, and to predict gene targets of miRNAs active in specific conditions of interest.

Chapter 2 presents SEECER, which we believe is the first error correction method for generic RNA-Seq data. Previous methods either assume a uniform coverage of reads [36, 37, 38, 39, 40] or were developed for a specific type of sequencing technology such as single molecule sequencing [193]. We demonstrated using three datasets of different sequencing depth and read length that SEECER outperforms other methods in correcting more errors while making a smaller number of false positive corrections. More importantly, *de novo* assembly of corrected reads leads to a more accurate transcriptome. We observed an improvement across the spectrum of expression levels, notably in lowly expressed genes which are more difficult to analyze. The method was used to perform *de novo* transcriptome correction and assembly of sea cucumber expression data providing new insights regarding the development of this species. We were also able to experimentally validated 14 highly expressed transcripts by RT-PCR analysis.

Even though expression data has become available for increasing number of species in recent years, it is still difficult to search for similar experiments in large expression databases across species. In Chapter 3, utilizing the ranking of orthologs in two species, we proposed a distance function which is learned using a training dataset of known similar pairs of experiments. To demonstrate its ability, we retrieved similar experiment pairs from GEO and asked a pathologist to evaluate their relevance. We also determined a set of mouse genes which are most coherent in expression values with a set of human cell cycle genes in these pairs of experiments. The results suggest that the identified pairs are meaningful and the set of mouse genes are significantly enriched with cell cycle related genes. We believe this is a helpful step toward building a query system of cross-species expression databases. Since the publication of our work [71], some approaches were introduced along this line of research such as [194] and [195], the latter of which was built upon the work presented in this thesis.

Building on our experience with cross-species analysis, Chapter 4 presents DPMMLM, a method for discovery of core and divergent sets of genes between two species. Most cross-species analyses assume a one-to-one mapping between genes in two species. This mapping is usually determined by a top match, for example sequence similarity, which in many cases is not the correct ortholog. While this assumption is acceptable for high level analysis such as querying large databases (Chapter 3), it may lead to wrong conclusions in other situations. To address this issue, DPMMLM allows soft matching of genes based on a prior given by sequence similarity and infers the best matching probabilistically using both the prior and the observed expression data. The method uses the Dirichlet Process to guide the selection of the number of sets of genes to report. The fact that DPMMLM

performed better on both simulated and immune response data suggests that probabilistic matching is suitable for cross-species analysis.

Lastly, we discuss in Chapter 5 and 6 two methods that integrate expression data with other evidence sources to infer miRNA regulatory networks. Determining such target set is important for fully understanding the role of various miRNAs, and to model the networks they regulate in a condition of interest. Since the effect of each miRNA on any single target is often limited, it often works cooperatively with multiple miRNAs targeting the same mRNA in a specific condition. To allow the use of such group- or module-based regulatory models, we introduced GroupMiR [115] which uses a nonparametric Bayesian prior based on the Indian Buffet Process (IBP [15]) to identify modules of co-regulated miRNAs and their target mRNAs. As we have shown, by using a module-based approach, we can improve upon methods that treat miRNAs or mRNAs individually, improving the set of correctly recovered miRNA-mRNA interactions [115]. With growing evidence that interacting proteins are more likely to be co-regulated by the same miRNAs, we extended GroupMiR in Chapter 6 to discover miRNA targets that are connected and participate in common pathways. We did so by formulating an optimization problem in PIMiM using an additional data source: protein-protein interaction data. Finally, we applied PIMiM to study miRNA regulation in several types of cancer, allowing us to identify novel regulators that either span multiple cancer types or are unique to specific cancers.

7.2 Themes shared by the methods in this thesis

In summary, these methods are successful for many reasons. Foremost, while developing these methods, we carefully considered specific data characteristics and incorporated these considerations in the probabilistic models. SEECER (Chapter 2) improved upon other existing methods for error correction significantly by adaptively and locally constructing alignments of reads without any assumption on the coverage of data. Other methods make many false negative error corrections resulting in lower rates of aligned reads because they assume uniform coverage. DPMMLM, which probabilistically matches genes, enables the discovery of gene sets that otherwise were masked because of incorrect orthology assignments. GroupMiR and PIMiM are superior in identifying key miRNA regulators because they specifically model the biological group structure of miRNA regulation, hence adhere more closely to the biology. We also want to emphasize the importance of abstraction in modeling these problems to make it possible to scale these methods to large biological data. Although it may be computationally attractive to design complex and rich models, striking a balance between complexity and data interpretation is challenging.

Secondly, a common theme running through this thesis is integrating many evidence sources: sequence data, prediction databases, sequence similarity scores, and protein interaction data. The main idea is that each source of data tells us a different aspect of the biological picture. Sequence data, which is static and less noisy than expression data, is useful to overcome the noise level. Prediction databases, which are built on many existing approaches based on conservation and sequence analysis, increase the confidence of our results. Utilizing protein interaction data in the work described in Chapter 6 was supported by biological evidence and experimental results. Combining many datasets such as the work on cancer data (Chapter 6), where we apply PIMiM to

compare and contrast three types of cancer, may lead to interesting hypotheses. This could allow researchers to determine the core cancer regulatory network and the differences in regulation among the various cancers.

Finally, preprocessing is an important factor leading to a successful use of expression data. For instance, a common practice in working with sequencing data is discarding reads with no or multiple alignments to a reference genome. By improving the quality of RNA-Seq data and the mappability of reads to the reference, SEECER should benefit many downstream analyses in data utility. One particular instance we have shown in this thesis is *de novo* transcriptome assembly, in which case, the assemblers could produce longer contigs yielding a more compact transcriptome.

7.3 Future work

We wrap up the thesis with some directions for future research on these topics. In general, we would like to add additional features to SEECER and investigate their effects on variant analyses such as SNP calling. SEECER can also be applied to other types of sequencing data. We also suggest using other information sources to extend the work in Chapter 3-6. A longterm direction is to include non-linear dynamics of expression data in these models.

7.3.1 Additional features for SEECER

Sequence data includes a quality value for each base of reads. This integer value indicates the level of confidence when a base is called by the sequencing machines and processing tools. Because quality values help identify erroneous locations in the reads, methods treating these locations with different levels of confidence can lead to more accurate assessment of the data [196, 33]. Many genome error correction methods already use this information to guide the search and remove sequencing errors [20, 34, 38]. The current version of SEECER does not use quality values in building contigs and estimating the HMM models. Quality values could be used to discard bad alignments, improving the filtering step of SEECER. We could also use them in estimating parameters by assigning weights to different alignments proportional to their qualities. These additions should result in a smaller number of false positive corrections made by SEECER. Especially in the regions of low coverage, rather than discarding these reads (because of few alignments), SEECER could construct a contig if the alignments are of high quality.

Currently, SEECER only supports outputs in FASTA format, which does not include quality values. SEECER does not compute and output quality values because it is unclear how to assess the quality of corrections (mismatches and indels). One option is to scale the likelihood of each read appropriately and use this as a quality value. For deletion corrections, we may also want to downgrade the quality values of adjacent bases to indicate a low confidence region. A more complicated option is to consider the quality of each base in addition to the likelihood of alignment; if these values disagree substantially, we should assign a low quality value.

7.3.2 Effect of error correction on SNP and other analyses

One of the major concerns in error correction is controlling the number of false positives. This is particularly important for sensitive analysis such as SNP calling or variant detection. There is possibility that SEECER makes some corrections that remove heterozygous SNP or other true variants in the data. This issue is difficult because low-coverage SNPs or variants are inevitably removed by most of the error correction methods since it is impossible to distinguish them from errors. We provide some crude assessment on the number of SNPs existing before and after error correction in Section 2.4.3. The current implementation of SEECER uses a simple heuristic dictating that correction is only made if by doing so the likelihood is increased by at least a certain margin. Nevertheless, we need to investigate the effect of error correction on SNP and variant detection more systematically and thoroughly. With the growth of SNP databases [197], this type of analysis could be more easily done in the future. In addition, it has been shown that *de novo* assemblies allow reliable detection of genes that are differentially expressed between two conditions [70]. Thus, by improving the resulting assembly, SEECER is likely to improve downstream differential expression analyses as well.

7.3.3 Extending SEECER to other data: Chip-Seq data

SEECER makes few assumptions about a particular sequencing technology. Most notable is the non-uniform coverage, which many other types of sequencing data possess. Many new experimental methods use deep sequencing technology as a way to analyze biological samples. Thus, SEECER can also be applied to these types of data.

ChIP-Seq [198], a technology combing immuno-precipitation and deep sequencing, is one potential candidate. The main application of ChIP-Seq is to survey interactions between proteins and DNA or to study histone modifications. The current processing pipeline starts with aligning reads to a reference genome. Then, the locations of enriched binding sites, called peak locations, are determined based on the abundance of reads. Sequences around these peak locations are extracted from the genome and then used as input to a motif discovery tool. SEECER can already be used to help align more reads to the reference, hence improving this processing pipeline. However, in some cases where there is no reference genome, we may be able to apply SEECER to ChIP-Seq data. We note that SEECER produces contigs from partial overlaps of reads as intermediate results. For ChIP-Seq data, where there is no alternative splicing events, most of these contigs would likely overlap with sequences identified by the peak locations. If we find the same motifs in these contigs as in the genome sequences, we can establish a new pipeline to analyze ChIP-Seq without a reference genome. This new technique opens the door for applications of ChIP-Seq data in new organisms for which no draft genome exits. Studies of cancer samples, in which mutations or genomic rearrangements make the reference unreliable, should also benefit from this new pipeline.

7.3.4 Improving cross-species analysis

Despite our encouraging results in Chapter 3, expression data itself seems insufficient for building a good retrieval system of cross-species expression databases. Expression

studies usually contain many samples corresponding to different time-points or stimuli. Our experience shows that while it is usually not difficult to retrieve pairs of studies that are similar, our method is incapable of discriminating among samples within these pairs. We envision a system that works more accurately at a finer granularity. A promising direction is to integrate other evidence sources into the model such as text information. One example is proposed in [195]. Their method uses our distance function coupled with a text-based classifier in a co-training framework to exploit the complimentary information between text and expression data, hence were able to improve the performance. There has been recent work in Machine Learning on developing distance functions or classification algorithms for distributions [199, 200]. Since each expression sample can be thought of as a distribution of expression values, we could apply some of these methods to our problem. We note that distributions of expression values are multi-modal and this fact should be taken into account properly.

The latent matching component is important to the success of DPMMLM. However, the matching is biased to one species. That means we only try to map genes of one species onto the other species. While this may result in many-to-many matchings, the probabilities are only shared among genes in the second species. A full treatment of this matching problem would be to match a groups of genes in both species together. Scaling DPMMLM to multiple species is also another interesting research problem.

7.3.5 MiRNAs, Transcription factors and combinatorial regulation

In addition, transcription factors (TFs) also play a major role in regulating gene expression and they have been shown to work combinatorially with miRNAs [201]. Our longterm goal is to develop a model that takes into account both TFs and miRNAs. For this, we would need to determine the role a specific TF plays (activator or repressor) and its activity level (either based on its expression levels or on the set of its targets [192]). With this information, we can incorporate TFs into our regression model to account for their part in regulating expression, which will hopefully lead to better results regarding the role played by specific miRNAs.

Moreover, the regression component that we considered in PIMiM uses a simple linear model to explain the regulation effect of multiple miRNAs. We could also extend this to incorporate other complex combinatorial analysis. MicroRNAs regulate their mRNA targets by base pairing. In plants, nearly perfect pairing of a miRNA to its mRNA target leads to mRNA cleavage [202]. We hypothesize that in other organisms, an mRNA can have multiple binding sites corresponding to many miRNAs. Any of these miRNAs that binds to the target is enough to trigger mRNA cleavage. We can think of this scenario as OR-like regulation of multiple miRNAs, where a change in expression of any of the miRNAs may lead to the negative regulation of the mRNA. One way to accommodate this phenomenon in a model is using the maximum of over-expression levels of multiple miRNAs to explain the down-expression of the target mRNA. On the other hand, while cleavage seems to be restricted to perfect pairing, translational repression may happen even when partial complementarity to mRNA occurs within the 3′ UTR [116]. Binding of a single miRNA may not be enough for repression and research [203] shows that multiple target sites can increase the level of translational repression. In this case, multiple miRNAs can cooperatively mediate the regulation of the same mRNA. This scenario corresponds to

an AND-like mechanism, where multiple miRNAs are necessary for meditating regulation of a transcript.

Finally, we would like to continue our work on cancer, where other genetic variants and mutation information can be incorporated into the model to reduce false predictions (for example, [204]).

Supplementary materials for Chapter 2

A.1 Detailed analysis of false positive and false negatives after TopHat alignment with the human data

Here we present a more detailed analysis of the number of false positive and false negative corrections on the 5 lane human data for all three investigated tools. The analysis is restricted to the set of reads that was uniquely aligned by all the methods tested for a dataset, to be fair, but the relative values of gain, sensitivity and specificity are similar even without this restriction. Table A.1 lists the number of true and false positives as well as true and false negatives for bases of aligned reads compared to the human reference sequence and the read sequence in the original experiment on the 5 lane human data. Among all methods, SEECER and Echo have the highest number of true positives. Echo has a larger number of true positives but it makes much more false positive corrections, about ~ 9 times more than SEECER. Therefore, despite a better sensitivity for Echo, SEECER provides the largest gain among all methods. Gain values for SEECER are roughly twice as high compared to Quake that has the second best gain value for all methods. All methods have similarly high specificity values. SEECER yields the largest gain in the other datasets we tested (Table A.2 and Table A.3).

A.2 De novo assembly results by expression

In order to understand the impact of error correction on *de novo* transcriptome assembly with Oases [27], we have computed the reconstruction accuracy (with and without correction) as a function of the expression levels. We used the express software with default parameters ([205] version 0.9.4) to quantify the expression of Ensembl (version 65) transcripts after alignment of the original reads with bowtie [66]. In Figure A.1 the results for all tested preprocessing methods are depicted. In general SEECER improves reconstruction for a wide range of expression levels compared to the original data, but most notably for the most highly expressed transcripts. The read clustering with SEED, gives good results, given that the number of reads is significantly reduced, and for the most highly expressed transcripts assembly improves over the original data for full length transcripts. The other genome error correction approaches, except HiTEC, generally lead to an improvement of reconstructed transcripts but all of them have the biggest gap compared to SEECER in the two second highest expression quantiles. This result demonstrates that genome error correction approaches have reduced sensitivity for the correction of errors in very highly expressed transcripts.

A.3 Detailed analysis of types of corrections made by SEECER

SEECER can make mismatch, insertion and deletion corrections to the reads. Figure A.2 shows the distribution of these three types of correction in terms of the read positions for the 55M paired-end 45bps reads of human T cells. As can be seen, mismatch corrections

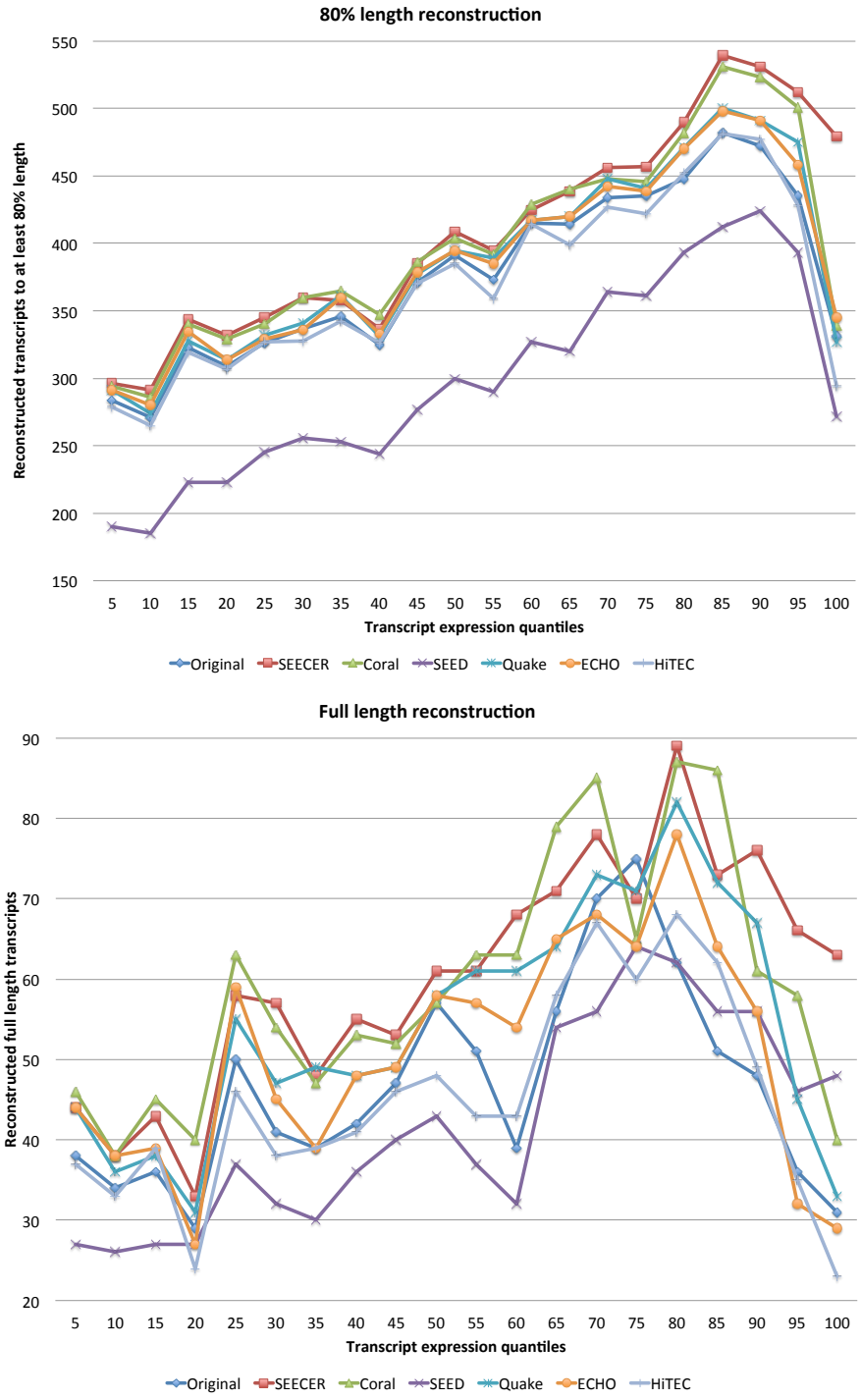


Figure A.1: Analysis of transcript reconstruction accuracy (y-axis) according to expression level by grouping Ensembl transcripts (v.65) with similar expression into quantiles of the same size (x-axis). The assembly performance with Oases on the original data is compared to preprocessing methods, SEED, ECHO, Quake,HiTEC, Coral and SEECER. The number of Ensembl transcripts covered to at least 80% (top) and full length (bottom) are shown.

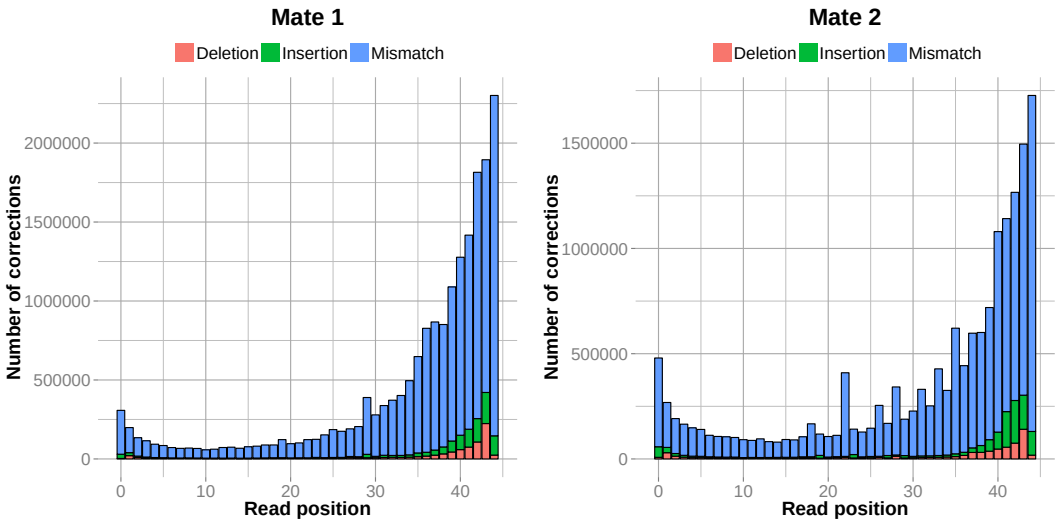


Figure A.2: The number of corrections that SEECER made to the 55M paired-end 45bps reads of human T cells.

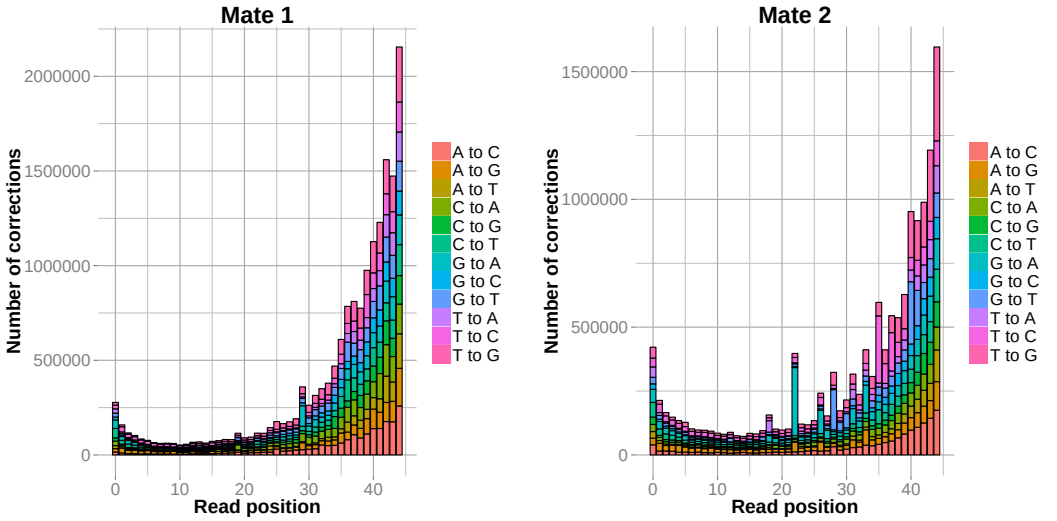


Figure A.3: The number of different types of *mismatch* corrections that SEECER made to the 55M paired-end 45bps reads of human T cells.

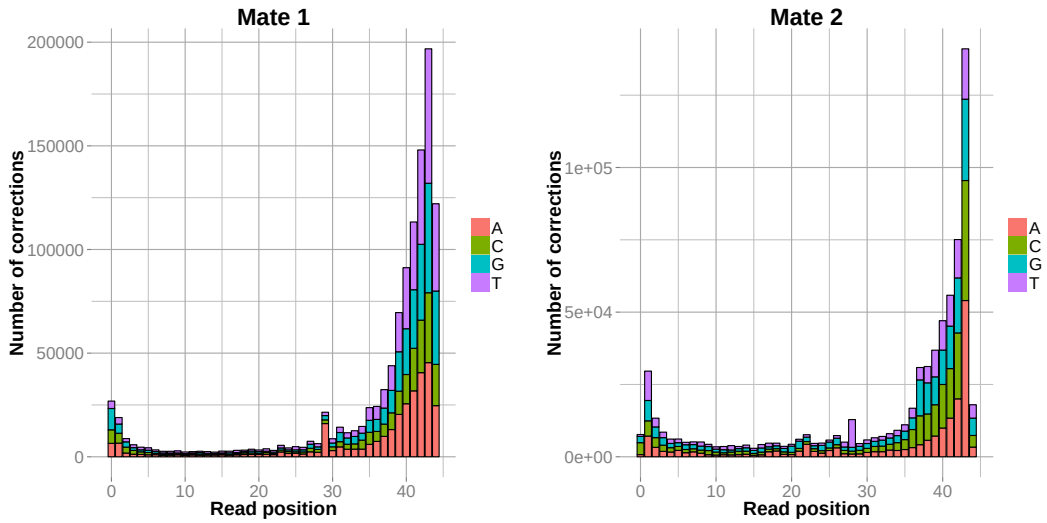


Figure A.4: The number of different types of *insertion* corrections that SEECER made to the 55M paired-end 45bps reads of human T cells.

are more common than the other types. In agreement with the fact that Illumina reads accumulate errors at the ends of reads [33], SEECER made many more corrections at the read ends.

We also show the frequency of different types of mismatch, insertion and deletion corrections in Figure A.3, A.4 and A.5. We observe that there are significant biases in substitution corrections such as A to C versus A to G/T, T to G versus T to A/C. Again recapitulating what was observed before for Illumina data [33].

A.4 Factors affecting running time of SEECER

Table A.5 summarizes runtime properties of SEECER when using different number of reads and read lengths. We fixed the sequencing throughput (total megabases = # reads $\times$ read length) to the same value of the T cells dataset by subsampling reads from the other two datasets (Hela cell lines and IMR90 cell lines). We find that different factors affect the run time and that it is hard to determine in advance how such factors will materialize in specific experiments:

1. The complexity of the transcriptome. This affects the amount of collisions and duplicated computational work due to random seeding performed by the algorithm.

2. The number of resulting contigs (which depends both on the species and the condition studied). Length of contigs rather then read length increases the run time. Thus, while the table can provide a rough guide as to what to expect when running

SEECER on comparable datasets, runtime is heavily experiment dependent and it is hard to interpret the running time as a function of read number or length.

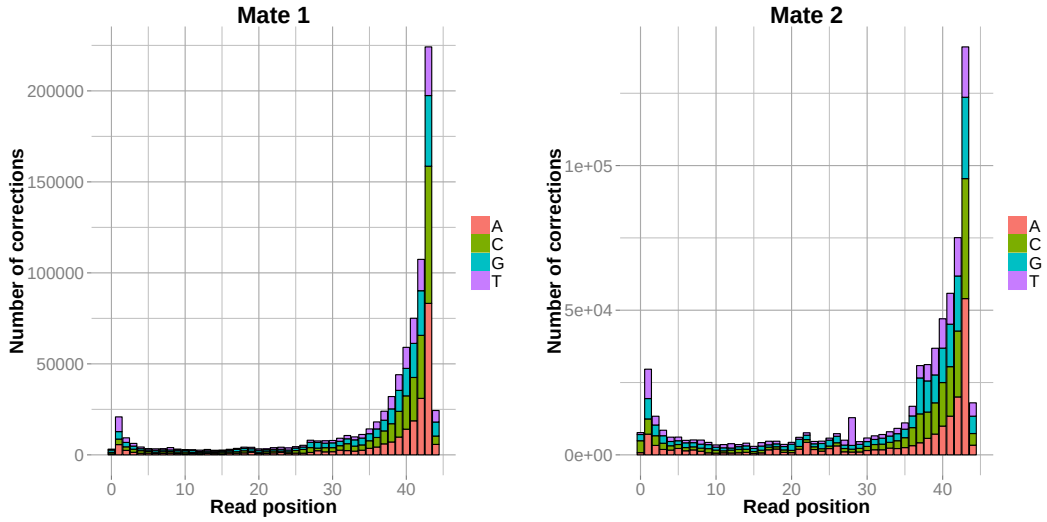


Figure A.5: The number of different types of *deletion* corrections that SEECER made to the 55M paired-end 45bps reads of human T cells.

Table A.1: Analysis of false positives and false negatives on the 5 lane human data.

metric	SEECER	Quake	Coral	HiTEC	Echo
# common aligned reads	10975133	10975133	10975133	10975133	10975133
True Positives	3768807	2038505	2887722	668387	4105244
True Negatives	487571328	487459161	487374550	487400380	484080485
False Positives	472479	584646	669257	643427	3963322
False Negatives	2068371	3798673	2949456	5168791	1731934
Sensitivity	0.6456557	0.3492278	0.494712	0.1145052	0.7032926
Specificity	0.9990319	0.9988021	0.9986287	0.9986816	0.9918792
Gain	0.5647126	0.2490688	0.3800578	0.004276039	0.02431346

Table A.2: Analysis of false positives and false negatives on the 64M paired-end 76bps reads of HeLa cell lines.

metric	SEECER	Quake	Coral
# common aligned reads	28214429	28214429	28214429
True Positives	8521449	5513033	4025984
True Negatives	2112496518	2112468492	2112751956
False Positives	2228188	2256214	1972750
False Negatives	21050449	24058865	25545914
Sensitivity	0.2881604	0.1864281	0.1361422
Specificity	0.9989463	0.998933	0.9990671
Gain	0.2128122	0.1101322	0.06943193

Table A.3: Analysis of false positives and false negatives on the dataset of 145M paired-end 101bps reads.

metric	SEECER	Quake
# common aligned reads	117345990	117345990
True Positives	41552282	23392458
True Negatives	11782184102	11783039248
False Positives	3193840	2338694
False Negatives	25014766	43174590
Sensitivity	0.62421698495628	0.351411977890322
Specificity	0.999728999781278	0.999801559694436
Gain	0.576237690456095	0.316279069487954

method	single	multi	mean	sd
original	437	140	1.3917	0.9423
SEECER	499	204	1.4993	1.0362
Coral	494	196	1.4826	1.0281
SEED	385	130	1.4369	1.0444
ECHO	460	154	1.4186	0.9519
HiTEC	434	134	1.3592	0.8086
Quake	466	177	1.4666	1.0078

Table A.4: Analysis of full length reconstruction of alternative isoforms with Oases [27] on the human T-cell data for different error correction methods. For each method, full length assembled Ensembl (v.65) transcripts were grouped into genes with only one isoform reconstructed (single), genes with at least two isoforms reconstructed (multi) and the mean and standard deviation (sd) of number of reconstructed transcripts per gene was computed. Only Ensembl transcripts with an estimated expression level ≥ 5 RPKM were used for the analysis. SEECER corrected reads lead to a higher number of reconstructed genes and more isoforms.

Dataset	length	# of reads	total megabases	Time (hours)
Human T cells	45	55394464	2492	12.25
HeLa cell line	76	32775682	2490	16.3
IMR90 cell line	101	24526546	2477	4

Table A.5: Running time of SEECER (5th col) for different datasets of similar total sequence throughput in megabases (4th col) but using different read length (2nd col) and absolute number of reads(3rd col).

Table A.6: Analysis of SNP calls on the T-cell dataset after TopHat alignments. For all methods tested (1st col), the minimum coverage threshold (2nd col) for SNP calling was varied and the number of SNP calls (3rd col), calls annotated in dbSNP (4th col) and the methods Precision are shown.

method	coverage c	total calls	overlap dbSNP	Precision
Quake	5	22092	3423	0.154942966
Quake	10	10963	1647	0.150232601
Quake	15	6793	992	0.146032681
SEECER	5	23539	3834	0.162878627
SEECER	10	11952	1950	0.16315261
SEECER	15	7381	1143	0.154857065
original	5	18178	2928	0.161073826
original	10	8767	1365	0.155697502
original	15	5084	764	0.150275374
HiTEC	5	18176	2931	0.161256602
HiTEC	10	8826	1374	0.155676411
HiTEC	15	5082	772	0.151908697
ECHO	5	19518	3298	0.168972231
ECHO	10	9171	1492	0.16268673
ECHO	15	5179	815	0.157366287
Coral	5	23221	3798	0.163558848
Coral	10	11069	1863	0.168307887
Coral	15	6310	1057	0.167511886

Table A.7: GO table for the top 200 expressed Sea urchin peptides matched in both time points.

N	X	P-value	corrected P-val	GO terms	Descriptions
69	154	2.81E-115	<0.001	GO:0003735	structural constituent of ribosome
67	148	5.41E-112	<0.001	GO:0005840	ribosome
68	233	7.90E-98	<0.001	GO:0006412	translation
68	1557	1.68E-39	<0.001	GO:0005622	intracellular
8	14	1.32E-14	<0.001	GO:0015935	small ribosomal subunit
8	92	2.59E-07	<0.001	GO:0007017	microtubule-based process
7	78	1.21E-06	<0.001	GO:0043234	protein complex
7	78	1.21E-06	<0.001	GO:0051258	protein polymerization
7	82	1.70E-06	<0.001	GO:0005874	microtubule
9	181	5.10E-06	<0.001	GO:0003924	GTPase activity
3	8	1.82E-05	3.00E-03	GO:0015934	large ribosomal subunit
3	10	3.87E-05	7.00E-03	GO:0004129	cytochrome-c oxidase activity
6	99	6.82E-05	2.10E-02	GO:0005198	structural molecule activity

Table A.8: GO table for the top 200 expressed Sea urchin peptides only matched in the first time point.

N	X	P-value	corrected P-val	GO terms	Descriptions
9	455	5.95E-06	1.00E-03	GO:0000166	nucleotide binding
5	92	7.10E-06	1.00E-03	GO:0007017	microtubule-based process
3	20	2.59E-05	7.00E-03	GO:0030414	peptidase inhibitor activity
4	78	7.89E-05	1.60E-02	GO:0051258	protein polymerization
4	78	7.89E-05	1.60E-02	GO:0043234	protein complex
4	82	9.59E-05	2.20E-02	GO:0005874	microtubule
5	181	1.82E-04	2.80E-02	GO:0003924	GTPase activity
4	99	1.99E-04	2.90E-02	GO:0005198	structural molecule activity
12	1253	2.06E-04	2.90E-02	GO:0003676	nucleic acid binding

Table A.9: GO table for the top 200 expressed Sea urchin peptides only matched in the second time point.

N	X	P-value	adj P-val	GO terms	Descriptions
6	37	6.85E-10	<0.001	0015986	ATP synthesis coupled proton transport
4	18	1.50E-07	<0.001	0015078	hydrogen ion transmembrane=transporter activity
5	99	7.51E-06	3.00E-03	0005198	structural molecule activity
4	78	6.20E-05	1.80E-02	0051258	protein polymerization
4	78	6.20E-05	1.80E-02	0043234	protein complex
2	5	7.38E-05	2.00E-02	0046034	ATP metabolic process
2	5	7.38E-05	2.00E-02	0016469	proton-transporting two-sector ATPase complex
4	82	7.54E-05	2.00E-02	0005874	microtubule
4	92	1.18E-04	2.40E-02	0007017	microtubule-based process
2	7	1.54E-04	3.40E-02	0015992	proton transport

Table A.10: Analysis of blastx alignment matches to sea urchin peptides after *de novo* assembly of sea cucumber transcripts for timepoint two data (larval stage). Column 3 and 4 report the number of sea urchin peptides that are covered to at least 50% (3rd col) and 60% (4th col) of their length by an assembled Oases transfrag after error correction with SEECER, Quake or Coral (1st col). We also contrast SEECER with k=17 (default) to k=21.

Method	transfrags	cov >= 0.5	cov >= 0.6
quake	1298830	690	96
SEECER-k17	628913	801	108
SEECER-k21	1097826	797	112
Coral	906846	777	101

Supplementary materials for Chapter 3

B.1 Metric properties

For completeness we list below the properties of distance metrics.

1. Non-negative: $d(\pi, \sigma) \geq 0$
2. Symmetric: $d(\pi, \sigma) = d(\sigma, \pi)$
3. Identity: $d(\pi, \sigma) = 0$ if and only if $\pi = \sigma$
4. Triangular inequality: $d(\pi, \sigma) \leq d(\pi, \tau) + d(\tau, \sigma)$ for any $\tau \in \mathcal{G}_m$

B.2 Proof of Asymptotic Normality

Proof Since $d(\pi, \sigma) = d(\pi\pi^{-1}, \sigma\pi^{-1}) = d(\mathbf{I}_m, \sigma\pi^{-1})$, the distribution D_d is the distribution of $d(I_m, \tau)$ when τ is a uniformly random permutation in $\mathcal{G}_m$. Applying Hoeffding's Combinatorial Central Limit Theorem [206] with $c_m(i,j) = (w_i - w_j)^2$, we only need to verify the condition (12) of the theorem 3.

Define $d_m(i,j)$ as in the equation (11). Let $\alpha = \min_{1 \leq i,j \leq m} d_m(i,j)$ and $\beta = \max_{1 \leq i,j \leq m} d_m(i,j)$. α and β exist because $-\infty < w(i) < \infty$ for all i .

$$\lim_{m \to \infty} \frac{\max_{1 \leq i,j \leq m} [d_m(i,j)]^2}{\frac{1}{m} \sum_{i=1}^{m} \sum_{j=1}^{m} [d_m(i,j)]^2} \leq \lim_{m \to \infty} \frac{\beta^2}{\frac{1}{m} \sum_{i=1}^{m} \sum_{j=1}^{m} \alpha^2} \tag{B.1}$$

$$= \lim_{m \to \infty} \frac{\beta^2}{m\alpha^2} \tag{B.2}$$

$$= 0 \tag{B.3}$$

B.3 Pseudometric properties of the relational weighted rank matrix

Below we prove that Equation 3.13 is a pseudometric in the original permutation space and a metric in the **W**-transformed space.

Lemma B.3.1

$$\mathbf{M}^F_{\pi,\sigma} = \mathbf{M}^{\mathrm{T}}_{\pi} \mathbf{M}_{\sigma} + \mathbf{M}^{\mathrm{T}}_{\sigma} \mathbf{M}_{\pi} \tag{B.4}$$

$$2\mathbf{I} - \mathbf{M}^F_{\pi,\sigma} = (\mathbf{M}_{\pi} - \mathbf{M}_{\sigma})^{\mathrm{T}} (\mathbf{M}_{\pi} - \mathbf{M}_{\sigma}) \tag{B.5}$$

Proof Since $\mathbf{M}_{\pi}$ and $\mathbf{M}_{\sigma}$ are permutation matrices, $\mathbf{M}^{\mathrm{T}}_{\pi} \mathbf{M}_{\sigma} = \mathbf{M}_{\sigma\pi^{-1}}$ and $\mathbf{M}^{\mathrm{T}}_{\sigma} \mathbf{M}_{\pi} = \mathbf{M}_{\pi\sigma^{-1}}$.

Therefore, by the definition of the permutation matrix in (3.5), $M_{\sigma\pi^{-1}}(i,j) = 1$ if and only if $\sigma\pi^{-1}(i) = j$ or $\pi^{-1}(i) = \sigma^{-1}(j)$. Similarly, $M_{\pi\sigma^{-1}}(i,j) = 1$ if and only if $\pi^{-1}(j) = \sigma^{-1}(i)$. Equation (B.4) follows from the definition of $\mathbf{M}^F_{\pi\sigma}$ in (3.12).

$$\begin{aligned} 2\mathbf{I} - \mathbf{M}^F_{\pi,\sigma} &= \mathbf{M}_\pi^{\mathrm{T}}\mathbf{M}_\pi + \mathbf{M}_\sigma^{\mathrm{T}}\mathbf{M}_\sigma - (\mathbf{M}_\pi^{\mathrm{T}}\mathbf{M}_\sigma + \mathbf{M}_\sigma^{\mathrm{T}}\mathbf{M}_\pi) \\ &= (\mathbf{M}_\pi - \mathbf{M}_\sigma)^{\mathrm{T}}(\mathbf{M}_\pi - \mathbf{M}_\sigma) \end{aligned}$$

Theorem B.3.2 *If the matrix* $\mathbf{W}$ *is positive semidefinite, the distance is a pseudometric.*

Proof

$$\begin{aligned} d(\pi,\sigma) &= \sqrt{\mathrm{tr}((2\mathbf{I} - \mathbf{M}^F_{\pi,\sigma})\mathbf{W})} \\ &= \sqrt{\mathrm{tr}\big(\mathbf{U}^{\mathrm{T}}(\mathbf{M}_\pi - \mathbf{M}_\sigma)^{\mathrm{T}}(\mathbf{M}_\pi - \mathbf{M}_\sigma)\mathbf{U}\big)} \\ &= \|(\mathbf{M}_\pi - \mathbf{M}_\sigma)\mathbf{U}\|_F \end{aligned}$$

Since the Frobenius norm $\|.\|_F$ is a metric, our distance $d(\pi,\sigma)$ satisfies non negativity, symmetry and triangular inequality. Therefore, the distance is a pseudometric. $d(\pi,\sigma) = 0$ implies $\mathbf{M}_\pi\mathbf{U} = \mathbf{M}_\sigma\mathbf{U}$, hence the distance is a metric in the W-transformed space.

B.4 Matrix and Vector Weight metrics

We show that the vector weight discussed in section 3.2.3 is a special case of the general weight matrix when that matrix has a rank of 1.

Proof Since $\mathbf{W}$ is ranked 1, $W = \mathbf{w}^{\mathrm{T}}\mathbf{w}$ with $\mathbf{w}$ is a vector of length n. Let d_1 and d_2 be the metric in Section 3.2.3 and Section 3.2.3 respectively. Recall from the proof of Theorem B.3:

$$d_1(\pi,\sigma) = \sqrt{\mathbf{w}^T(\mathbf{M}_\pi - \mathbf{M}_\sigma)^T(\mathbf{M}_\pi - \mathbf{M}_\sigma)\mathbf{w}} \tag{B.6}$$

$$d_2(\pi,\sigma) = \sqrt{\mathrm{tr}\left(\mathbf{w}^{\mathrm{T}}(\mathbf{M}_\pi - \mathbf{M}_\sigma)^{\mathrm{T}}(\mathbf{M}_\pi - \mathbf{M}_\sigma)\mathbf{w}\right)} \tag{B.7}$$

$$= \sqrt{\mathbf{w}^{\mathrm{T}}(\mathbf{M}_\pi - \mathbf{M}_\sigma)^{\mathrm{T}}(\mathbf{M}_\pi - \mathbf{M}_\sigma)\mathbf{w}} \tag{B.8}$$

Therefore, the metric in Section 3.2.3 is a special case of the metric in Section 3.2.3.

B.5 Normality of the null distribution

Figure B.1 experimentally confirms that the null model follows a normal distribution. The red curve is a normal distribution fit using Matlab.

B.6 Robustness of the methods

B.6.1 Effect of ortholog assignment on the performance of the Matrix method

Inparanoid contains over 10000 known orthologs between human and mouse making them one of the best annotated pairs of species. As noted above, from this set we select a subset of 500 genes and

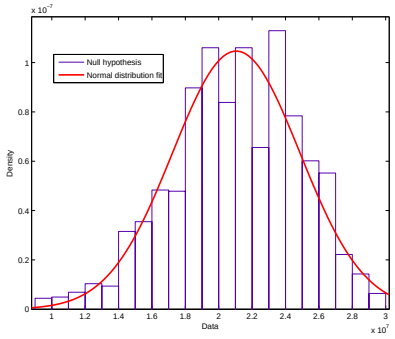


Figure B.1: The histogram of the Spearman correlation of 2000 random pairs of microarrays and the Gaussian distribution fit using Matlab.

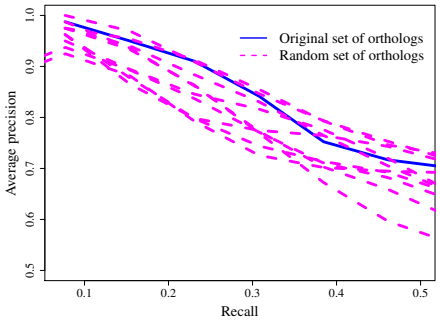


Figure B.2: PR curves for the Matrix Weight metric when starting with fewer orthologs. The blue curve is the result when starting with all orthologs (same curve as in Figure 3.3).

use these in our algorithms. To test whether our methods would be appropriate to other species pairs for which much fewer orthologs are known we repeated the analysis discussed above starting with a smaller set of orthologs. We selected random sets of 2000 orthologs (roughly 12% of all orthologs) and then reran our method using this initial set (selecting the top 500 varying genes from this smaller subset and running the matrix algorithm discussed above). Figure B.2 presents results for seven of these random sets. The blue curve are the results when starting with the full set of orthologs. As can be seen our method is robust and is appropriate for pairs of species with much fewer known orthologs as well.

B.6.2 Comparison of cross species comparison metrics using 1000 most variant genes

We reran experiments with 1000 orthologs and the results are presented in Figure B.3 . Indeed, as the reviewer suspected the matrix method did slightly worse when compared to the results using 500 genes. However, for the highest precision rates Matrix was still the best method (though by a much lower margin when compared to the vector method which requires far fewer parameters). The results of using 500 genes are slightly better than using 1000 genes at the 0.9 precision range (for a recall of 0.21 the 500 genes method achieves a 0.92 precision whereas the 1000 genes achieves 0.91). Of course, these results are also a function of the training data size. With a larger training datasets the ability to fit parameters to more sophisticated models increases and so more complex methods, like the Matrix method, are likely to outperform the simpler methods.

B.6.3 Randomized dataset

To demonstrate that how well different methods perform relative to random prediction, we have carried out the experiment on a randomized dataset, by randomly permuting expression values in each array. The results are presented as Figure B.4. As can be seen, all methods do very badly and the results are essentially a flat PR curve as expected from random data.

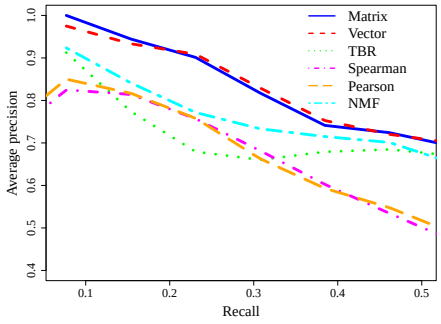


Figure B.3: PR curves of the metrics using 1000 most variant genes.

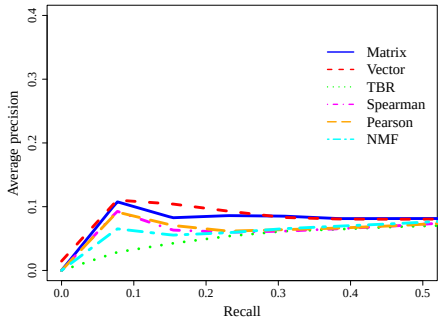


Figure B.4: PR curves of the metrics on a randomized dataset.

B.7 Human and mouse tissue list

Table B.1 shows the list of 26 human and mouse tissues used in this analysis.

B.8 Identifying similar experiments in GEO

B.8.1 Histogram of the correlation of 500 selected genes

Figure B.5 shows distributions of correlations for the selected highly varying 500 genes. When using the 500 selected genes the results look pretty similar to the results presented in the paper though the mean correlation is slightly higher (0.1057 vs. 0.1021).

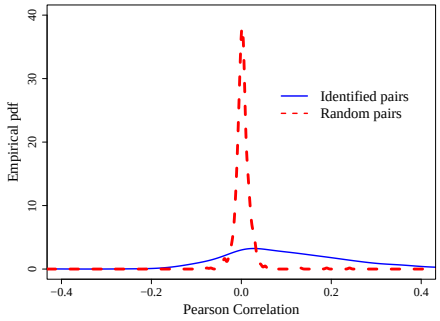


Figure B.5: Blue curve: Correlation of 500 orthologs used for training in a random sample of 301,453 microarray pairs from human and mouse. Red curve: Correlation of 500 orthologs used for training in the set of microarray pairs selected by our method.

Human	Mouse
Adrenal Cortex	Adrenal
Bladder	Bladder
Bone Marrow	Bone Marrow
Brain	Brain
Brain Cerebellum	Cerebellum
Brain Cerebral cortex	Cortex
Epididymis	Epididymus
Heart	Heart
Kidney	Kidney
Liver	Liver
Lung	Lung
Pancreas	Pancreas
Placenta	Placenta 12.5
Prostate	Prostate
Salivary Gland	Salivary
Skeletal Muscle	Skeletal Muscle
Small Intestine	Small Intestine
Spinal Cord	Spinal Cord
Spleen	Spleen
Stomach	Stomach
Testis	Testis
Thymus	Thymus
Thyroid	Thyroid
Tongue	Tongue
Trachea	Trachea
Uterus	Uterus

Table B.1: The one-one similarity list of human and mouse tissues.

B.8.2 Description analysis on random sets of array pairs

We repeated the analysis with random sets of array pairs. As can be seen in Table B.2, for these pairs the p-values are much higher (less significant). Specifically, there are no matched terms with a p-value lower than 10^{-10} (whereas in the identified matching there are 4 such words) and only 3 of the top random match words would be ranked in the top 10 of the words identified using the matches made by the algorithm. Thus, such p-values are significant and would not be expected from random assignments.

Rank	P-value	Word	#Pairs	
			Identified	Expected
1	1.13469e-09	BONE	51	19.13650
2	3.91648e-09	ACUTE	43	15.18268
3	1.26953e-06	MARROW	15	3.16306
4	1.49012e-05	GASTROCNEMIUS	8	1.05435
5	2.02795e-05	STEROID	5	0.31631
6	7.76604e-05	METAPLASIA	3	0.07908
7	1.34712e-04	LIPOPOLYSACCHARIDE	8	0 1.44973
8	2.29396e-04	PULMONARY	15	05.00818
9	3.32228e-04	PROGENITOR	8	0 1.66061
10	5.00167e-04	IFN-GAMMA	5	0.63261
11	7.80427e-04	DYSTROPHY	14	5.06089
12	7.86850e-04	DUCHENNE	8	1.89783
13	7.94474e-04	REGIONS	9	02.37229
14	1.34160e-03	LEUKEMIAS	2	0.05272

Table B.2: Top 14 words identified in titles of pairs determined to be similar. #Pairs Identified is the number of time this pair was observed. #Pairs Expected is the number of time expected based on single species occurrences.

B.8.3 Heat map of similarity between 3416 human and 2991 mouse microarrays

Figure B.6 presents a heatmap showing all human by mouse arrays where the color indicates the level of similarity from the Weight Matrix metric. Smaller value means more similarity.

B.8.4 Human assessment of identified matched dataset pairs.

To test whether the identified matched pairs are indeed a feasible solution we have asked an expert pathologist (Oltvai, a co-author of the paper) to examine the top 100 matched dataset pairs identified by our method. Based on the description for that dataset the expert assigned each match to one of three categories: A correct match (Y), an incorrect match (N) and an inconclusive. As can be seen in Table B.3, there were 83 Y assignments in the top 100 matches with the other 17 determined to either be mistakes (N, 13) or inconclusive (4). Given that almost all random matches would not make sense this is a very high accuracy rate and it clearly indicates that this method can be use to help improve, and speed up, human assessment of similarity. We have changed the introduction and results sections to reflect this idea and to highlight the ability of the method to aid in human assessment of similarity.

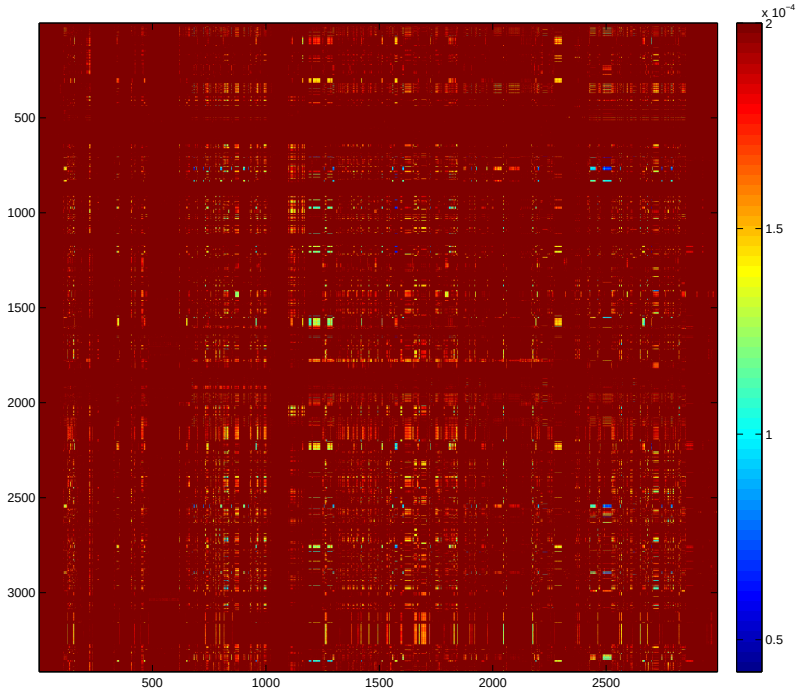


Figure B.6: The similarity between 3416 human and 2991 mouse microarrays.

Table B.3: The result of human assessment of identified matched dataset pairs.

Human Dataset	Description	Mouse Dataset	Description	Assessment
GDS2767	Blood response to various beverages: time course	GDS1077	Hematopoietic stem cells from different recombinant inbred strains	Y/ inconcl.
GDS2767	Blood response to various beverages: time course	GDS2047	Lipopolysaccharide effect on macrophages pretreated with carbon monoxide: time course	Y/ inconcl.
GDS2772	Sevoflurane and propofol effect on the heart during off-pump coronary artery bypass graft surgery	GDS2329	Acute myocardial infarction model: time course (MG-U74A)	Y
GDS2772	Sevoflurane and propofol effect on the heart during off-pump coronary artery bypass graft surgery	GDS2330	Acute myocardial infarction model: time course (MG-U74B)	Y
GDS2055	Skeletal muscle types (HG-U133A)	GDS2329	Acute myocardial infarction model: time course (MG-U74A)	Y

Table B.3: The result of human assessment of identified matched dataset pairs.

GDS1815	High-grade gliomas (HG-U133A)	GDS2159	Spinal cord injury model: time course	Y
GDS2055	Skeletal muscle types (HG-U133A)	GDS2330	Acute myocardial infarction model: time course (MG-U74B)	Y
GDS2772	Sevoflurane and propofol effect on the heart during off-pump coronary artery bypass graft surgery	GDS488	Myocardial infarction time course	Y
GDS2056	Skeletal muscle types (HG-U133B)	GDS2330	Acute myocardial infarction model: time course (MG-U74B)	Y
GDS2740	Lengthening and shortening contractions effect on the muscle: time course	GDS2329	Acute myocardial infarction model: time course (MG-U74A)	Y
GDS2740	Lengthening and shortening contractions effect on the muscle: time course	GDS2330	Acute myocardial infarction model: time course (MG-U74B)	Y
GDS2678	Brain regions of humans and chimpanzees	GDS2159	Spinal cord injury model: time course	Y
GDS2055	Skeletal muscle types (HG-U133A)	GDS488	Myocardial infarction time course	Y
GDS2767	Blood response to various beverages: time course	GDS2150	Spleens of males and females at puberty	N
GDS2255	Transmigrated neutrophils in the alveolar space of endotoxin-exposed lung	GDS1077	Hematopoietic stem cells from different recombinant inbred strains	Y
GDS2255	Transmigrated neutrophils in the alveolar space of endotoxin-exposed lung	GDS2047	Lipopolysaccharide effect on macrophages pretreated with carbon monoxide: time course	Y
GDS2373	Squamous cell lung carcinomas	GDS2334	Myod and Myog expression effect on myogenesis: time course	N
GDS2373	Squamous cell lung carcinomas	GDS981	Uterine response to physiologic and plant-derived estrogen: time course	N
GDS2373	Squamous cell lung carcinomas	GDS1244	Phosgene effect on lungs: time course	Y
GDS2055	Skeletal muscle types (HG-U133A)	GDS234	Muscle regeneration (U74Av2)	Y
GDS2767	Blood response to various beverages: time course	GDS1336	T cell anergy induction regulation by Egr-2 and Egr-3 (MG-U74A)	Y/ inconcl.
GDS1673	Non-diseased lung tissue	GDS1244	Phosgene effect on lungs: time course	Y
GDS2373	Squamous cell lung carcinomas	GDS1072	Platelet derived growth factor effect in the presence of Src family kinase inhibitors (MOE430A)	Inconcl.
GDS2740	Lengthening and shortening contractions effect on the muscle: time course	GDS488	Myocardial infarction time course	Y

Table B.3: The result of human assessment of identified matched dataset pairs.

GDS2772	Sevoflurane and propofol effect on the heart during off-pump coronary artery bypass graft surgery	GDS2335	Exercise effect on the diabetic cardiac muscle: time course	Y
GDS2772	Sevoflurane and propofol effect on the heart during off-pump coronary artery bypass graft surgery	GDS627	Cardiac development in embryo	Y
GDS2767	Blood response to various beverages: time course	GDS1514	Interferon-gamma tolerogenic effect on CD8+ dendritic cells	Y/ inconcl.
GDS2767	Blood response to various beverages: time course	GDS2408	B cell-activating factor of the TNF family effect on B cells	Y/ inconcl.
GDS2767	Blood response to various beverages: time course	GDS993	Naive CD8+ T cells proliferative response to lymphopenia: time course	Y/ inconcl.
GDS2055	Skeletal muscle types (HG-U133A)	GDS1541	Exercise effect on diabetic skeletal muscle: time course	Y
GDS2056	Skeletal muscle types (HG-U133B)	GDS1541	Exercise effect on diabetic skeletal muscle: time course	Y
GDS2678	Brain regions of humans and chimpanzees	GDS2917	Various brain regions of several inbred strains	Y
GDS2055	Skeletal muscle types (HG-U133A)	GDS2335	Exercise effect on the diabetic cardiac muscle: time course	Y
GDS596	Large-scale analysis of the human transcriptome (HG-U133A)	GDS2159	Spinal cord injury model: time course	Inconcl.
GDS2678	Brain regions of humans and chimpanzees	GDS1406	Brain regions of various inbred strains	Y
GDS2373	Squamous cell lung carcinomas	GDS1058	Uterus response to 17beta-estradiol: time course	N
GDS2772	Sevoflurane and propofol effect on the heart during off-pump coronary artery bypass graft surgery	GDS1766	Extraocular and hindlimb skeletal muscle cell differentiation: time course (MG-430B)	Y
GDS1340	Exercise effect on aged muscle	GDS2329	Acute myocardial infarction model: time course (MG-U74A)	Y
GDS1340	Exercise effect on aged muscle	GDS2330	Acute myocardial infarction model: time course (MG-U74B)	Y
GDS198	Inflammatory myopathy	GDS2329	Acute myocardial infarction model: time course (MG-U74A)	Y
GDS2373	Squamous cell lung carcinomas	GDS1277	Obliterative bronchiolitis and tracheal allograft	Y
GDS1673	Non-diseased lung tissue	GDS251	Pulmonary fibrosis	Y
GDS2767	Blood response to various beverages: time course	GDS882	Neuromedin U effect on type-2 Th cells: time course	Y/ inconcl.
GDS2168	HIV viremia effect on monocytes	GDS1077	Hematopoietic stem cells from different recombinant inbred strains	Y

Table B.3: The result of human assessment of identified matched dataset pairs.

GDS707	Aging brain: frontal cortex expression profiles at various ages	GDS2159	Spinal cord injury model: time course	Y
GDS2055	Skeletal muscle types (HG-U133A)	GDS1765	Extraocular and hindlimb skeletal muscle cell differentiation: time course (MG-430A)	Y
GDS2373	Squamous cell lung carcinomas	GDS1631	Osteoblast differentiation (MG-U74A)	N
GDS2373	Squamous cell lung carcinomas	GDS1071	Platelet derived growth factor effect in the presence of Src family kinase inhibitors (MG-U74A)	Y
GDS198	Inflammatory myopathy	GDS234	Muscle regeneration (U74Av2)	Y
GDS2767	Blood response to various beverages: time course	GDS1285	Macrophage response to lipopolysaccharide and CstF-64 overexpression	Y/ inconcl.
GDS2767	Blood response to various beverages: time course	GDS1315	Immune response to suppressive vs. stimulatory immunomodulators	Y/ inconcl.
GDS2767	Blood response to various beverages: time course	GDS1654	Dendritic cell subpopulations: spleen (MG-U74A)	Y/ inconcl.
GDS2767	Blood response to various beverages: time course	GDS2741	TCR-alpha/beta CD8-alpha/alpha intestinal intraepithelial lymphocytes	Y/ inconcl.
GDS2767	Blood response to various beverages: time course	GDS2957	Resting and activated natural killer cells	Y/ inconcl.
GDS2767	Blood response to various beverages: time course	GDS658	Thymocyte selection by agonist	Y/ inconcl.
GDS2767	Blood response to various beverages: time course	GDS827	Acute ethanol administration effect on Toll-like receptor 3 signaling in macrophages	Y/ inconcl.
GDS2056	Skeletal muscle types (HG-U133B)	GDS2329	Acute myocardial infarction model: time course (MG-U74A)	Y
GDS2083	Limb immobilization effect on skeletal muscle	GDS2329	Acute myocardial infarction model: time course (MG-U74A)	Y
GDS2083	Limb immobilization effect on skeletal muscle	GDS2330	Acute myocardial infarction model: time course (MG-U74B)	Y
GDS2772	Sevoflurane and propofol effect on the heart during off-pump coronary artery bypass graft surgery	GDS40	Cardiac development, maturation and aging	Y
GDS2373	Squamous cell lung carcinomas	GDS1865	Chondrocyte differentiation: time course	N
GDS2767	Blood response to various beverages: time course	GDS2521	Megakaryocytes at successive stages of maturation	Y
GDS395	Biomaterial engineering	GDS981	Uterine response to physiologic and plant-derived estrogen: time course	N
GDS2113	Pheochromocytomas of various genetic origins	GDS2159	Spinal cord injury model: time course	Y

Table B.3: The result of human assessment of identified matched dataset pairs.

GDS1036	Microglial cell response to interferon-gamma: time course	GDS2047	Lipopolysaccharide effect on macrophages pretreated with carbon monoxide: time course	Inconcl.
GDS1684	Cardiac allograft rejection: time course	GDS2329	Acute myocardial infarction model: time course (MG-U74A)	Y
GDS1684	Cardiac allograft rejection: time course	GDS2330	Acute myocardial infarction model: time course (MG-U74B)	Y
GDS2740	Lengthening and shortening contractions effect on the muscle: time course	GDS1541	Exercise effect on diabetic skeletal muscle: time course	Y
GDS2740	Lengthening and shortening contractions effect on the muscle: time course	GDS2335	Exercise effect on the diabetic cardiac muscle: time course	Y
GDS833	Alternative pre-mRNA splicing in various tissues and cell lines (Rosetta/Merck Splicing Chip 5)	GDS2162	CH1 domain deletion, p300 and CBP heterozygous null mutant hypoxic fibroblasts response to trichostatin A	N
GDS833	Alternative pre-mRNA splicing in various tissues and cell lines (Rosetta/Merck Splicing Chip 5)	GDS1244	Phosgene effect on lungs: time course	N
GDS1284	Multiple myeloma molecular classification	GDS1077	Hematopoietic stem cells from different recombinant inbred strains	Y
GDS198	Inflammatory myopathy	GDS488	Myocardial infarction time course	Y
GDS2055	Skeletal muscle types (HG-U133A)	GDS627	Cardiac development in embryo	Y
GDS2373	Squamous cell lung carcinomas	GDS951	Hormone-induced adipogenesis suppressed by 2,3,7,8-tetrachlorodibenzo-p-dioxin and EGF	N
GDS424	Normal human tissue expression profiling (HG-U95C)	GDS2329	Acute myocardial infarction model: time course (MG-U74A)	Inconcl.
GDS2255	Transmigrated neutrophils in the alveolar space of endotoxin-exposed lung	GDS1336	T cell anergy induction regulation by Egr-2 and Egr-3 (MG-U74A)	Y/ inconcl.
GDS2528	Basal plate of the placenta from midgestation to term (HG-U133A)	GDS981	Uterine response to physiologic and plant-derived estrogen: time course	Y
GDS1340	Exercise effect on aged muscle	GDS488	Myocardial infarction time course	Y
GDS2056	Skeletal muscle types (HG-U133B)	GDS488	Myocardial infarction time course	Y
GDS1340	Exercise effect on aged muscle	GDS234	Muscle regeneration (U74Av2)	Y
GDS2373	Squamous cell lung carcinomas	GDS857	Corneal stromal cell differentiation	N
GDS1815	High-grade gliomas (HG-U133A)	GDS2917	Various brain regions of several inbred strains	Y

Table B.3: The result of human assessment of identified matched dataset pairs.

GDS2772	Sevoflurane and propofol effect on the heart during off-pump coronary artery bypass graft surgery	GDS1541	Exercise effect on diabetic skeletal muscle: time course	Y
GDS2106	Lymphoblastoid cell lines from various CEPH pedigrees	GDS2047	Lipopolysaccharide effect on macrophages pretreated with carbon monoxide: time course	Y
GDS2310	Exercise effect on white blood cells	GDS2047	Lipopolysaccharide effect on macrophages pretreated with carbon monoxide: time course	Y
GDS2772	Sevoflurane and propofol effect on the heart during off-pump coronary artery bypass graft surgery	GDS388	Cardiac remodeling (Mu11K-B)	Y
GDS1962	Glioma-derived stem cell factor effect on angiogenesis in the brain	GDS2159	Spinal cord injury model: time course	Y
GDS2255	Transmigrated neutrophils in the alveolar space of endotoxin-exposed lung	GDS1514	Interferon-gamma tolerogenic effect on CD8+ dendritic cells	Y
GDS2255	Transmigrated neutrophils in the alveolar space of endotoxin-exposed lung	GDS2408	B cell-activating factor of the TNF family effect on B cells	Y
GDS738	Intervertebral disc cells and osmotic loading	GDS981	Uterine response to physiologic and plant-derived estrogen: time course	N
GDS395	Biomaterial engineering	GDS2162	CH1 domain deletion, p300 and CBP heterozygous null mutant hypoxic fibroblasts response to trichostatin A	N
GDS2435	Male and female venous blood	GDS1077	Hematopoietic stem cells from different recombinant inbred strains	Y
GDS2959	Granulocyte colony-stimulating factor mobilized leukocytes	GDS1077	Hematopoietic stem cells from different recombinant inbred strains	Y
GDS2767	Blood response to various beverages: time course	GDS2011	Lupus-prone BWF1 males and females: spleen (MG-U74A)	Y/ inconcl.
GDS2767	Blood response to various beverages: time course	GDS2041	Type II activated macrophage	Y/ inconcl.
GDS2767	Blood response to various beverages: time course	GDS2651	Macrophage cell line response to Chlamydia pneumoniae infection	Y/ inconcl.
GDS2767	Blood response to various beverages: time course	GDS433	CD8+ effector and central memory T cells (MG-U74A)	Y/ inconcl.
GDS2767	Blood response to various beverages: time course	GDS684	T regulatory and T effector cells in prediabetic lesion	Y/ inconcl.
GDS2055	Skeletal muscle types (HG-U133A)	GDS2001	Utrophin/dystrophin-deficient double mutant and dystrophin-deficient mdx mutant skeletal muscles	Y

Supplementary materials for Chapter 5

C.1 Taking the infinite limit

Lemma C.1.1 *For any real numbers $a_k (k \geq 1)$, which are constants with respect to n and $1 < T < \infty$,*

$$\lim_{n\to\infty}(1+\sum_{k=1}^{T}\frac{a_k}{n^k})^n = \exp(a_1) \tag{C.1}$$

Proof The limit is in the indeterminate form 1^∞, we apply a transformation and L'Hôpital's rule:

$$\lim_{n\to\infty}(1+\sum_{k=1}^{T}\frac{a_k}{n^k})^n = \exp\lim_{n\to\infty}\frac{\ln(1+\sum_{k=1}^{T}\frac{a_k}{n^k})}{1/n} \quad \text{(transformation)} \tag{C.2}$$

$$= \exp\lim_{n\to\infty}\frac{-\sum_{k=1}^{T}\frac{ka_k}{n^{k+1}}/(1+\sum_{k=1}^{T}\frac{a_k}{n^k})}{-1/n^2} \quad \text{(L'Hôpital's rule)} \tag{C.3}$$

$$= \exp\lim_{n\to\infty}\frac{\sum_{k=1}^{T}\frac{ka_k}{n^{k-1}}}{1+\sum_{k=1}^{T}\frac{a_k}{n^k}} \tag{C.4}$$

$$= \exp(a_1) \tag{C.5}$$

Here we show that:

$$\lim_{K\to\infty}\frac{K!}{\prod_{h=0}^{2^N-1}K_h!}\prod_{k=1}^{K_+}\Phi_{z,k}\frac{B(m_k+\frac{\alpha}{K},N-m_k+1)}{B(\frac{\alpha}{K},N+1)}\prod_{k=1}^{K}\frac{1}{Z'}B(\frac{\alpha}{K},N+1) \tag{C.6}$$

$$= \frac{\alpha^{K+}}{\prod_{h=1}^{2^N-1}K_h!}\prod_{k=1}^{K_+}\Phi_{z,k}\frac{(N-m_k)!(m_k-1)!}{N!}\exp\left(-\alpha\Psi\right) \tag{C.7}$$

We consider each term separately.

$$\frac{K!}{\prod_{h=0}^{2^N-1}K_h!}\prod_{k=1}^{K_+}\Phi_{z,k}\frac{B(m_k+\frac{\alpha}{K},N-m_k+1)}{B(\frac{\alpha}{K},N+1)} \tag{C.8}$$

$$= \frac{K!}{\prod_{h=0}^{2^N-1}K_h!}\prod_{k=1}^{K_+}\Phi_{z,k}\frac{\Gamma(\frac{\alpha}{K}+m_k)\Gamma(N-m_k+1)}{\Gamma(\frac{\alpha}{K})\Gamma(N+1)} \tag{C.9}$$

$$= \frac{K!}{\prod_{h=0}^{2^N-1}K_h!}\prod_{k=1}^{K_+}\Phi_{z,k}\frac{(N-m_k)!\frac{\alpha}{K}\prod_{j=1}^{m_k-1}(j+\frac{\alpha}{K})}{N!} \tag{C.10}$$

$$= \frac{\alpha^{K+}}{\prod_{h=1}^{2^N-1}K_h!}\frac{K!}{K_0!K^{K+}}\prod_{k=1}^{K_+}\Phi_{z,k}\frac{(N-m_k)!\prod_{j=1}^{m_k-1}(j+\frac{\alpha}{K})}{N!} \tag{C.11}$$

By the same argument as shown in [15],

$$\lim_{K\to\infty}\frac{K!}{\prod_{h=0}^{2^N-1}K_h!}\prod_{k=1}^{K_+}\Phi_{z,k}\frac{B(m_k+\frac{\alpha}{K},N-m_k+1)}{B(\frac{\alpha}{K},N+1)} \tag{C.12}$$

$$= \frac{\alpha^{K+}}{\prod_{h=1}^{2^N-1}K_h!}\prod_{k=1}^{K_+}\Phi_{z,k}\frac{(N-m_k)!(m_k-1)!}{N!} \tag{C.13}$$

$$\prod_{k=1}^{K} \frac{Z'}{B(\frac{\alpha}{K}, N+1)} = \prod_{k=1}^{K} \frac{\sum_{h=0}^{2^N-1} \Phi_h B(\frac{\alpha}{K} + m_h, N - m_h + 1)}{B(\frac{\alpha}{K}, N+1)} \tag{C.14}$$

$$= \left(\sum_{h=0}^{2^N-1} \Phi_h \frac{\Gamma(\frac{\alpha}{K} + m_h)\Gamma(N - m_h + 1)}{\Gamma(\frac{\alpha}{K})\Gamma(N+1)} \right)^K \tag{C.15}$$

$$= \left(1 + \frac{\alpha}{K} \sum_{h=1}^{2^N-1} \Phi_h \frac{(N - m_h)! \prod_{j=1}^{m_h-1}(j + \frac{\alpha}{K})}{N!} \right)^K \tag{C.16}$$

$$= \left(1 + \frac{\alpha}{K} \sum_{h=1}^{2^N-1} \Phi_h \frac{(N - m_h)!(m_h - 1)!}{N!} + (\frac{\alpha}{K})^2 \cdots + \ldots \right)^K \tag{C.17}$$

Using Lemma C.1.1, we get:

$$\lim_{K\to\infty} \prod_{k=1}^{K} \frac{Z'}{B(\frac{\alpha}{K}, N+1)} = \exp\left(\alpha \sum_{h=1}^{2^N-1} \Phi_h \frac{(N - m_h)!(m_h - 1)!}{N!} \right) = \exp\left(\alpha \Psi\right) \tag{C.18}$$

Combining (C.13) and (C.18), we arrive at (C.7).

C.2 The generative process

In Section 5.2.2, we described the generative process using a culinary metaphor. The customers select dishes one after the other as follows. The first customer tries Poisson$(\alpha\Psi_1)$ dishes. The remaining customers enter one after the others. Customer i selects dishes with a probability that partially depends on the selection of the previous customers. For each dish, the probability that it would be selected is specified by: $\sum_{h:\, h_i = z_{<ik} \text{ and } h(i)=1} \overline{\Phi}_h / \sum_{h:\, h_i = z_{<ik}} \overline{\Phi}_h$. He then samples a Poisson$(\alpha\Psi_i)$ number of new dishes. This process repeats until all customers have made their selections.

We show here that this process simplifies to the Indian Buffet Process when $\Phi_h = 1$ for all h.

Theorem C.2.1 *If $\Phi_h = 1$ for all h,*

$$\Psi_i = \frac{1}{i} \tag{C.19}$$

Therefore, each customer selects Poisson$(\frac{\alpha}{i})$ new dishes as in the IBP.

Proof

$$\Psi_i = \sum_{h:\, h_i=0 \text{ and } h(i)=1} \overline{\Phi}_h \tag{C.20}$$

$$= \sum_{h:\, h_i=0 \text{ and } h(i)=1} \frac{(N-m_h)!(m_h-1)!}{N!} \tag{C.21}$$

$$= \sum_{t=0}^{N-i} \binom{N-i}{t} \frac{(N-t-1)!t!}{N!} \tag{C.22}$$

$$= \frac{(i-1)!(N-i)!}{N!} \sum_{t=0}^{N-i} \binom{N-t-1}{i-1} \tag{C.23}$$

$$= \frac{(i-1)!(N-i)!}{N!} \frac{N\binom{N-1}{i-1}}{i} \tag{C.24}$$

$$= \frac{1}{i} \tag{C.25}$$

Theorem C.2.2 *If $\Phi_h = 1$ for all h,*

$$\frac{\sum_{h:\, h_i=z_{<ik} \text{ and } h(i)=1} \overline{\Phi}_h}{\sum_{h:\, h_i=z_{<ik}} \overline{\Phi}_h} = \frac{m_k}{i} \tag{C.26}$$

Therefore, each customer selects an old dish with probability $\frac{m_k}{i}$ as in the IBP.

Proof

$$\sum_{h:\, h_i=z_{<ik}, h(i)=1} \overline{\Phi}_h = \sum_{h:\, z_{<ik}, h(i)=1} \frac{(N-m_h)!(m_h-1)!}{N!} \tag{C.27}$$

$$= \sum_{t=0}^{N-i} \binom{N-i}{t} \frac{(N-t-m_k-1)!(t+m_k)!}{N!} \tag{C.28}$$

$$= \frac{1}{(m_k+1)\binom{i}{m_k+1}} \tag{C.29}$$

$$\sum_{h:\, h_i=z_{<ik}} \overline{\Phi}_h = \sum_{h:\, z_{<ik}} \frac{(N-m_h)!(m_h-1)!}{N!} \tag{C.30}$$

$$= \sum_{t=0}^{N-i+1} \binom{N-i+1}{t} \frac{(N-t-m_k)!(t+m_k-1)!}{N!} \tag{C.31}$$

$$= \sum_{t=0}^{N-i+1} \binom{N-i+1}{t} \frac{(N-t-m_k)!(t+m_k-1)!}{N!} \tag{C.32}$$

$$= \frac{1}{m_k\binom{i-1}{m_k}} \tag{C.33}$$

Together,

$$\frac{\sum_{h:\, h_i=z_{<ik}, h(i)=1} \overline{\Phi}_h}{\sum_{h:\, h_i=z_{<ik}} \overline{\Phi}_h} = \frac{m_k\binom{i-1}{m_k}}{(m_k+1)\binom{i}{m_k+1}} \tag{C.34}$$

$$= \frac{m_k}{i} \tag{C.35}$$

Furthermore, an equivalence class $[\mathbf{Z}]$ can be represented by a frequency vector $\mathbf{K} = (K_1, \ldots, K_{2^N-1})$. We can define a distribution on $\mathbf{K}$ by assuming that each K_h is generated independently by a Poisson distribution with parameters $\alpha\overline{\Phi}_h$. The probability is given by:

$$P(\mathbf{K}) = \prod_{h=1}^{2^N-1} \frac{(\alpha\overline{\Phi}_h)^{K_h}}{K_h!} \exp(-\alpha\overline{\Phi}_h) \tag{C.36}$$

This could be easily seen to be the same as Equation (5.10).

C.3 GO results for clusters in Figure 5.5

Table C.1 show the GO enrichment results for cluster (b), (c) and (d) in Figure 5.5 by GOstat[145]. We only show terms with corrected P-value less than 0.01. Cluster (a), (e) and (f) have no significant terms.

C.4 Comparison with GenMiR++, K-means, and IBP

Figure C.1 shows the network inferred by GenMiR++ with threshold of 0.9. We did not find any significant enrichment with corrected P-value less than 0.01. We ran K-means on the same set of mRNAs in Figure 5.5 using $k = 6$ as inferred by GroupMiR. We did not find any GO enrichment indicating that only by integrating sets of miRNAs with the mRNAs for this data we can find functional biological groupings.

We also tested with the original IBP ($\mathbf{W} = 0$). Not surprisingly, the results for both the synthetic and real data were weak (the IBP is of course not intended for our data since it cannot use the prior interaction information). Specifically, for the synthetic data the average F1 when using a noise level of 0.4 (a high but reasonable level) is 0.8418 for our method and only 0.5163 for the original IBP. For the real data, the IBP failed to recover any significant groupings. Without the priors the ability to identify significant interactions is greatly weakened.

C.5 Networks at 60% posterior probability.

We also report networks constructed with 60% posterior probability by GroupMiR in Figure C.2 and 0.6 threshold by GenMiR++ in Figure C.3.

Table C.2 show the GO enrichment results for two connected componentsin Figure C.3 by GOstat[145]. We only show terms with corrected P-value less than 0.01.

Term ID	Description	P value
GO:22402	cell cycle process	7.32E-05
GO:7049	cell cycle	1.77E-04
GO:22403	cell cycle phase	3.17E-04
GO:278	mitotic cell cycle	2.94E-03
GO:279	M phase	2.94E-03

5.5b

Term ID	Description	P value
GO:45859	regulation of protein kinase activity	8.83E-03
GO:51338	regulation of transferase activity	8.83E-03
GO:165	MAPKKK cascade	8.83E-03

5.5c

Term ID	Description	P value
GO:724	double-strand break repair via homologous recombination	3.29E-03
GO:725	recombinational repair	3.29E-03
GO:6281	DNA repair	3.29E-03
GO:6974	response to DNA damage stimulus	3.93E-03
GO:6310	DNA recombination	3.93E-03
GO:9314	response to radiation	3.93E-03
GO:9719	response to endogenous stimulus	3.93E-03
GO:51053	negative regulation of DNA metabolic process	3.93E-03
GO:8630	DNA damage response, signal transduction resulting in induction of apoptosis	4.13E-03
GO:6302	double-strand break repair	4.78E-03
GO:51052	regulation of DNA metabolic process	4.78E-03
GO:10212	response to ionizing radiation	4.78E-03
GO:9411	response to UV	6.28E-03
GO:7568	aging	7.34E-03
GO:8629	induction of apoptosis by intracellular signals	7.87E-03
GO:6996	organelle organization	9.95E-03
GO:9628	response to abiotic stimulus	9.95E-03
GO:42770	DNA damage response, signal transduction	9.95E-03

5.5d

Table C.1: GO enrichment analysis of clusters in Figure 5.5.

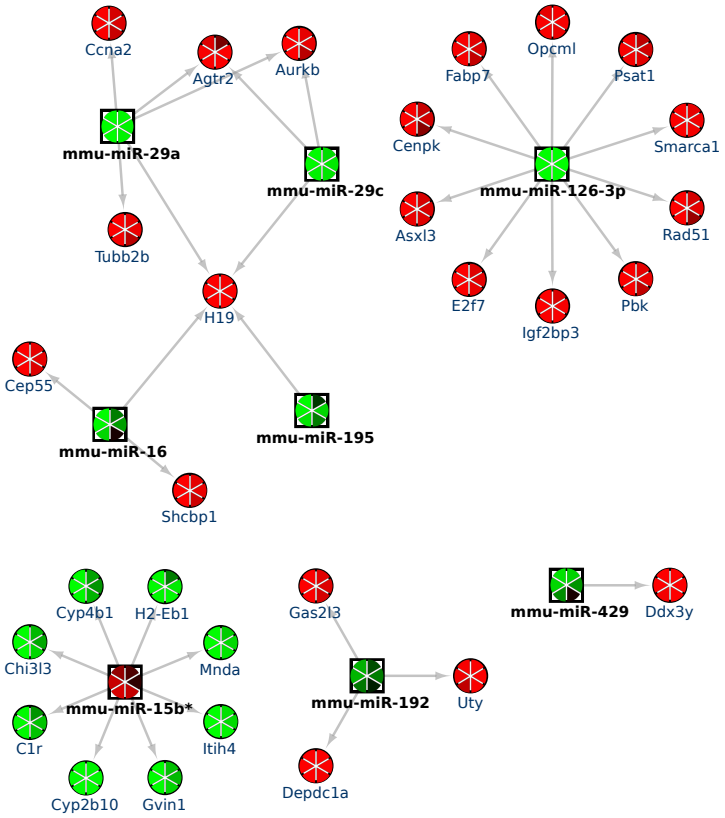


Figure C.1: GenMiR++: only interactions whose scores ≥ 0.9 are shown.

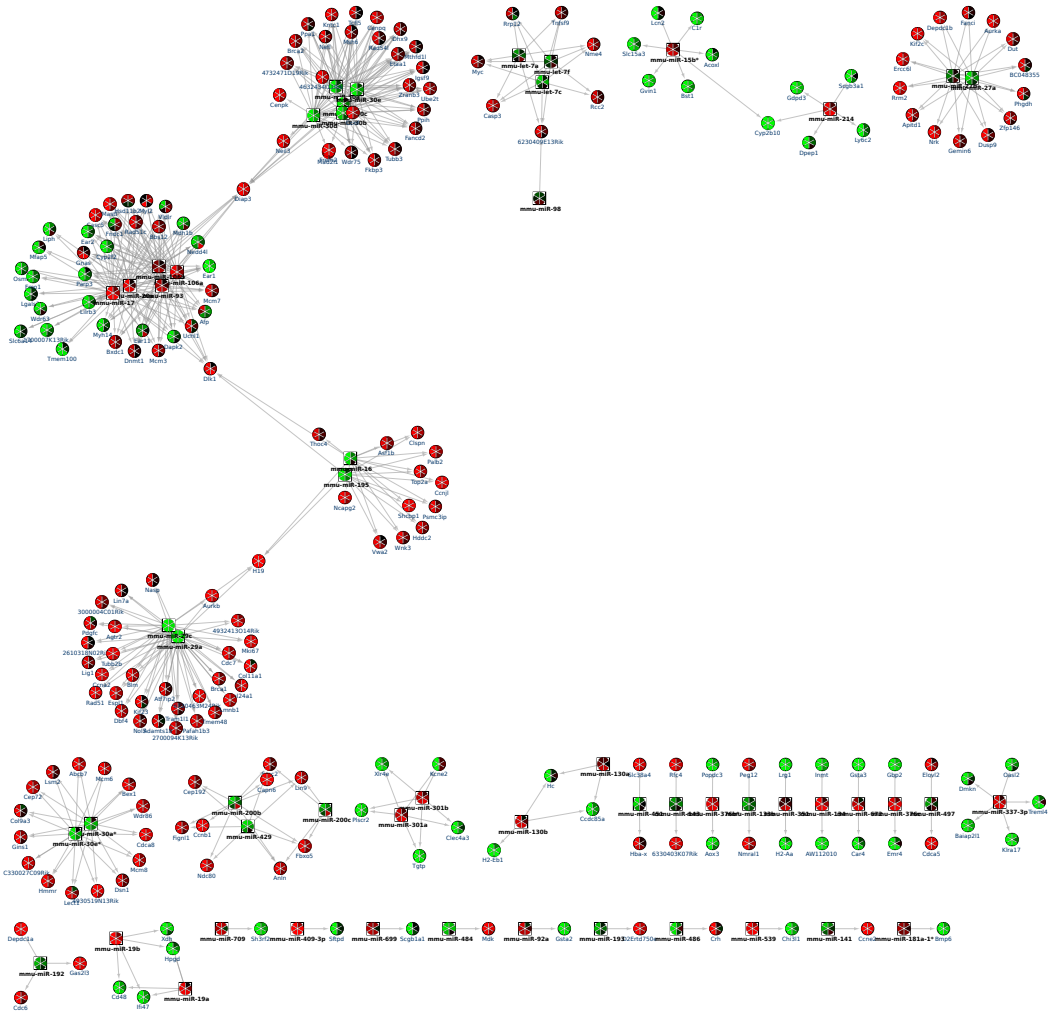


Figure C.2: Network inferred by GroupMiR with 60% posterior probability.

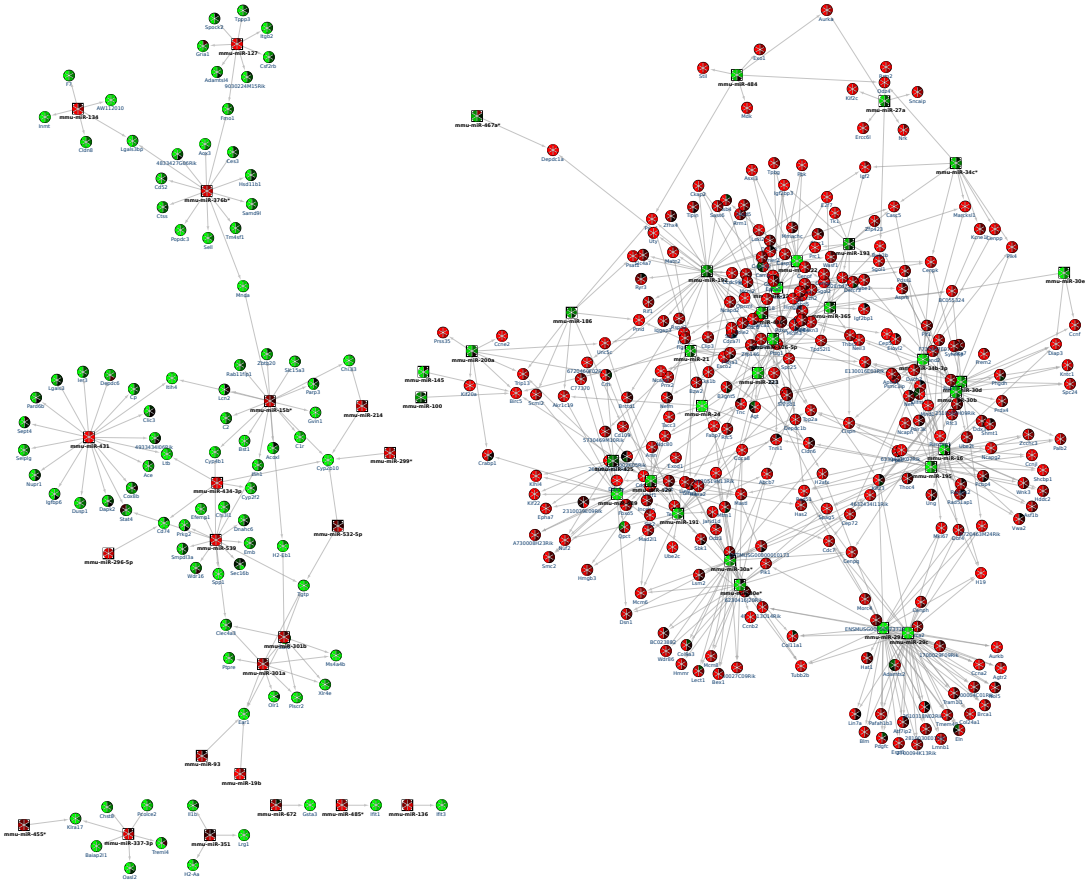


Figure C.3: Network inferred by GenMiR++ with threshold of 0.6.

Term ID	Description	P value
GO:0044421	extracellular region part	0.00541
GO:0005615	extracellular space	0.00541
GO:0005624	membrane fraction	0.00857
GO:0016798	hydrolase activity, acting on glycosyl bonds	0.00857
GO:0005529	sugar binding	0.00907

Component 1

Term ID	Description	P value
GO:0022402	cell cycle	3.79e-71
GO:0006259	DNA metabolic	3.25e-31
GO:0000279	cell cycle phase M	7.76e-29
GO:0000278	mitotic cell	4.86e-27
GO:0000087	M phase of mitotic cell	1.61e-26
GO:0005524	adenyl ribonucleotide binding	4.16e-12
GO:0051726	regulation of cell cycle	1.64e-10
GO:0044430	response to DNA damage	1.87e-08
GO:0032555	purine ribonucleotide	2.47e-08

Component 2

Table C.2: GO results for genes in Figure C.3.

Supplementary materials for Chapter 6

D.1 Solving the optimization problem (6.4)

As discussed in Section 6.2.6, we solve the optimization (6.4) by an iterative procedure. We show here how to compute the gradients of the objective function, which are required for using `minPQN`. We begin with the objective function:

$$\mathcal{F} = -\log p(\mathbf{Y}|\mathbf{U},\mathbf{V},\mathbf{X},\boldsymbol{\mu},\boldsymbol{\Sigma}) - \sum_{i,j}\log p(\phi_{ij}|\mathbf{U},\mathbf{V}) - \sum_{j\neq j'}\log p(\omega_{jj'}\neq 0|\mathbf{V}) \tag{D.1}$$

The first term can be expanded to :

$$\begin{aligned}
&\log p(\mathbf{Y}|\mathbf{U},\mathbf{V},\mathbf{X},\boldsymbol{\mu},\boldsymbol{\Sigma}) \\
=&\sum_j \log \mathcal{N}(\mathbf{y}_j \mid \boldsymbol{\mu} - \mathbf{X}^{\mathrm{T}}((\mathbb{1}_{\boldsymbol{\Phi}})_{,j}\circ(\mathbf{U}\mathbf{v}_j)),\boldsymbol{\Sigma}) \\
=&-\sum_j\sum_p \log(\sqrt{2\pi}\sigma_p) - \frac{1}{2}\Big(\boldsymbol{\mu} - \mathbf{X}^{\mathrm{T}}((\mathbb{1}_{\boldsymbol{\Phi}})_{,j}\circ(\mathbf{U}\mathbf{v}_j)) - \mathbf{y}_j\Big)^{\mathrm{T}}\boldsymbol{\Sigma}^{-1}\Big(\boldsymbol{\mu} - \mathbf{X}^{\mathrm{T}}((\mathbb{1}_{\boldsymbol{\Phi}})_{,j}\circ(\mathbf{U}\mathbf{v}_j)) - \mathbf{y}_j\Big)
\end{aligned}$$

(We abuse the notation in $(\boldsymbol{\mu}^{\mathrm{T}} - \mathbf{Y})$ a bit. When subtracting a matrix from a row vector, we need to vertically replicate the row vector.)

$$\begin{aligned}
=&-N\sum_p \log(\sqrt{2\pi}\sigma_p) - \frac{1}{2}\operatorname{tr}\{(\boldsymbol{\mu}^{\mathrm{T}} - \mathbf{Y})\boldsymbol{\Sigma}^{-1}(\boldsymbol{\mu}^{\mathrm{T}} - \mathbf{Y})^{\mathrm{T}}\} \\
&+\operatorname{tr}\{(\boldsymbol{\mu}^{\mathrm{T}} - \mathbf{Y})\boldsymbol{\Sigma}^{-1}\big((\mathbf{V}\mathbf{U}^{\mathrm{T}}\circ\mathbb{1}_{\boldsymbol{\Phi}}^{\mathrm{T}})\mathbf{X}\big)^{\mathrm{T}}\} - \frac{1}{2}\operatorname{tr}\{\big((\mathbf{V}\mathbf{U}^{\mathrm{T}}\circ\mathbb{1}_{\boldsymbol{\Phi}}^{\mathrm{T}})\mathbf{X}\big)\boldsymbol{\Sigma}^{-1}\big((\mathbf{V}\mathbf{U}^{\mathrm{T}}\circ\mathbb{1}_{\boldsymbol{\Phi}}^{\mathrm{T}})\mathbf{X}\big)^{\mathrm{T}}\}
\end{aligned}$$

Define:

$$\begin{aligned}
\boldsymbol{\Phi}_* &= \boldsymbol{\Phi} + (1 - \mathbb{1}_{\boldsymbol{\Phi}}) \\
\boldsymbol{\Omega}_* &= \boldsymbol{\Omega} + (1 - \mathbb{1}_{\boldsymbol{\Omega}})
\end{aligned}$$

(basically replacing the zero entries with ones.)

We take the derivatives:

Procedure ProjectOnSimplex($\mathbf{v}, C$)

output: $\arg\min_{\mathbf{w}} \|\mathbf{w} - \mathbf{v}\|_2$ s.t. $\sum_i w_i \leq C, \mathbf{w} \geq 0$

Procedure Project($\mathbf{U}, \mathbf{V}, C_1, C_2$)

```
// Threshold
for i, k do
    u_ik arrow 0 if u_ik < ε ;
for j, k do
    v_jk arrow 0 if v_jk < ε ;
for k do
    // Remove redundant entries in U
    for i ∈ {i : u_ik > 0 and u_ik v_,k^T Œ_i = 0} do
        u_ik arrow 0 ;
    // Projection
    u_,k arrow ProjectOnSimplex(u_,k, C_1);
    v_,k arrow ProjectOnSimplex(v_,k, C_2);
```

Figure D.1: Projection procedure to solve the optimization problem (6.4).

$$\frac{\partial \mathcal{F}}{\partial \mathbf{U}} = \underbrace{-\big(\mathbf{X}\boldsymbol{\Sigma}^{-1}(\boldsymbol{\mu}^{\mathrm{T}} - \mathbf{Y})^{\mathrm{T}}\big) \circ \mathbb{1}_{\boldsymbol{\Phi}}\mathbf{V} + \mathbf{X}\boldsymbol{\Sigma}^{-1}\mathbf{X}^{\mathrm{T}}\big((\mathbf{U}\mathbf{V}^{\mathrm{T}}) \circ \mathbb{1}_{\boldsymbol{\Phi}}\big) \circ \mathbb{1}_{\boldsymbol{\Phi}}\mathbf{V}}_{\text{from } \partial \log p(\mathbf{Y}|\mathbf{U},\mathbf{V},\mathbf{X},\boldsymbol{\mu},\boldsymbol{\Sigma})} + \underbrace{\alpha(\sigma^{\boldsymbol{\Phi}} - \mathbb{1}_{\boldsymbol{\Phi}}) \circ \boldsymbol{\Phi}_*\mathbf{V}}_{\text{from } \partial \sum_{i,j} \log p(\mathbb{1}_{\phi_{ij}}|\mathbf{U},\mathbf{V})}$$

where

$$\sigma^{\boldsymbol{\Phi}} = \sigma\big(\alpha\ \boldsymbol{\Phi}_* \circ (\mathbf{U}\mathbf{V}^{\mathrm{T}})\big)$$

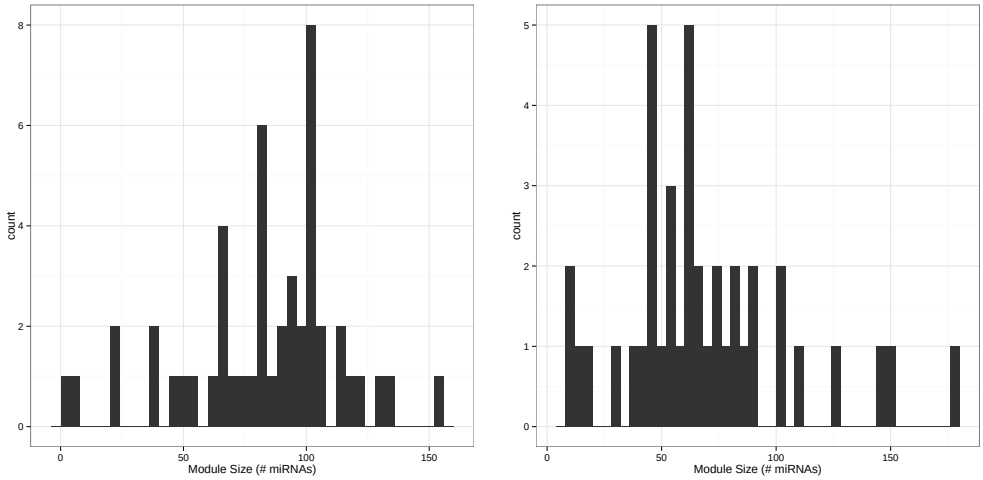
$$\frac{\partial \mathcal{F}}{\partial \mathbf{V}} = \underbrace{-\big((\boldsymbol{\mu}^{\mathrm{T}} - \mathbf{Y})\boldsymbol{\Sigma}^{-1}\mathbf{X}^{\mathrm{T}}\big) \circ \mathbb{1}_{\boldsymbol{\Phi}}^{\mathrm{T}}\mathbf{U} + \Big(\big((\mathbf{U}\mathbf{V}^{\mathrm{T}}) \circ \mathbb{1}_{\boldsymbol{\Phi}}\big)^{\mathrm{T}}\mathbf{X}\boldsymbol{\Sigma}^{-1}\mathbf{X}^{\mathrm{T}}\Big) \circ \mathbb{1}_{\boldsymbol{\Phi}}^{\mathrm{T}}\mathbf{U}}_{\text{from } \partial \log p(\mathbf{Y}|\mathbf{U},\mathbf{V},\mathbf{X},\boldsymbol{\mu},\boldsymbol{\Sigma})} + \underbrace{\beta(\sigma^{\boldsymbol{\Omega}} - \mathbb{1}_{\boldsymbol{\Omega}}) \circ \boldsymbol{\Omega}\mathbf{V}}_{\text{from } \partial \sum_{i,j} \log p(\mathbb{1}_{\omega_{jj'}=1}|\mathbf{V})} + \underbrace{\alpha(\sigma^{\boldsymbol{\Phi}} - \mathbb{1}_{\boldsymbol{\Phi}})^{\mathrm{T}} \circ \boldsymbol{\Phi}_*{}^{\mathrm{T}}\mathbf{U}}_{\text{from } \partial \sum_{i,j} \log p(\mathbb{1}_{\phi_{ij}}|\mathbf{U},\mathbf{V})}$$

where

$$\sigma^{\boldsymbol{\Omega}} = \sigma\big(\beta\ \boldsymbol{\Omega}_* \circ (\mathbf{V}\mathbf{V}^{\mathrm{T}})\big)$$

D.2 Distribution of module sizes

Figure D.2 shows the size of modules identified by SNMNMF and PIMiM. Modules identified by both methods have comparable size.



(a) SNMNMF (b) PIMiM

Figure D.2: The histogram of the size of modules of SNMNMF and PIMiM.

D.3 Choosing the parameters K and α

We select the parameter K of PIMiM that yields the best F1 score as shown in Figure D.3. In addition, we varied the values of α to examine the interplay effect of predictions of miRNA targets and protein interaction data. The result is shown in Figure D.4.

D.4 Enrichment results of several modules from TCGA dataset

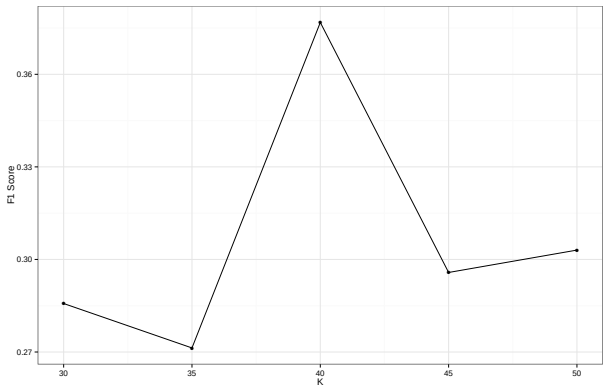


Figure D.3: Performance of PIMiM with different values of K.

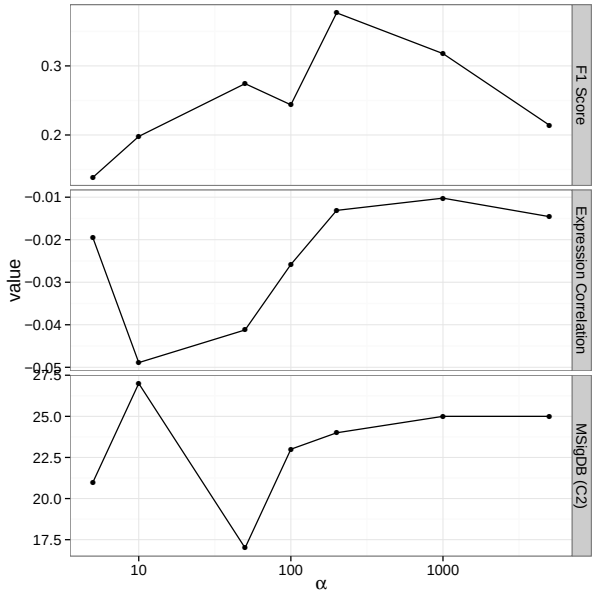


Figure D.4: We varied the value of α and tested the different metrics. On one hand, low values and high values lead to smaller F1 score. On the other hand, small values lead to more coherent gene modules, which explains the better expression correlation.

ID	Name	Adj.P-value
GO:0033276	transcription factor TFTC complex	<0.001
GO:0070461	SAGA-type complex	<0.001
GO:0000123	histone acetyltransferase complex	<0.001
GO:0005669	transcription factor TFIID complex	<0.001
GO:0005667	transcription factor complex	<0.001
GO:0044428	nuclear part	<0.001
GO:0016578	histone deubiquitination	<0.001
GO:0006352	transcription initiation, DNA-dependent	<0.001
GO:0019219	regulation of nucleobase, nucleoside, nucleotide and nucleic acid metabolic process	<0.001
GO:0051171	regulation of nitrogen compound metabolic process	<0.001

GO

Gene Set Name	Description	P value
MIPS TFTC COMPLEX	TFTC complex (TATA-binding protein-free TAF-II-containing complex)	0E0
MIPS GCN5 TRRAP HISTONE ACETYLTRANSFERASE COMPLEX	GCN5-TRRAP histone acetyltransferase complex	0E0
KEGG BASAL TRANSCRIPTION FACTORS	Basal transcription factors	5.55E-16
MIPS TFIID BETA COMPLEX	TFIID-beta complex	1.86E-13
MIPS TFIID BETA COMPLEX 1	TFIID-beta complex	1.86E-13
MIPS STAGA COMPLEX	STAGA complex (SPT3-TAF9-GCN5 acetyltransferase complex)	3.02E-13
MIPS DA COMPLEX	DA complex	7.05E-13
MIPS PCAF COMPLEX	PCAF complex	7.85E-11
MIPS TFIID COMPLEX	TFIID complex	1.85E-10
MIPS TFIID COMPLEX B CELL SPECIFIC	TFIID complex, B-cell specific	1.85E-10

MSigDB

Table D.1: Enrichment analysis of the set of genes in Module 11.

ID	Name	Adj. P-value
GO:0010941	regulation of cell death	<0.001
GO:0010942	positive regulation of cell death	<0.001
GO:0035631	CD40 receptor complex	<0.001
GO:0042981	regulation of apoptosis	<0.001
GO:0043065	positive regulation of apoptosis	<0.001
GO:0043067	regulation of programmed cell death	<0.001
GO:0043068	positive regulation of programmed cell death	<0.001
GO:0008624	induction of apoptosis by extracellular signals	0.004
GO:0009898	internal side of plasma membrane	0.005
GO:0006917	induction of apoptosis	0.015
GO:0012502	induction of programmed cell death	0.015
GO:0048522	positive regulation of cellular process	0.015
GO:0048518	positive regulation of biological process	0.031
GO:0004842	ubiquitin-protein ligase activity	0.032
GO:0019787	small conjugating protein ligase activity	0.041
GO:0051090	regulation of sequence-specific DNA binding transcription factor activity	0.047
GO:0051092	positive regulation of NF-kappaB transcription factor activity	0.047
GO:0035304	regulation of protein dephosphorylation	0.05

GO

Gene Set Name	Description	P value
KEGG SMALL CELL LUNG CANCER	Small cell lung cancer	1.62E-9
BIOCARTA TALL1 PATHWAY	TACI and BCMA stimulation of B cell immune responses	1.02E-7
BIOCARTA TNFR2 PATHWAY	TNFR2 Signaling Pathway	1.82E-7
PID CD40 PATHWAY	CD40/CD40L signaling	9.97E-7
SIG CD40 PATHWAY MAP	Genes related to CD40 signaling	1.33E-6
KEGG PATHWAYS IN CANCER	Pathways in cancer	1.48E-6
PID TNF PATHWAY	TNF receptor signaling pathway	3.35E-6
PID CERAMIDE PATHWAY	Ceramide signaling pathway	3.81E-6
LAU APOPTOSIS CDKN2A UP	Genes up-regulated by UV-irradiation in cervical cancer cells after knockdown of CDKN2A	5.77E-6
REACTOME CELL DEATH SIGNALLING VIA NRAGE NRIF AND NADE	Genes involved in Cell death signalling via NRAGE, NRIF and NADE	7.51E-6

MSigDB

Table D.2: Enrichment analysis of the set of genes in Module 23.

ID	Name	Adj.P-value
GO:0071930	negative regulation of transcription involved in G1/S phase of mitotic cell cycle	0.005
GO:0035189	Rb-E2F complex	0.008
GO:0000122	negative regulation of transcription from RNA polymerase II promoter	0.01
GO:0005634	nucleus	0.032
GO:0003700	sequence-specific DNA binding transcription factor activity	0.033
GO:0001071	nucleic acid binding transcription factor activity	0.033
GO:0003677	DNA binding	0.038
GO:0019219	regulation of nucleobase, nucleoside, nucleotide and nucleic acid metabolic process	0.045

GO

Gene Set Name	Description	P value
PID HES HEYPATHWAY	Notch-mediated HES/HEY network	1.1E-8
MARKS ACETYLATED NON HISTONE PROTEINS	Non-histone proteins that are acetylated	6.14E-8
PARK TRETINOIN RESPONSE AND RARA PLZF FUSION	Genes up-regulated by tretinoin (all-trans retinoic acid, ATRA) in U937 cells (acute promyelocytic leukemia, APL) made resistant to the drug by expression of the PLZF-RARA fusion	2.08E-7
PARK TRETINOIN RESPONSE AND PML RARA FUSION	Genes up-regulated by tretinoin (all-trans retinoic acid, ATRA) in U937 cells (acute promyelocytic leukemia, APL) made sensitive to the drug by expression of the PML-RARA fusion	5.46E-7
MAGRANGEAS MULTIPLE MYELOMA IGLL VS IGLK UP	Up-regulated genes discriminating multiple myeloma samples by the ype of immunoglobulin light chain they produce: Ig lambda (IGLL) vs Ig kappa (IGLK)	1.54E-6
TONKS TARGETS OF RUNX1 RUNX1T1 FUSION HSC DN	Genes down-regulated in normal hematopoietic progenitors by RUNX1-RUNX1T1 fusion.	2.67E-6
PID RB 1PATHWAY	Regulation of retinoblastoma protein	5.81E-6
RAMJAUN APOPTOSIS BY TGFB1 VIA MAPK1 DN	Apoptotic genes dependent on MAPK1 and down-regulated in AML12 cells (hepatocytes) after stimulation with TGFB1	8.09E-6
QI PLASMACYTOMA UP	Up-regulated genes that best disciminate plasmablastic plasmacytoma from plasmacytic plasmacytoma tumors	9.69E-6
PID CMYB PATHWAY	C-MYB transcription factor network	1.26E-5

MSigDB

Table D.3: Enrichment analysis of the set of genes in Module 48.

Bibliography

[1] Travis C Glenn. Field guide to next-generation dna sequencers. *Mol Ecol Resour*, 11(5):759–69, Sep 2011.

[2] Samik Ghosh, Yukiko Matsuoka, Yoshiyuki Asai, Kun-Yi Hsin, and Hiroaki Kitano. Software for systems biology: from tools to integrated platforms. *Nature Reviews Genetics*, 12(12):821–832, 2011.

[3] Daehee Hwang, Alistair G Rust, Stephen Ramsey, Jennifer J Smith, Deena M Leslie, Andrea D Weston, Pedro De Atauri, John D Aitchison, Leroy Hood, Andrew F Siegel, et al. A data integration methodology for systems biology. *Proceedings of the National Academy of Sciences of the United States of America*, 102(48):17296–17301, 2005.

[4] H. Lodish and J.E. Darnell. *Molecular cell biology*, volume 5. Wiley Online Library, 2000.

[5] Todd R Golub, Donna K Slonim, Pablo Tamayo, Christine Huard, Michelle Gaasenbeek, Jill P Mesirov, Hilary Coller, Mignon L Loh, James R Downing, Mark A Caligiuri, et al. Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. *science*, 286(5439):531–537, 1999.

[6] V. Emilsson, G. Thorleifsson, B. Zhang, A.S. Leonardson, F. Zink, J. Zhu, S. Carlson, A. Helgason, G.B. Walters, S. Gunnarsdottir, et al. Genetics of gene expression and its effect on disease. *Nature*, 452(7186):423–428, 2008.

[7] L. He and G.J. Hannon. Micrornas: small rnas with a big role in gene regulation. *Nature Reviews Genetics*, 5(7):522–531, 2004.

[8] J. Krol, I. Loedige, and W. Filipowicz. The widespread regulation of microrna biogenesis, function and decay. *Nature Reviews Genetics*, 11(9):597–610, 2010.

[9] N. Rajewsky. microRNA target predictions in animals. *Nature genetics*, 38:S8–S13, 2006.

[10] S.L. Yu, H.Y. Chen, G.C. Chang, C.Y. Chen, H.W. Chen, S. Singh, C.L. Cheng, C.J. Yu, Y.C. Lee, H.S. Chen, et al. Microrna signature predicts survival and relapse in lung cancer. *Cancer Cell*, 13(1):48–57, 2008.

[11] L. Shi, L.H. Reid, W.D. Jones, R. Shippy, J.A. Warrington, S.C. Baker, P.J. Collins, F. De Longueville, E.S. Kawasaki, K.Y. Lee, et al. The microarray quality control (maqc) project shows inter-and intraplatform reproducibility of gene expression measurements. *Nature biotechnology*, 24(9):1151–1161, 2006.

[12] Michael R Stratton, Peter J Campbell, and P Andrew Futreal. The cancer genome. *Nature*, 458(7239):719–724, 2009.

[13] Frederick Sanger, Steven Nicklen, and Alan R Coulson. Dna sequencing with chain-terminating inhibitors. *Proceedings of the National Academy of Sciences*, 74(12):5463–5467, 1977.

[14] Z. Wang, M. Gerstein, and M. Snyder. Rna-seq: a revolutionary tool for transcriptomics. *Nature Reviews Genetics*, 10(1):57–63, 2009.

[15] T. Griffiths and Z. Ghahramani. Infinite latent feature models and the Indian buffet process. *In Advances in Neural Information Processing Systems*, 18:475, 2006.

[16] Hai-Son Le, Marcel H Schulz, Brenna M McCauley, Veronica F Hinman, and Ziv Bar-Joseph. Probabilistic error correction for rna sequencing. *Nucleic acids research*, 2013.

[17] Zhong Wang, Mark Gerstein, and Michael Snyder. RNA-Seq: a revolutionary tool for transcriptomics. *Nature reviews. Genetics*, 10(1):57–63, January 2009.

[18] John Marioni, Christopher Mason, Shrikant Mane, Matthew Stephens, and Yoav Gilad. RNA-seq: An assessment of technical reproducibility and comparison with gene expression arrays. *Genome research*, June 2008.

[19] Marc Sultan, Marcel H Schulz, Hugues Richard, Alon Magen, Andreas Klingenhoff, Matthias Scherf, Martin Seifert, Tatjana Borodina, Aleksey Soldatov, Dmitri Parkhomchuk, Dominic Schmidt, Sean O'Keeffe, Stefan Haas, Martin Vingron, Hans Lehrach, and Marie-Laure Yaspo. A global view of gene activity and alternative splicing by deep sequencing of the human transcriptome. *Science (New York, N.Y.)*, 321(5891):956–960, August 2008.

[20] Ali Mortazavi, Brian A Williams, Kenneth McCue, Lorian Schaeffer, and Barbara Wold. Mapping and quantifying mammalian transcriptomes by RNA-Seq. *Nature methods*, May 2008.

[21] Zhiyu Peng, Yanbing Cheng, Bertrand Chin-Ming Tan, Lin Kang, Zhijian Tian, Yuankun Zhu, Wenwei Zhang, Yu Liang, Xueda Hu, Xuemei Tan, Jing Guo, Zirui Dong, Yan Liang, Li Bao, and Jun Wang. Comprehensive analysis of RNA-Seq data reveals extensive RNA editing in a human transcriptome. *Nature biotechnology*, 30(3):253–260, March 2012.

[22] Graham A Heap, Jennie H M Yang, Kate Downes, Barry C Healy, Karen A Hunt, Nicholas Bockett, Lude Franke, Patrick C Dubois, Charles A Mein, Richard J Dobson, Thomas J Albert, Matthew J Rodesch, David G Clayton, John A Todd, David A van Heel, and Vincent Plagnol. Genome-wide analysis of allelic expression imbalance in human primary cells by high-throughput transcriptome resequencing. *Hum Mol Genet*, 19(1):122–134, January 2010.

[23] Hugues Richard, Marcel H Schulz, Marc Sultan, Asja Nürnberger, Sabine Schrinner, Daniela Balzereit, Emilie Dagand, Axel Rasche, Hans Lehrach, Martin Vingron, Stefan A Haas, and Marie-Laure Yaspo. Prediction of alternative isoforms from exon expression levels in RNA-Seq experiments. *Nucleic acids research*, 38(10):e112, June 2010.

[24] Adam Roberts, Cole Trapnell, Julie Donaghey, John L Rinn, and Lior Pachter. Improving RNA-Seq expression estimates by correcting for fragment bias. *Genome biology*, 12(3):R22, March 2011.

[25] Gordon Robertson, Jacqueline Schein, Readman Chiu, Richard Corbett, Matthew Field, Shaun D. Jackman, Karen Mungall, Sam Lee, Hisanaga M. Okada, Jenny Q. Qian, Malachi Griffith, Anthony Raymond, Nina Thiessen, Timothee Cezard, Yaron S. Butterfield, Richard Newsome, Simon K. Chan, Rong She, Richard Varhol, Baljit Kamoh, Anna-Liisa Prabhu, Angela Tam, YongJun Zhao, Richard A. Moore, Martin Hirst, Marco A. Marra, Steven J. M. Jones, Pamela A. Hoodless, and Inanc Birol. De novo assembly and analysis of RNA-seq data. *Nature methods*, 7(11):909–912, 2010.

[26] Manfred G Grabherr, Brian J Haas, Moran Yassour, Joshua Z Levin, Dawn A Thompson, Ido Amit, Xian Adiconis, Lin Fan, Raktima Raychowdhury, Qiandong Zeng, Zehua Chen, Evan Mauceli, Nir Hacohen, Andreas Gnirke, Nicholas Rhind, Federica di Palma, Bruce W Birren, Chad Nusbaum, Kerstin Lindblad-Toh, Nir Friedman, and Aviv Regev. Full-length transcriptome assembly from RNA-Seq data without a reference genome. *Nature biotechnology*, 29(7):644–652, July 2011.

[27] Marcel H Schulz, Daniel R. Zerbino, Martin Vingron, and Ewan Birney. Oases: Robust de novo RNA-seq assembly across the dynamic range of expression levels. *Bioinformatics (Oxford, England)*, January 2012.

[28] Jun Li, Hui Jiang, and Wing Hung Wong. Modeling non-uniformity in short-read rates in RNA-Seq data. *Genome biology*, 11(5):R50, 2010.

[29] Kasper D Hansen, Steven E Brenner, and Sandrine Dudoit. Biases in Illumina transcriptome sequencing caused by random hexamer priming. *Nucleic acids research*, 38(12):gkq224, July 2010.

[30] Alicia Oshlack and Matthew J Wakefield. Transcript length bias in RNA-seq data confounds systems biology. *Biology direct*, 4:14, 2009.

[31] James H Bullard, Elizabeth Purdom, Kasper D Hansen, and Sandrine Dudoit. Evaluation of statistical methods for normalization and differential expression in mRNA-Seq experiments. *BMC bioinformatics*, 11:94, 2010.

[32] Davide Risso, Katja Schwartz, Gavin Sherlock, and Sandrine Dudoit. GC-Content Normalization for RNA-Seq Data. *BMC bioinformatics*, 12:480, 2011.

[33] J.C. Dohm, C. Lottaz, T. Borodina, and H. Himmelbauer. Substantial biases in ultra-short read data sets from high-throughput dna sequencing. *Nucleic acids research*, 36(16):e105–e105, 2008.

[34] Linnéa Smeds and Axel Künstner. ConDeTri–a content dependent read trimmer for Illumina data. *PloS one*, 6(10):e26314, 2011.

[35] Xiao Yang, Sriram P Chockalingam, and Srinivas Aluru. A survey of error-correction methods for next-generation sequencing. *Briefings in bioinformatics*, April 2012.

[36] J. Schröder, H. Schröder, S.J. Puglisi, R. Sinha, and B. Schmidt. SHREC: a short-read error correction method. *Bioinformatics (Oxford, England)*, 25(17):2157–2163, September 2009.

[37] Lucian Ilie, Farideh Fazayeli, and Silvana Ilie. HiTEC: accurate error correction in high-throughput sequencing data. *Bioinformatics (Oxford, England)*, 27(3):295–302, February 2011.

[38] David R Kelley, Michael C Schatz, and Steven L Salzberg. Quake: quality-aware detection and correction of sequencing errors. *Genome biology*, 11(11):R116, 2010.

[39] Wei-Chun Kao, Andrew H Chan, and Yun S Song. ECHO: a reference-free short-read error correction algorithm. *Genome research*, 21(7):1181–1192, July 2011.

[40] Leena Salmela and Jan Schröder. Correcting errors in short reads by multiple alignments. *Bioinformatics (Oxford, England)*, 27(11):1455–1461, June 2011.

[41] Paul Medvedev, Eric Scott, Boyko Kakaradov, and Pavel Pevzner. Error correction of high-throughput sequencing datasets with non-uniform coverage. *Bioinformatics (Oxford, England)*, 27(13):i137–41, July 2011.

[42] Wei Qu, Shin-Ichi Hashimoto, and Shinichi Morishita. Efficient frequency-based de novo short-read clustering for error trimming in next-generation sequencing. *Genome research*, 19(7):1309–1315, July 2009.

[43] Edward Wijaya, Martin C Frith, Yutaka Suzuki, and Paul Horton. Recount: expectation maximization based error correction tool for next generation sequencing data. *Genome Inform*, 23(1):189–201, October 2009.

[44] Ergude Bao, Tao Jiang, Isgouhi Kaloshian, and Thomas Girke. SEED: efficient clustering of next-generation sequences. *Bioinformatics (Oxford, England)*, 27(18):2502–2509, September 2011.

[45] S.R. Eddy. Profile hidden Markov models. *Bioinformatics (Oxford, England)*, 14(9):755–763, 1998.

[46] Guillaume Marçais and Carl Kingsford. A fast, lock-free approach for efficient parallel counting of occurrences of k-mers. *Bioinformatics (Oxford, England)*, 27(6):764–770, March 2011.

[47] Andreas Döring, David Weese, Tobias Rausch, and Knut Reinert. SeqAn an efficient, generic C++ library for sequence analysis. *BMC bioinformatics*, 9:11, 2008.

[48] A.Y. Ng, M.I. Jordan, and Y. Weiss. On spectral clustering: Analysis and an algorithm. *Advances in neural information processing systems*, 2:849–856, 2002.

[49] H. Zha, C. Ding, M. Gu, X. He, and H. Simon. Spectral relaxation for k-means clustering. *Advances in neural information processing systems*, 14:1057–1064, 2001.

[50] U. Von Luxburg. A tutorial on spectral clustering. *Statistics and Computing*, 17(4):395–416, 2007.

[51] P. Liang and D. Klein. Online EM for unsupervised models. In *Proceedings of human language technologies: The 2009 annual conference of the North American chapter of the association for computational linguistics*, pages 611–619. Association for Computational Linguistics, 2009.

[52] Cole Trapnell, Lior Pachter, and Steven L Salzberg. TopHat: discovering splice junctions with RNA-Seq. *Bioinformatics (Oxford, England)*, 25(9):1105–1111, May 2009.

[53] Moran N Cabili, Cole Trapnell, Loyal Goff, Magdalena Koziol, Barbara Tazon-Vega, Aviv Regev, and John L Rinn. Integrative annotation of human large intergenic noncoding RNAs reveals global properties and specific subclasses. *Genes & development*, 25(18):1915–1927, September 2011.

[54] Tanya Barrett, Dennis B Troup, Stephen E Wilhite, Pierre Ledoux, Carlos Evangelista, Irene F Kim, Maxim Tomashevsky, Kimberly A Marshall, Katherine H Phillippy, Patti M Sherman, Rolf N Muertter, Michelle Holko, Oluwabukunmi Ayanbule, Andrey Yefanov, and Alexandra Soboleva. NCBI GEO: archive for functional genomics data sets–10 years on. *Nucleic acids research*, 39(Database issue):D1005–10, January 2011.

[55] Xiao Yang, Karin S Dorman, and Srinivas Aluru. Reptile: representative tiling for short read error correction. *Bioinformatics (Oxford, England)*, 26(20):2526–2533, October 2010.

[56] W James Kent. BLAT–the BLAST-like alignment tool. *Genome research*, 12(4):656–664, April 2002.

[57] Scott F. Saccone, Jiaxi Quan, Gaurang Mehta, Raphael Bolze, Prasanth Thomas, Ewa Deelman, Jay A. Tischfield, and John P. Rice. New tools and methods for direct programmatic access to the dbsnp relational database. *Nucleic Acids Research*, 39(suppl 1):D901–D907, 2011.

[58] Anne-Katrin Emde, Marcel H. Schulz, David Weese, Ruping Sun, Martin Vingron, Vera M. Kalscheuer, Stefan A. Haas, and Knut Reinert. Detecting genomic indel variants with exact breakpoints in single- and paired-end sequencing data using splazers. *Bioinformatics*, 2012.

[59] Eric H Davidson, Jonathan P Rast, Paola Oliveri, Andrew Ransick, Cristina Calestani, Chiou-Hwa Yuh, Takuya Minokawa, Gabriele Amore, Veronica Hinman, Cesar Arenas-Mena, Ochan Otim, C Titus Brown, Carolina B Livi, Pei Yun Lee, Roger Revilla, Alistair G Rust, Zheng jun Pan, Maria J Schilstra, Peter J C Clarke, Maria I Arnone, Lee Rowen, R Andrew Cameron, David R McClay, Leroy Hood, and Hamid Bolouri. A genomic regulatory network for development. *Science (New York, N.Y.)*, 295(5560):1669–1678, March 2002.

[60] Veronica F Hinman and Eric H Davidson. Evolutionary plasticity of developmental gene regulatory network architecture. *Proceedings of the National Academy of Sciences of the United States of America*, 104(49):19404–19409, December 2007.

[61] Andreas Untergasser, Harm Nijveen, Xiangyu Rao, Ton Bisseling, René Geurts, and Jack A M Leunissen. Primer3Plus, an enhanced web interface to Primer3. *Nucleic acids research*, 35(Web Server issue):W71–4, July 2007.

[62] Christiam Camacho, George Coulouris, Vahram Avagyan, Ning Ma, Jason Papadopoulos, Kevin Bealer, and Thomas L Madden. BLAST+: architecture and applications. *BMC bioinformatics*, 10:421, 2009.

[63] UniProt Consortium. Ongoing and future developments at the Universal Protein Resource. *Nucleic acids research*, 39(Database issue):D214–9, January 2011.

[64] Michael Y Galperin and Xosé M Fernández-Suárez. The 2012 Nucleic Acids Research Database Issue and the online Molecular Biology Database Collection. *Nucleic acids research*, 40(Database issue):D1–8, January 2012.

[65] Bo Li and Colin N Dewey. RSEM: accurate transcript quantification from RNA-Seq data with or without a reference genome. *BMC bioinformatics*, 12:323, 2011.

[66] Ben Langmead, Cole Trapnell, Mihai Pop, and Steven L Salzberg. Ultrafast and memory-efficient alignment of short DNA sequences to the human genome. *Genome biology*, 10(3):R25, 2009.

[67] Gabriel F Berriz, John E Beaver, Can Cenik, Murat Tasan, and Frederick P Roth. Next generation software for functional trend analysis. *Bioinformatics (Oxford, England)*, 25(22):3043–3044, November 2009.

[68] H Wada and N Satoh. Phylogenetic relationships among extant classes of echinoderms, as inferred from sequences of 18S rDNA, coincide with relationships deduced from the fossil record. *Journal of molecular evolution*, 38(1):41–49, January 1994.

[69] Huixia Du, Zhenmin Bao, Rui Hou, Shan Wang, Hailin Su, Jingjing Yan, Meilin Tian, Yan Li, Wen Wei, Wei Lu, Xiaoli Hu, Shi Wang, and Jingjie Hu. Transcriptome sequencing and characterization for the sea cucumber Apostichopus japonicus (Selenka, 1867). *PloS one*, 7(3):e33311, 2012.

[70] A comprehensive comparison of rna-seq-based transcriptome analysis from reads to differential gene expression and cross-comparison with microarrays: a case study in saccharomyces cerevisiae. *Nucleic Acids Research*, 2012.

[71] H.-S. Le, Z.N. Oltvai, and Z. Bar-Joseph. Cross-species queries of large gene expression databases. *Bioinformatics*, 26(19):2416, 2010.

[72] D. Lee, O. Redfern, and C. Orengo. Predicting protein function from sequence and structure. *Nature reviews. Molecular cell biology*, 8(12):995–1005, December 2007.

[73] S. F. Altschul, T. L. Madden, A. A. Schäffer, J. Zhang, Z. Zhang, W. Miller, and D. J. Lipman. Gapped blast and psi-blast: a new generation of protein database search programs. *Nucleic acids research*, 25(17):3389–3402, September 1997.

[74] Art B. Owen, Josh Stuart, Kathy Mach, Anne M. Villeneuve, and Stuart Kim. A gene recommender algorithm to identify coexpressed genes in c. elegans. *Genome Res.*, 13(8):1828–1837, August 2003.

[75] Lars J. Jensen, Thomas S. Jensen, Ulrik de Lichtenberg, Søren Brunak, and Peer Bork. Co-evolution of transcriptional and post-translational cell-cycle regulation. *Nature*, September 2006.

[76] G. Lelandais, V. Tanty, C. Geneix, C. Etchebest, C. Jacq, and F. Devaux. Genome adaptation to chemical stress: clues from comparative transcriptomics in Saccharomyces cerevisiae and Candida glabrata. *Genome Biol.*, 9:R164, 2008.

[77] Joshua M. Stuart, Eran Segal, Daphne Koller, and Stuart K. Kim. A gene-coexpression network for global discovery of conserved genetic modules. *Science*, 302(5643):249–255, October 2003.

[78] Yong Lu, Peter Huggins, and Ziv Bar-Joseph. Cross species analysis of microarray expression data. *Bioinformatics*, 25(12):1476–1483, June 2009.

[79] J. L. Bussiere. Species selection considerations for preclinical toxicology studies for biotherapeutics. *Expert Opin Drug Metab Toxicol*, 4:871–877, Jul 2008.

[80] P. Tamayo, D. Scanfeld, B. L. Ebert, M. A. Gillette, C. W. Roberts, and J. P. Mesirov. Metagene projection for cross-platform, cross-species characterization of global transcriptional states. *Proc. Natl. Acad. Sci. U.S.A.*, 104:5959–5964, Apr 2007.

[81] W. Fujibuchi, L. Kiseleva, T. Taniguchi, H. Harada, and P. Horton. CellMontage: similar expression profile search server. *Bioinformatics*, 23:3103–3104, Nov 2007.

[82] L. Hunter, R. C. Taylor, S. M. Leach, and R. Simon. GEST: a gene expression search tool based on a novel Bayesian similarity metric. *Bioinformatics*, 17 Suppl 1:S115–122, 2001.

[83] Persi Diaconis. *Group representations in probability and statistics*. Institute of Mathematical Statistics Lecture Notes—Monograph Series, 11. Institute of Mathematical Statistics, Hayward, CA, 1988.

[84] T. Hastie, R. Tibshirani, and J. H. Friedman. *The Elements of Statistical Learning*. Springer, corrected edition, July 2009.

[85] Jorge Nocedal and Stephen J. Wright. *Numerical Optimization*. Springer Series in Operations Research. Springer, New York, 2006.

[86] E. T. Chan, G. T. Quon, G. Chua, T. Babak, M. Trochesset, R. A. Zirngibl, J. Aubin, M. J. Ratcliffe, A. Wilde, M. Brudno, Q. D. Morris, and T. R. Hughes. Conservation of core gene expression in vertebrate tissues. *J. Biol.*, 8:33, 2009.

[87] J. P. Brunet, P. Tamayo, T. R. Golub, and J. P. Mesirov. Metagenes and molecular pattern discovery using matrix factorization. *Proc. Natl. Acad. Sci. U.S.A.*, 101:4164–4169, Mar 2004.

[88] Jesse Davis and Mark Goadrich. The relationship between precision-recall and roc curves. In *ICML '06: Proceedings of the 23rd international conference on Machine learning*, pages 233–240, New York, NY, USA, 2006. ACM.

[89] Andrew I. Su, Tim Wiltshire, Serge Batalov, Hilmar Lapp, Keith A. Ching, David Block, Jie Zhang, Richard Soden, Mimi Hayakawa, Gabriel Kreiman, Michael P. Cooke, John R. Walker, and John B. Hogenesch. A gene atlas of the mouse and human protein-encoding transcriptomes. *Proceedings of the National Academy of Sciences of the United States of America*, 101(16):6062–6067, April 2004.

[90] ScienceDaily Boston College. Biologists build a better mouse model for cancer research, 2008.

[91] N. E. Sharpless and R. A. Depinho. The mighty mouse: genetically engineered mouse models in cancer drug development. *Nat Rev Drug Discov*, 5:741–754, Sep 2006.

[92] M. L. Whitfield, G. Sherlock, A. J. Saldanha, J. I. Murray, C. A. Ball, K. E. Alexander, J. C. Matese, C. M. Perou, M. M. Hurt, P. O. Brown, and D. Botstein. Identification of genes periodically expressed in the human cell cycle and their expression in tumors. *Mol. Biol. Cell*, 13:1977–2000, Jun 2002.

[93] J. Ernst and Z. Bar-Joseph. STEM: a tool for the analysis of short time series gene expression data. *BMC Bioinformatics*, 7:191, 2006.

[94] Aharon Bar-Hillel, Tomer Hertz, Noam Shental, and Daphna Weinshall. Learning a mahalanobis metric from equivalence constraints. *J. Mach. Learn. Res.*, 6:937–965, 2005.

[95] H.-S. Le and Z. Bar-Joseph. Cross species expression analysis using a dirichlet process mixture model with latent matchings. In *Advances in Neural Information Processing Systems, to appear*, volume 22, 2010.

[96] J. M. Stuart, E. Segal, D. Koller, and S. K. Kim. A gene-coexpression network for global discovery of conserved genetic modules. *Science*, 302:249–255, Oct 2003.

[97] Sven Bergmann, Jan Ihmels, and Naama Barkai. Similarities and differences in genome-wide expression data of six organisms. *PLoS Biol*, 2(1):e9, 12 2003.

[98] G. Quon, Y. W. Teh, E. Chan, M. Brudno, T. Hughes, and Q. D. Morris. A mixture model for the evolution of gene expression in non-homogeneous datasets. In *Advances in Neural Information Processing Systems*, volume 21, 2009.

[99] Y. Lu, R. Rosenfeld, and Z. Bar-Joseph. Identifying cycling genes by combining sequence homology and expression data. *Bioinformatics*, 22:e314–322, Jul 2006.

[100] Y. Lu, R. Rosenfeld, G. J. Nau, and Z. Bar-Joseph. Cross species expression analysis of innate immune response. *J. Comput. Biol.*, 17:253–268, Mar 2010.

[101] R. Sharan et al. Conserved patterns of protein interaction in multiple species. *Proc. Natl. Acad. Sci. U.S.A.*, 102:1974–1979, Feb 2005.

[102] Thomas S. Ferguson. A bayesian analysis of some nonparametric problems. *The Annals of Statistics*, 1(2):209–230, 1973.

[103] J. Sethuraman. A constructive definition of dirichlet priors. *Statistica Sinica*, 4:639–650, 1994.

[104] Michael I. Jordan, Zoubin Ghahramani, Tommi S. Jaakkola, and Lawrence K. Saul. An introduction to variational methods for graphical models. *Machine Learning*, 37(2):183–233, November 1999.

[105] Martin J. Wainwright and Michael I. Jordan. Graphical models, exponential families, and variational inference. *Found. Trends Mach. Learn.*, 1(1-2):1–305, 2008.

[106] H. Ishwaran and James. Gibbs sampling methods for stick breaking priors. *Journal of the American Statistical Association*, pages 161–173, March 2001.

[107] Zoubin Ghahramani and Matthew J. Beal. Propagation algorithms for variational bayesian learning. In *In Advances in Neural Information Processing Systems 13*, pages 507–513. MIT Press, 2001.

[108] Sanjoy Dasgupta. Learning mixtures of gaussians. In *FOCS '99: Proceedings of the 40th Annual Symposium on Foundations of Computer Science*, Washington, DC, USA, 1999.

[109] M. Meila. Comparing clusterings by the variation of information. In *Learning theory and Kernel machines: 16th Annual Conference on Learning Theory and 7th Kernel Workshop, COLT/Kernel 2003, Washington, DC, USA, August 24-27, 2003: proceedings*, page 173. Springer Verlag, 2003.

[110] C. S. Detweiler et al. Host microarray analysis reveals a role for the Salmonella response regulator phoP in human macrophage cell death. *Proc. Natl. Acad. Sci. U.S.A.*, 98:5850–5855, May 2001.

[111] K. van Erp et al. Role of strain differences on host resistance and the transcriptional response of macrophages to infection with Yersinia enterocolitica. *Physiol. Genomics*, 25:75–84, 2006.

[112] D. M. Monack, B. Raupach, et al. Salmonella typhimurium invasion induces apoptosis in infected macrophages. *Proc. Natl. Acad. Sci. U.S.A.*, 93:9833–9838, Sep 1996.

[113] O. O. Zharskaia et al. [Activation of transcription of ribosome genes following human embryo fibroblast infection with cytomegalovirus in vitro]. *Tsitologiia*, 45:690–701, 2003.

[114] J. W. Gow, S. Hagan, P. Herzyk, C. Cannon, P. O. Behan, and A. Chaudhuri. A gene signature for post-infectious chronic fatigue syndrome. *BMC Med Genomics*, 2:38, 2009.

[115] H.-S. Le and Z. Bar-Joseph. Inferring interaction networks using the ibp applied to microrna target prediction. In *Advances in Neural Information Processing Systems, to appear*, 2011.

[116] D.P. Bartel. Micrornas: target recognition and regulatory functions. *Cell*, 136(2):215–233, 2009.

[117] N.Y. Shao, H.Y. Hu, Z. Yan, Y. Xu, H. Hu, C. Menzel, N. Li, W. Chen, and P. Khaitovich. Comprehensive survey of human brain microrna by deep sequencing. *BMC genomics*, 11(1):409, 2010.

[118] N. Meola, V.A. Gennarino, and S. Banfi. micrornas and genetic diseases. *Pathogenetics*, 2(1):7, 2009.

[119] A. Krek, D. Grün, M.N. Poy, R. Wolf, L. Rosenberg, E.J. Epstein, P. MacMenamin, I. da Piedade, K.C. Gunsalus, M. Stoffel, et al. Combinatorial microrna target predictions. *Nature genetics*, 37(5):495–500, 2005.

[120] S. Motameny, S. Wolters, P. Nürnberg, and B. Schumacher. Next generation sequencing of mirnas–strategies, resources and methods. *Genes*, 1(1):70–84, 2010.

[121] Praveen Sethupathy, Molly Megraw, and Artemis G Hatzigeorgiou. A guide through present computational approaches for the identification of mammalian microrna targets. *Nature methods*, 3(11):881–886, 2006.

[122] J.C. Huang, T. Babak, T.W. Corson, et al. Using expression profiling data to identify human microRNA targets. *Nature methods*, 4(12):1045–1049, 2007.

[123] C. Cheng and L.M. Li. Inferring microRNA activities by combining gene expression with microRNA target prediction. *PLoS One*, 3(4):1989, 2008.

[124] J.G. Joung, K.B. Hwang, J.W. Nam, S.J. Kim, and B.T. Zhang. Discovery of microRNA–mRNA modules via population-based probabilistic learning. *Bioinformatics*, 23(9):1141, 2007.

[125] C.H. Ooi, H.K. Oh, H.Z.A. Wang, A.L.K. Tan, J. Wu, M. Lee, S.Y. Rha, H.C. Chung, D.M. Virshup, and P. Tan. A densely interconnected genome-wide network of micrornas and oncogenic pathways revealed using gene expression signatures. *PLoS Genetics*, 7(12):e1002415, 2011.

[126] G.T. Huang, C. Athanassiou, and P.V. Benos. mirconnx: condition-specific mrna-microrna network integrator. *Nucleic acids research*, 39(suppl 2):W416–W423, 2011.

[127] H. Wang and W.H. Li. Increasing MicroRNA target prediction confidence by the relative R2 method. *Journal of theoretical biology*, 259(4):793–798, 2009.

[128] J.C. Huang, Q.D. Morris, and B.J. Frey. Bayesian inference of microRNA targets from sequence and expression data. *J of Computational Biology*, 14(5):550–563, 2007.

[129] ME Peter. Targeting of mrnas by multiple mirnas: the next step. *Oncogene*, 29(15):2161–2164, 2010.

[130] K.T. Miller, T.L. Griffiths, and M.I. Jordan. The phylogenetic indian buffet process: A non-exchangeable nonparametric prior for latent features. In *UAI*, 2008.

[131] E.M. Airoldi, D.M. Blei, S.E. Fienberg, and E.P. Xing. Mixed membership stochastic blockmodels. *The Journal of Machine Learning Research*, 9:1981–2014, 2008.

[132] K.T. Miller, T.L. Griffiths, and M.I. Jordan. Nonparametric latent feature models for link prediction. *In Advances in Neural Information Processing Systems*, 2009.

[133] Z. Bar-Joseph, G.K. Gerber, T.I. Lee, et al. Computational discovery of gene modules and regulatory networks. *Nature biotechnology*, 21(11):1337–1342, 2003.

[134] Y.J. Wang and G.Y. Wong. Stochastic blockmodels for directed graphs. *Journal of the American Statistical Association*, 82(397):8–19, 1987.

[135] C. Kemp, J.B. Tenenbaum, T.L. Griffiths, , et al. Learning systems of concepts with an infinite relational model. In *Proc. 21st Natl Conf. Artif. Intell.(1)*, page 381, 2006.

[136] E. Meeds, Z. Ghahramani, R.M. Neal, and S.T. Roweis. Modeling dyadic data with binary latent factors. *In Advances in NIPS*, 19:977, 2007.

[137] P. Orbanz and J.M. Buhmann. Nonparametric bayesian image segmentation. *International Journal of Computer Vision*, 77(1):25–45, 2008.

[138] David M. Blei and Peter Frazier. Distance dependent chinese restaurant processes. In *ICML*, pages 87–94. Omnipress, 2010.

[139] M. Zhou, H. Yang, G. Sapiro, D. Dunson, and L. Carin. Dependent hierarchical beta process for image interpolation and denoising. *Proc. Artificial Intelligence and Statistics (AISTATS)*, 2011.

[140] Mingyuan Zhou, Hongxia Yang, Guillermo Sapiro, David B. Dunson, and Lawrence Carin. Covariate-dependent dictionary learning and sparse coding. In *ICASSP*, pages 5824–5827. IEEE, 2011.

[141] S. Chib and E. Greenberg. Understanding the metropolis-hastings algorithm. *Amer. Statistician*, 49(4):327–335, 1995.

[142] J. Dong, G. Jiang, Y.W. Asmann, S. Tomaszek, et al. MicroRNA Networks in Mouse Lung Organogenesis. *PloS one*, 5(5):4645–4652, 2010.

[143] P. Shannon, A. Markiel, O. Ozier, et al. Cytoscape: a software environment for integrated models of biomolecular interaction networks. *Genome research*, 13(11):2498, 2003.

[144] C. Xiao and K. Rajewsky. MicroRNA control in the immune system: basic principles. *Cell*, 136(1):26–36, 2009.

[145] T. Beißbarth and T.P. Speed. GOstat: find statistically overrepresented Gene Ontologies within a group of genes. *Bioinformatics*, 20(9):1464, 2004.

[146] A. Ventura, A.G. Young, M.M. Winslow, et al. Targeted Deletion Reveals Essential and Overlapping Functions of the miR-17-92 Family of miRNA Clusters. *Cell*, 132:875–886, 2008.

[147] C.W. Hsu, H.F. Juan, and H.C. Huang. Characterization of microrna-regulated protein-protein interaction network. *Proteomics*, 8(10):1975–1979, 2008.

[148] H. Liang and W.H. Li. Microrna regulation of human protein–protein interaction network. *Rna*, 13(9):1402, 2007.

[149] S. Sass, S. Dietmann, U. Burk, S. Brabletz, D. Lutter, A. Kowarsch, K. Mayer, T. Brabletz, A. Ruepp, F. Theis, et al. Microrna coordinately regulate protein complexes. *BMC systems biology*, 5(1):136, 2011.

[150] S. Zhang, Q. Li, J. Liu, and X.J. Zhou. A novel computational framework for simultaneous integration of multiple types of genomic data to identify microrna-gene regulatory modules. *Bioinformatics*, 27(13):i401, 2011.

[151] C. Stark, B.J. Breitkreutz, A. Chatr-Aryamontri, L. Boucher, R. Oughtred, M.S. Livstone, J. Nixon, K. Van Auken, X. Wang, X. Shi, et al. The biogrid interaction database: 2011 update. *Nucleic acids research*, 39(suppl 1):D698–D704, 2011.

[152] E. Wingender, X. Chen, R. Hehl, H. Karas, I. Liebich, V. Matys, T. Meinhardt, M. Prüß, I. Reuter, and F. Schacherer. Transfac: an integrated system for gene expression regulation. *Nucleic acids research*, 28(1):316–319, 2000.

[153] S. Griffiths-Jones, R.J. Grocock, S. Van Dongen, A. Bateman, and A.J. Enright. mirbase: microrna sequences, targets and gene nomenclature. *Nucleic acids research*, 34(suppl 1):D140–D144, 2006.

[154] M. Schmidt, E. Van Den Berg, M. Friedlander, and K. Murphy. Optimizing costly functions with simple constraints: A limited-memory projected quasi-newton algorithm. In *Proc. of Conf. on Artificial Intelligence and Statistics*, pages 456–463, 2009.

[155] R. Caruana. Multitask learning. *Machine learning*, 28(1):41–75, 1997.

[156] G. Obozinski, B. Taskar, and M.I. Jordan. Joint covariate selection and joint subspace selection for multiple classification problems. *Statistics and Computing*, 20(2):231–252, 2010.

[157] J. Liu, S. Ji, and J. Ye. Multi-task feature learning via efficient l 2, 1-norm minimization. In *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence*, pages 339–348. AUAI Press, 2009.

[158] I. Koturbash, F.J. Zemp, I. Pogribny, and O. Kovalchuk. Small molecules with big effects: the role of the micrornaome in cancer and carcinogenesis. *Mutation Research/Genetic Toxicology and Environmental Mutagenesis*, 722(2):94–105, 2011.

[159] A. Subramanian, P. Tamayo, V.K. Mootha, S. Mukherjee, B.L. Ebert, M.A. Gillette, A. Paulovich, S.L. Pomeroy, T.R. Golub, E.S. Lander, et al. Gene set enrichment analysis: a knowledge-based approach for interpreting genome-wide expression profiles. *Proceedings of the National Academy of Sciences of the United States of America*, 102(43):15545–15550, 2005.

[160] Adrian Alexa and Jorg Rahnenfuhrer. *topGO: topGO: Enrichment analysis for Gene Ontology*, 2010. R package version 2.2.0.

[161] S. Pelengaris and M. Khan. Oncogenic co-operation in beta-cell tumorigenesis. *Endocrine-Related Cancer*, 8(4):307–314, 2001.

[162] B. John, A.J. Enright, A. Aravin, T. Tuschl, C. Sander, and D.S. Marks. Human microrna targets. *PLoS biology*, 2(11):e363, 2004.

[163] B.P. Lewis, C.B. Burge, and D.P. Bartel. Conserved seed pairing, often flanked by adenosines, indicates that thousands of human genes are microrna targets. *Cell*, 120(1):15–20, 2005.

[164] SM Khoshnaw, AR Green, DG Powe, and IO Ellis. Microrna involvement in the pathogenesis and management of breast cancer. *Journal of clinical pathology*, 62(5):422–428, 2009.

[165] Dasong Hua, Fan Mo, Dong Ding, Lisa Li, Xu Han, Na Zhao, Gregory Foltz, Biaoyang Lin, Qing Lan, and Qiang Huang. A catalogue of glioblastoma and brain micrornas identified by deep sequencing. *Omics: a journal of integrative biology*, 16(12):690–699, 2012.

[166] R. Garzon, F. Pichiorri, T. Palumbo, M. Visentini, R. Aqeilan, A. Cimmino, H. Wang, H. Sun, S. Volinia, H. Alder, et al. Microrna gene expression during retinoic acid-induced differentiation of human acute promyelocytic leukemia. *Oncogene*, 26(28):4148–4157, 2007.

[167] Fengyan Yu, Herui Yao, Pengcheng Zhu, Xiaoqin Zhang, Qiuhui Pan, Chang Gong, Yijun Huang, Xiaoqu Hu, Fengxi Su, Judy Lieberman, and Erwei Song. let-7 regulates self renewal and tumorigenicity of breast cancer cells. *Cell*, 131(6):1109 – 1123, 2007.

[168] S.T. Lee, K. Chu, H.J. Oh, W.S. Im, J.Y. Lim, S.K. Kim, C.K. Park, K.H. Jung, S.K. Lee, M. Kim, et al. Let-7 microrna inhibits the proliferation of human glioblastoma cells. *Journal of neuro-oncology*, 102(1):19–24, 2011.

[169] M. Jongen-Lavrencic, S.M. Sun, M.K. Dijkstra, P.J.M. Valk, and B. Löwenberg. Microrna expression profiling in relation to the genetic heterogeneity of acute myeloid leukemia. *Blood*, 111(10):5078–5085, 2008.

[170] S.L. Lin, D.C. Chang, S.Y. Ying, D. Leu, and D.T.S. Wu. Microrna mir-302 inhibits the tumorigenecity of human pluripotent stem cells by coordinate suppression of the cdk2 and cdk4/6 cell cycle pathways. *Cancer research*, 70(22):9473–9482, 2010.

[171] Irene K Guttilla and Bruce A White. Coordinate regulation of foxo1 by mir-27a, mir-96, and mir-182 in breast cancer cells. *Journal of Biological Chemistry*, 284(35):23204–23216, 2009.

[172] Haifeng Zhao, Donghai Wang, Weiting Du, Dongsheng Gu, and Renchi Yang. Microrna and leukemia: tiny molecule, great function. *Critical reviews in oncology/hematology*, 74(3):149–155, 2010.

[173] Elizabeth O'Day and Ashish Lal. Micrornas and their target gene networks in breast cancer. *Breast Cancer Research: BCR*, 12(2):201, 2010.

[174] Yunqing Li, Fadila Guessous, Ying Zhang, Charles DiPierro, Benjamin Kefas, Elizabeth Johnson, Lukasz Marcinkiewicz, Jinmai Jiang, Yanzhi Yang, Thomas D Schmittgen, et al. Microrna-34a inhibits glioblastoma growth by targeting multiple oncogenes. *Cancer research*, 69(19):7569–7576, 2009.

[175] Thorsten Zenz, Julia Mohr, Eric Eldering, Arnon P Kater, Andreas Bühler, Dirk Kienle, Dirk Winkler, Jan Dürig, Marinus HJ van Oers, Daniel Mertens, et al. mir-34a as part of the resistance network in chronic lymphocytic leukemia. *Blood*, 113(16):3801–3808, 2009.

[176] George A Calin, Amelia Cimmino, Muller Fabbri, Manuela Ferracin, Sylwia E Wojcik, Masayoshi Shimizu, Cristian Taccioli, Nicola Zanesi, Ramiro Garzon, Rami I Aqeilan, et al. Mir-15a and mir-16-1 cluster functions in human leukemia. *Proceedings of the National Academy of Sciences*, 105(13):5166–5171, 2008.

[177] Joshua T Mendell. miriad roles for the mir-17-92 cluster in development and disease. *Cell*, 133(2):217–222, 2008.

[178] A Ernst, B Campos, J Meier, F Devens, F Liesenberg, M Wolter, G Reifenberger, C Herold-Mende, P Lichter, and B Radlwimmer. De-repression of ctgf via the mir-17-92 cluster upon differentiation of human glioblastoma spheroid cultures. *Oncogene*, 29(23):3411–3422, 2010.

[179] Shuangli Mi, Zejuan Li, Ping Chen, Chunjiang He, Donglin Cao, Abdel Elkahloun, Jun Lu, Luis A Pelloso, Mark Wunderlich, Hao Huang, et al. Aberrant overexpression and function of the mir-17-92 cluster in mll-rearranged acute leukemia. *Proceedings of the National Academy of Sciences*, 107(8):3710–3715, 2010.

[180] Manav Korpal, Esther S Lee, Guohong Hu, and Yibin Kang. The mir-200 family inhibits epithelial-mesenchymal transition and cancer cell migration by direct targeting of e-cadherin transcriptional repressors zeb1 and zeb2. *Journal of Biological Chemistry*, 283(22):14910–14914, 2008.

[181] Marcus E Peter. Let-7 and mir-200 micrornas: guardians against pluripotency and cancer progression. *Cell Cycle*, 8(6):843–852, 2009.

[182] S Uhlmann, JD Zhang, A Schwäger, H Mannsperger, Y Riazalhosseini, S Burmester, A Ward, U Korf, S Wiemann, and Ö Sahin. mir-200bc/429 cluster targets plcγ1 and differentially regulates proliferation and egf-driven invasion than mir-200a/141 in breast cancer. *Oncogene*, 29(30):4297–4306, 2010.

[183] Gabriel F Berriz, John E Beaver, Can Cenik, Murat Tasan, and Frederick P Roth. Next generation software for functional trend analysis. *Bioinformatics*, 25(22):3043–3044, 2009.

[184] N Kurabe, K Katagiri, Y Komiya, R Ito, A Sugiyama, Y Kawasaki, and F Tashiro. Deregulated expression of a novel component of tftc/staga histone acetyltransferase complexes, rat sgf29, in hepatocellular carcinoma: possible implication for the oncogenic potential of c-myc. *Oncogene*, 26(38):5626–5634, 2007.

[185] O Wada-Hiraike, T Yano, T Nei, Y Matsumoto, K Nagasaka, S Takizawa, H Oishi, T Arimoto, S Nakagawa, T Yasugi, et al. The dna mismatch repair gene hmsh2 is a potent coactivator of oestrogen receptor α. *British journal of cancer*, 92(12):2286–2291, 2005.

[186] Susceptibility Gene. Breast and ovarian cancer susceptibility gene brca1. *Science*, 266:7, 1994.

[187] Jiaqiang Ren, Ping Jin, Ena Wang, Francesco M Marincola, and David F Stroncek. Microrna and gene expression patterns in the differentiation of human embryonic stem cells. *Journal of Translational Medicine*, 7(1):20, 2009.

[188] S. Volinia, M. Galasso, S. Costinean, L. Tagliavini, G. Gamberoni, A. Drusco, J. Marchesini, N. Mascellani, M.E. Sana, R.A. Jarour, et al. Reprogramming of mirna networks in cancer and leukemia. *Genome research*, 20(5):589–599, 2010.

[189] Yu-Zhuo Pan, Marilyn E Morris, and Ai-Ming Yu. Microrna-328 negatively regulates the expression of breast cancer resistance protein (bcrp/abcg2) in human cancer cells. *Molecular pharmacology*, 75(6):1374–1379, 2009.

[190] Wenwei Hu, Chang S Chan, Rui Wu, Cen Zhang, Yvonne Sun, Jun S Song, Laura H Tang, Arnold J Levine, and Zhaohui Feng. Negative regulation of tumor suppressor p53 by microrna mir-504. *Molecular cell*, 38(5):689–699, 2010.

[191] Pauline Peyrouze, Soizic Guihard, Nathalie Grardel, Céline Berthon, Nicolas Pottier, Arnaud Pigneux, Jean-Yves Cahn, Marie Christine Béné, Véronique Lhéritier, Eric Delabesse, et al. Genetic polymorphisms in arid5b, cebpe, ikzf1 and cdkn2a in relation with risk of acute lymphoblastic leukaemia in adults: a group for research on adult acute lymphoblastic leukaemia (graall) study. *British journal of haematology*, 159(5):599–613, 2012.

[192] Y. Shi, M. Klutstein, I. Simon, T. Mitchell, and Z. Bar-Joseph. A combined expression-interaction model for inferring the temporal activity of transcription factors. *Journal of Computational Biology*, 16(8):1035–1049, 2009.

[193] Sergey Koren, Michael C Schatz, Brian P Walenz, Jeffrey Martin, Jason T Howard, Ganeshkumar Ganapathy, Zhong Wang, David A Rasko, W Richard McCombie, Erich D Jarvis, et al. Hybrid error correction and de novo assembly of single-molecule sequencing reads. *Nature biotechnology*, 2012.

[194] Guy E Zinman. Analysis of high throughput genomic datasets across species. *Ph.D. Thesis*, 2012.

[195] Aaron Wise, Zoltán N Oltvai, et al. Matching experiments across species using expression values and textual information. *Bioinformatics*, 28(12):i258–i264, 2012.

[196] Heng Li, Jue Ruan, and Richard Durbin. Mapping short dna sequencing reads and calling variants using mapping quality scores. *Genome research*, 18(11):1851–1858, 2008.

[197] Jennifer Couzin. The hapmap gold rush: researchers mine a rich deposit. *Science*, 312(5777):1131–1131, 2006.

[198] Peter J Park. Chip–seq: advantages and challenges of a maturing technology. *Nature Reviews Genetics*, 10(10):669–680, 2009.

[199] Arthur Gretton, Kenji Fukumizu, Choon Hui Teo, Le Song, Bernhard Schölkopf, and Alexander J Smola. A kernel statistical test of independence. 2008.

[200] Barnabás Póczos, Liang Xiong, Dougal J Sutherland, and Jeff Schneider. Support distribution machines. *arXiv preprint arXiv:1202.0302*, 2012.

[201] J. Sun, X. Gong, B. Purow, and Z. Zhao. Uncovering microrna and transcription factor mediated regulatory networks in glioblastoma. *PLoS Computational Biology*, 8(7):e1002488, 2012.

[202] Witold Filipowicz, Suvendra N Bhattacharyya, and Nahum Sonenberg. Mechanisms of post-transcriptional regulation by microrna: are the answers in sight? *Nature Reviews Genetics*, 9(2):102–114, 2008.

[203] John G Doench, Christian P Petersen, and Phillip A Sharp. sirnas can function as mirnas. *Genes & development*, 17(4):438–442, 2003.

[204] Manu Setty, Karim Helmy, Aly A Khan, Joachim Silber, Aaron Arvey, Frank Neezen, Phaedra Agius, Jason T Huse, Eric C Holland, and Christina S Leslie. Inferring transcriptional and microrna-mediated regulatory programs in glioblastoma. *Molecular Systems Biology*, 8(1), 2012.

[205] Adam Roberts and Lior Pachter. Streaming fragment assignment for real-time analysis of sequencing experiments. *Nature Methods*, 10(1):71–73, 2013.

[206] Wassily Hoeffding. A combinatorial central limit theorem. *The Annals of Mathematical Statistics*, 22(4):558–566, 1951.

MACHINE LEARNING
DEPARTMENT

Carnegie Mellon University
5000 Forbes Avenue
Pittsburgh, PA15213

Carnegie Mellon