Systems modeling and network inference for genetic and epigenetic regulation of gene expression

Chen, Haifen

2016

Chen, H. (2016). Systems modeling and network inference for genetic and epigenetic regulation of gene expression. Doctoral thesis, Nanyang Technological University, Singapore.

<https://hdl.handle.net/10356/66633>

<https://doi.org/10.32657/10356/66633>

Systems Modeling and Network Inference for Genetic and Epigenetic Regulation of Gene Expression

A THESIS SUBMITTED
FOR THE DEGREE OF DOCTOR OF PHILOSOPHY

by

HAIFEN CHEN (G1101346J)

SCHOOL OF COMPUTER ENGINEERING
NANYANG TECHNOLOGICAL UNIVERSITY

AUGUST 2015

Acknowledgments

First, I would like to express my deepest gratitude to my supervisor Prof. Jie Zheng for introducing me into this exciting field and giving me numerous invaluable advice and guidance. I really admire his vast skill in many areas and the unreserved sharing of his extensive expertise. I truly appreciate his careful proofreading of my papers and many valuable suggestions in writing, which would have a far-reaching impact on my academic life. He always places himself in my position and cares the future of my scholarly career. His profound knowledge, grand passion for research, immense patience and great kindness make him the best PhD supervisor one could ask for.

During my PhD study, I met some erudite professors who gave very useful lectures and helpful advice. In particular, I want to thank Prof. Chee Keong Kwoh, Prof. Mark Featherstone, Prof. Zbynek Bozdech, Prof. Mahesan Niranjan, Dr. Paul Robson, Prof. Alioune Ngom and Prof. Feng Lin for their valuable guidance and support. Thank Prof. J.C. Rajapakse for his great lectures on Computational Biology. I also met many wonderful colleagues and friends, Piyushkumar A. Mundra, Qian Liu, Yifeng Li, Ragini Rai, Min Wu, Zhenhua Li, Liang Zhao, Haiyan Piao, Fan Zhang, Xiaomeng Zhang, and other guys in the lab. Thank all of them for the illuminating discussion and selfless help.

I am grateful to Nanyang Technological University (NTU) and School of Computer Engineering (SCE) for offering me the opportunity to study in this wonderful environment. Thank Mr. SingYau Tan, our lab technician, for his kind and timely assistance when every technical problem comes up.

Last but not least, my special thanks go to my family for their continuous and unconditional love and support.

Contents

Acknowledgments	i
Abstract	vii
Author's Publications	ix
Abbreviation and Symbol	xi
List of Tables	xiii
List of Figures	xv
1 Introduction	1
1.1 Background	1
1.1.1 Gene Regulation and Gene Regulatory Network	1
1.1.2 Epigenetic Regulation of Transcription	3
1.1.3 Epigenetic Landscape	6
1.2 Motivation and Objectives	7
1.3 Contributions	8
1.4 Organization	10
2 Literature Review	13
2.1 Inference of Gene Interactions from Gene Expression Data	13
2.1.1 Gene Regulatory Network Inference	13
2.1.2 Inference of Time-Delayed Gene Regulation	17
2.1.3 Evaluation of GRN Inference	18
2.2 Epigenetic Landscape and Gene Regulation During Embryonic Development	20
2.2.1 Modeling Epigenetic Landscape	20
2.2.2 GRN Inference from Single-cell Transcriptional Data During Embryonic Development	22
2.3 Global Epigenetic Regulation of Gene Expression	24
2.4 Integration of Epigenetic Data in GRN Inference	27

3	Reconstructing GRNs from Single-Cell Gene Expression Data	31
3.1	Introduction	31
3.1.1	Background	31
3.1.2	Motivation	32
3.2	Materials	32
3.2.1	Single-cell Data of Mouse Pre-implantation Development	32
3.2.2	Benchmark Network	34
3.3	Methods	35
3.3.1	Approach Overview	35
3.3.2	Structure Inference of GRN	36
3.3.3	Identifying Operational Interactions in GRN	38
3.4	Experimental Results	40
3.4.1	Results of Structure Inference	40
3.4.2	Analysis of Parameters	45
3.4.3	Results of Identifying Operational Interactions	48
3.5	Conclusions, Discussion and Future Work	50
4	Inference of Time-Delayed GRNs Based on Network Deconvolution	53
4.1	Introduction	53
4.1.1	Background	53
4.1.2	Motivation	54
4.2	Methods	54
4.2.1	Cross-correlation	55
4.2.2	Network Deconvolution	56
4.2.3	Time-delayed ND	57
4.3	Experiments and Results	59
4.3.1	Benchmark Networks and Data	60
4.3.2	Results of Network Inference	62
4.3.3	Results of Time-Delay Inference	69
4.4	Conclusions and Discussion	71
5	Mining Regulatory Patterns between HMs and Gene Expression	73
5.1	Introduction	73
5.1.1	Background	73
5.1.2	Motivation	74
5.2	Data	74
5.3	Methods	75
5.3.1	Approach Overview	75
5.3.2	Association Rule Mining	77

5.3.3	Problem Formulation	78
5.3.4	P-values and Multiple-testing Correction	80
5.3.5	Pseudo Code of Histone Code Mining	81
5.3.6	Gene Expression Prediction Based on Rules	82
5.4	Experimental Results	83
5.4.1	Rules of HMs and Gene Expression	83
5.4.2	Robustness of Rules	86
5.4.3	Comparison with Existing Methods	87
5.5	Conclusions and Future Work	91
6	Integrating Epigenetic Prior in DBN for GRN Inference	93
6.1	Introduction	93
6.1.1	Background	93
6.1.2	Motivation	94
6.2	Methods	94
6.2.1	Learning GRN Structure Based on DBN and MCMC	94
6.2.2	Integration of Epigenetic Prior into GRN Inference	97
6.2.3	Epigenetic Prior Generation	99
6.2.4	Synthetic Expression Data Generation	102
6.3	Experiments and Results	102
6.4	Conclusions and Future Work	108
7	Conclusions and Future Work	109
7.1	Conclusions	109
7.2	Future Directions	111
7.2.1	Cell Cycle	112
7.2.2	Cellular Reprogramming	113
7.2.3	Inference of Large-Scale GRNs	114
7.2.4	Towards Whole-Cell Modeling	115
	References	117

Abstract

Gene expression is a fundamental activity in cellular environment. Due to the complexity of gene regulation, the regulatory mechanisms of gene expression in many cellular processes remain elusive despite a long time of research. Computational methods have been indispensable companions to experimental work by providing predictions and guidance. In this thesis, we propose computational methods to elucidate the regulation of gene expression on two basic and interacting layers, namely genetic and epigenetic layers.

First we focus on the inference of gene regulatory networks (GRNs) from gene expression data, which is a central and challenging problem in systems biology. Here we aim to reconstruct the transcriptional regulatory networks controlling cell fate decisions in mammalian embryonic development. We propose a novel method that integrates the structure of a cell lineage tree with transcriptional patterns from single-cell data. This method adopts probabilistic Boolean network for network modeling, and genetic algorithm as search strategy. Guided by the “directionality” of cell development along branches of the cell-lineage tree, our method is able to accurately infer the regulatory circuits from single-cell gene expression data, in a holistic way. Applied on the single-cell transcriptional data of mouse preimplantation development, our algorithm outperforms the state-of-the-art methods of GRN inference. In addition, we develop an algorithm to infer time-delayed GRNs from time-series gene expression data. Time delay is an essential characteristic of gene regulation. However, the inference of time-delayed GRNs is even more challenging because it is a higher-order inference from less information (i.e. time points in time series data). We propose an algorithm based on cross-correlation and network deconvolution to infer time-delayed GRNs. Experiments on time-series gene expression datasets of yeast show that our method has significantly higher sensitivity and better F-measure than other methods for time-delayed GRN inference.

Then we investigate the relationships between epigenetic modifications and the global gene expression in an organism. Based on the technique of association rule mining, we propose a framework to uncover the relationships between combinatorial histone modifications and the global transcriptional state. Our findings reveal that histone modifications regulate transcription on a large scale and some of the rules are consistent with

the literature, while other rules are *de novo* and could guide further experiments. We also build a model to predict gene expression based on our association rules. Our model outperforms a published Bayesian network model for gene expression prediction using histone modifications. Our results bring new support for the hypothesis of “histone code”, and provide insight into the study of epigenetic regulatory mechanisms of transcription as well as other biological processes.

Furthermore, we integrate epigenetic data into GRN inference to improve the performance. We choose Bayesian network to model a GRN for its strength in integrating prior knowledge. Here dynamic Bayesian network (DBN) is employed to capture the dynamic gene regulation from time series gene expression data. And epigenetic data (histone modifications here) are incorporated into the prior probability of the Bayesian model, using Gibbs distribution. The effectiveness of our method has been evaluated on both simulated data and real-life gene expression data in yeast cell cycle. Experimental results show that the integration of epigenetic data improves the performance of GRN inference, which indicates the intimate relationship between epigenetic modifications and gene regulation.

To conclude, we have made the following contributions in this thesis: proposing a novel method to infer GRNs from single-cell gene expression data, providing an effective method to infer time-delayed GRNs with high sensitivity, developing a framework for histone code mining, and integrating epigenetic data into GRN inference to improve the performance. The mechanisms of gene regulation have been investigated on both genetic and epigenetic layers to achieve more accurate models and give more precise predictions. Our work would help biologists elucidate the complex program of gene regulation, which could provide insight into the molecular basis of normal development as well as diseases.

Author's Publications

Journal Papers

- 1 **Haifen Chen**, Lonardi Stefano, Jie Zheng. Deciphering Histone Code of Transcriptional Regulation in Malaria Parasites by Large-scale Data Mining. *Computational Biology and Chemistry* 2014, 50: 3—10.
- 2 **Haifen Chen**, Piyushkumar A. Mundra, Li Na Zhao, Feng Lin, Jie Zheng. Highly sensitive inference of time-delayed gene regulations by network deconvolution. *BMC Systems Biology* 2014 8(Suppl 4), S6.
- 3 Ragini Rai, Lei Zhu, **Haifen Chen**, Archana Patkar Gupta, Siu Kwan Sze, Jie Zheng, Christiane Ruedl, Zbynek Bozdech, and Mark Featherstone. Genome-wide analysis in Plasmodium falciparum reveals early and late phases of RNA polymerase II occupancy during the infectious cycle. *BMC Genomics* 2014, 15(1): 959.
- 4 **Haifen Chen**, Jing Guo, Paul Robson, Mahesan Niranjan, Jie Zheng. Single-cell transcriptional analysis to uncover regulatory circuits driving cell fate decisions in early mouse development. *Bioinformatics* 2015, 31(7): 1060—1066.
- 5 Yifeng Li¹, **Haifen Chen**¹ (joint first authors), Jie Zheng, Alioune Ngom. The Max-Min High-Order Dynamic Bayesian Network for Learning Gene Regulatory Networks with Time-Delayed Regulations. *IEEE/ACM Transactions on Computational Biology and Bioinformatics* 2015. Accepted.

Conference Papers

- 6 **Haifen Chen**¹, D.A.K. Maduranga¹ (joint first authors), Piyushkumar A. Mundra, Jie Zheng*. Integrating Epigenetic Prior in Dynamic Bayesian Network for Gene Regulatory Network Inference. *IEEE Symposium on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB)*, pp. 76—82. 2013.

Abbreviation and Symbol

GRN	<u>G</u> ene <u>R</u> egulatory <u>N</u> etwork
DBN	<u>D</u> ynamic <u>B</u> ayesian <u>N</u> etwork
MCMC	<u>M</u> arkov <u>C</u> hain <u>M</u> onte <u>C</u> arlo
DAG	<u>D</u> irected <u>A</u> cylic <u>G</u> raph
HM	<u>H</u> istone <u>M</u> odification
GE	<u>G</u> ene <u>E</u> xpression
NGS	<u>N</u> ext <u>G</u> eneration <u>S</u> equencing
TF	<u>T</u> ranscription <u>F</u> actor
SVM	<u>S</u> upport <u>V</u> ector <u>M</u> achine
RF	<u>R</u> andom <u>F</u> orests
PCA	<u>P</u> rincipal <u>C</u> omponent <u>A</u> nalys <u>i</u> s
ND	<u>N</u> etwork <u>D</u> econvolution
RNAPII	RNA <u>P</u> olymerase II
mRNA	<u>m</u> essenger RNA
ODE	<u>O</u> rdinary <u>D</u> ifferential <u>E</u> quation
PBN	<u>P</u> robabilistic <u>B</u> oolean <u>N</u> etwork
STG	<u>S</u> tate <u>T</u> ransition <u>G</u> raph
GA	<u>G</u> enetic <u>A</u> lgorithm
ESC	<u>E</u> mbryonic <u>S</u> tem <u>C</u> ell
TE	<u>T</u> rophectoderm
PE	<u>P</u> rimitive <u>E</u> ndoderm
EPI	<u>E</u> piblast
ICM	<u>I</u> nn <u>e</u> r <u>c</u> ell <u>m</u> ass
iPSC	<u>i</u> nduced <u>P</u> luripotent <u>S</u> tem <u>C</u> ells

List of Tables

2.1	A summary of methods for GRN inference	16
2.2	Methods of modeling epigenetic landscape	22
2.3	Methods of predicting gene expression with histone modifications	25
3.1	Performance comparison of different methods on structure inference . . .	42
3.2	Performance of SingCellNet with different values of K	47
3.3	Performance of SingCellNet on data with unique and duplicate states . .	48
3.4	Rules and their references	49
3.5	Significant rules for lineage decisions	49
4.1	Performance comparison of different methods on <i>yeast5on</i> dataset.	64
4.2	Performance comparison of different methods on <i>yeast5off</i> dataset. . . .	64
4.3	Performance comparison of different methods on <i>yeast9Spellman</i> dataset.	65
4.4	Performance comparison of different methods on <i>yeast9Breedon1</i> dataset.	65
4.5	Performance comparison of different methods on <i>yeast9Breedon2</i> dataset.	65
4.6	Validating the inferred time delays on <i>Spellman</i> data and <i>Breedon</i> data .	70
5.1	Integers corresponding to the events of changes in GE and HMs	79
5.2	Association rules of HMs and GE	83
5.3	Comparison between Bayesian network and our approach in predicting GE based on HMs.	90
6.1	DBN performance of different priors and different numbers of time points	104

List of Figures

1.1	An example of GRN	2
1.2	A schematic of epigenetic modifications	4
1.3	The structural components of a nucleosome	5
1.4	Various types of histone modifications	5
2.1	A depiction of Waddington’s epigenetic landscape	20
3.1	The cell lineage tree and the network for the formation of three cell lineages: TE, PE and EPI	34
3.2	Benchmark networks for (a) $16C \rightarrow TE + ICM$, (b) $ICM \rightarrow PE + EPI$	41
3.3	The inferred networks for $16C \rightarrow TE + ICM$	43
3.4	The inferred networks for $ICM \rightarrow PE + EPI$	44
3.5	The convergence of scores over GA iterations	46
4.1	The flow chart of time-delayed ND.	58
4.2	Alignment of time samples based on time lags.	60
4.3	Benchmark networks of <i>yeast5</i> and <i>yeast9</i>	61
4.4	Comparison between time-delayed ND and the original ND.	63
4.5	Inferred networks on <i>yeast5on</i> by four methods	66
4.6	Inferred networks on <i>yeast5off</i> by four methods	67
4.7	Inferred networks on <i>yeast9Spellman</i> by four methods	68
5.1	Heat-map of transcription levels and HMs of <i>P. falciparum</i>	76
5.2	Distributions of missing data in mRNA and different HMs.	76
5.3	Bayesian network of HMs and GE	86
5.4	The frequency of the significant rules from original dataset also occurring as significant rules in the 20 bootstrapping datasets.	87
5.5	P-values (-log) distribution of the same set of rules in the original dataset and bootstrapping datasets	88
5.6	Comparison of p-values (-log) between random rules and true rules . . .	89
5.7	Comparison between Bayesian network and our method in predicting gene expression	90

6.1	An example of DBN modeling for a two-gene network	95
6.2	Comparison of correlations of HM vectors between random gene-pairs and confirmed gene pairs in yeast.	100
6.3	Comparison of normalized correlations of HM vectors between random gene-pairs and confirmed gene-pairs in yeast.	101
6.4	Performance comparison of different priors for GRN inference on both real data and synthetic data	103
6.5	Performance comparison between GRN inference without prior and with the <i>GIn</i> prior	105
6.6	Comparison of inferred networks by DBN without and with prior	106
6.7	F-measure variation of GRN inference using benchmark as prior with β from 1 to 10.	107

Chapter 1

Introduction

1.1 Background

1.1.1 Gene Regulation and Gene Regulatory Network

Gene expression is a process where genes transfer their genetic information from DNA sequence to functional products such as proteins and functional RNAs. Gene expression is a fundamental activity in cellular environment. The regulation of gene expression (or gene regulation) is critical for cells to maintain their identities and carry out diverse functions. Gene regulation is also crucial to cells responding to environmental signals during development or exposure to pathogens and so on. In this thesis, we focus on gene regulation during transcription, which is the first step of gene expression. Hereafter, we use “transcription” and “gene expression” interchangeably.

During transcription, the messenger RNA (i.e. mRNA) of a gene is generated through a number of steps and undergoes various regulatory mechanisms. In genetic layer, the transcription rates are affected by the bindings of some general transcription factors (TFs), RNA polymerase II (RNAPII), specific TFs, mediator complex and so on. Gene expression could also be regulated in epigenetic layer, where the epigenetic modifications in the DNA sequence or in the histones where DNA sequence is wrapped could greatly influence the expression of genes. This thesis studies gene regulation in both genetic and epigenetic layers. In genetic layer, we focus on the bindings of specific TFs, which will be introduced as follows. The epigenetic regulation will be introduced in Section 1.1.2.

The regulation of gene expression is through the interaction between the *cis*-acting sequences of genes and the *trans*-acting factors. The *trans*-acting factors bind to the *cis*-acting sequences and regulate the expression of genes. Since the *trans*-acting factors are proteins encoded by genes, whose expression is also controlled by the interplay of *cis*-acting sequences and other *trans*-acting factors, the gene expression within an organism is under intricate control by an elaborate gene regulatory network (GRN). Currently there is no uniform definition for GRN. Since there are so many elements that could affect the expression of genes, the structure of GRN would be extremely complicated if all those things are included. In the classic definition of a GRN, an edge often represents the interaction between a specific TF and its target gene. Figure 1.1 shows an example of GRN, where nodes are genes and each edge means that the protein encoded by the source gene acts as an TF of the target gene.

Figure 1.1: An example of GRN. It is a GRN in mouse embryonic stem cells [1]. Arrows denote regulatory interactions, while edges without directions indicate protein-protein interactions. Different colors mean different techniques used to obtain corresponding edges. The figure is reprinted with permission from the corresponding author of [1].

The inference of GRNs (or genetic interactions) is a major task in modern biology. It is the key step to understand the mechanisms in biological systems. GRNs could be detected by wet-lab experiments, e.g. knock-out experiments. However, experimental detection of GRNs is often time-consuming and expensive, especially when the size of GRN is large. Furthermore, knock-out experiments cannot differentiate between direct and indirect interactions of genes. In recent decades, computational inference (or reconstruction) of

GRNs from gene expression data has drawn increasing research interest. Meanwhile, the technology of gene expression profiling is evolving fast, from low-throughput technologies to high-throughput microarray, and to next generation sequencing (NGS) which measures gene expression by *de novo* assembly of sequences. Furthermore, single-cell sequencing technology was developed in recent years to measure gene expression for individual cells, which enable high-accuracy expression profiling for a particular cell stage. These technological advancements offer a better characterization of cellular activities, and call for new computational approaches for the inference of GRNs. The survey of existing methods for GRN inference is provided in Section 2.1.1 of this thesis.

1.1.2 Epigenetic Regulation of Transcription

Epigenetics is an emerging field which studies the heritable changes in cellular activities (e.g. gene expression) caused by functional modifications to the genome that do not involve any change in the primary DNA sequence. Such modifications include DNA methylation, histone modifications, non-coding RNAs (see Figure 1.2) and so on. Epigenetic modifications play an essential role in many cellular processes such as transcription [2], X-chromosome inactivation [3], imprinting [4], meiotic recombination [5], alternative splicing [6] and carcinogenesis [7]. One famous example of epigenetic regulation is cellular differentiation, where pluripotent cells differentiate into a wide variety of specialized cell types including skin cells, muscle cells, neurons, etc. All these cell types possess identical DNA sequence, but they show different phenotypes and carry out diverse functions, which is achieved by activating some genes while inhibiting the others through epigenetic modifications. The study of epigenetics aims to explain how cells with identical genome generate distinct expression of genes and present various phenotypes, and thus is considered a bridge between genotype and phenotype [8, 9].

DNA methylation and histone modification are two major epigenetic mechanisms and are intensively studied. DNA methylation is the first recognized epigenetic modification [11]. It is a type of modification on DNA sequence by adding a methyl group to the cytosine or adenine. As a critical component of gene regulation, DNA methylation is often associated with gene repression or silencing. DNA methylation plays an important part in many cellular processes such as the formation of heterochromatin, silencing of repetitive sequences, X-chromosome inactivation, carcinogenesis, etc. The modifications of DNA methylation are stable through cell divisions and heritable to offspring.

Being another major epigenetic mechanism, histone modification draws increasing attention for its dynamic and diverse regulation on gene expression. Histones are highly conserved proteins in eukaryotic cell nuclei. They are the key components of chromatin. Histones can be classified into five families: H1/H5, H2A, H2B, H3 and H4, where H1/H5 are linker histones between two consecutive nucleosomes. Other four types of Histones

Figure 1.2: A schematic of epigenetic modifications [10]. The figure is reprinted with permission from the author of [10].

H2A, H2B, H3 and H4 are core histones serving as the backbone of a nucleosome (which is the fundamental unit of chromatin, see Figure 1.2), where approximately 147 base pairs of DNA double helix is wrapped around an octamer of core histones, see Figure 1.3. The amino-terminal tails of histones are subject to different posttranslational covalent modifications at different residues, including but not limited to, lysine (K) acetylation, lysine and arginine (R) methylation, see Figure 1.4. The common name of a histone modification is composed of three parts: (1) the name of the histone (e.g. H4); (2) the name of the residue and its position or site in the histone, for example K20, which means the 20th residue (a lysine) from the start (i.e., the N-terminal) of the histone; (3) the type of modification (e.g. Ac: acetyl, Me: methyl, Ub: ubiquitin, P: phosphate) and number of modifications (only applicable to methylation) which could be 1, 2, or 3, denoting mono-, di- or tri-methylation. For example, H4K20me1 represents the monomethylation of the 20th residue (a lysine) of the H4 protein.

The modifications to the histone tails would alter the interaction between histones and the DNA sequence which is wrapped around histones. Hence, histone modifications are crucial to many chromatin-templated processes such as transcription, cell division, DNA repair, DNA replication, etc. The relationship between histone modifications and gene expression has been under intensive research. In contrast to the case that DNA methylation is often related to gene silencing, histone modifications could be linked to

Figure 1.3: The structural components of a nucleosome [12]. The figure is reused with permission from the author of [12].

Figure 1.4: Various types of histone modifications on different sites of residues [13] (A: acetylation, M: methylation, P: phosphorylation, U: ubiquitination). The figure is reused with permission from one of the authors of [13].

gene activation or suppression. Histone modifications mediate gene expression by changing the structure of chromatin and hence affect the accessibility or affinity of DNA to transcription-related proteins such as transcription factors. The acetylation of histones is often associated with transcriptional activation, for it removes the positive charges on the histones and relaxes the affinity between histones and the negatively charged DNA. Histone methylation, on the other hand, can activate or suppress gene expression, depending on the specific site and residue modified, as well as the number of modifications on this

residue, which could be one (mono-methylation), two (di-methylation) or three (tri-methylation). For example, H3K9me1, H3K27me1 and H4K20me1, H3K4me3 are often associated with gene activation, whereas H3K9me2, H3K9me3, H3K27me2, H3K27me3 and H4K20me3 are usually linked to transcriptional repression [14, 15].

Studies have shown that histone modifications regulate gene expression in a cooperative manner [16, 17, 18, 14], which leads to the hypothesis of “histone code”. Similar to the genetic code, the hypothesis of histone code suggests that combinations of different histone modifications correspond to unique chromatin states, which would be read by other proteins and give rise to distinct downstream effects. The deciphering of histone code is essential to shed light on many chromatin-dependent processes such as DNA replication and RNA transcription. The outline of this epigenetic system will bring about far-reaching effects for cell fate decisions and the development of both normal and diseased individuals. In recent years, histone code has evolved into a “language” instead of a simple code [19, 20]. It has been indicated that the effects brought by the interplay of histone modifications are context-dependent (e.g. chromatin signaling pathways), making it more like a complex language.

Histone modifications described above are covalent mechanisms of chromatin variation. On the other hand, the noncovalent mechanisms include chromatin remodeling, histone variants and so on. ATP-dependent chromatin remodeling complexes can slide or eject nucleosomes, which would change the interactions between histones and DNA sequence, thus alter the chromatin accessibility of a particular region [9]. Chromatin remodeling complex (with histone chaperones) can also replace or remove histones with their variants (e.g. H2A.Z, H3.3). The covalent and noncovalent mechanisms cooperate with each other to achieve dynamic regulation of chromatin states.

1.1.3 Epigenetic Landscape

Epigenetic regulation plays a critical role in achieving diverse functions of different cells with identical DNA sequence. By adding modifications such as methylation to DNA sequence or histones, epigenetic regulation determines different sets of genes being expressed in different cell types. For example, some genes are suppressed by the hyper-methylation on the DNA sequences of their promoter regions. DNA methylation often results in gene silencing, because the methylated DNA sequences in promoter regions would prevent the binding of transcriptional proteins.

However, the role of epigenetic modifications in regulating gene expression has been questioned in recent years [21, 22, 23, 24, 25], based on the following justifications. The first one is that epigenetic modifications are not specific to genes, because the enzymes of epigenetic modifications are general for all genes. Transcription factors, on the other hand, have specificity in regulating the expression of genes, due to the specific binding

between a transcription factor and a particular segment of DNA sequence related to a gene [21]. Therefore, epigenetic modifications is unlikely to play a dominating role in gene regulation. They may have to cooperate with transcription factors or other factors with specific binding sites [26, 27, 28]. Another reason for the controversy about epigenetic regulation is that epigenetic modifications such as histone modifications may not be stable through cell cycle, i.e. not heritable [22, 29]. It would be hard to justify that those epigenetic modifications have stable mechanisms to regulate gene expression. Finally, it is still not clear that epigenetic modifications are causative or consequent of gene regulation. They probably play in both ways, according to current experimental evidence [30, 31, 32].

To obtain a deeper understanding of epigenetic regulation, we need to have a systematic view on the mechanisms of gene regulation. Actually in the origin of the word “epigenetics” from C.H. Waddington over half a century ago [8, 33], epigenetics was referred as “the interactions of genes”. In recent years, Waddington’s epigenetic landscape was tested and enriched by modern theories as well as data generated from modern technologies such as microarray and NGS [34, 35, 36, 37, 38]. Models from physics (e.g. the theory of dynamical systems [39]) have been applied to describe and capture the characteristics of epigenetic landscape. Readers are referred to Section 2.2.1 for more information about epigenetic landscape modeling. It is believed that the shape of epigenetic landscape (including valleys and ridges) is determined by the underlying genetic network [36, 21]. Therefore, epigenetic modifications as part of the products of the network probably lie in the intermediate layer for regulating gene expression [28].

1.2 Motivation and Objectives

Understanding gene regulation is critical to elucidating the mechanisms of biological phenomena such as diseases. Gene regulation is a highly complex process. Experimental detection of gene interactions is expensive and time-consuming, especially for a large set of genes. Therefore, computational inference of gene regulatory networks has drawn much interest.

However, GRN inference from gene expression data is a very challenging task, due to the limited data, noise or bias in experiment measurements and so on. Although many methods have been proposed to infer GRNs, the accuracy of GRN inference still has much room for improvement. On the other hand, understanding the underlying mechanism of gene regulation can offer great help to GRN inference. Current methods of GRN inference are mostly data-driven. Although some of them are quite effective in reconstructing genetic relationships, a holistic and systematic view of gene regulation is still missing. Furthermore, these data-driven approaches may have limited power when

the dataset is small or the signals of gene regulation are too weak to detect. Therefore, we need to synthesize the data-driven and knowledge-driven modeling approaches to enhance the performance of reconstructing GRN. In addition, epigenetic modifications have been observed to be closely associated with gene expression. Therefore, an appropriate integration of epigenetic data is promising to improve the accuracy of GRN inference.

Our objective here is to investigate the mechanism of gene regulation and thereby improve the performance of GRN inference. First we want to combine data-driven and knowledge-driven approaches into one model, to improve the accuracy of GRN inference especially when the data are limited, and more importantly to have a better understanding of the regulatory mechanisms of gene expression. In addition, characteristics of gene regulation such as time delay should also be taken into account, for better inference of GRN. Then we explore the relationships between epigenetic modifications and global transcriptional state, to provide evidence for the “histone code” hypothesis. Finally, we aim to improve the inference of gene interactions by incorporating the data of epigenetic modifications.

1.3 Contributions

The main contributions of this thesis are summarized as follows.

- **GRN inference from single-cell transcriptional data using an ensemble approach**

We propose a novel method that integrates the structure of a cell lineage tree with transcriptional patterns from single-cell data. This method adopts probabilistic Boolean network for network modeling, and genetic algorithm as search strategy. Guided by the “directionality” of cell development along branches of the cell lineage tree, our method is able to accurately infer the regulatory circuits from single-cell gene expression data, in a holistic way. Applied on the single-cell transcriptional data of mouse preimplantation development, our algorithm outperforms conventional methods of network inference. Given the network topology, our method can also identify the operational interactions in the gene regulatory network, corresponding to specific cell fate determination. This is one of the first attempts to infer gene regulatory networks from single-cell transcriptional data, incorporating dynamics of cell development along a cell lineage tree.

- **Inferring time-delayed GRN based on network deconvolution (ND)**

We tackle the challenges of inferring higher-order (i.e. time-delayed) gene regulation by developing an algorithm based on cross-correlation and network deconvolution. Using cross-correlation and data processing, we first determine the probable time

lags of gene regulation. Then based on the inferred time delays, the technique of ND is applied to identify direct interactions between the target gene and its regulators. By taking into account the information of time delays among gene interactions, our method is able to infer GRNs more accurately. The effectiveness of our method has been shown by the experiments on three real-life gene expression datasets of yeast. Compared with other existing methods which were designed for learning time-delayed GRNs, our method has significantly higher sensitivity without much reduction of specificity.

- **“Histone code” mining**

We propose a framework based on association rule mining, to discover the relationship between histone modifications and gene expression. Compared with other methods of machine learning such as SVM, our approach provides a straightforward way to obtain unambiguous and interpretable rules which describe how various combinations of histone modifications regulate gene expression. Furthermore, we utilize statistical techniques to calculate the p-value for each rule and give an appropriate criterion to filter out the false-positive rules. In addition, we build a model to predict gene expression using our association rules, which outperforms a published Bayesian network model for gene expression prediction with histone modifications. Our results show that histone modifications regulate gene expression on a large scale and some of the rules are supported by literature, while other rules which are *de novo* identified could guide further experiments. Our findings provide new support for the hypothesis of “histone code”, and offer an insight into the study of epigenetic regulatory mechanisms for transcription as well as other biological processes.

- **Integration of epigenetic data into GRN inference**

We integrate epigenetic data as “prior” into a dynamic Bayesian network based model to improve the performance of GRN inference. We study the underlying characteristics of histone modifications profiles in different genomic regions and propose several options of “prior”. Then the epigenetic data are incorporated into the prior probability of the Bayesian model for GRN inference, using Gibbs distribution. Experiment results show that the integration of epigenetic prior can improve the performance of GRN inference on both synthetic and real-life gene expression data of yeast cell cycle, indicating the close relationship between epigenetic factors and gene regulation.

1.4 Organization

This thesis is organized as follows. In this chapter (i.e. Chapter 1), we have introduced some background information about gene regulation, GRN, epigenetics and so on. Motivation of the studies in this thesis has been described, as well as our objectives and main contributions.

Chapter 2 is the literature review of existing works related to our studies in this thesis. First we summarize the existing methods about GRN inference from gene expression data, including the inference of time-delayed GRNs. Next some performance measures for evaluating the results of GRN inference are introduced. Then we review current studies of epigenetic landscape modeling and the methods of GRN inference from single-cell transcriptional data during mammalian embryonic development. And then we conduct a survey of the existing approaches proposed to uncover the regulatory relationship between epigenetic modifications and gene expression. Finally, we review the methods of integrating prior knowledge and data (including epigenetic data) into the process of inferring GRNs from gene expression data.

In Chapter 3, we propose a novel method which synthesizes the data-driven and knowledge-driven approaches to infer GRNs from single-cell gene expression data. First we introduce our method for reconstructing network structures, which adopts probabilistic Boolean network for network modeling, and genetic algorithm as search strategy. Then we show that our method (after slight adaption) can also be adapted to identify the operational interactions in a given network topology, corresponding to specific cell fate decision. We assess the performance of our algorithm on the single-cell transcriptional data of mouse preimplantation development. Results show that our method outperforms other methods in reconstructing GRNs from single-cell gene expression data. And the operational components identified by our method from a given GRN are mostly consistent with the literature of experiments.

Chapter 4 introduces an algorithm of inferring time-delayed GRNs. We first utilize the technique of cross-correlation to determine probable time lag for the interaction between each pair of genes. Then ND algorithm is applied to identify the direct genetic relationships. The effectiveness of our method has been evaluated by the experiments on three real-life gene expression datasets of yeast.

In Chapter 5, we propose a framework to analyze how combinatorial histone modifications (HMs) regulate gene expression. First, a data mining based technique is applied to obtain the interesting rules between HMs and gene expression. Then we utilize a statistical method to calculate the p-values of the rules, which are further adjusted by a permutation-based multiple testing technique. After filtering out the insignificant rules, we output the rules with statistical significance as “histone code”, which describe how combinatorial HMs influence gene expression. Our rules are validated by the literature.

In addition, we build a model to predict gene expression using our association rules, which shows better accuracy than a published Bayesian model.

Chapter 6 presents a method of integrating epigenetic data into dynamic Bayesian network for GRN inference. First, a normal procedure of GRN inference using DBN is described. Then we introduce our method of integrating epigenetic data into DBN for GRN reconstruction. By taking advantage of the different characteristics of epigenetic data in different genomic regions, we construct a new prior to Bayesian inference of GRNs. Experiments on both the synthetic and real time-series gene expression data of yeast cell cycle show that our method can improve the performance of GRN inference.

Chapter 7 concludes this thesis, and lists a few directions for the future study.

Chapter 2

Literature Review

In this chapter, we first review existing methods of inferring gene regulatory networks (GRNs) from gene expression data. We also summarize the approaches for the inference of time-delayed GRNs. Next, some commonly used performance measures of GRN inference are reviewed. Then we give an overview of current studies on epigenetic landscape modeling and their application to the inference of gene regulation during embryonic development. After that, we survey existing works on exploring the relationships between epigenetic modifications and gene expression. Finally, a literature survey is given about data integration in GRN inference, especially the integration of epigenetic data.

2.1 Inference of Gene Interactions from Gene Expression Data

2.1.1 Gene Regulatory Network Inference

Gene regulatory network is in the fundamental layer of the complex cellular network which achieves the functions of cells. An edge in a GRN often refers to the relationship where the protein translated from the source gene acts as the transcription factor of the target gene. The inference (or reverse engineering) of GRNs is critical to understanding the mechanisms of various cellular processes, as well as the genesis of dysfunctional activities. GRN inference as a central problem in systems biology has drawn a lot of attention in years. Many approaches have been proposed to reconstruct GRNs from high throughput biological data such as microarray data and data from next generation sequencing (NGS). These approaches could be divided into four categories: Boolean

networks, information theory models, differential equations, and Bayesian networks (see [40, 41], and references therein). They all have their own advantages and disadvantages.

Boolean network was proposed by Kauffman in 1969 [42], and he pioneered the use of Boolean networks in modeling genetic regulation. Boolean networks use binary variables to represent the states of genes, and Boolean operations such as AND, OR, NOT to model the interactions among genes. Efficient algorithms have been developed to learn GRNs in the framework of Boolean network [43, 44, 45]. In recent years, Boolean networks are widely applied to analyse regulatory networks and their dynamics (see [46, 47, 48, 49] and reference therein). Boolean networks are simple, easy to interpret, and very efficient for computational modeling. Moreover, they can model the dynamic relationships among genes across different time points. However, their limitations are also obvious. Boolean networks require discretization to transform the continuous data into binary values, which may lose important information. Also, the Boolean functions may oversimplify the relations between genes. Furthermore, Boolean networks are deterministic whereas biological events such as transcription are inherently stochastic. In recent years, probabilistic Boolean network (PBN) [50] was proposed to introduce randomness into the dynamics of Boolean networks, which brings broader application of Boolean network in modeling biological systems. However, PBN is only a trade-off between deterministic and stochastic model. Its power in capturing the stochastic nature of biological systems is still limited.

Information theory models [51] calculate mutual information for each pair of genes and determine connection between two genes if their mutual information is above a predefined threshold. Mutual information of two not-directly-connected genes may still be high when these two genes interact through a third gene. Therefore, some techniques such as data processing inequality (DPI) are utilized to filter out the indirect edges [52]. GRN inference based on information theory is simple and fast, thus it is applicable to the inference of large-scale networks. However, this type of methods is often static and therefore cannot infer the dynamic regulations among genes. Although a dynamic version of this has been developed recently [53], information theory models do not consider multiple regulators thus can only model pairwise regulations between genes.

Differential equations based approaches [54, 55, 56] employ linear or non-linear ordinary differential equations (ODEs) to model the expression changes of a gene as a function of the expression levels of other genes. This type of approaches is able to model the dynamic behavior of a GRN in a quantitative way. Its flexibility enables it to handle the complex relations between genes. However, ODEs are not suitable for modeling large-scale networks, because the differential equations are often difficult to solve or hard to obtain solutions unequivocally. In addition, differential equation models need accurate estimation of parameter values, which remains a challenge in many cases [57].

Bayesian networks [58, 59, 60] use random variables to represent the expression rates of genes, and conditional probabilities to simulate the regulatory relations among genes. Compared to the other three types of approaches, Bayesian networks are more widely used because of at least five advantages [58]: (1) the ability to model randomness and noise, which are inherent in biological processes; (2) the power to learn causal relationships between variables; (3) the flexibility to integrate prior knowledge and heterogeneous data; (4) the capability to handle incomplete data; (5) the capacity to avoid over-fitting the data. Nevertheless, the classic Bayesian networks cannot handle feedback loops because they use directed acyclic graphs (DAGs) as network structures. Therefore, dynamic Bayesian network (DBN) [61] was proposed to cope with such a problem. DBN applies Bayesian network to infer relationships of variables over adjacent time steps, and thus can handle feedback loops among variables. Another limitation of Bayesian networks is that they are computationally expensive. Because Bayesian networks have to search all possible network structures for a given set of nodes (i.e. genes) to achieve the optimal value of the posterior probability, which is the probability of a structure given the expression data of genes. And the number of network structures grows super-exponentially in the number of nodes. Therefore, heuristic or stochastic algorithms are needed to reduce the computational cost of Bayesian learning. These algorithms include genetic algorithms [62], simulated annealing [63], Markov Chain Monte Carlo (MCMC)[64], etc.

There are still other approaches for GRN inference which do not fall into the above four categories. For example, as reviewed by Schlitt and Brazma in [41], Petri nets are dynamic network models which are often used to simulate metabolic networks. Petri nets can quantitatively characterize the dynamic processes as mass flows in a network. Therefore, they have been successfully applied in modeling metabolic reactions. Petri nets are also suitable for the modeling and analysis of GRNs [65, 66].

A more extensive review of GRN inference methods was done by Marbach *et al.* [67] in 2012. In this paper, they assessed over 30 methods of GRN inference on four microarray datasets, namely *Saccharomyces cerevisiae* (*S. cerevisiae*), *Escherichia coli* (*E. coli*), *Staphylococcus aureus* (*S. aureus*), and *in silico* microarray data. Their results showed that no single method has superior performance on all datasets, and the community network based on the integration of predictions from multiple approaches shows high confidence and robustness across different datasets. This paper classified the methods of network inference into six categories: regression-based, correlation-based, mutual information-based, Bayesian networks, meta methods and other methods. Marbach *et al.* presented detailed comparison and analysis on all these methods. However, they missed out the discussion on two major types of GRN inference methods, i.e. Boolean networks and differential equations based methods.

Here we give a more generalized categorization of GRN inference methods as follows: network-based, statistics-based, equations-based, meta and other methods. Network-based methods include Boolean networks, Bayesian networks and other networks such as

Petri nets [65], neural networks [68] and so on. Statistics-based methods mainly measure the pairwise statistical relationships between two genes, such as correlations and mutual information. Statistical tests (e.g. χ^2 test, t-test and ANOVA) can also be applied to estimate the associations between genes. Equations-based methods include regression models (linear or non-linear) and differential equations. Non-linear regression models may partly overlap with the regression models based on machine learning techniques such as Random Forests and support vector machine. Meta methods refer to such type of methods that combine multiple approaches from different categories or subcategories [69, 70]. For example, a combination of different approaches was proposed to infer a GRN from heterogeneous gene expression datasets [69] (see also the Supplementary Note of [67]). Marbach *et al.* [67] integrated the predictions from an ensemble of GRN inference approaches based on the Borda count election method, where the confidence of an edge was measured by the (weighted or non-weighted) average rank over the predictions (i.e. a list of ranking gene pairs) of those approaches. The proposed method in [67] can be viewed as a meta method of GRN inference, although in a higher level. The category of “other methods” refers to those methods that do not fall into any of the previous four categories. One famous method in this category is the tree-based methods (e.g. GENIE3 [71]), which infers gene interactions based on the regression of target genes using Random Forests. The tree-based methods turn the problem of GRN inference into feature selection problems, where the transcription factors for each target gene are selected based on their importance in predicting the expression level of the target gene. Other feature selection algorithms (e.g. support vector regression) can also be applied to infer GRNs [72]. However, results showed that they are not as successful as Random Forests, probably because some randomization techniques (e.g. data resampling) used in Random Forests are crucial for the performance of GRN inference [67]. We summarize the methods of GRN inference in Table 2.1.

Table 2.1: A summary of methods for GRN inference.

Category	Subcategory (Reference)	Dis. ¹	Eff.	Dir.	Dyn.	Sto.	Remarks
Network-based	Boolean networks ([43, 44, 50])	$\checkmark$ ²	$\checkmark$	$\checkmark$	$\checkmark$	—	not as efficient as statistic methods; mostly deterministic
	Bayesian networks ([58, 61])	$\checkmark$	$\times$	$\checkmark$	—	$\checkmark$	more advantages include modelling randomness and prior integration
	Other networks ([65, 68])						depend on specific methods to construct networks

Statistics-based	Correlations ([73, 74])	×	✓	×	×	×	may include many indirect edges
	Mutual information ([51, 52, 53])	×	✓	×	—	×	infer only pairwise relationships
	Statistic tests ([75, 76])	×	✓	×	×	×	require large sample size
Equations-based	Linear regression ([77, 78])	×	✓	✓	✓	×	oversimplify gene regulation
	Non-linear regression ([79, 80])	×	✓	✓	✓	×	incline to overfitting
	Differential equations ([55, 56, 57])	×	×	✓	✓	—	heavily depend on parameters
Meta	([70, 69, 67])						depend on individual methods and the way to combine them
Other methods	([71, 81, 72, 82])						

¹“**Dis.**, **Eff.**, **Dir.**, **Dyn.**, **Sto.**” denote “Discrete, Efficient, Directed, Dynamic, Stochastic”;

² The notations “✓, ×, —” represent a method is with, without, partly with the particular feature.

2.1.2 Inference of Time-Delayed Gene Regulation

GRNs are often inferred from time-series gene expression datasets using machine learning algorithms. Given the time-series data, the gene regulation could be inferred in two ways: one is assuming instantaneous or first order regulation, and the other is considering higher order regulation. In many cases, a gene regulates the expression of another gene by its products (RNAs or proteins). Since it takes time to generate those products and different processes (e.g. transcription, translation) need different amounts of time, time-delayed regulation is ubiquitous in cellular processes. Thus, inferring time-delayed gene interactions is essential to accurately reconstructing GRNs.

One of the challenges of GRN inference is that time-series gene expression datasets have much less time points compared to the number of genes measured. The problem of inferring higher-order time delays is even more challenging, due to the tremendous search space when the numbers of time lags are unknown. For the r -th order system with totally T time points in the dataset, the available number of time points for inference reduces to $T - r$. This poses a serious computational challenge resulting in more false predictions.

While many methods have been proposed to reconstruct GRNs without delays (see Section 2.1.1) or with first-order delays (e.g. DBN-MCMC [83, 84, 85], dynamic RandomForest [81]), the methods for inferring time-delayed GRNs with high-order delays are

relatively limited. In 2010, a dynamic version of ARACNE (Algorithm for the Reconstruction of Accurate Cellular Networks) was introduced to infer time-delayed dependencies among genes [53]. Their method, called TimeDelay ARACNE (or TD-ARACNE), is able to reconstruct time-delayed dependencies effectively. In 2012, Morshed *et al.* proposed a framework to infer instantaneous and time-delayed genetic interactions at the same time [86]. Their approach was shown to outperform some existing methods such as TD-ARACNE and BANJO. In 2013, Li *et al.* presented a method to infer high-order gene regulation, named MMHODBN (max-min high-order dynamic Bayesian network) [87]. MMHODBN is a hybrid Bayesian network method, which includes two steps: first it learns the skeleton (i.e. an undirected network) of GRN using constraint-based Bayesian learning ([88]); then it performs a search-and-score technique to orient the edges in the skeleton of GRN. It was shown that MMHODBN was able to learn high-order gene interactions effectively and efficiently [87]. Mundra *et al.* proposed a method for inferring time-delayed GRN based on cross-correlation and LASSO [89]. This method has been tested on real-life yeast pathways in G1 phase to show its effectiveness in identifying time-delayed regulation among genes. There are still some other methods for inferring time-delayed GRNs, for example, a linear optimization model [90], decision tree analysis [91], gapped local alignment [92], time-delayed DBN [93], and an adapted PC algorithm to time-series data [94]. Despite all these efforts, the performance of inferring time-delayed genetic regulation is yet to be further improved.

2.1.3 Evaluation of GRN Inference

The performance of GRN inference is often evaluated by comparing the reconstructed GRN with the benchmark network. Let $G = \{g_1, g_2, \dots, g_N\}$ be a set of genes under study, N_G be the reconstructed GRN and N_B be the benchmark network of G . If an edge $g_i \rightarrow g_j$, $i, j \in [1, N]$ exists in N_B and is also inferred in N_G , then we get one “true positive”. Likewise, if an edge exists in N_B but is not found in N_G , then it is a “false negative”; a “false positive” happens when an edge is not in N_B but is found in N_G ; and “true negative” means that an edge is in neither N_B nor N_G . Let TP, TN, FP and FN be the numbers of true positives, true negatives, false positives and false negatives respectively. Then the common measures of performance, including accuracy, sensitivity (also known as recall, true positive rate or TPR), specificity (true negative rate or TNR), precision (i.e. positive predicted value or PPV) and F-measure (also named F-score) are defined as follows:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (2.1)$$

$$Sensitivity = \frac{TP}{TP + FN} \quad (2.2)$$

$$Specificity = \frac{TN}{TN + FP} \quad (2.3)$$

$$Precision = \frac{TP}{TP + FP} \quad (2.4)$$

$$F\text{-measure} = 2 \times \frac{Precision \times Sensitivity}{Precision + Sensitivity}. \quad (2.5)$$

By definition, accuracy is the fraction of correct predictions. In the case of GRN reconstruction, because GRN is inherently sparse, TN and FP tend to be big (generally $TN \gg TP$, $FP > FN$), especially for a large-scale GRN. Thus, an inferred GRN with high TN is not necessarily a good reconstruction of the benchmark GRN, because its TP could be very low. Accuracy could be biased by such reasons and hence may not be a good measure for the performance of GRN inference. Likewise, specificity is not suitable for the evaluation of GRN inference. On the other hand, sensitivity and precision are used more frequently. Because the main goal of GRN reconstruction is to increase TP while controlling FP and FN, sensitivity and precision can serve as good measures for the performance of GRN reconstruction. F-measure considers both sensitivity and precision and thus provides a balanced measure for the evaluation of GRN inference. In this thesis, we mainly use sensitivity, precision and F-measure to evaluate the performance of GRN inference. In addition, a receiver operating characteristic (ROC) curve can be used to show the overall performance of a method when we can adjust the parameters in this method to obtain networks with different degree of connectivity [83]. The ROC curve visualises TPR (i.e. sensitivity) versus FPR (i.e. 1-specificity) at different threshold settings of a parameter. For an ROC curve, we often calculate area under the curve (AUC) to provide a single metric to show the overall performance of a network inference method.

The approaches discussed above are based on the binary (i.e. whether an edge exists or not) classification of edges in a network. These approaches are often sufficient to evaluate the performance of methods for GRN structure inference. Of course, there are other approaches for the evaluation of GRN inference methods. A GRN inference method is often assessed through the validation of the inferred network. The methods for network validation include two types: internal validation and external validation [40]. In internal validation, the data used to infer the network are utilized again by resampling or perturbation techniques to evaluate the robustness or generalization ability of a network inference method. The inferred network can also be validated by comparing the simulated data from this network with the real gene expression data. External validation assesses the inferred network using information from literature or public database, as well as additional experiments.

2.2 Epigenetic Landscape and Gene Regulation During Embryonic Development

Epigenetic factors have been observed to be involved in many cellular processes. However, the whole picture of epigenetic regulation is still far from clear. To obtain a systematic understanding of epigenetic mechanisms, an integrative model of epigenetic regulation is needed.

2.2.1 Modeling Epigenetic Landscape

In 1950s, C.H. Waddington proposed the model of epigenetic landscape [33] to describe the process of cellular differentiation. In this conceptional model, he made a metaphor that the totipotent cell was like a ball on the top of a hill (see Figure 2.1), and cell development could be viewed as the ball rolling down the hill with many ridges and valleys. Although simple and intuitive, Waddington’s epigenetic landscape was found very useful in elucidating the complicated process of cell differentiation. The event that a stem cell transforms into its final differentiated state (e.g. a neuron or a skin cell) is represented by the movement of a ball coming down through many bifurcate paths, where its potential for differentiation gradually diminishes (after making many cell fate decisions). Furthermore, this model inspires many insightful questions, such as where the “gravity” comes from, what determines the shape of the landscape and what may change the landscape.

Figure 2.1: A depiction of Waddington’s epigenetic landscape [33].

Although Waddington’s epigenetic landscape (referred as “epigenetic landscape” hereafter) was proposed over 50 years ago, before many concepts in developmental biology were established, this model has survived and held true ever since. As a matter of fact,

epigenetic landscape has been applied widely to model not only stem cell differentiation [95], but also tissue morphogenesis [34], carcinogenesis [96, 35] and reprogramming [97, 37]. In recent years, many mathematical and computational models have been developed based on the epigenetic landscape to integrate the experimental data and the knowledge concerning those processes.

Current models of epigenetic landscape are mostly based on the theoretical framework of *dynamical systems* [98, 95]. Under this framework, a cell state at time t is represented as a vector of gene expression levels: $\mathbf{x}(t) = [x_1(t), x_2(t), \dots, x_n(t)]$, where $x_i(t)$ is the expression value of the i th gene at time t , n is the number of genes in the GRN within the cell. Hence, each cell state is a point in the n -dimensional *state space*, which is the set of all possible states, corresponding to all possible phenotypes that a cell can achieve. There are some special states called *attractors*, which are both static (i.e. not changing with time) and stable (i.e. robust to perturbations). Caused by the driving force (or “gravity”), other states would normally move towards attractors. The movements of the states form *trajectories*. The trajectories are determined by the architecture of a GRN, including its structure and interaction functions. Kauffman proposed that cell types can be modelled as attractors over 40 years ago, and he pioneered the use of Boolean networks [42]. Among numerous applications, Boolean networks have been applied to studies of cancer cell death [99, 100] and stem cells [101, 102]. In addition to the discrete model of Boolean network, ordinary differential equations (ODEs) based approaches that include kinetic models of molecular interactions are popular for cell fate modeling [103, 37]. The continuous models can represent the physical processes of gene regulation precisely and quantitatively, and they are amenable to theoretical analysis of the underlying systems (e.g. steady-state analysis). However, differential equation models need accurate estimation of parameter values, which remains a challenge in many cases [57].

Based on the framework of dynamical systems, researchers have proposed many mathematical and computational models of epigenetic landscape, which could be classified into two categories: differential equation (DE)-based and Boolean network (BN)-based models [104]. In a DE-based model, the landscape is constructed either by a potential or quasi-potential function [105, 106], or by calculating the probabilities of states in a given interval of space and time [37, 107]. In a BN-based model, a discrete version of landscape is constituted of the state space and the states transitions generated by Boolean rules. Each of the BN-based and DE-based models can further be categorized into two types: deterministic models and stochastic models. In deterministic BN-based models, the states of genes are determined and updated by some predefined rules. Stochasticity could be introduced to BN-based models in several ways. One is the stochasticity in nodes (SIN) model [108]. In SIN model, a probability ξ is introduced to define the probability of each gene not following its Boolean function for state update at each time step. Another approach to stochastic BN modeling is through the threshold network

formalism [109, 110], where the update of gene state is not by explicit Boolean functions but by the sum-up values of all positive and negative inputs (i.e. regulators). For the continuous DE-based models, stochasticity is included in differential equations of states variation along time. For example, the Langevin equation as a stochastic DE-based model is obtained by introducing the driving stochastic forces [37]; the Fokker-Plank equation, which describes the distribution of the cell state probability evolving over time, integrates a drift vector and a diffusion matrix to model the stochastic temporal evolution of cell state [111]. The approaches of modeling epigenetic landscape are summarized in Table 2.2.

Table 2.2: Methods of modeling epigenetic landscape

Category	Subcategory	References	Advantages	Limitations
BN [†] -based	Deterministic models	(very few reference, just for comparison)	simple and efficient	not accurate; cannot model transitions between different attractors, nor the dynamic transitions
	Stochastic models	[109, 110, 112]	efficient; effective in capturing some essential events in cell development etc	not accurate or stochastic enough
DE [‡] -based	Deterministic models	[106, 105]	fairly accurate; suited for analyzing small networks	may not consistent with real physical processes
	Stochastic models	[37, 111, 95]	accurately and quantitatively capture the physical processes of cell differentiation etc	depend on accurate estimation of parameters; complicated approaches; suited for analysis but not for <i>de novo</i> inference

† Boolean Network; ‡ Differential Equation

2.2.2 GRN Inference from Single-cell Transcriptional Data During Embryonic Development

Embryonic development is a major application field of epigenetic landscape models. Based on the inherent relationship between epigenetic landscape and genetic networks [33, 36], epigenetic landscape model can be applied to uncover the regulatory interactions among genes during development as well as other cellular processes.

Understanding gene regulatory networks (GRNs) that control cell fate decisions is of paramount importance in biology. It can provide critical insights into many questions in developmental biology [113], stem cell [114], cancer [115, 116], synthetic biology ([117]), etc. Mouse preimplantation development is a prominent example for the study of cell fate decisions [118, 119]. Starting from the 1-cell fertilized zygote it proceeds to the 64-cell blastocyst, consisting of three distinct cell types, i.e. trophectoderm (TE), primitive endoderm (PE) and epiblast (EPI) [120]. For this classic developmental process, many results and insights have been obtained through decades of research. However, the regulatory mechanism for differential gene expression that leads to the formation of the three cell types remains to be elucidated. One hurdle lies in that many analyses were conducted on the average expression of a population of cells in mouse as well as other species [121, 122], ignoring the heterogeneity among individual cells, which however is critical for the cell lineage formation. Recently, the technology of single-cell RNA profiling promises to overcome this hurdle and provides new insights [123].

In recent years, increasing datasets of single-cell RNA profiling have been generated and analyzed for the study of early development of mouse, human and so on [118, 124, 125, 126, 127]. Intriguing patterns that shed light on the dynamics of cell fate decisions have been identified using statistical analysis techniques on the datasets, which demonstrated the power of single-cell expression analysis. However, these papers mainly utilize data-driven approaches, which extract information entirely from data without guidelines from existing knowledge. The data-driven approaches in these papers were limited to descriptive statistical relations of gene expression, interpretation of which requires highly specialized knowledge and experiences of researchers. To gain a systematic and holistic understanding of the causal mechanisms, an integrative network model which synthesizes data-driven and knowledge-driven approaches and is “executable” through computer simulations would be highly desirable. In conjunction with experimental works, mathematical models have been proposed for the study of cell fate determination. Many models are based on the theory of dynamical systems [39] (Related works have been reviewed in Section 2.2.1). Machine learning models have also been developed to elucidate gene regulation in cell development. For example, in [128] and [129], cell lineage trees were incorporated into machine learning algorithms to infer gene networks and expression programs, respectively.

Although many methods for GRN inference have been proposed (see Section 2.1.1), they were mostly designed for the average expression in a population of cells. These methods may not be suitable for GRN inference from single-cell gene expression data, because they were not designed to capture the heterogeneity among individual cells in single-cell data. As single-cell data become prevalent in the research of embryonic development, new approaches are needed for the inference of gene regulation from single-cell data during cell development. So far, only a few methods have been proposed for GRN

inference from single-cell transcriptional data. In a recent paper [130], Moignard *et al.* employed a Boolean network based approach to reconstruct the regulatory network from single-cell gene expression data of early blood development. They first constructed a state transition graph from the single-cell data, and then searched the network modeled by asynchronous Boolean network that best fitted the state transition graph. Boolean network was also applied by Xu *et al.* to infer GRNs from single-cell gene expression data of mouse embryonic stem cells in [102], where the network structure was obtained by other experimental data (e.g. ChIP-seq, knockdown), and Boolean network was employed to learn the regulatory logic functions among genes. A framework was designed in [131] to reconstruct gene regulatory dynamics and network structures from single-cell gene expression data. Firstly, a dimensionality reduction technique and a time-ordering method were applied to generate a dataset with pseudo time series. Then a model selection approach based on Bayesian framework was used to select the ODE model that best fitted the observations in the dataset. In addition, several methods have been proposed to infer signaling pathways from single-cell protein phosphorylation data. A Bayesian network based approach was proposed in [132], where each cell in the flow cytometry data was treated as an independent observation (i.e. sample) in a particular condition. Therefore, the single-cell dataset with hundreds of samples provided a robust source of statistical dependencies which enable accurate inference of causal relationships by Bayesian network. A similar application of Bayesian network can be found in [133]. The above Bayesian network based approaches may be limited in analyzing the single-cell data if the cells are not totally independent. For example, some cells are supposed to be the descendants of a particular set of cells during embryonic development.

2.3 Global Epigenetic Regulation of Gene Expression

In recent years, epigenetics has been under intense research. Epigenetic modifications such as histone modifications are found to be involved in many cellular processes. The profiles of histone modifications have been used to predict many cellular activities or genomic regions, such as transcription levels [134], transcription factor binding sites [135], mRNA stability [136], meiotic recombination hotspots [137], boundary elements [138], promoters and gene structure [139], and enhancers [140]. The prediction of gene expression based on histone modifications has drawn particular attention. Various methods have been proposed to discover the regulatory relations between epigenetic modifications and gene expression, i.e. to mine the “histone code” in transcription. We summarize these methods in Table 2.3.

Table 2.3: Methods of predicting gene expression with histone modifications

Authors (year)	N.*	M.L.†	R.‡	Method	Remarks	Ref.
Yu <i>et al.</i> (2008)	✓			Bayesian network		[30]
Su <i>et al.</i> (2012)	✓			Correlation network		[141]
Xu <i>et al.</i> (2010)			✓	Multiple regression models		[142]
Cheng <i>et al.</i> (2011)		✓		Clustering, SVM, and linear models		[143]
Cheng <i>et al.</i> (2012)		✓		SVM		[144]
Teng <i>et al.</i> (2012)		✓		Semi-supervised bi-clustering		[145]
Ha <i>et al.</i> (2011)		✓	✓	Cluster and regression analysis		[146]
Karlic <i>et al.</i> (2010)			✓	Linear Regression	First paper which found HMs were predictive for GE	[134]
Wilczynski <i>et al.</i> (2012)	✓			Bayesian network		[147]
Markowetz <i>et al.</i> (2010)		✓	✓	SVM, Random Forest, etc.		[148]
do Rego <i>et al.</i> (2012)			✓	Mixture of regression models		[149]
Mcleay <i>et al.</i> (2012)			✓	PCA + regression model		[150]
Costa <i>et al.</i> (2011)			✓	Mixture of regression models		[151]
Dong <i>et al.</i> (2012)		✓	✓	Random Forest and regression		[152]
Wang <i>et al.</i> (2012)			✓		mRNA stability	[136]
Heintzman <i>et al.</i> (2009)		✓		Entropy based clustering		[140]
Jung <i>et al.</i> (2012)			✓		Function-specific HMs	[153]
Bianco-Miotto <i>et al.</i> (2010)			✓	Regression and statistical analysis		[154]

* Network-based; † Machine-learning; ‡ Regression model

Currently the methods to predict gene expression based on histone modifications can be categorized into three types: network-based methods, machine-learning methods, and regression models. Although regression models can be viewed as a subset of machine learning methods, we treat this method as a separate type here because of its prevalent use in predicting gene expression based on histone modifications. Also, these regression models include linear, non-linear regression model, or mixture of them, aside from those

regression models based on well-known machine-learning methods such as support vector machine (SVM) and Random Forest (RF).

Networks have been built to analyze the regulatory relations between gene expression and histone modifications. In 2008, Yu *et al.* [30] constructed a Bayesian network based on the data of histone modifications and protein binding (such as RNA polymerase II (RNAP II) binding) and gene expression data, to infer causal and combinatorial relationships between gene expression and histone modifications. Gene expression was treated as a node in the Bayesian network. This network was able to describe the causal relations among histone modifications and more importantly, the causal relations between combinatorial histone modifications and gene expression. Existing experimental observations were provided to validate the inferred relationships. However, they did not show to what extent the network can represent the data (or how well gene expression levels can be predicted by histone modifications using this network). Su *et al.* [141] used partial correlation and Pearson correlation to construct an epigenetic interaction network of histone modifications, DNA methylation and gene expression. A combination strategy was introduced to analyze the synergetic modules in the network, in order to show how those factors regulate gene expression cooperatively. They observed some interesting phenomena such as DNA methylation regulates gene expression via H3K4me3 and RNAP II (with experimental support), although their network cannot represent causal relationships, as done by Bayesian network. In [147], Bayesian network was employed to integrate several types of genomics data, including histone modifications, transcription factor (TF) binding and RNAP II binding, to predict temporal and spatial gene expression in metazoan development. This paper suggests that integrative models such as Bayesian networks are particularly suitable for predicting global gene expression patterns during development, because they can accurately represent the relations between TF occupancy and cis-regulatory modules (CRMs) in specific tissues and stages.

Machine-learning techniques have also been adopted to study the relationship between histone modifications and gene expression. These machine-learning methods include unsupervised techniques such as clustering, and supervised models such as SVM and Random Forest. Cheng *et al.* [143] applied both unsupervised clustering and supervised models such as SVM to study the relationship between gene expression and chromatin features. Both types of methods revealed the capability of histone features in predicting gene expression. SVM was used by Cheng *et al.* [144] to also predict gene expression levels based on TF binding and histone modifications in murine embryonic stem cells. They found that their models for TF binding and HMs are cell line specific whereas models on HMs are not, and the models trained only on protein-coding genes can also predict microRNAs expression levels. In [148], many classification methods such as SVM and Random Forest were employed to predict gene expression changes based on the variations of histone acetylation in mouse embryonic stem cells. The application of

RF on gene expression prediction can also be found in [152]. Unsupervised clustering methods find their frequent use in this area too. Teng *et al.* [145] proposed a semi-supervised biclustering method to find the combinatorial chromatin modifications which are correlated with gene expression in human enhancers. Ha *et al.* [146] used both clustering analysis and multiple linear regression to observe that coordinated histone modifications are associated with gene expression within and between species.

Regression models have been used frequently to predict gene expression with histone modification levels. The quantitative models derived by Karlic *et al.* [134] were linear regression models. In their experiments, Karlic *et al.* observed that gene expression can be accurately predicted by only a small number of histone modifications. In addition, they found that the relationships between gene expression and histone modifications are conserved among different cell types. Xu *et al.* [142] applied both linear and non-linear regression models to estimate gene expression using histone modifications. They concluded that nonlinear models had better prediction than linear models. On the other hand, linear regression model has been improved in several studies to enhance prediction accuracy. For instances, McLeay *et al.* [150] applied PCA to the data matrix which contained HMs levels before performing the regression to predict gene expression. Costa *et al.* [151] and Do-Rego *et al.* [149] used a mixture of linear regression models to achieve better accuracy of prediction. They utilized several linear regression models to estimate the expression level for each gene, and assumed the errors followed a Normal distribution. Then, the posterior probability of a given gene would be used to select the linear model with the best performance for predicting its expression.

All the studies discussed above show that existing methods have achieved good performance in predicting gene expression based on epigenetic profiles. However, most of these methods revealed the relationships between histone modifications and gene expression using “black box” models, which makes it difficult for human to understand and interpret the underlying physical processes and biological meanings. Hence, methods with interpretability would be highly desirable. For example, some rules, which interpret the relationship between histone modifications and gene expression in a straightforward way, would be much easier to understand.

2.4 Integration of Epigenetic Data in GRN Inference

Gene regulation is a complex process, with many elements involved. To improve the performance of inferring gene interactions, prior knowledge from various sources has been incorporated into the process of GRN inference. One type of prior knowledge is protein-DNA interactions (or transcription factor binding). Studies have shown that the performance of GRN inference could be improved by incorporating known protein-DNA

interactions, extracted from ChIP-chip or ChIP-seq experiments [155] or public databases [156, 157, 158]. Networks such as protein-protein interactions (PPI) networks [159, 160], biological pathways [161, 162, 163] and functional association networks [164] have also been used as prior to help the reconstruction of GRN from gene expression data. Prior knowledge for GRN inference can also be extracted from the literature (e.g. abstracts from PubMed) [156, 159, 165, 166], gene ontology [163] and so on.

Bayesian networks have the intrinsic superiority of integrating data from different sources. Heterogeneous data can be integrated into the prior probability of Bayesian network. The prior probability can be modelled by Gibbs distribution [167, 162] or by the structure prior [156]. Besides Bayesian model, prior can also be integrated in the sparsity term of the penalty function in ordinary differential equations (ODE) [155, 158], or in the objective function of convex optimization [165, 164].

Being closely related with gene expression, epigenetics also provides a good source of prior knowledge. Epigenetic data such as histone modifications have been integrated into the process of GRN inference [168, 85]. With recent advancement in sequencing technologies, more and more epigenetic data are gathered which could help GRN inference and provide new insights. The first study on the integration of epigenetic data into GRN reconstruction is from Zheng *et al.* [168]. In their work, they first observed that the correlations of histone modification profiles of the confirmed gene-pairs in benchmark GRN are significantly higher than those of random gene-pairs. Therefore, they used the correlations of histone modification profiles as prior for GRN inference based on Bayesian networks. Results from [168] show that, the integration of epigenetic data (i.e. histone modification profiles) using Gibbs prior in Bayesian network can significantly increase the accuracy of reconstructing a GRN in yeast cell cycle.

However, the experiments in [168] were limited to static Bayesian networks which cannot detect the feedback regulations between genes. How the epigenetic priors work on dynamic Bayesian networks is still unknown. Moreover, they took the average histone modification profiles across all loci for each gene, which may smooth out some important information behind different locations. The paper [85] extended the work of [168] and integrated the epigenetic prior into the dynamic Bayesian network. The authors of [85] observed different characteristics of histone profiles in different loci of genes, based on which they designed more appropriate “prior” for GRN inference using DBN.

Chapter 3

Reconstructing Regulatory Networks from Single-Cell Gene Expression Data During Embryonic Development

3.1 Introduction

3.1.1 Background

Embryonic development is a process in which a zygote (i.e. a fertilized egg) gradually evolves into a set of somatic cells which form a whole organism, through cell divisions and cellular differentiation. Early development, or preimplantation development, is referred to the period of a zygote (1-cell stage) transforming into a 64-cell blastocyst, which consists of three distinct cell lineages, i.e. trophoctoderm (TE), primitive endoderm (PE) and epiblast (EPI) [120]. During the process of development, cells convert to different cell types (or cell fates) driven by the combined effect of cellular regulatory networks, external signals, cell-cell communications and so on.

Understanding gene regulatory networks (GRNs) that control cell fate decisions can provide critical insights into many questions in developmental biology [113], stem cell [114], cancer [115], etc. However, transcriptional regulatory networks controlling cell fate decisions in mammalian embryonic development remain elusive despite many efforts over decades. The average expression of cell populations used in previous analysis cannot capture the heterogeneity among individual cells, which however is critical for the cell

lineage formation. The recent technology of single-cell RNA profiling brings promise for new discoveries in embryonic development [123]. Single-cell technology separates a population of cells (by flow cytometry, etc) and measures gene expression levels of individual cells from one to another, providing a high resolution of expression measures. Single-cell data analysis is especially important for the fields where the heterogeneity of cells plays critical functions, e.g. stem cell and cancer research.

3.1.2 Motivation

Although experimental works have obtained intriguing insights into the mouse early development, a holistic and systematic view is still missing. Mathematical models of cell fates tend to be concept-based, not designed to learn from real data. To elucidate the regulatory mechanisms behind cell fate decisions, it is highly desirable to synthesize the data-driven and knowledge-driven modeling approaches. Reconstructing GRNs is of vital importance for elucidating the regulatory mechanisms of biological processes. Existing methods of GRN inference (see [40, 41], and references therein) were mostly designed for the average expression in a population of cells, and rely on local dependence among variables (e.g. genes). However, single-cell expression data pose new challenges for GRN inference. The heterogeneity among cells could make the average of expression levels misleading. Moreover, many existing methods of GRN inference require long and dense times series data of RNA profiling, which are often unavailable, especially when the experimental cost is high. Therefore, alternative strategies of GRN inference are needed for single-cell transcriptional data with short and sparse time series.

In this chapter, we propose a novel method that integrates the structure of a cell lineage tree with transcriptional patterns from single-cell data. This method adopts probabilistic Boolean network for network modeling, and genetic algorithm as search strategy. Guided by the “directionality” of cell development along branches of the cell lineage tree, our method is able to accurately infer the regulatory circuits from single-cell gene expression data, in a holistic way. Applied on the single-cell transcriptional data of mouse preimplantation development, our algorithm outperforms conventional methods of network inference. Given the network topology, our method can also identify the operational interactions in the network, corresponding to specific cell fate determination. This is one of the first attempts to infer gene regulatory networks from single-cell transcriptional data, incorporating dynamics of cell development along a cell lineage tree.

3.2 Materials

3.2.1 Single-cell Data of Mouse Pre-implantation Development

The single-cell gene expression data used in this study are from Robson lab [118]. The dataset (which will be referred as “Robson’s dataset” hereafter) includes RNA expression

profiles of 48 genes in 442 single cells from 1-cell (1C) zygote to 64-cell (64C) blastocyst. In order to capture cell identities based on gene expression, it is essential to measure multiple genes in parallel. Therefore, Guo *et al.* [118] selected 48 genes and quantitatively profiled their expression levels in parallel at single-cell level, using the 48.48 Dynamic Array chips (Fluidigm). The 48 genes were selected because they are transcription factors (TFs) functional in the formation of cell types, or they are differentially expressed in the blastocyst, and/or the genes are known to have functions in early development.

The single-cell data of gene expression were generated by high-throughput single-cell qPCR. For 48 selected genes, single-cell TaqMan real-time PCR assays were applied on more than 500 cells harvested in the first 4 days of mouse development, from 1C zygote to 64C blastocyst. After the manual separation of blastomeres and the lysis, Guo *et al.* proceeded with cDNA synthesis, and sequence specific preamplification in a single tube, and then measured the gene expression using TaqMan qPCR on the BioMark system (Fluidigm). The Ct values in the published datasets by Guo *et al.* are already in logarithmic scale. To transform the Ct values to relative gene expression levels, we only need to subtract the values from the baseline value of 28. To normalize the gene expression levels to the endogenous control, the resulting values would be subtracted by the average expression levels of *Actb* and *Gapdh* in each cell. For more details about data preparation and pre-processing, please refer to [118].

Two cell fate decisions are made during this period (see Figure 3.1(a)). The first decision is made when cells transform from 16-cell stage to 32-cell stage, where two distinct cell lineages take shape: trophectoderm (TE) and inner cell mass (ICM). The second decision is made during the transitions from the 32-cell stage to 64-cell stage, and two cell types are generated from ICM: primitive endoderm (PE) and epiblast (EPI). To focus on the cell fate regulation during these two processes of cell lineage formation, we extracted data from the three stages, i.e. 16-, 32- and 64-cell stages. In Robson's dataset, the numbers of cells in 16-, 32- and 64-cell stages are 75, 113 and 159 respectively. By applying PCA as in the original paper (i.e. [118]), we identified the two cell types (i.e. TE and ICM) from the cells in the 32-cell stage (i.e. 32C), and the three cell types (i.e. TE, PE and EPI) from the cells in the 64-cell stage (i.e. 64C). The numbers of TE and ICM cells in 32C, and PE and EPI cells in 64C are 60, 50, 38 and 17 respectively.

As Boolean network was used to model the GRN, we first discretized the continuous gene expression levels into two states: 0 (lowly expressed) and 1 (highly expressed), for the data of three stages (i.e. the 16-, 32- and 64-cell stages) separately. Let $X = \{x_{ij}\}_{n \times m}$ be the gene expression dataset with n genes and m samples. For each gene, by assuming Gaussian distribution, we calculated the mean μ and standard deviation σ , and set the discrete value d_{ij} of the i th gene in the j th sample as

Figure 3.1: The structure of cell lineage tree (a) and the gene regulatory network (b) for the formation of three distinct cell lineages: TE, PE and EPI (adapted from [119]).

$$d_{ij} = \begin{cases} 1, & \text{if } x_{ij} \geq \mu_i + w \times \sigma_i \\ 0, & \text{otherwise} \end{cases} \quad (3.1)$$

where $\mu_i = \frac{\sum_{j=1}^m x_{ij}}{m}$, $\sigma_i = \sqrt{\frac{\sum_{j=1}^m (x_{ij} - \mu_i)^2}{m-1}}$, $i = 1, \dots, n$, and w is the weight which was set to 0.5 [169, 170] in our experiments.

To infer a network from a dataset of single-cell gene expression, first we need to decide what genes are involved in the network. These genes are often selected from the core regulators and/or lineage specifiers in a particular process. Here, the input data to our method (see Algorithm 3.1) is a Boolean dataset with rows and columns representing cells and (selected) genes respectively. The Boolean dataset is obtained by applying Eq. (3.1) to the continuous single-cell transcriptional data. Each cell in the Boolean dataset is represented by a row vector of states of genes. The cell state vectors from real data will be compared with the simulated cell states by our model (see Section 3.3.2 for details).

3.2.2 Benchmark Network

The benchmark network used in this study is shown in Figure 3.1(b), which accounts for two cell fate decisions from the 16-cell stage to 64-cell stage. The GRN in Figure 3.1(b) is composed of two smaller networks (shown in Figure 3.2), which are benchmark networks for $16C \rightarrow TE + ICM$ and $ICM \rightarrow PE + EPI$. Here we focus on the ten genes (see Figure 3.1(b)), because they are the core regulators and lineage specifiers in those particular cell

types [118]. Moreover, the interactions among the ten genes are known to play a critical role in the lineage specifications from 16C to 64C stages [119]. We believe that the rest genes are relatively peripheral compared with the ten core genes.

All the interactions of genes in the benchmark network are supported by wet-lab experiments (see [119] and references therein). Furthermore, experimental support of the edges in this network are also well documented in other papers. For example, the mutual antagonism between *Cdx2* and *Oct4*, which determines the trophectoderm (i.e. TE) differentiation, has been described with details in [171]. Another mutual antagonism between *Nanog* and *Gata6*, which is the key to the bifurcation of primitive endoderm (PE) and epiblast (EPI), has also been supported by experimental literature [172, 173, 174].

3.3 Methods

3.3.1 Approach Overview

We propose a novel approach to infer GRNs from single-cell expression data. It takes into account the relative positions of cells on the cell lineage tree. Our assumption is that GRNs (together with signaling pathways) provide the driving force that pushes cells from ancestral states towards descendant states [21]. Such “directionality” of cell development is imposed by the logical constraints of the underlying GRN. For more background information about our approach, please refer to Section 2.2.1 and Section 2.2.2. From Robson’s dataset of mouse early development, we can observe which cell states belong to a certain node in the cell lineage tree. Given a candidate GRN, we can simulate the dynamic trajectories of cells following the regulatory rules encoded in the GRN. If the simulated trajectories are compatible with the directed paths from the root towards leaves of the cell lineage tree, the GRN is likely to be true. Conversely, if a candidate GRN drives cells from descendant states backward to ancestral states, it is likely to be false. To implement this strategy, we choose the framework of Boolean network for its strength in state transition analysis, and genetic algorithms (GA) for optimization (see Section 3.3.2 for details). A similar idea for GRN inference based on Boolean network was proposed in [175]. However, Pal *et al.* assumed that the expression data should be sampled from attractor states, which may be too restrictive for single-cell data. Relying on the concordance of attractor states of the model with states of the data, they did not consider fully the “directionality” of cell development along branches of a cell lineage tree. Therefore, although the method of Pal *et al.* is very insightful, it is not directly applicable to the single-cell expression data of mouse embryonic development.

Running on the Robson’s dataset, our method for GRN structure inference achieved better performance than most of the other methods (see Section 3.4.1), validated using a published benchmark GRN constructed manually from experiments [119]. Moreover,

given the structure of GRN, our method is able to predict the operational regulatory interactions responsible for a specific cell lineage formation. Most of the predictions are consistent with literature of experimental observations (see Section 3.4.3). To the best of our knowledge, the method proposed here is one of the first attempts to infer GRN from single-cell expression data. Unlike existing methods, it does not rely on local dependence of variables, which is difficult to detect from short and sparse time series data. Instead, it is a holistic approach that utilizes global information about a system (e.g. network stability). As more single-cell data become available in near future, our method is promising to help decipher the mechanisms of cell fate decisions.

3.3.2 Structure Inference of GRN

In this section, our goal is to reconstruct the topological structures of GRNs underlying cell fate decisions. The hierarchical structure of the cell lineage tree is used to guide the GRN inference, based on the premise that a true GRN should have simulated trajectories consistent with the temporal order of the developmental process.

To simulate the dynamic trajectories of cell state transitions, we used simple and general rules to update the states of genes [109], based on the assumption that the behaviors of a cellular system depend more on the topology of the regulatory network and less on the parameters of the interactions [176, 116]. Let $g_i(t+1)$ be the state of the i th gene at time $t+1$, $N_i^I(t)$ be the number of inhibitory edges incident to the i th gene at time t , and $N_i^A(t)$ be the number of activating edges at time t . Note that the “time” here means an updating step in a Boolean model. The updating rule for the i th gene is defined as follows:

$$g_i(t+1) = \begin{cases} 0, & \text{if } N_i^I(t) > N_i^A(t) \\ 0.5, & \text{if } N_i^I(t) = N_i^A(t) \\ 1, & \text{if } N_i^I(t) < N_i^A(t) \end{cases} \quad (3.2)$$

Since the state of a gene in a Boolean network can only be 0 or 1, when $N_i^I(t) = N_i^A(t)$, $g_i(t+1)$ will be randomly assigned to 0 or 1 with equal probability.

For a proposed network structure, we generate its state transition graph (STG) by asynchronously updating the states of genes using the rule described in Eq. (3.2). For a particular cell state (i.e. a vector of states of genes), in each step, we randomly pick one gene with equal probability and update its state based on its inputs (i.e. regulators) and the rule in Eq. (3.2), thereby making a transition from this cell state to another. After u updates, we calculate the transition probabilities from this cell state to others. In this way, we obtain an STG (represented by its adjacency matrix A), of which each entry contains the transition probability between cell states. For the i th cell state, we have

$$\sum_{j=1}^{2^n} A_{ij} = 1, \quad (3.3)$$

where n is the number of genes and $i = 1, \dots, 2^n$. Here A is a Markov matrix, since the transition probabilities do not change over time. Therefore, by calculating A^l , we can obtain the end states after l time points from any starting state.

After the STG is generated from a proposed network, we extract only the transitions of the cell states which appear in the real data. Note that we can observe the ancestor-descendant relationships of cell states from real data. Then we examine the directions of the transitions of the cell states in the STG, and compare them with the relationships of cell states in real data. Following the “directionality” of cell development, during ICM $\rightarrow$ PE+EPI (similarly for 16C $\rightarrow$ TE+ICM), we expect the states in PE and EPI to be stabler than the ICM states. As such, the “forward” transitions from ICM to PE or EPI should have higher probabilities than the “backward” transitions. Therefore, we give bonus to the transitions from ICM to PE or EPI (we also give bonus to self-transitions within each cell type), while impose a penalty on any transition from PE or EPI back to ICM. In addition, we also put penalties on trans-transitions across different cell types at the same stage, e.g. PE $\rightarrow$ EPI or EPI $\rightarrow$ PE. Accordingly, the fitness score for structure inference is defined as

$$\begin{aligned} \text{Score} = & \sum_{k=1}^2 \sum_{j \in W} \sum_{i \in M_k} (p_{ij} \times \text{Bonus}(\text{UpLevel} \rightarrow \text{LowLevel})) - \\ & \sum_{j \in W} \sum_{k=1}^2 \sum_{i \in M_k} (p_{ij} \times \text{Penalty}(\text{LowLevel} \rightarrow \text{UpLevel})) - \\ & \sum_{k=1}^2 \sum_{j \in M_k} \sum_{i \in M_{k'}} (p_{ij} \times \text{Penalty}(\text{trans-transitions})), \end{aligned} \quad (3.4)$$

where W is the set of indices of cell states belonging to the upper-level cell type; M_k is the set of indices of states in the k th lower-level cell type (here we assume there are two cell types at the lower level, i.e. $k = 1$ or 2 , although our method can be easily generalized to more than two descendant cell types); p_{ij} is the transition probability from i th state to j th state (i.e. p_{ij} is an element of the matrix of STG); $M_{k'}$ is the alternative representation of M_k , which means when $k = 1$, $k' = 2$, and when $k = 2$, $k' = 1$.

To learn the optimal GRN structure that best fits the data, a genetic algorithm (GA) is designed to optimize the scores of candidate topologies. GA is often used to solve optimization and search problems, by simulating the process of natural selection, crossover

and mutation to reach an advanced species [177, 178]. The pseudocode of GRN structure inference based on GA is shown in Algorithm 3.1. Firstly, we create a population with U candidate individuals (i.e. the initial population). Each GA individual, which corresponds to a network, is a binary string obtained by concatenating the columns of the adjacency matrix of the network. The maximum number of regulators for each gene is K , as a constraint for each individual. Then we calculate the fitness scores based on Eq. (3.4), which are used to select the top U' individuals as candidates for the next GA iteration. In “crossover”, two parents are randomly chosen by Roulette wheel selection based on their scores. A typical “recombination” procedure is performed to generate their children. In “mutation”, dependent on the mutation rate, we randomly choose chromosomes (i.e. individuals) and randomly decide the mutation sites to be flipped. As such, a new population is generated and the algorithm enters the next iteration till the maximum number of iterations is reached. Since the time complexity of Algorithm 3.1 is exponential in the number of genes n , our method is currently more applicable to small networks (e.g. $n < 10$).

3.3.3 Identifying Operational Interactions in GRN

3.3.3.1 Network Encoding

Here, we will demonstrate that the same framework of inference as in the previous section can also be used to identify the operational interactions responsible for a specific cell fate decision, from a given network topology. The GRN is again encoded as an asynchronous Probabilistic Boolean network $\mathbb{N}$. The states of genes are updated based on their inputs (regulators) and Boolean functions (or rules). Since the incoming edges for each gene are known for the given network topology, we only need to determine the Boolean rules for generating the dynamic behaviors of $\mathbb{N}$. Here the rules were derived based on the knowledge from literature (see Section 3.4.3 for details).

For a particular network $\mathbb{N}$, the collection of regulatory rules R consists of a set of Boolean functions F and their corresponding probabilities in set P . Thus we have $R = \{F, P\}$, where $F = \{F^1, F^2, \dots, F^n\}$, $P = \{P^1, P^2, \dots, P^n\}$, and n is the number of genes. For the i th gene, $F^i = \{f_1^i, f_2^i, \dots, f_{L_i}^i\}$ and $P^i = \{p_1^i, p_2^i, \dots, p_{L_i}^i\}$, where $i = 1, 2, \dots, n$. Here f_j^i is the j th regulatory rule of the i th gene with probability p_j^i , and L_i is the total number of rules for the i th gene. The probabilities of rules should satisfy

$$\sum_{i=1}^n \sum_{j=1}^{L_i} p_j^i = 1. \quad (3.5)$$

The dynamic behavior of a GRN is not only determined by the Boolean functions in F , but also influenced by the probability parameters in P . Since a rule with a higher probability is more likely to be chosen to update the states of genes, the probabilities of rules affect the dynamics of the network.

Algorithm 3.1. GA-based GRN inference from single-cell transcription data

INPUT: Single-cell gene expression data (a Boolean dataset obtained by Eq. (3.1)), population size U , number of selected individuals U' , mutation rate μ , permutations per chromosome η , and maximum iteration times $MaxIt$

OUTPUT: A vector P , which consists of regulators of each gene in the network (i.e. P is the representation of a network topology)

```

 $P_0(U) \leftarrow$  initial population
for each  $i \in P_0$  do
    Calculate  $score(i)$  using Eq. (3.4)
end for
Select top  $U'$  individuals based on their scores and insert them into a set  $Q$ 
for  $k = 1$  to  $MaxIt$  do
     $P_k \leftarrow$  a null set
    // Crossover
    while ( $size(P_k) < U$ )
        Choose two parents  $P_{k-1}^i$  and  $P_{k-1}^j$  from  $Q$  by Roulette wheel selection
        Generate two children by  $Recombination(P_{k-1}^i, P_{k-1}^j)$ 
        Check and modify children according to the constraint for individual
        Add children to  $P_k$ 
    end while
    // Mutation
    Select  $U \times \mu$  members from  $P_k$ 
    Invert  $\eta$  randomly selected site for each chromosome
    Check and modify children according to the constraint for individual
    Add new individuals to  $P_k$ 
    // Scoring
    for each  $i \in P_k$  do
        Calculate  $score(i)$  using Eq. (3.4)
    end for
     $Q \leftarrow$  a null set
    // Selection
    Select  $U'$  top score individuals from  $P_k$  and insert them into  $Q$ 
    Save a copy of the  $U'$  top score individuals to  $\mathbb{P}$ 
end for
Determine  $P$  by taking the “consensus” of high-score networks in  $\mathbb{P}$ 
Return  $P$ 
    
```

3.3.3.2 Probability Inference

We employ GA again to infer the probability vector P . Without loss of generality, we assume that cell fate decision is a bifurcation in the cell lineage tree: $Up \rightarrow Low_1 + Low_2$ (e.g. ICM $\rightarrow$ PE + EPI). Since we aim to identify the operational components for a specific lineage decision (e.g. $Up \rightarrow Low_1$), the GA should maximize the transition probabilities for this lineage, and minimize the transition probabilities in the alternative direction (i.e. $Up \rightarrow Low_2$). Hence, the fitness score is defined as

$$\begin{aligned} \text{Score} = & \sum_{j \in M_1} \sum_{i \in W} (p_{ij} \times \text{Bonus}(Up \rightarrow Low_1)) - \\ & \sum_{j \in M_2} \sum_{i \in W} (p_{ij} \times \text{Penalty}(Up \rightarrow Low_2)), \end{aligned} \quad (3.6)$$

where M_1 and M_2 are the sets of indices of cell states belonging to the two lower cell types; W is the set of indices of cell states of the upper-level cell type; p_{ij} is the transition probability from the i th state to the j th state.

To calculate the transition probabilities among cell states, we employ an asynchronous PBN to model GRN and obtain the STG by asynchronously updating the activities of nodes. In each step, we randomly select one rule f ($f \in F$) based on its probability p ($p \in P$) and update the corresponding target gene. After the STG is obtained, we map the cell states from real data to the STG. Then a fitness score will be calculated for a particular setting of P using Eq. (3.6). Through GA, the final probability vector P would show which rules are more frequently used for the cells to commit to a specific lineage.

The process of probability inference is very similar to Algorithm 3.1, except for the codification and constraint of GA individuals, and the calculation of fitness scores. Here the GA individual is a vector of the probabilities of all rules for all genes, and each individual is constrained by Eq. (3.5).

3.4 Experimental Results

3.4.1 Results of Structure Inference

To validate our algorithm (named SingCellNet hereafter) for GRN structure inference, we applied SingCellNet to real single-cell expression data and compared the inferred networks with benchmark networks derived from experimental evidence [119]. The benchmark networks for $16C \rightarrow TE + ICM$ and $ICM \rightarrow PE + EPI$ are shown in Figure 3.2. Note that the edge $Fgf4 \rightarrow Fgfr2$ in Figure 3.2(b) is a signal transduction event which occurs among cells (i.e. from EPI progenitors to PE progenitors) [172]. Since we aim to reconstruct GRN within cells rather than across cells, we do not count it when evaluating the performance of GRN inference.

To evaluate the strength of SingCellNet, we applied other methods of GRN inference on the same dataset and compared their performance with our method. These methods include Bayesian network (using WinMine package [179]), Boolean network (using BoolNet package [180]), Random Forest [71, 81], LASSO [181], Elastic-Net [182], and *Treegl* [128]. The results of comparisons are shown in Table 3.1. TP, FP, FN and TN

Figure 3.2: Benchmark networks for (a) 16C→TE+ICM, (b) ICM→PE+EPI, (adapted from [119]). Since the link $Fgf4 \rightarrow Fgfr2$ is a signal transduction event, it is represented by a dashed line.

represent the numbers of true positives, false positives, false negatives and true negatives respectively. Pre, Sen, Fm, Spe and Acc denote precision, sensitivity, F-measure, specificity and accuracy respectively. Their definitions can be found in Section 2.1.3. The “consenNet” in the last row of the two tables denotes the consensus network obtained by the majority vote from the six other methods (i.e. except SingCellNet). The consensus networks above use 0.5 as the thresholds, i.e. every edge in the consensus network must be supported by at least half (50%) of the methods. Figure 3.3 and Figure 3.4 show the inferred networks by different methods for 16C→TE+ICM and ICM→PE+EPI respectively. Note that the *SingCellNet** in Table 3.1(a) and Figure 3.3 denotes a variant of SingCellNet that integrates signal perturbation during structure inference. Specifically, we send an inhibition signal to gene Cdx2 at the beginning of generating the transitions from each cell state. Then after obtaining the STG, we assign bonus to the transitions to the states in ICM while impose penalty on the transitions to the TE states, using the score function similar as Eq. (3.6). The idea of integrating signals into structure inference is motivated by the biological process of lineage formation from 16C to 32C stage. Signaling pathways such as the Hippo pathway are known to play important roles in TE/ICM segregation, and cells adopt the ICM cell fate mainly by negative regulation of TE regulators [183, 119]. Since adding the signal perturbation requires prior knowledge, we only discuss the performance of SingCellNet (instead of SingCellNet*) here, for fair comparison with other methods.

As shown in Table 3.1(a), our method SingCellNet can achieve performance close to Random Forest which ranks first in the GRN inference for 16C→TE+ICM. It suggests that SingCellNet is competitive with Random Forest, even though it probably loses some information in data due to discretization. Table 3.1(b) shows that SingCellNet

Table 3.1: Performance comparison of different methods on structure inference

(a) 16C→TE+ICM									
Methods	TP	FP	FN	TN	Pre	Sen	Fm	Spe	Acc
SingCellNet*	6	1	4	14	0.86	0.60	0.71	0.93	0.80
Random Forest	6	4	4	11	0.60	0.60	0.60	0.73	0.68
SingCellNet	6	6	4	9	0.50	0.60	0.55	0.60	0.60
BoolNet	4	4	6	11	0.50	0.40	0.44	0.73	0.60
Treegl	3	1	7	14	0.75	0.30	0.43	0.93	0.68
LASSO	4	6	6	9	0.40	0.40	0.40	0.60	0.52
Elastic-Net	4	8	6	7	0.33	0.40	0.36	0.47	0.44
Bayesian	1	5	9	10	0.17	0.10	0.13	0.67	0.44
consenNet	5	5	5	10	0.50	0.50	0.50	0.67	0.60

(b) ICM→PE+EPI									
Methods	TP	FP	FN	TN	Pre	Sen	Fm	Spe	Acc
SingCellNet	6	2	2	15	0.75	0.75	0.75	0.88	0.84
BoolNet	5	7	3	10	0.42	0.63	0.50	0.59	0.60
Treegl	3	5	5	12	0.38	0.38	0.38	0.71	0.60
Random Forest	3	7	5	10	0.30	0.38	0.33	0.59	0.52
Bayesian	2	4	6	13	0.33	0.25	0.29	0.76	0.60
LASSO	3	11	5	6	0.21	0.38	0.27	0.35	0.36
Elastic-Net	3	13	5	4	0.19	0.38	0.25	0.24	0.28
consenNet	3	8	5	9	0.27	0.38	0.32	0.53	0.48

outperforms all other methods significantly in the structure inference for ICM→PE+EPI. We notice that SingCellNet performs better in ICM→PE+EPI than in 16C→TE+ICM, while most other methods have lower performance in ICM→PE+EPI. One possible reason could be that the performance of other methods might be affected by the reduced number of samples, as the number of cells (i.e. samples) in ICM→PE+EPI is 56.75% of that in 16C→TE+ICM.

The good performance of SingCellNet is probably due to its ability to utilize the characteristics of single-cell expression data in the context of cell development. Among cells of the same cell type in the single-cell datasets there is heterogeneity of cellular gene expression. Furthermore, there are only two time points with a wide interval in each of the two datasets used here, which provides insufficient information of variation along

Figure 3.3: The inferred networks for 16C→TE+ICM by: (a) Bayesian network, (b) BoolNet, (c) Random Forest, (d) Elastic-Net, (e) LASSO, (f) Treegl, (g) SingCellNet and (h) SingCellNet*, where solid green and red lines denote TPs and FPs respectively, and grey dashed lines denote FNs.

time points for conventional methods to capture the dependency among variables (i.e. genes). Bayesian network can learn causal relationships among variables and is widely used in GRN inference. However, when applied to the single-cell expression dataset, Bayesian network would assume each cell is an independent sample without considering the heterogeneity among cells and their clustering into cell types. Boolean network is

Figure 3.4: The inferred networks for ICM→PE+EPI by: (a) Bayesian network, (b) BoolNet, (c) Random Forest, (d) Elastic-Net, (e) LASSO, (f) Treegl and (g) SingCellNet, where solid green and red lines denote TPs and FPs respectively, and grey dashed lines denote FNs.

a simple yet effective method for GRN inference. To compare Boolean network with our method, we used the *BestFit* algorithm to represent Boolean network approaches.

Since this method does not take into account the inherent relationship among cell types, its performance is lower than SingCellNet. Random Forest, LASSO and Elastic-Net are regression-based methods, which are quite effective for GRN inference partly because they do not require data discretization. Similar as Bayesian network, these methods would consider a single cell as an independent sample and build the regression models based on the samples. As a result, they may have high false positive rates. Among the six methods, *Treegl* is probably the closest to SingCellNet in that it also considers the cell lineage tree for reverse engineering of gene networks. However, *Treegl* was not designed for single-cell expression data. Moreover, it does not take into account the dynamic “directionality” of cell development, but instead relies on the similarity among networks.

Compared with other methods, SingCellNet takes advantage of additional information supplied by the single-cell gene expression data. For example, the heterogeneity among individual cells can provide statistical signals of state transitions; the directionality of trajectories along branches of cell lineage tree can help filter out many spurious network topologies. Running on the dataset with short and sparse time series, SingCellNet is less affected by the lack of sufficient signals of local dependence among variables, because it captures the global and qualitative properties of a system (e.g. the stability of a network state).

3.4.2 Analysis of Parameters

Here we discuss how the changes of parameters in our method (Algorithm 3.1, Eq. (3.4) and Eq. (3.6)) affect the performance of SingCellNet. Our method contains the following parameters: population size U , number of selected individuals U' , mutation rate μ , the number of permutations per chromosome η , the maximum number of iterations *MaxIt* (denoted as M here), the maximum number of regulators for each gene K , and u , which is the number of asynchronous updates to obtain the transition probabilities of the state transition graph (STG). The results above were obtained based on the following parameter settings: $U = 1000, U' = 100, \mu = 0.01, \eta = 1, M = 50, K = 3, u = 100$. In addition, we have also tested different settings of parameters and analysed whether our algorithm is sensitive to the settings, as follows.

First, the following four parameters are commonly used in GA: the population size U , the number of selected individuals U' , mutation rate μ and permutation rate η . These parameters would influence the performance of GA itself, e.g. the running time and rate of convergence. However, our experiments show that they have little effect on the performance of our algorithm (data not shown). Therefore, here we set these parameters according to the standard settings in GA.

Setting the maximum number of iterations M to 1000, 500, 100, and 50, we observed that the algorithm can always achieve convergence before 50 iterations (see Figure 3.5).

Therefore we set $M = 50$ to save the running time. Since the highest score may not be unique, we obtain the final inferred network by taking the “consensus” of the networks with the highest scores, using a standard voting procedure (e.g. an edge is accepted if it is supported by over half of the networks).

Figure 3.5: The convergence of scores over GA iterations. “Mean” and “Median” denote the mean and median scores of the U' individuals in each iteration, respectively.

The parameter u denotes the number of asynchronous updates to obtain the transition probabilities of the STG. We have tried 1000 and found that the transition probabilities obtained are very similar to those with $u = 100$ (data not shown). So we used $u = 100$ to reduce the running time.

The parameter K represents the maximum number of regulators for each gene. According to the power-law degree distribution in gene regulatory networks ([184, 185]), K is typically set to 3, 4, or 5 for GRN inference. An appropriate choice of K depends on some prior knowledge about the benchmark network. Here we chose $K = 3$. We have also tested the cases for $K = 4$ or 5, and found the results to be similar, as shown in Table 3.2.

In addition, the “bonus” and “penalty” in the fitness scores (i.e. Eq. (3.4) and Eq. (3.6)) are also parameters of our method. As seen from Eq. (3.4), there are three parameters: bonus for positive transition (denoted as “*bon_pos*” here), penalty for backward transitions (denoted as “*pen_back*”), and penalty for trans-transitions (denoted as “*pen_tran*”). Among positive transitions, a special type is self-transitions with the bonus “*bon_self*”. Currently, we set their values empirically. The results shown in the main text correspond to the following parameter settings: *bon_self*=1, *bon_pos*=3, *pen_back*=-2, *pen_tran*=-3. Normally we set these parameters with values ranging from 1 to 5, although they could take any real numbers. We observed that the performance of our method would not change significantly as long as the parameters satisfy the following

Table 3.2: Performance of SingCellNet with different values of K

(a) 16C→TE+ICM			
Parameter K	Precision	Sensitivity	F-measure
$K = 3$	0.4167	0.5000	0.4545
$K = 4$	0.4000	0.4000	0.4000
$K = 5$	0.3571	0.5000	0.4167

(b) ICM→PE+EPI			
Parameter K	Precision	Sensitivity	F-measure
$K = 3$	0.7143	0.6250	0.6667
$K = 4$	0.6000	0.7500	0.6667
$K = 5$	0.5000	0.7500	0.6000

conditions: $bon_pos \geq bon_self > 0$. In Eq. (3.6), the “bonus” parameter is for the chosen lineage, and the “penalty” parameter is for the alternative lineage. We set both parameters to 1 to obtain the results presented in the main text; other values could also be assigned to these two parameters. In the future, we will use a machine learning method to infer the values of these parameters from data.

All the results of our method SingCellNet above are based on the data with unique states. Since we use a vector of gene expression levels to represent a cell state, when the number of genes n is small (here $n = 5$), it is possible that multiple cells correspond to the identical state vector. Previously we counted each unique cell state only once (i.e. multiple cells mapping to the same state vector will be considered as one state) and applied SingCellNet on the dataset of unique states (denoted as the “unique” dataset). We have also tested our method on “duplicate” dataset (i.e. even if two cells have identical state, we still feed both as input to the algorithm). However, we found that the performance on duplicate dataset is slightly worse than on the unique dataset (see Table 3.3). This could be due to the fact that the number of cells in Robson’s dataset is relatively small, which may not contain sufficient statistical signals for accurate modeling of cell state distributions. Moreover, the distribution of cells might be mediated by external perturbations (e.g. signal transduction) during embryonic development, but our work is currently focused on internal gene regulatory network. Nonetheless, considering the number of cells and signaling perturbations would be promising future work to yield new insights into cell fate decisions.

Table 3.3: Performance of SingCellNet on data with unique and duplicate states

(a) 16C→TE+ICM			
Data	Precision	Sensitivity	F-measure
“Unique”	0.4167	0.5000	0.4545
“Duplicate”	0.4000	0.4000	0.4000

(b) ICM→PE+EPI			
Data	Precision	Sensitivity	F-measure
“Unique”	0.7143	0.6250	0.6667
“Duplicate”	0.5455	0.7500	0.6316

3.4.3 Results of Identifying Operational Interactions

To validate our method for probability inference of rules on real expression data from 16-cell to 64-cell stage, we adapted the network derived from experimental evidence [119] as our benchmark network (see Figure 3.1(b)).

By encoding a network with rules and corresponding probabilities, and comparing simulated trajectories of cells with expected paths on the cell lineage tree, our method can infer which part of the network is responsible for a specific cell fate decision. The encoded rules for this network (i.e. Figure 3.1(b)) are provided in Table 3.4. All the rules were derived based on the knowledge from literature. Note that when encoding the network, we removed the gene *Sall4* from Figure 3.1(b) because it has no in-coming edge and thus the system of this network cannot update its state. In addition, we added two edges, *Oct4* → *Sox2* and *Nanog* → *Sox2*, because these interactions have been observed in mouse embryonic development ([186, 187, 188, 189]).

Since there are four cell fate decisions (i.e. 16C→TE, 16C→ICM, ICM→PE, ICM→EPI), we conducted GA searching for probability inference four times, one for each of the four cell fate decisions during mouse early development. Applied to Robson’s single-cell gene expression data, our method is able to identify the significant rules (i.e. rules with high probabilities) for each lineage formation, shown in Table 3.5. Results show that, the rules with significantly higher probabilities in 16C→TE are $1 \rightarrow Cdx2$ and $Oct4|Nanog \rightarrow Sox2$. The significant rules for the other three processes of cell fate decisions are $Cdx2 \ \& \ \sim(Oct4|Nanog) \rightarrow Cdx2$, $Nanog \rightarrow Fgf4$, and $\sim Nanog \ \& \ (Gata6|Fgfr2) \rightarrow Gata6$, respectively.

As shown in Table 3.5, the most significant rule for 16C→TE is $1 \rightarrow Cdx2$. Gene *Cdx2* is the main regulator for TE cell type [193, 173]. Signaling pathways such as the Hippo

Table 3.4: Rules and their references

Rules	References [†]
$0 \rightarrow Cdx2$	([183])
$1 \rightarrow Cdx2$	([183])
$Cdx2 \ \& \ \sim(Oct4 Nanog) \rightarrow Cdx2$	([171]; [119])
$Cdx2 \rightarrow Cdx2$	([190])
$Cdx2 \mid Eomes \rightarrow Eomes$	([191]) [*]
$Cdx2 \sim(Oct4 Nanog Sox2)\rightarrow Eomes$	([119]) [*]
$Nanog Sox2 \rightarrow Oct4$	([188]; [187])
$\sim Cdx2 \ \& \ (Nanog Sox2) \rightarrow Oct4$	([171]; [119])
$Oct4 Sox2 \rightarrow Nanog$	([186])
$\sim Cdx2 \ \& \ (Oct4 Sox2) \rightarrow Nanog$	([119]; [173])
$\sim Gata6 \ \& \ (Oct4 Sox2) \rightarrow Nanog$	([189]; [119])
$\sim Fgfr2 \ \& \ (Oct4 Sox2) \rightarrow Nanog$	([172]; [119])
$Oct4 Nanog \rightarrow Sox2$	([188]; [187])
$Gata6 Fgfr2 \rightarrow Gata6$	([172]; [119])
$\sim Nanog \ \& \ (Gata6 Fgfr2) \rightarrow Gata6$	([172]; [119])
$Nanog \rightarrow Fgf4$	([172], [173])
$1 \rightarrow Fgf4$	([192], [173])
$Fgf4 \ \& \ \sim Nanog \rightarrow Fgfr2$	([172], [173]) [*]
$1 \rightarrow Fgfr2$	([172]) [*]
$Gata6 Fgfr2 \rightarrow Gata4$	([172]; [119])
$Gata6 \ \& \ Fgfr2 \rightarrow Gata4$	([172]; [119])

[†]References with * denote the corresponding rules have been revised

Table 3.5: Significant rules for lineage decisions

Lineage decisions	Significant rules
$16C \rightarrow TE$	$1 \rightarrow Cdx2; Oct4 Nanog \rightarrow Sox2$
$16C \rightarrow ICM$	$Cdx2 \ \& \ \sim(Oct4 Nanog) \rightarrow Cdx2$
$ICM \rightarrow PE$	$Nanog \rightarrow Fgf4$
$ICM \rightarrow EPI$	$\sim Nanog \ \& \ (Gata6 Fgfr2) \rightarrow Gata6$

pathway are known to play important roles in TE/ICM segregation, by activating *Cdx2* in TE while blocking it in ICM [119]. Consistent with the observations, our algorithm

assigned the highest probability to the rule $1 \rightarrow Cdx2$ in $16C \rightarrow TE$. The second significant rule $Oct4|Nanog \rightarrow Sox2$ may suggest that, during the TE formation, *Sox2* needs to be suppressed, which is achieved by the inhibition of its activator *Oct4* or *Nanog* by *Cdx2*. In other words, *Oct4* or *Nanog* passes the inhibition from *Cdx2* to *Sox2*. The active rule for $16C \rightarrow ICM$ is $Cdx2 \& \sim (Oct4|Nanog) \rightarrow Cdx2$, which shows the inhibition of *Cdx2* by *Oct4* (as well as *Nanog*). It has been shown that the antagonism between *Cdx2* and *Oct4* (as well as *Nanog*) directs and reinforces the cell fate to be either TE or ICM [193, 173, 119]. The target of this rule is *Cdx2*, which indicates that the driving force of ICM formation may be from the suppression of *Cdx2*. Unlike TE formation, where the driving force is from signaling pathways, cells adopt the ICM cell fate mainly by negative regulation of TE regulators [183, 119].

For $ICM \rightarrow PE$, the rule with the highest probability is $Nanog \rightarrow Fgf4$, which reflects the following aspect of PE/EPI specification. *Fgf4/Fgfr2* is a ligand-receptor pair in FGF signaling pathway, which is essential for PE differentiation. First *Fgf4* and *Fgfr2* are both activated. Then *Fgfr2* activates *Gata6* which is the marker gene of the PE cell type. *Nanog* in EPI progenitors activates *Fgf4* (i.e. $Nanog \rightarrow Fgf4$), which would further induce PE-specific genes in PE progenitors through $Fgf4 \rightarrow Fgfr2$ and complete the program of PE differentiation [194, 192, 172]. The significant rule for $ICM \rightarrow EPI$ inferred by our algorithm is $\sim Nanog \& (Gata6|Fgfr2) \rightarrow Gata6$, which shows the inhibition of *Gata6* from *Nanog*. This is consistent with literature that EPI formation is obtained by the suppression of *Gata6* [172, 119].

Overall, our method can infer the operational interactions in the GRN for specific cell fate decisions, which are consistent with experimental literature. Our approach sheds new light on mechanism of cell development by analysis of the single-cell data.

3.5 Conclusions, Discussion and Future Work

In this chapter, we have proposed a novel method to uncover regulatory circuits of cell fate decisions from single-cell transcriptional data. We mapped the transcriptional dynamics of cell development to the state transition graph generated by an asynchronous probabilistic Boolean network, and defined a score to evaluate how well a candidate network fits the data. To learn the optimal network structure, a genetic algorithm was employed to search for the network with the most natural cell state transitions according to the cell lineage tree. Applied to real single-cell expression data and validated against benchmark networks from biological experiments, our method outperformed most of the other methods for GRN inference. Through simple adaption, our method could also be used to identify operational interactions in GRN responsible for cell fate decisions.

As one of the first methods for network analysis of single-cell data, our method still needs to address several issues. First, it has been verified only on Robson’s dataset,

partly due to the dearth of publicly available single-cell data and benchmark networks. Nevertheless, we believe that the strategy of our method can be naturally applied to other single-cell datasets in many biological contexts (e.g. cancer, stem cell). Second, the scoring function for a network in our algorithm is heuristic and with predefined parameters. In the future, we will generate the scoring function more objectively and use machine-learning techniques to infer the values of parameters from data. Third, although our method achieved good prediction performance (measured by precision, sensitivity, and F-measure), it is currently slower than the other methods, partly because our method was implemented in Matlab, while others were in Java or C++. Also, we used genetic algorithm, which is known to be computationally intensive. In future, we will speed up our method by parallel computing, and other strategies (e.g. [195]). Fourth, the procedure of cell lineage commitment is often triggered by external stimulus, e.g. signaling events from neighbouring cells, which has not been explicitly included in our current model. In future, this could be modelled as a perturbation that pushes a cell state to a different point in a state space without following any internal regulatory rules.

Chapter 4

Inference of Time-Delayed Gene Regulation Based on Network Deconvolution

4.1 Introduction

4.1.1 Background

Gene regulatory network (GRN) is a fundamental topic in systems biology. The dynamics of GRNs can shed light on the cellular processes, which facilitates the understanding of the mechanisms of diseases when the related processes are dysregulated. Accurate reconstruction of GRNs could also provide guidelines for experimental biologists. Therefore, inferring GRNs from high-throughput gene expression data has drawn much research interest for over a decade. However, due to the inherent complexity of gene regulation, noise in the data measurement, the short length of time-series data and so on, it is very challenging to accurately reconstruct GRNs. On the other hand, a better understanding into the mechanisms of gene regulation could offer help to improve the performance of GRN inference. Time delay is an essential characteristic of gene regulation. Generally a gene regulates the expression of another gene by its products such as RNAs or proteins. And it takes different amounts of time to generate these products through different processes (e.g. transcription, translation). Therefore, time delay is inherent in gene regulation. By incorporating the information of time delays, we could achieve more accurate inference of GRNs.

4.1.2 Motivation

Although quite a few methods have been proposed to infer time-delayed GRNs, the accuracy of inference is yet to be improved. Recently, a novel algorithm named network deconvolution (ND) [196] was developed for GRN inference. ND is a technique to identify direct dependencies in an observed network (e.g. correlation-based network) which contains both direct and indirect interactions. By assuming that the indirect edges could be estimated from the products of the direct edges and the observed network is the sum of the direct and indirect edges, ND can recover the direct interactions from the observed network through the process of deconvolution. ND has many superior merits in inferring genetic relationships, such as good accuracy and robustness. Nevertheless, the original ND algorithm did not consider time delays in gene regulation, i.e. all direct interactions were assumed to take equal time, which is unlikely in the real biological systems. In addition, ND cannot infer the directions of gene interactions either.

In this chapter, we propose a simple yet effective and efficient method to tackle the challenges of inferring high-order (i.e. time-delayed) gene regulation, based on ND. Using cross-correlation [197, 198] and data processing, we first determine the probable time lag of each pair of genes. Then ND algorithm is applied to infer the time-delayed GRN. Our method integrates time delay inference and adjustment into the ND approach, to further increase its power. By taking into account the time delays among gene interactions, our method is able to infer GRNs more accurately. The effectiveness of our method has been shown by the experiments on three real-life gene expression datasets of yeast. Compared with other existing methods which were designed for learning time-delayed GRNs, our method has significantly higher sensitivity without much reduction of specificity. Overall, our method achieves better performance (in terms of F-measure) than other methods.

4.2 Methods

We propose a method to infer time-delayed gene regulation based on cross-correlation and network deconvolution. First, we utilize cross-correlation to obtain the probable time delays for the interactions between each gene and its potential regulators. Then we align the samples in the data based on the identified time delays, and calculate the correlations of genes based on the aligned samples. After that, we apply ND to the correlation matrix and identify the direct interactions between each target gene and its regulators. The network inferred by our method would be directed and includes time delays.

The time-delayed GRNs are inferred from time-series gene expression data. Let $X_{T \times N} = (x_1, x_2, \dots, x_N)$ be a time-series gene expression dataset where N is the number of genes and T is the number of time samples. Let $x_{t,i}$ denote the expression level of

the i -th gene at time t . Then $x_i = (x_{1,i}, x_{2,i}, \dots, x_{T,i})^T$, where $1 \leq i \leq N$ denotes the expression profile of the i -th gene across T time points.

To infer regulatory interactions among genes, the most straightforward way is by using correlations. However, there are two major issues about correlations: (1) time-delayed regulation is not likely to be inferred by simple correlations; (2) the relationships based on correlations may not be direct and would suffer from a large number of false positive predictions. The two issues can be coped with by cross-correlation and network deconvolution respectively, which are described in the following two sections.

4.2.1 Cross-correlation

To infer a time-delayed regulation between the i -th gene and j -th gene, we need to determine the number of time lags first. This could be achieved by applying cross-correlation [197, 198] on the expression profiles of the two genes. The lag that gives the maximum absolute cross-correlation is the most likely time lag.

For two real continuous functions x and y , the cross-correlation ϕ_{xy} is defined as [197]:

$$\phi_{xy}(\tau) = \int_{-\infty}^{\infty} x(t - \tau)y(t)dt. \quad (4.1)$$

The cross-correlation $\phi_{xy}(\tau)$ can be obtained by inverse Fourier transform (iFT) as follows:

$$\phi_{xy} = iFT[\Phi_{xy}(\tau)] = iFT[X^*(f)Y(f)]. \quad (4.2)$$

where Φ_{xy} , X and Y are ϕ_{xy} , x and y in frequency domain respectively, X^* is the conjugate of X . The derivation of Eq. (4.2) is shown in the footnote of this page.¹

The cross-correlation could also be obtained in another way as follows. Let r be the maximum length of time delays, and τ be one possible lag in a time delay, i.e. $\tau = 0, 1, 2, \dots, r$. For the i -th gene and the j -th gene (with expression profiles denoted as x_i and x_j respectively), the unbiased cross-correlation is defined as in [197, 89]:

$$C(x_i, x_j, \tau) = \frac{1}{T - |\tau|} \sum_{t=1}^{T-\tau-1} x_{t+\tau,i} x_{t,j}, \text{ where } \tau = 0, 1, 2, \dots, r. \quad (4.3)$$

Note that we usually normalize the gene expression data for each gene to have “zero” mean and “one” standard deviation before calculating the cross-correlation.

¹The derivation of Eq. (4.2): Using the convolution expression, we have $\phi_{xy}(\tau) = x(-\tau) * y(\tau)$; Converting $x(-\tau)$ to frequency domain using Fourier transform $FT[x(-\tau)] = \int_{-\infty}^{\infty} x(-\tau)\exp(-i2\pi f\tau)d\tau$; Substituting $\tau' = -\tau$ we get $FT[x(-\tau)] = \int_{\infty}^{-\infty} -x(\tau')\exp(i2\pi f\tau')d\tau' = \int_{-\infty}^{\infty} x(\tau')\exp(i2\pi f\tau')d\tau' = X^*(f)$; Combining the equations above we have $\Phi_{xy} = FT[\phi_{xy}(\tau)] = X^*(f)Y(f)$; Applying inverse Fourier transform we can obtain ϕ_{xy} as in Eq. (4.2).

In this study, we utilize the Matlab function *xcorr* which adopts the previous way (i.e. Eq. (4.2)) to calculate cross-correlations. For each target gene, cross-correlation is calculated between this gene and all the other genes. From the maximum possible r time lags, we identify the lag τ which corresponds to the maximum absolute value of $C(x_i, x_j, \tau)$. This time lag is denoted by l_{ij} and it represents the most probable time lag of regulation between the i -th gene and the j -th gene.

4.2.2 Network Deconvolution

After determining the probable time lag for each gene pair, we proceed to determine the regulators for each target gene (i.e. to identify the direct interactions among genes). As mentioned above, using correlations is a natural way to identify interactions among genes; but such approach may suffer from a large number of false positive predictions. The main reason is that many correlations represent “indirect” instead of “direct” dependencies. A direct dependency between two variables mean that the interaction between the two variables does not depend on any intermedium. On the other hand, the indirect dependency is caused by direct dependencies through some intermediate nodes. For example, if A regulates B and B regulates C , even though there is no direct relationship between A and C , the correlation between A and C could be high because there is an intermediate node B between them. Network deconvolution [196] is a technique to infer direct dependencies among variables. Let us use a matrix to represent a network. Starting from the matrix of correlations (or other similarity metrics such as mutual information) which could include both direct and indirect dependencies, ND is able to filter out the indirect dependencies through a process called network deconvolution.

Let S_o be the observed network (i.e. the correlation matrix), and S_d be true network with only direct edges. ND assumes that the indirect edges could be derived from the product of direct edges, and the observed network S_o is the sum of the direct and indirect edges, as follows.

$$S_o = S_d + S_d^2 + S_d^3 + S_d^4 + \dots \quad (4.4)$$

The task of obtaining S_d from S_o seems intractable because of the infinite sum. However, by using the closed form solution of infinite Taylor series, we can obtain

$$S_o = S_d(I + S_d + S_d^2 + S_d^3 + \dots) = S_d(I - S_d)^{-1}, \quad (4.5)$$

where I is the identity matrix. Note that Eq. (4.5) holds with requirement that the largest absolute eigenvalue of S_d is less than one (more details are shown subsequently). Through simple transformation, we have

$$S_d = S_o(I + S_o)^{-1}. \quad (4.6)$$

It has been shown in [196] that all symmetric matrices and even some asymmetry matrices can be decomposed into their eigenvectors and eigenvalues. Let U be the matrix of eigenvectors, and Σ_o be the diagonal matrix of eigenvalues of S_o . Then $S_o = U\Sigma_o U^{-1}$. Similarly, we have $S_d = U\Sigma_d U^{-1}$. Let λ_i^o be the i th eigenvalue of S_o , and λ_i^d be the i th eigenvalue of S_d . Through Eq. (4.6), we have

$$\lambda_i^d = \frac{\lambda_i^o}{1 + \lambda_i^o}. \quad (4.7)$$

To guarantee the convergence of Eq. (4.4), which requires that the largest absolute eigenvalue of S_d is less than one, the authors of ND introduced a scaling factor α . Then Eq. (4.7) is changed to

$$\lambda_i^d = \frac{\lambda_i^o}{\frac{1}{\alpha} + \lambda_i^o}. \quad (4.8)$$

In our code, the scaled version (i.e. Eq. (4.8)) of ND was implemented. We used the default value for the parameter of α .

Experimental results in [196] (i.e. the original paper of ND) show that when their assumptions hold that the indirect edges can be derived from the product of direct edges and the observed network is the sum of direct and indirect edges, ND can remove all indirect edges and recover all direct edges; even when the assumptions do not hold, it still can infer most of direct interactions, as shown by simulation experiments on various network structures. For more details about ND, please refer to [196].

4.2.3 Time-delayed ND

We incorporate time delays into network deconvolution to enhance its strength in GRN inference. The flow chart of our method named time-delayed ND is shown in Figure 4.1. Starting with the time-series gene expression data, for each target gene we perform the following steps. First we identify the time lags of regulations based on cross-correlation. Then we align the time samples based on the inferred time lags. After that, we calculate the correlations for each gene pairs based on the aligned samples, and apply ND on the correlation matrix. Finally, we obtain a GRN with time-delayed regulation.

Algorithm 4.1 gives more technical details about the procedure of time-delayed ND. For each gene (as the target gene) in the set of N genes, the same process is carried out, which consists of three steps. First, with a fixed maximum time delay r , we identify the most likely time lags which corresponds to the maximum absolute cross-correlation for the interaction between the target gene and the potential regulator (note that all other genes are potential regulators). Second, we align the time samples for the target gene based on the time delays (see Algorithm 4.2), and compute correlations between this gene and the other genes based on the aligned samples. Third, we apply ND on the matrix

Figure 4.1: The flow chart of time-delayed ND.

of correlations and obtain a new matrix (as the output of ND) with direct dependencies among genes. Then the direct dependencies between the target gene and the other genes are extracted and stored. Note that the dependencies here are with directions. In this way, we infer directed time-delayed regulation among genes (i.e. a time-delayed GRN) from time-series gene expression data.

In aligning time samples based on the inferred time lags, we assume that the time-series data are not periodic. Recall that r is the maximum order of regulation between the target gene and its regulators. Then effectively we have $T - r$ samples based on which we calculate the correlations between the target gene and its possible regulators. The procedure of aligning time samples is presented in Algorithm 4.2. Figure 4.2 is an example showing how to do the alignment of time samples based on the lags between the target gene g and its possible regulators (gene a , b , c or d). The symbol $\surd$ inside a slot indicates

Algorithm 4.1. Inferring time-delayed regulation based on Network Deconvolution

INPUT: Time-series gene expression data $X_{(T \times N)}$ with N genes and T time samples; the maximum possible time lag r

OUTPUT: An $N \times N$ matrix S with weights to show the strength of interactions among genes; An $N \times N$ matrix D with the information of time delay for each interaction

Normalize X so that the expression data for each gene have “zero” mean and “one” standard deviation

for each gene i **do**

Initialize a temporary vector $L_i^* = []$

for each gene j in the rest genes **do**

Calculate the cross-correlation $C(x_i, x_j)$ for gene i and gene j

Identify the time lag l_{ij} with the maximum absolute cross-correlation $C(x_i, x_j)$ among r choices of time lags

Store l_{ij} into L_i^*

end for

Obtain the matrix X'_i with aligned time samples for target gene i according to Algorithm 4.2

Initialize a temporary $N \times N$ matrix $C^* = []$

for each gene i' **do**

for each gene j' **do**

Compute the correlation $corr_{ij}$ between gene i' and gene j' based on the corresponding time samples in X'_i .

Store $corr_{ij}$ into C^*

end for

end for

Apply ND on C^* to obtain S^* which denotes the direct interactions among genes based on the extracted time samples.

Extract the vector S_i^* which contains the direct dependencies between gene i and its potential regulators and store it into S , $S = [S \ S_i^*]$

Append the values of time lags to D , $D = [D \ L_i^*]$

end for

Return S and D

that the corresponding time sample will be used to calculate the correlation, while the empty slots mean that those samples will not be used to calculate the correlations between this target gene and its potential regulators. When the time-series data are periodic, a similar method of alignment can be used, except that in this case we can use all the T time samples, in which the time points are shifted circularly [86].

4.3 Experiments and Results

We have proposed a method to infer time-delayed gene regulation based on cross-correlation and network deconvolution. Compared to the original ND approach, our method (i.e. time-delayed ND) can infer directed networks with time delays. We have evaluated the performance of our method on three real-life time-series datasets of gene expression. We

Algorithm 4.2. Aligning time samples based on the delays inferred by cross-correlation

INPUT: Time-series gene expression data $X_{(T \times N)}$; the matrix of time delays L_i^* for target gene (denoted as the i -th gene); the maximum possible time lag r

OUTPUT: An $(T - r) \times N$ matrix X'_i with aligned time samples for the target gene

Initialize $X'_i = []$

Extract the vector of time samples for the target gene (the i -th gene) from X , $y = X(r + 1 : T, i)$

Append the expression of the target gene to X'_i , $X'_i = [X'_i \ y]$

for each gene j ($j \neq i$) **do**

 Find time delay l_{ij} of interaction between the target gene i and gene j from L_i^*

 Extract and align time samples for gene j from X based on l_{ij} , $x_j = X(r - l_{ij} + 1 : T - l_{ij}, j)$

 Append the expression of the j -th gene to X'_i , $X'_i = [X'_i \ x_j]$

end for

Return X'_i

	<i>g</i>	<i>a</i>	<i>b</i>	<i>c</i>	<i>d</i>
1			✓		
2		✓	✓		✓
3		✓	✓	✓	✓
4	✓	✓	✓	✓	✓
⋮	⋮	⋮	⋮	⋮	⋮
$T-3$	✓	✓	✓	✓	✓
$T-2$	✓	✓		✓	✓
$T-1$	✓			✓	
T	✓				

Figure 4.2: Alignment of time samples based on time lags.

first compare the performance of time-delayed ND and the original ND. Then we compare time-delayed ND with other methods for inferring time-delayed GRNs.

4.3.1 Benchmark Networks and Data

The performance of our method has been evaluated on two real-life benchmark networks and compared with other methods. One benchmark network (see Figure 4.3.a) is a five-gene network in yeast, from the experiment of *in vivo* reverse-engineering and modeling assessment (IRMA) [199]. This network was carefully constructed to include the interactions among five genes (CBF1, GAL4, SWI5, GAL80 and ASH1) in *Saccharomyces cerevisiae* and made sure that the influence from endogenous genes is negligible. Two

datasets of gene expression were measured for this network. One dataset was obtained when the cell culture was shifted from glucose to galactose. This dataset was named “switch-on” because the network would be triggered by galactose. The other dataset was named “switch-off” since it is obtained by shifting the cell culture from galactose to glucose. The “switch-on” dataset (denoted as *yeast5on*) consists of 16 equally distributed time points, and the “switch-off” dataset (denoted as *yeast5off*) contains 21 equally distributed time points.

The other benchmark network (see Figure 4.3.b) is experimentally identified in *Saccharomyces cerevisiae* (yeast) cell cycle [200]. This network consists of nine genes (ACE2, CLN3, FKH2, MBP1, MCM1, NDD1, SWI4, SWI5 and SWI6). The real expression data (denoted as *yeast9Spellman*) of genes in this network were taken from Spellman *et al.* [201], which consists of the transcription profiles in yeast cell cycle. We extracted the time-series data from “cdc-15” cell cycle arrest which contains 24 equally distributed time points. In addition, we also extract two datasets of the nine genes from Breeden’s lab (GEO accession number GSE5376). The two datasets (which are technically duplicate) are denoted as *yeast9Breeden1* and *yeast9Breeden2*. Similar to Spellman’s data, the Breeden data are also transcription profiles of yeast genes across the yeast cell cycle. Cells were synchronized with alpha factor in YEPD media and sampled every 5 minutes for 2 hours. Hence there are 25 time points in each dataset. The Breeden dataset, which was published in 2006 [202], is less noisy than the Spellman dataset. All these datasets would be used to validate the performance of our method.

 4.3.a Benchmark network of *yeast5*

 4.3.b Benchmark network of *yeast9*

 Figure 4.3: Benchmark networks of *yeast5* [199] and *yeast9* [200] .

4.3.2 Results of Network Inference

Before assessing the proposed method (i.e. time-delayed ND), let us look at the parameters that may affect its performance. There are two parameters in time-delayed ND. One is the threshold θ ($0 \leq \theta \leq 1$) for the matrix output by ND. Since this matrix is a weighted matrix where each entry represents the strength of interaction between the corresponding gene pair, we need to set up a threshold to obtain a connectivity matrix from this weighted matrix. After that we compare the connectivity matrix with the benchmark network and calculate performance metrics such as *sensitivity* and *F-measure*. In all the following experiments (except the ROC curves), we set the threshold θ to a moderate value (i.e. 0.5). We also show the results of time-delayed ND with θ equal to 0.6 and 0.7 (see Tables 4.1 ~ 4.5). The inferred networks by ND in Figure 4.5, Figure 4.6 and Figure 4.7 are based on the results with $\theta = 0.5$. The other parameter is the maximum time lag r . Since a big r would lead to a small number of samples available to infer gene interactions, we usually set $r \leq 5$. We have done experiments with r taking values from 2 to 5. The results of comparing to other methods are similar (data not shown). Here we only show the results with $r = 5$ for all experiments.

First we compared time-delayed ND with the original ND (no delay). By evenly changing the values of the threshold θ ($0 \leq \theta \leq 1$), we can generate a set of performance metrics. Then a receiver operating characteristic (ROC) curve can be plotted to show the overall performance of a method with the results obtained from the whole set of different values for the threshold. The comparison between time-delayed ND with the original ND (no-delay ND) is shown in Figure 4.4, where “False positive rate (FPR)” is defined as “ $1 - \text{specificity}$ ”, and “True positive rate (TPR)” is equal to “*sensitivity*”. Overall, time-delayed ND outperforms the original ND in that the AUC of the former is larger than the latter for all three datasets. Moreover, when TPR (i.e. *sensitivity*) is high (e.g. ≥ 0.5), the FPR of time-delayed ND is lower than the original ND most of the time, which means time-delayed ND has a higher *specificity*. These results show that the accuracy of GRN inference using ND is improved by taking into account the time delays among gene interactions.

To compare time-delayed ND with other existing methods which are designed for inferring time-delayed gene regulation, we applied these methods on the same dataset and compare their performance using *sensitivity*, *precision*, and *F-measure*. There are three existing methods available for our comparisons, namely time-delayed ARACNE (TD-ARACNE) [53], Xcorr+LASSO (TD-LASSO) [89, 203], and MMHODBN [87].

We first performed experiments on the benchmark network with five genes (shown in Figure 4.3.a), which contains two datasets: *yeast5on* and *yeast5off*. Table 4.1 and Table 4.2 show the performance comparison of the four methods on *yeast5on* and *yeast5off* respectively. In Table 4.1, the *F-measure* of time-delayed ND (when $\theta = 0.5$) ranks the

4.4.a Based on *yeast5on* (The AUC for TimeDelay-ND versus the original ND is 0.8493 vs 0.8051)

4.4.b Based on *yeast5off* (The AUC for TimeDelay-ND versus the original ND 0.6397 vs 0.6013)

4.4.c Based on *yeast9Spellman* data (The AUC for TimeDelay-ND versus the original ND is 0.6475 vs 0.6222)

Figure 4.4: Comparison between time-delayed ND and the original ND.

second place among the four methods. Although the *F-measure* of time-delayed ND is slightly lower than MMHODBN, it still has higher *sensitivity* than the latter. For *yeast5off* dataset, as shown in Table 4.2, the *F-measure* of time-delayed ND is again the highest among the four methods. The inferred networks by the four methods on the *yeast5on* and *yeast5off* are presented in Figure 4.5 and Figure 4.6 respectively. Note that

solid lines (green), dashed lines and dot lines denote true positives, false positives and false negatives respectively (likewise for Figure 4.7). We then carried out experiments on the *yeast9Spellman* dataset. The performance comparison between time-delayed ND and the other three methods in Table 4.3 shows that our method has significantly higher *F-measure* than all other methods, which means our method can infer time-delayed GRN more accurately. Although the *precision* of MMHODBN is higher than time-delayed ND, its *sensitivity* is lower. The inferred networks by the four methods are presented in Figure 4.7. We also compared time-delayed ND with other methods on the two datasets from Breeden’s lab, namely *yeast9Breeden1* and *yeast9Breeden2*. The comparison results of time-delayed ND and the other three methods on *yeast9Breeden1* and *yeast9Breeden2* are shown in Table 4.4 and Table 4.5 respectively. Similar to the results on the *yeast9Spellman* dataset, time-delayed ND has significantly higher *sensitivity* and better *F-measure* than other methods on the two Breeden datasets.

As the experimental results suggested, time-delayed ND has a high *sensitivity* in detecting time-delayed gene interactions, which yields a better performance (in terms of *F-measure*) than other methods.

Table 4.1: Performance comparison of different methods on *yeast5on* dataset.

	Precision	Sensitivity	F-measure
time-delayed ND¹	0.5000	0.8750	0.6364
time-delayed ND²	0.7000	0.8750	0.7778
time-delayed ND³	0.7500	0.7500	0.7500
TD-ARACNE	0.6667	0.2500	0.3636
TD-LASSO	0.4000	0.2500	0.3077
MMHODBN	1.000	0.5000	0.6667

time-delayed ND^{1,2,3}: $\theta = 0.5, 0.6, 0.7$ respectively; similarly for the tables below.

Table 4.2: Performance comparison of different methods on *yeast5off* dataset.

	Precision	Sensitivity	F-measure
time-delayed ND¹	0.3571	0.6250	0.4545
time-delayed ND²	0.4444	0.5000	0.4706
time-delayed ND³	0.4286	0.3750	0.4000
TD-ARACNE	0.5000	0.1250	0.2000
TD-LASSO	0.2500	0.1250	0.1667
MMHODBN	0.6667	0.2500	0.3636

Table 4.3: Performance comparison of different methods on *yeast9Spellman* dataset.

	Precision	Sensitivity	F-measure
time-delayed ND¹	0.2917	0.8235	0.4308
time-delayed ND²	0.3158	0.7059	0.4364
time-delayed ND³	0.3333	0.5882	0.4255
TD-ARACNE	0.2307	0.1764	0.2000
TD-LASSO	0.2500	0.2353	0.2424
MMHODBN	0.5000	0.1765	0.2609

Table 4.4: Performance comparison of different methods on *yeast9Breedden1* dataset.

	Precision	Sensitivity	F-measure
time-delayed ND¹	0.2623	0.9412	0.4103
time-delayed ND²	0.2955	0.7647	0.4262
time-delayed ND³	0.3158	0.7059	0.4364
TD-ARACNE	0.2632	0.2941	0.2778
TD-LASSO	0.1622	0.3529	0.2222
MMHODBN	0.1429	0.0588	0.0833

Table 4.5: Performance comparison of different methods on *yeast9Breedden2* dataset.

	Precision	Sensitivity	F-measure
time-delayed ND¹	0.2500	0.9412	0.3951
time-delayed ND²	0.2857	0.8235	0.4242
time-delayed ND³	0.3250	0.7647	0.4561
TD-ARACNE	0.3333	0.2353	0.2759
TD-LASSO	0.3095	0.7647	0.4407
MMHODBN	0.2222	0.1176	0.1538

4.5.a Inferred by time-delayed ND

4.5.b Inferred by TD-ARACNE

4.5.c Inferred by TD-LASSO

4.5.d Inferred by MMHODBN

Figure 4.5: Inferred networks on *yeast5on* by four methods of time-delayed GRN inference.

4.6.a Inferred by time-delayed ND

4.6.b Inferred by TD-ARACNE

4.6.c Inferred by TD-LASSO

4.6.d Inferred by MMHODBN

Figure 4.6: Inferred networks on *yeast5off* by four methods of time-delayed GRN inference.

4.7.a Inferred by time-delayed ND

4.7.b Inferred by TD-ARACNE

4.7.c Inferred by TD-LASSO

4.7.d Inferred by MMHODBN

Figure 4.7: Inferred networks on *yeast9Spellman* by four methods of time-delayed GRN inference.

4.3.3 Results of Time-Delay Inference

In this section, we analyze the inferred delays by our methods (i.e. time-delayed ND) and validate the results using literature. We could not provide the comparison between time-delayed ND and other methods on the inference of time delays, because the results of time delays from other methods are either unavailable (e.g., TDARACNE and MMHODBN) or based on the same technique as our method (TDLASSO also uses cross-correlation).

Using the *yeast9* network (i.e. Figure 4.3.b) as the benchmark network, we first extract the information of time delay for each edge in this network. According to [200], the nine genes are active at different phases (G1, S, G2 or M) during cell cycle. The delays of interactions among the nine genes can be estimated by the time intervals of those phases. The cycle time of budding yeast is about 81 minutes, where the lengths of G1, S and G2/M are 21 minutes, 31 minutes and 28 minutes respectively [204]. We applied time-delayed ND on two yeast cell cycle datasets, namely *Spellman* data and *Breeden* data (*yeast9Breeden1*). The maximal value of time delay was set to 5 (time points). Results of inferring time delays by time-delayed ND are shown in Table 4.6, where the interactions in the first column are edges in the benchmark network, and the information about gene interactions in different phases (i.e. the second column) is from [200]. Note that the predicted time delays are measured in the numbers of time lags. We highlight (in parenthesis) the inferred delays that may be “wrong”. As shown in Table 4.6, our method (i.e. time-delayed ND) can infer time delays of gene interaction reasonably well.

Table 4.6: Validating the inferred time delays on *Spellman* data and *Breeden* data

Interactions	Phases	Predicted delays ¹	Predicted delays ²
ACE2 $\rightarrow$ CLN3	M $\rightarrow$ G1	2	3
CLN3 $\rightarrow$ MBP1	G1 $\rightarrow$ post G1	3	(0)
CLN3 $\rightarrow$ SWI4	G1 $\rightarrow$ post G1	3	2
FKH2 $\rightarrow$ SWI5	G2 $\rightarrow$ M	(5)	0
FKH2 $\rightarrow$ ACE2	G2 $\rightarrow$ M	2	0
MBP1 $\rightarrow$ NDD1	G1 $\rightarrow$ G2	3	(1)
MBP1 $\rightarrow$ SWI4	G1 $\rightarrow$ G1	0	2
MCM1 $\rightarrow$ ACE2	G2 $\rightarrow$ M	(5)	(5)
MCM1 $\rightarrow$ CLN3	M $\rightarrow$ G1	1	(0)
MCM1 $\rightarrow$ SWI4	M $\rightarrow$ G1	4	2
MCM1 $\rightarrow$ SWI5	G2 $\rightarrow$ M	4	0
NDD1 $\rightarrow$ ACE2	G2 $\rightarrow$ M	(5)	(4)
NDD1 $\rightarrow$ SWI5	G2 $\rightarrow$ M	(4)	(4)
SWI4 $\rightarrow$ NDD1	G1 $\rightarrow$ G2	4	3
SWI5 $\rightarrow$ CLN3	M $\rightarrow$ G1	2	4
SWI6 $\rightarrow$ NDD1	G1 $\rightarrow$ G2	3	5
SWI6 $\rightarrow$ SWI4	G1 $\rightarrow$ G1	(4)	1

Predicted delays^{1,2} are from *Spellman* and *Breeden* data respectively.

4.4 Conclusions and Discussion

In this chapter, we proposed a method named time-delayed ND to infer time-delayed gene interactions based on cross-correlation and network deconvolution. We first inferred the probable time delays for the interactions between each target gene and its potential regulators, using cross-correlation. Then based on the inferred time delays, we aligned the time samples for each target gene, and calculated the correlations of genes based on the aligned samples. After that, we employed the ND algorithm on the correlation matrix to identify direct interactions between each target gene and its regulators. The performance of time-delayed ND has been evaluated on three real-life gene expression datasets. Compared with three other methods for inferring time-delayed GRNs, our method achieved overall better performance (in terms of F-measure).

The ND algorithm is a nonlinear filter which could be applied to any symmetric (and some asymmetric) matrix (which represents a network) to filter out the indirect edges. Using correlations in the original ND method, only an undirected network could be inferred. In comparison, our method is able to infer directed networks. In the proposed method, the correlations are calculated between each target gene and the rest of genes. The correlation matrix also includes correlations between all the other genes using the time samples aligned based on the inferred time delays between these genes and the target genes (see Algorithms 4.1 and 4.2). ND is then applied to such a correlation matrix, i.e., the filter for indirect edges is determined using correlations not only between a target gene and the rest of genes but also among the rest of genes. This step helps in the determination of regulatory directions while considering redundancy in the possible regulators.

In the proposed method, the networks are inferred with a *priori* determined maximum time lag r and a threshold θ . Given the short time series, increasing maximum possible time lag will reduce the available number of time points to compute the correlations. This may ultimately result in an increase in the numbers of false predictions. Hence, we restrict the maximum possible time lag to 5 in our experiments. An increase in the value of θ means increasing the cut-off to infer a regulatory edge. Generally, this increase will result in only partial recovery of true positive edges. Hence in our experiments, we have kept the value of θ to a moderate level so that most of the true positives edges could be recovered while keeping the false positives in check.

Chapter 5

Mining Regulatory Patterns between Histone Modifications and Gene Expression

5.1 Introduction

5.1.1 Background

In the nucleus of a eukaryotic cell, DNA double helix is wrapped around the conserved proteins named histone in the way that a segment of approximately 147bp DNA binds to an octamer of four histones to form a nucleosome, which is the fundamental subunit of chromatin. The N-terminal tails of histones protrude out of the nucleosome core and are subject to various types of post-translational modifications such as methylation and acetylation. The modifications of histone tails could affect the accessibility of genes to the transcription-related proteins such as transcription factors. Histone modifications can mediate gene expression through direct change of chromatin structure or by recruiting effector proteins [19]. It has been observed that histone modifications often regulate gene expression in a cooperative manner. Therefore, a hypothesis of “histone code” was proposed, which suggests that a certain combination of histone modifications corresponds to a unique chromatin state which causes distinct effect on cellular activities such as transcription [17]. Understanding the histone code of transcriptional regulation is of great importance in molecular biology. Since the “histone code” hypothesis was proposed, it has been under intense research [205, 206, 207, 143, 142]. Nevertheless, the existence of histone code remains an open question. On the other hand, the availability

of related high-throughput data such as microarray, ChIP-chip and ChIP-seq provides an opportunity to decipher the histone code by large-scale data mining and modeling.

5.1.2 Motivation

A lot of approaches have been proposed to reveal the regulatory relationship between epigenetic modifications and gene expression in recent years (see Section 2.3). However, most of these methods produced the relationships between histone modifications and gene expression with a “black box” model which makes it difficult for human to understand and interpret the underlying physical processes and biological meanings. Another limitation of these methods is that they did not use time course data of histone modifications so that they cannot capture the dynamic relationships between histone modifications and transcription.

Here, we propose a framework based on association rule mining to explore the regulatory patterns between histone modifications and transcription levels. Our method provides a straightforward strategy to obtain easy-to-understand rules which describe how combinations of histone modifications regulate gene expression. The rules output by our method are statistically significant. Some of the rules are supported by literature of experimental results. Other rules which are *de novo* discovered can guide further research in epigenetic regulation of transcription. Based on our association rules, we build a model to predict gene expression, which outperforms a published Bayesian network model for predicting gene expression based on histone modifications. Our findings reveal that histone modifications regulate transcription on a large scale. To our knowledge, this is one of the first computational methods to explore “histone code” on transcription, which takes into account the dynamic changes of both gene expression and histone modifications over time points.

5.2 Data

In this work, we aim to study the transcriptional regulation of the most deadly malaria parasite, *Plasmodium falciparum*, which claims about a million human deaths worldwide each year [208]. Despite intense research for decades, the mechanism of gene regulation in this species remains a mystery. *Plasmodium* genomes are known for AT-richness and lack of transcription factors, and its special cyclic patterns of gene expression along its life cycle [209, 210]. So far there is not much evidence of DNA methylation in *Plasmodium*, hence histone modifications are most likely to be critical for transcriptional regulation in this species [211, 212, 213].

The input data to our analysis are the transcriptional levels and histone modification enrichment profiles across the genome of *P. falciparum* from [212] and [214] (with

GEO accession number GSE39238). The transcriptional levels were obtained from long oligonucleotides based microarray experiments and the histone modification profiles were from ChIP-on-chip (or ChIP-chip) experiments. The dataset includes 14 variables: gene expression (denoted as cDNA, RNA or mRNA) and 13 histone modification marks, which are H4K5ac, H4K8ac, H4K12ac, H4K16ac, H4ac4, H3K9ac, H3K14ac, H3K56ac, H4K20me1, H4K20me3, H3K4me3, H3K79me3, and H4R3me2. The data are time-series, with six time points for each variable, which show the variation of transcription and histone modification profiles along 48 hours (six time points, with an interval of 8 hours) of *P. falciparum* intra-erythrocytic developmental cycle (IDC). The whole dataset can be viewed as a $N \times M$ matrix, where N is the number of probes (several probes map to one gene), and $M = T \times V$ where T is the number of time points and V is the number of variables (here $T = 6, V = 14$). Figure 5.1 depicts a heat map of gene expression (cDNA) and histone modification profiles of *Plasmodium*. In this heat map, each row represents one probe (i.e. gene). Each variable (i.e. cDNA, H4K5ac, etc) is represented by a submatrix consisting of six consecutive columns corresponding to the six time points. Note that the rows (i.e. probes or genes) in this heat map are sorted according to the phase values calculated from the gene expression levels of the six time points. The phase value indicates the timing of maximal expression. For more information about the phase value, please refer to [210]. The red-and-green stripe which separates every two neighboring variables shows the phase values, which range from -3.14 (green) to 3.14 (red), or from $-\pi$ to π .

Since it is a microarray dataset, missing values are quite commonly distributed. The distribution of missing values in this dataset is shown in Figure 5.2, where the vertical axis is the number of probes, ‘NA’ means the values are missing for all six time points, ‘blank’ means there are 2~5 missing values (note that the single missing values among the six time points have been imputed). Here ‘NA’ and ‘blank’ are treated equally as missing data.

5.3 Methods

5.3.1 Approach Overview

We propose a simple method based on association rule mining to uncover the regulatory relations between histone modifications and gene expression. Through proper transformation, we turn our problem into a task of association rule mining, where the set of interesting strong rules would be output as the “histone code”. As a data mining technique, association rule mining can discover the relationship between histone modifications and gene expression directly from the transcription data and histone modification profiles, and thus provides a straightforward way to decipher the “histone code” without

Figure 5.1: Heat-map of transcription levels (cDNA) and histone modification profiles of *P. falciparum*.

Figure 5.2: Distributions of missing data in mRNA and different HMs.

any assumption of models and the tuning of parameters. After obtaining rules with high *support* and *confidence*, we calculate the p-value for each rule and apply permutation-based multiple testing to adjust the p-values. Finally, rules with high adjusted p-values are output as significant rules. We have verified our significant rules using literature, and also demonstrated that these rules are robust to bootstrapping (i.e. changing distribution

in data items) and are not obtained by chance. Moreover, experiments showed that, the prediction of gene expression based on our rules is more consistent with observed gene expression than the prediction from a published Bayesian network model.

5.3.2 Association Rule Mining

We apply association rule mining to explore the relations between combinatorial histone modifications and gene expression. As a well-studied technique of data mining, association rule mining was firstly proposed to discover the regularities among products in large transaction database of supermarkets [215]. It has also been applied to problems in Bioinformatics [216, 217, 218]. Results from these research show that association rule mining is a promising technique in biological data analysis. Here we introduce some basics of association rule mining. More details of association rule mining can be found in reviews [219, 220].

Association rule mining was firstly proposed to discover interesting relations between variables in large transaction database for market basket analysis. A transaction database is a collection of records each of which is composed of binary values which denote the occurrence of variables, where ‘1’ means the corresponding variable occurs in this transaction while ‘0’ means not. Let $I = \{i_1, i_2, \dots, i_n\}$ be a set of binary attributes called *items*, $TD = \{td_1, td_2, \dots, td_m\}$ be a set of transactions (i.e. records) called *transaction database*. Each record in TD is assigned with a unique transaction id and contains a subset of items in I , which are those with values equal to 1 in this transaction. In real case such as supermarket sales data, the number of items could be thousands and the number of transactions in database could be millions or more. Association rule mining was designed for such large-scale data analysis. As this technique become more and more sophisticated, it has been used in many other applications such as web usage mining and Bioinformatics.

An association rule is represented in the form $B \Rightarrow A$, where A and B are item-sets (or sets of items, i.e. $A, B \subseteq I$), and $A \cap B = \emptyset$. An interesting association rule $B \Rightarrow A$ means that the itemset B occurs frequently in the database and when B occurs A also appears most of the time. Two concepts *support* and *confidence* are employed the most often as constraints for the selection of interesting rules from all possible combinations of items or rules. The *support* of an itemset A , denoted as $supp(A)$, is defined as the proportion of transactions in the dataset which contain the itemset A (or A occurs in these transactions). Let N be the number of all transactions in the database and N_A be the number of transactions which contain A , then $supp(A) = N_A/N$. The *support* of itemset A is used to describe how frequent A happens among the transactions. The *confidence* of a rule is defined as $conf(B \Rightarrow A) = supp(A \cup B)/supp(B)$, which is the proportion of the *support* of A and B to the *support* of B . Let $N_{A \cup B}$ be the number of

transactions which contain both A and B , and N_B be the number of transactions which contain B , then $\text{conf}(B \Rightarrow A) = N_{A \cup B} / N_B$. The concept *confidence* measures how often A appears when B already happens. The idea of *confidence* is the estimation of the conditional probability $P(A|B)$, the probability of finding itemset A in transactions under the condition that those transactions also contain B . In order to identify interesting rules, *support* and *confidence* must each be higher than a lower bound, where the lower bound thresholds are specified by the users. Association rule mining mainly consists of two steps: (1) retrieve all frequent itemsets with *support* higher than a threshold in the database; (2) from the frequent itemsets found in the previous step, form rules with *confidence* larger than a threshold.

The main challenge of association rule mining is to obtain all frequent itemsets from all possible item combinations, the size of which would grow exponentially with the number of items in I . Let n be the number of items in I , then the number of all possible item combinations is $2^n - 1$, where the empty set which includes no items is not included. So when n is large, say thousands, the time complexity of association rule mining would be very high. Fortunately, there is a property called “*downward-closure property*” (also named “*anti-monotonicity*”) which enable efficient search for frequent itemsets. This property guarantees that for an infrequent itemset, its all supersets must also be infrequent. The downward-closure property can be used to excluding lots of infrequent itemsets in the $2^n - 1$ number of item combinations. This property has been used in many association rule mining algorithms such as Apriori [221] and Eclat [222]. The efficiency of association rule mining is further improved by a method named “*Frequent pattern growth*” (*FP-growth*) [223]. This method compresses the original transaction database into a highly compact structure called *FP-tree*. Then it finds frequent itemsets by frequent pattern growth instead of costly candidate generation, and thus raises the search efficiency substantially. LCM (short for Linear time Closed item set Miner) is yet another efficient algorithm which was proposed by T. Uno *et al.* in 2003 [224]. Based on a parent-child relationship that is defined on frequent closed itemsets, they construct a tree which is composed by only frequent closed item sets. Then by traversing the tree in a depth-first manner, LCM can find all frequent closed itemsets in polynomial time per item set. In this work, we apply LCM to retrieve all rules with high *support* and *confidence* from the data of gene expression and histone modifications in *Plasmodium falciparum*.

5.3.3 Problem Formulation

Before applying association rule mining to the data, we need to discretize the continuous data of gene expression and HMs enrichment first. Here we discretize the values for each gene into three states: -1, 0, 1, which represent down-regulate ($\downarrow$), unchanged ($-$),

upregulate ($\uparrow$) respectively. We utilize an “equal-width binning” technique to discretize the data. The discretization is carried out among the six time points for each variable of each gene. So the discrete values represent the variation of the corresponding variable among the six time points.

After discretization, each variable would have three states, each of which corresponds to an attribute of this variable. We use a unique integer to represent the state of each variable, as shown in Table 5.1. By treating each state of each variable as an item, we transform the original dataset into a transactional dataset, with each gene as one transaction. If the total number of variables in the original dataset is n , there would be $3 \times n$ items (or attributes) in the transaction dataset. But since the three states of each variable are mutually exclusive, there would be at most n items in a transaction.

Table 5.1: Integers corresponding to the events of changes in gene expression and histone modifications

Variables	Attributes	Integers	Variables	Attributes	Integers
Gene Expr.	$\downarrow, -, \uparrow$	1,2,3	H3K14ac	$\downarrow, -, \uparrow$	22,23,24
H4K5ac	$\downarrow, -, \uparrow$	4,5,6	H3K56ac	$\downarrow, -, \uparrow$	25,26,27
H4K8ac	$\downarrow, -, \uparrow$	7,8,9	H4K20me1	$\downarrow, -, \uparrow$	28,29,30
H4K12ac	$\downarrow, -, \uparrow$	10,11,12	H4K20me3	$\downarrow, -, \uparrow$	31,32,33
H4K16ac	$\downarrow, -, \uparrow$	13,14,15	H3K4me3	$\downarrow, -, \uparrow$	34,35,36
H4ac4	$\downarrow, -, \uparrow$	16,17,18	H3K79me3	$\downarrow, -, \uparrow$	37,38,39
H3K9ac	$\downarrow, -, \uparrow$	19,20,21	H4R3me2	$\downarrow, -, \uparrow$	40,42,42

Let the data matrix $D = [d_{i,j}]_{n \times m}$ represent the values of the n variables on the m time points, and let V be a set of variables $\{v_1, v_2, \dots, v_n\}$, and T be a set of consecutive time points $\{t_0, t_1, \dots, t_{m-1}\}$, then $d_{i,j}$ denotes the value of variable v_i at time point t_j . The data D can be transformed to a transaction dataset TD using the method described above. Let $I = \{i_1, i_2, \dots, i_n\}$ be a set of attributes (called *items*) in TD . For a transaction dataset TD , we define a rule as $R = \langle (I_1, I_2), F, C \rangle$, where I_1 and I_2 are subsets of I , and $I_1 \cap I_2 = \emptyset$; F is the frequency of itemsets I_1 appearing in TD , C is the ratio of the transactions containing both I_1 and I_2 to the transactions which contains I_1 . So R is equivalent to the rule $I_1 \Rightarrow I_2$ with *support* F and *confidence* C . Here, for the data of *P. falciparum*, since we want to discover the relations between transcription level and histone modifications (HMs), I_2 would include only the attributes corresponding to gene expression (GE), and I_1 could be any combinations of the attributes corresponding to the 13 HMs. Therefore, our rules would be in this form: $f(HMs^{attr}) \Rightarrow GE^{attr}$, where GE^{attr} represents the attributes corresponding to GE, $f(HMs^{attr})$ denotes the combinations of HMs attributes. Since we aim to explore the effect of HMs to transcription, it is natural

to fine rules in this form $f(HMs^{attr})\Big|_{t=t_i} \Rightarrow GE^{attr}\Big|_{t=t_{i+1}}$. However, we still pursue rules like $f(HMs^{attr})\Big|_{t=t_i} \Rightarrow GE^{attr}\Big|_{t=t_i}$ because in our dataset $t_{i+1} - t_i = 8$ hours. The interval between two consecutive time points may be too long for the effect of a particular HM event to last. Therefore, here we assume the HMs state still exists when its effect on transcription appears, which is reasonable because HMs are relatively stable albeit dynamic.

In a transaction database, if one itemset A occurs often enough, and when A occurs another itemset B which has no overlap with A also happens most of the time, we then decide that A implies B with significant support. Note that “implies” here does not necessarily mean “causal”. We restrict the left-hand side to be HMs attributes and the right-hand side to be GE attributes, and since biologically HMs regulate RNA expression, we can assume for simplicity that the frequent co-occurrence of HMs with a state of GE implicates “histone code”. Therefore, if for a significant number of genes there exists a group of histone modifications events, which coincides frequently with a particular state of gene expression, then we conclude that it is a “histone code”. In this way, the problem of deciphering histone code can be achieved by association rule mining. Here we use LCM [224] to mine all frequent closed item sets which have *support* larger than a pre-set threshold $supp_{min}$, and then obtain all interesting rules with *confidence* higher than our pre-set threshold $conf_{min}$. These rules would reveal regulatory relations between histone modifications and gene expression.

5.3.4 P-values and Multiple-testing Correction

Using fixed thresholds of *support* and *confidence* to find interesting rule may be a bit arbitrary, because the selection of such thresholds might depend on the data. Thus we calculate p-values to assess the significance of rules. For a rule $R: f(HMs) \Rightarrow GE$, which stands for $f(HMs^{attr}) \Rightarrow GE^{attr}$, its p-value is the probability of observing a rule more extreme (which means with higher *support* or *confidence*) than R under the null hypothesis that the $f(HMs)$ and gene expression are independent (i.e. the $f(HMs)$ imply the three possible changes of gene expression with equal probabilities). Here we used one-tailed Fisher’s exact test [225, 226] to calculate the p-value of R . Let $s_R = supp(R)$ be the support of the rule R , and $s_M = supp(HMs)$ be the support of the combinatorial HMs (i.e. the LHS of the rule). Then the p-value of R is given by:

$$\begin{aligned} p(R) &= p(s_R; n, n_g, s_M) \\ &= \sum_{k \leq s_M - s_R} H(k; n, n_g, n - s_M) \\ &= \sum_{k \leq s_M - s_R} \frac{\binom{n_g}{k} \binom{n - n_g}{s_M - k}}{\binom{n - s_M}{s_M}}, \end{aligned} \tag{5.1}$$

where n is the total number of records in the dataset, n_g is the number of records that contain the item g , and H is the hypergeometric distribution. Note that here $s_M = \text{supp}(HMs)$ is the number (not proportion) of transactions which contain HMs. The value of $s_R = \text{supp}(R)$ can be calculated as: $\text{supp}(R) = \text{confidence} * \text{supp}(HMs)$.

Since the number of rules generating in the previous step could be thousands, we might expect many “false positives” if we use a standard cut-off (e.g. 0.05) for p-values. Therefore, we resort to multiple-testing correction techniques to cope with this problem. Two measures are often used to control false positive errors, namely family-wise error rate (FWER) and false discovery rate (FDR). FWER is the probability of getting at least one false positive. FDR is the proportion of false positives among the set of rejected hypotheses (i.e. the rules which are treated as statistically significant) [227]. FWER is a strict criterion and is often used when we try to get definitive results. On the other hand, FDR is preferred for exploratory studies, where the goal is to control the proportion of false positives among the reported “positives”. Here we choose FDR to control false positives in our study.

We adopt a permutation-based approach [227] to adjust the p-values of the rules calculated as above. The permutation-based approach randomly shuffles the class labels of items, which can destroy the association between the patterns and class labels (which are the two sides of rules). Then the p-values after the random shuffling can approximate the null distribution in which the two sides of the rules are independent. Because the permutation approach preserves the relationships among patterns (e.g. sub-patterns and super-patterns), it is more powerful (than other approaches such as Benjamini-Hochberg’s method) to detect the true association rules [227]. First we randomly generate N permutations for each rule by randomly shuffling the class labels of items (i.e. the states of gene expression). Let N_0 be the number of rules we have generated from the original dataset, and $RD = \{p_1, p_2, \dots, p_{N \times N_0}\}$ be the p-values of the N_0 rules with N permutations. For each rule, we re-calculate its p-value as follows: let p be the p-value of this rule, then the new p-value is

$$p_{new} = \frac{|\{p_i | p_i \leq p, p_i \in RD\}|}{N \times N_0}, \quad (5.2)$$

where the numerator is the number of elements in RD which are smaller than or equal to p .

5.3.5 Pseudo Code of Histone Code Mining

The process of our method for mining histone code is summarized in Algorithm 5.1.

Algorithm 5.1. Mining histone code from time-series data of gene expression and histone modifications

INPUT: A dataset D with m variables and n time points

Steps:

- 1: Discretize D if necessary, and transform D into a 3-symbol matrix D_{sbl3} ;
- 2: Convert D_{sbl3} into a transaction dataset TD using attributes of variables as items;
- 3: Apply LCM on TD and obtain all frequent closed itemsets with $support\ supp \geq supp_{min}$, where $supp_{min}$ is a pre-set threshold for $support$;
- 4: Form rules with $confidence\ conf \geq conf_{min}$ from the frequent closed itemsets, where $conf_{min}$ is a pre-set threshold for $confidence$;
- 5: **For all** rules obtained in step 4 **do**
- 6: Calculate their p-values using

$$p(R) = \sum_{k \leq supp(HMs) - supp(R)} H(k; n, n_g, n - supp(HMs)) \quad (5.3)$$

7: **end for**

8: **For all** p-values calculated in step 5~7 **do**

9: Adjust the p-values using permutation-based multiple testing:

$$Adj(p) = \frac{|\{p_i | p_i \leq p, p_i \in RD\}|}{N \times N_0}. \quad (5.4)$$

10: **end for**

11: **Output** the significant rules with adjusted p-values larger than the threshold PV_{min} .

5.3.6 Gene Expression Prediction Based on Rules

Based on a set of rules, gene expression of each record in the dataset can be predicted as follows. Here we only consider ‘up-rules’ (i.e. rules with GE $\uparrow$) and ‘down-rules’ (i.e. rules with GE $\downarrow$), because no ‘flat-rules’ (i.e. rules with GE $-$) are significant in our dataset. For each record (i.e. a gene) in the dataset, we define an up-score S_u and a down-score S_d , which represent how strongly this record is supported by up-rules and down-rules respectively. To calculate S_u , let A be the occurrence vector of up-rules: $A = [a_1, a_2, \dots, a_n]^T$, where n is the total number of rules up-rules we have obtained from the whole dataset, and a_i ($1 \leq i \leq n$) equals to 1 when the i th up-rule occurs in this record, and 0 otherwise. Then a weight vector of up-rules is defined as follows: $W = [w_1, w_2, \dots, w_n]$, where w_i ($1 \leq i \leq n$) equals to the confidence of the i th up-rule.

Then, the up-score S_u is calculated using $S_u = W * A$. The down-score S_d of this record is calculated similarly. After that, gene expression can be predicted in this way: When the up-score of one record is larger than its down-score, its corresponding gene expression is predicted as ‘up-regulated’; otherwise, its gene expression is predicted as ‘down-regulated’.

For each gene we classify its original gene expression levels of six time points into six groups: $GE = 1, GE = 2, \dots, GE = 6$, which correspond to the lowest, second lowest, ..., largest expression value of the gene among the six time points. For each of the six groups, we calculate the average of the observed gene expression and normalize them into the range between 0 and 1. The normalized average expression value is regarded as the observed gene expression level of each of the six groups. For the predicted gene expression, we calculate the percentage of genes which are predicted as ‘up-regulated’ for each of the six groups, and take this percentage value as the predicted gene expression level of this group.

5.4 Experimental Results

After transformation, the data of *P. falciparum* turn into a transaction dataset with 15028 transactions for each time point. Since we aim to find the global histone code among all the six time points, we put the six transaction datasets together to form one large dataset, which has $15028 \times 6 = 90168$ transactions. Applying our method on this large transaction dataset of *P. falciparum*, we obtained 2015 rules with *support* and *confidence* larger than 1000 and 0.4 respectively. After p-values calculation and correction, we extract 60 rules by applying a threshold 0.01 on the corrected p-values.

5.4.1 Rules of HMs and Gene Expression

From the 60 rules which have adjusted p-values less than 0.01, we select those having both directions in regulating gene expression and show them in Table 5.2. For example, the rule “H3K56ac $\uparrow \Rightarrow$ GE $\uparrow$ ” is included, if and only if “H3K56ac $\downarrow \Rightarrow$ GE $\downarrow$ ” also exists in the 60 rules. We did this selection because bi-directions would give more certainty to the rules about the role of combinatorial HMs in regulating gene expression

Table 5.2: Association rules of HMs and gene expression (GE)

Rules	Support	Conf.	P-values
H3K56ac $\uparrow \Rightarrow$ GE $\uparrow$	22108	0.4314	0.000353
H3K56ac $\downarrow \Rightarrow$ GE $\downarrow$	22115	0.4287	0.000938
H4K8ac $\downarrow \Rightarrow$ GE $\downarrow$	12538	0.4471	0.000614

H4K8ac $\uparrow \Rightarrow$ GE $\uparrow$	12409	0.4435	0.000672
H3K4me3 $\downarrow \Rightarrow$ GE $\downarrow$	11699	0.4462	0.001265
H3K4me3 $\uparrow \Rightarrow$ GE $\uparrow$	11648	0.4414	0.00207
H3K9ac $\uparrow \Rightarrow$ GE $\uparrow$	16543	0.4328	0.001355
H3K9ac $\downarrow \Rightarrow$ GE $\downarrow$	16167	0.4332	0.00307
H4K20me1 $\downarrow \Rightarrow$ GE $\downarrow$	16003	0.4288	0.00398
H4K20me1 $\uparrow \Rightarrow$ GE $\uparrow$	16079	0.425	0.005183
H3K9ac $\uparrow$, H3K56ac $\uparrow \Rightarrow$ GE $\uparrow$	9062	0.4542	0.001008
H3K9ac $\downarrow$, H3K56ac $\downarrow \Rightarrow$ GE $\downarrow$	8927	0.4527	0.002151
H4R3me2 $\uparrow$, H3K56ac $\downarrow \Rightarrow$ GE $\downarrow$	6634	0.4543	0.004927
H4R3me2 $\downarrow$, H3K56ac $\uparrow \Rightarrow$ GE $\uparrow$	6260	0.4594	0.002766
H4K20me1 $\downarrow$, H3K56ac $\downarrow \Rightarrow$ GE $\downarrow$	5764	0.4585	0.00584
H4K20me1 $\uparrow$, H3K56ac $\uparrow \Rightarrow$ GE $\uparrow$	5922	0.4589	0.004159
H4K20me1 $\downarrow$, H4K5ac $\uparrow \Rightarrow$ GE $\downarrow$	5559	0.4588	0.00617
H4K20me1 $\uparrow$, H4K5ac $\downarrow \Rightarrow$ GE $\uparrow$	5094	0.4628	0.004852
H3K56ac $\downarrow$, H4K5ac $\uparrow \Rightarrow$ GE $\downarrow$	5030	0.4811	0.001549
H3K56ac $\uparrow$, H4K5ac $\downarrow \Rightarrow$ GE $\uparrow$	4468	0.4836	0.001719
H3K9ac $\downarrow$, H4R3me2 $\uparrow \Rightarrow$ GE $\downarrow$	4692	0.4641	0.007133
H3K9ac $\uparrow$, H4R3me2 $\downarrow \Rightarrow$ GE $\uparrow$	4887	0.4585	0.006956
H3K4me3 $\downarrow$, H3K56ac $\downarrow \Rightarrow$ GE $\downarrow$	4271	0.4874	0.001855
H3K4me3 $\uparrow$, H3K56ac $\uparrow \Rightarrow$ GE $\uparrow$	4173	0.468	0.005847
H3K4me3 $\downarrow$, H4K5ac $\uparrow \Rightarrow$ GE $\downarrow$	3930	0.4666	0.009993
H3K4me3 $\uparrow$, H4K5ac $\downarrow \Rightarrow$ GE $\uparrow$	3593	0.4823	0.003789
H3K4me3 $\downarrow$, H4K8ac $\downarrow \Rightarrow$ GE $\downarrow$	3825	0.5071	0.000309
H3K4me3 $\uparrow$, H4K8ac $\uparrow \Rightarrow$ GE $\uparrow$	3794	0.4815	0.003444
H4K20me1 $\downarrow$, H3K9ac $\downarrow \Rightarrow$ GE $\downarrow$	3840	0.4731	0.006805
H4K20me1 $\uparrow$, H3K9ac $\uparrow \Rightarrow$ GE $\uparrow$	4098	0.4729	0.004502
H3K9ac $\downarrow$, H4K5ac $\uparrow \Rightarrow$ GE $\downarrow$	3776	0.4907	0.002756
H3K9ac $\uparrow$, H4K5ac $\downarrow \Rightarrow$ GE $\uparrow$	3626	0.4884	0.002425
H3K56ac $\downarrow$, H4K12ac $\uparrow \Rightarrow$ GE $\downarrow$	3760	0.4702	0.008417
H3K56ac $\uparrow$, H4K12ac $\downarrow \Rightarrow$ GE $\uparrow$	3999	0.4693	0.006235
H3K4me3 $\downarrow$, H3K9ac $\downarrow \Rightarrow$ GE $\downarrow$	3436	0.4921	0.00338
H3K4me3 $\uparrow$, H3K9ac $\uparrow \Rightarrow$ GE $\uparrow$	3394	0.4675	0.009406
H4K8ac $\downarrow$, H4K5ac $\uparrow \Rightarrow$ GE $\downarrow$	2688	0.5111	0.002441
H4K8ac $\uparrow$, H4K5ac $\downarrow \Rightarrow$ GE $\uparrow$	2396	0.4974	0.005507
H3K9ac $\downarrow$, H3K56ac $\downarrow$, H4K5ac $\uparrow \Rightarrow$ GE $\downarrow$	1798	0.5189	0.005544
H3K9ac $\uparrow$, H3K56ac $\uparrow$, H4K5ac $\downarrow \Rightarrow$ GE $\uparrow$	1705	0.5026	0.009744

First let us look at the relations between a single HM and gene expression (i.e. the first

ten rules in Table 5.2). These rules show that H3K56ac, H4K8ac, H3K4me3, H3K9ac and H4K20me1 are positively correlated with gene expression globally with statistical significance. Among the five HMs, H4K20me1 and H3K4me3 are histone methylation marks that are well known as activating histone modification marks [14, 19, 15, 228], which is also confirmed by [30] and [142] with computational methods. Our method on *Plasmodium* data also show the two HMs are strongly correlated with gene activation. The other three HMs are histone acetylation marks, which are often observed to be associated with gene activation since acetylation removes the positive charges on histones and relaxes the affinity between histones and DNA. However, only H3K56ac, H4K8ac and H3K9ac are found to be significantly linked to transcription activation in our dataset. Although other histone acetylation marks such as H4K16ac and H3K14ac may also have activating trends, their trends are not significant enough to be detected by our approach.

We have observed three special HMs from our rules: H4K5ac, H4K12ac, and H4R3me2. H4K5ac and H4K12ac as histone acetylation marks are generally linked to gene activation. However in our dataset, they are observed to be negatively correlated with gene expression (in the rules with multiple HMs). Similar observations have been found in literature. H4K12ac was observed negatively correlated with gene expression in yeast [229], which was confirmed in *P. falciparum* [211]. Although H4K5ac is generally considered as a mark for activating transcription which was tested in many species such as human [14], it has a clear sign of globally transcription suppression in our dataset of *P. falciparum*. It is therefore interesting to test if H4K5ac and H4K12ac are transcriptional suppressors (maybe by cooperating with other factors) in *P. falciparum*, in contrast to other species. In addition, our method discovers that H4R3me2 is negatively associated with gene expression (in the rules with multiple HMs). H4R3me2 was revealed to globally repress gene expression by [142]. Hence it is worth further study to confirm the role of H4R3me2 in the gene regulation of *P. falciparum*.

The ability to discover rules with multiple HMs is one of the main advantages of our method. This kind of rules shows how the combinations of HMs cooperatively regulate gene expression, as a clear sign of histone code on transcription. However, compared to the rules with single HMs, the experimental support of such rules can hardly be found in the literature. To compare these multiple-HMs rules with causal relations, we construct a Bayesian network as in [30] based on our dataset, see Figure 5.3. Red edges are compelled edges, which retain their directions in all equivalent DAGs (directed acyclic graphs). Green edges are reversible edges, which may change their directionality among different DAGs in the same equivalence class. According to [230], compelled edges denote causal relationships, whereas reversible edges might be just correlation but not necessarily causality. In Figure 5.3, green edges all occur among the nodes on the top level. Here we only focus on the red edges. From Figure 5.3, we can see that H4K5ac, H3K56ac, and H3K4me3 have direct causal relations with mRNA expression,

which implies the three HMs are closely related with gene expression. Note that almost all the association rules with multiple HMs in Table 5.2 include at least one of the three HMs: H3K56ac, H4K5ac and H3K4me3. This indicates that the rules we retrieved reveal some causal relations between histone modifications and gene expression. Some, but not all, rules include HMs with causal relations inferred by the Bayesian network. These HMs with causal relations are “H4K20me1 and H4K5ac”, “H3K4me3 and H4K8ac”, “H3K56ac and H4K12ac” and “H4K8ac and H4K5ac”. This suggests that HMs could regulate gene expression cooperatively.

Figure 5.3: Bayesian network of HMs and gene expression, where red edges are compelled edges, and green edges are reversible edges.

5.4.2 Robustness of Rules

The rules we discovered might be biased by the missing data (see Figure 5.2). Because missing data would lead to different percentages of different features (e.g. HMs) in the records, rules consisting of the HMs with more missing data would have lower support and hence less significant p-values. To test if our rules are robust to the different percentages of HMs in the records, we apply bootstrapping to the original dataset. First we randomly generate 20 datasets from the original dataset by sampling with replacement. Each of these bootstrapping datasets contain 90,168 rows, as the original dataset (15,028probes $\times$ 6timepoints). The process of bootstrapping could bring randomization to the percentage of HMs in the records. Next we are going to test if our rules are robust to such resampling.

We apply our method on the 20 bootstrapping datasets and obtain 20 sets of significant rules respectively. From the 20 sets of significant rules, we count the occurrences

of the significant rules from the original dataset (i.e. rules in Table 5.2). As shown in Figure 5.4, most significant rules from the original dataset also appear as significant rules in the bootstrapping datasets. Then we calculate the p-values of the rules in these 20 datasets. Figure 5.5 shows the distributions of p-values of the rules in Table 5.2 from the original dataset(left box) and 20 bootstrapping datasets (right box). The difference between the two groups is not significant (p-value = 0.7709), which indicates that the re-sampling may not affect the detection of significant rules.

Figure 5.4: The frequency of the significant rules from original dataset also occurring as significant rules in the 20 bootstrapping datasets.

For comparison, we generate a set of random rules which have the same number as the true rules (i.e. rules in Table 5.2), and calculate p-values of these random rules in both the original dataset and the 20 bootstrapping datasets. The two sub-figures in Figure 5.6 show the comparisons of p-values from these two groups of rules, in the original dataset(left) and in the 20 bootstrapping datasets (right) respectively. In both original dataset and the bootstrapping datasets, true rules have p-values far more significant than the random rules, which indicate that the rules we discovered are not obtained by chance.

5.4.3 Comparison with Existing Methods

We have compared our method with Bayesian network [30], which is a pioneering approach in the field. Similar to our method, Bayesian network applies discretization on

Figure 5.5: P-values ($-\log$) distribution of the same set of rules (rules in Table 5.2) in the original dataset and bootstrapping datasets; Wilcoxon test p-value 0.7709.

data, and then it infers causality among variables. Bayesian network can output rules just as our method does. One advantage of Bayesian network is that it can infer hierarchical and causal interactions among HMs, while right now our rules only include the associations between HMs and GE. However, our method can obtain not only the rules with causal relations among HMs (which are like the rules output by Bayesian network) but also the rules in which HMs have other kind of relationships such as cooperation. It is known that HMs regulate transcription in a cooperatively way, so our method may be more suitable for the decoding of histone code. We have compared our rules with Bayesian network in Section 5.4.1. Next we will compare the performance of our method with Bayesian network in the prediction of gene expression. We have introduced the process of predicting gene expression given a set of rules in Section 5.3.6. For comparison, Bayesian network is also used to predict gene expression by applying Libra [231]. We have built a Bayesian network on our dataset, shown in Figure 5.3. The structure of Bayesian network and the discrete values of HMs are input to Libra, which would then output the predicted gene expression level for each record. Then for each of the six groups of gene expression (see Section 5.3.6), the percentage of genes which are predicted as ‘up-regulated’ is calculated and treated as the predicted gene expression level of this group.

The plot of “predicted gene expression” versus “observed gene expression” is shown in Figure 5.7, where the two straight lines are obtained by linear regression, and the six

Figure 5.6: Comparison of p-values ($-\log$) between random rules and true rules, in the original dataset and bootstrapping datasets (Wilcoxon test p-values are $2.5889\text{e-}14$ and $6.0629\text{e-}253$ respectively).

dots (blue or green) from left to right correspond to the six groups of genes with expression levels from highest to lowest respectively among the six time points (see Section 5.3.6). The detailed results of linear regression are shown in Table 5.3, where both linear regression and Pearson correlation are applied to the two vectors, i.e. the predicted and observed expression levels. As seen, our method has significantly higher R^2 than Bayesian network, which indicates that our method is able to predict gene expression more consistently. The coefficients and p-values show that our method can predict the trend of gene expression more accurately. In addition, Pearson correlation coefficients were calculated between the observed and predicted gene expression, which are 0.7407 (p-value = 0.0921) for Bayesian network and 0.9879 (p-value = 0.0002) for our method. These results indicate that our method has better performance than Bayesian network in predicting gene expression. This is probably due to the fact that our rules are selected from all possible combinations of different states of HMs and gene expression (i.e. global searching), while Bayesian network is constructed based on local structure searching.

However, it is generally difficult to compare our method with most existing methods such as regression models because they are different at the technical level. For example, although SVM has good performance in non-linear regression, it is inefficient for large-scale data analysis especially when the number of samples is very large. Our dataset

Figure 5.7: Comparison between Bayesian network and our method in predicting gene expression. Blue dots and the green line denote the prediction by Bayesian network, while red dots and red line denote the prediction by our method.

Table 5.3: Comparison between Bayesian network and our approach in predicting GE based on HMs.

		Bayesian network	Our approach
Linear regression	R^2	0.5487	0.9760
	Coefficient	0.0142	0.1432
	p-Value	0.0921	0.0002
Pearson correlation	PCC	0.7407	0.9879
	p-Value	0.0921	0.0002

includes 90, 168 records (i.e. samples), which would take a lot of time for SVM to do the training. Moreover, there are a lot of missing data in our dataset, which is not unusual for microarray data. The missing data in synchronized multi-variables dataset make it difficult for regression using SVM or Random Forests. Because there are missing values of at least one variable for almost all records, the accuracy of regression could easily be affected. In contrast, our method is relatively robust to missing data, partly because data discretization is carried out independently for each variable. Even if we are using a new dataset which is suitable for both our method and those regression models such as SVM, it is still difficult to compare these two kinds of methods. It could be unfair to compare their prediction accuracy on gene expression because our method requires data discretization which would cause loss of some information. But our method can provide results with interpretability, which those regression models lack.

5.5 Conclusions and Future Work

In this chapter, we proposed a framework based on association rule mining to discover in large-scale the regulatory relations between histone modifications and gene expression. Based on the technique of association rule mining, we designed a method to identify interesting relations by the measures of *support* and *confidence*. Then we used Fisher's exact test and multiple testing correction to calculate p-values for all the rules obtained from the previous step. A threshold for p-values (here we use the widely accepted value of 0.01) was used to select the most significant rules. Applied to the data of *P. Falciparum*, our method is able to discover rules that are consistent with literature of experimental biology. We demonstrated that our rules are not obtained by chance and they are robust to random re-sampling. Compared with Bayesian network, our model based on the significant association rules could predict gene expression more accurately and consistently.

The proposed method can generate rules with interpretable biological meanings. Driven by the power of data mining, our method is able to detect significant patterns of histone modifications regulating transcription, from all possible combinations. The direct mining from data could provide a reliable way to study and interpret the underlying biological processes. Because our method is data-driven and unsupervised, it can uncover novel regulatory rules between histone modification and gene expression to guide further biological experiments and discoveries. To the best of our knowledge, this is the first method to apply association rule mining to reveal regulatory relations between gene expression and histone modifications in *P. Falciparum*, while taking into account the dynamic epigenetic regulation of gene expression.

One interesting future work is to apply our method to data of other species, with the aim to find evolutionarily conserved regulatory patterns. Another future work is to extend simple rules with only a few histone modifications on the left hand side to incorporate hierarchical and combinatorial interactions among histone modifications.

Chapter 6

Integrating Epigenetic Prior in DBN for GRN Inference

6.1 Introduction

6.1.1 Background

In cellular environment, genes are interacting with each other to achieve their functions. The interactions among a group of genes constitute a gene regulatory network (GRN), which affects the transcription rates of genes in this network. GRNs are critical for understanding the mechanisms of cellular processes, in both normal and abnormal states. Since abnormal changes in GRNs can link to some notorious diseases such as cancers, the accurate reconstruction of GRNs could help to identify the target genes and related pathways of the diseases, and lead to the design of effective drugs. Therefore, GRN inference is a major topic in modern biology. Many approaches have been proposed in this field. Among them, Bayesian networks are widely used. First it is because the randomness of Bayesian networks can properly model the stochastic phenomena in biological processes. Other advantages of Bayesian networks include the ability to infer causal relations and the robustness to noise or missing data and so on.

GRN reconstruction is a very challenging task, partly due to that biological processes are inherently complicated, and the lack of suitable data and benchmark networks. The recent advancements in new technologies in genomics raise hope for new discoveries by bringing new data such as epigenetic data. Data from various sources should be taken together to achieve more accurate reconstruction of GRNs. The integration of epigenetic data is promising to improve GRN inference because of the close relationship between epigenetic factors and transcription.

6.1.2 Motivation

It has been shown that incorporating epigenetic data could improve GRN inference [168]. By integrating the data of histone modifications into the Bayesian inference of GRNs, Zheng *et al.* in [168] demonstrated that their method can improve the performance of GRN reconstruction significantly. However, the Bayesian model used in [168] was static, which cannot capture the feedback regulatory relations between genes. The effect of such epigenetic prior on dynamic Bayesian network (DBN) based GRN inference has not been tested yet. Furthermore, in [168], the enrichment of histone modifications for each gene was averaged across all genomic regions of this gene, which may smooth out some important information hidden in different regions. Hence, there could be better methods to encode the epigenetic knowledge into GRN inference.

In this chapter, we take a step forward and test the effect of epigenetic prior in GRN inference based on DBN. In order to take advantage of the information behind different regions of a gene, we treat two types of genomic regions differently: intragenic (or genic) and intergenic regions. Based on their distinct characteristics, we propose several new epigenetic priors and analyze their effect on GRN reconstruction using DBN. Experiments on both synthetic and real-life gene expression data of yeast cell cycle show that our method can improve the performance of GRN reconstruction by increasing the number of true positives while keeping the number of false positives under control.

6.2 Methods

In this section, we first introduce the strategy of reconstructing GRNs using DBN and Markov chain Monte Carlo (MCMC) simulation. Then we describe our method of integrating epigenetic prior into the GRN inference based on DBN and MCMC.

6.2.1 Learning GRN Structure Based on DBN and MCMC

For a set of random variables X , a Bayesian network is defined by a graphic model S , which consists of a set of nodes V and a set of directed edges E : $S = (V, E)$, where each node in V is associated with a random variable in X , and E describes the conditional dependence relations among the random variables. The structure of a Bayesian network is a direct acyclic graph (DAG), with no feedback loops in the network. Bayesian networks assume the Markov property, which means each variable in Bayesian network is only dependent on its parents. Therefore, given the structure S , the joint probability distribution for $X = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N\}$ is given by [232]:

$$p(\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N) = \prod_{i=1}^N p(\mathbf{x}_i | \mathbf{a}_i), \quad (6.1)$$

where $\mathbf{a}_i$ denotes the set of parents of the variable $\mathbf{x}_i$ in the structure S . Bayesian networks have many advantages, such as the ability to learn causal relationships, to combine prior knowledge, to handle incomplete dataset and to avoid over-fitting [58]. They also have some limitations. The major problem of Bayesian networks is that they cannot model feedback regulations among genes, which however are very important for GRNs.

Dynamic Bayesian networks [61] were proposed to cope with the feedback-loops problem of static Bayesian networks. One major application of DBNs is to infer regulatory relationships among genes (i.e. GRNs) from time-series gene expression data. DBNs only model the regulations of genes across time points, see Figure 6.1. The left figure in Figure 6.1 is the DBN modeling for a two-gene network which is shown in the right figure. The order of a DBN is the maximum number of time points ahead are considered by the DBN to infer the gene regulations. For example, a first-order DBN is to capture the regulations from genes across the consecutive time points (i.e. only consider one time point ahead). Thus, by treating one gene in two different time points as “two” genes, DBNs take into consideration the loops of regulation among genes.

Figure 6.1: An example: DBN modeling (left) for a two-gene network (right).

For a network structure s , let θ be the parameters of this structure, i.e. conditional probabilities associated with the edges in s . Let $\mathbf{x} = [x(t)]_{t=1}^T = [(x_i(t))_{i=1}^N]_{t=1}^T$ be the time-course gene expression data, $x_i(t)$ is the expression of the i th gene at time t , N and T are the total number of genes and time points respectively. By applying the decomposability property [233] on the Eq. (6.1), the likelihood of gene expression $\mathbf{x}$ generated by this network is given by:

$$P(\mathbf{x}|s, \theta) = \prod_{i=1}^N \prod_{j=1}^J \prod_{k=1}^K \theta_{ijk}^{N_{ijk}}, \quad (6.2)$$

where $\theta_{i,j,k} = p(x_i(t) = k | a_i(t-1) = j)$, a_i denotes the parent of the i th gene; N_{ijk} is the number of instances of θ_{ijk} occurring in the expression data (actually $\theta_{ijk} = \frac{N_{ijk}}{\sum_{k'} N_{ijk'}}$);

N , J and K are the total number of genes, number of states of the i th gene's parents at the previous time point, number of states of the i th gene, respectively.

According to Bayes' rule, the posterior probability of network structure s given data can be obtained by:

$$p(s|\mathbf{x}) = \frac{p(s)p(\mathbf{x}|s)}{\sum_{s' \in S'} p(s')p(\mathbf{x}|s')}, \quad (6.3)$$

where $p(s)$ is the prior probability of the network structure s and $\sum_{s' \in S'} p(s')p(\mathbf{x}|s') = p(\mathbf{x})$ is a normalization factor, and S' denotes the set of all possible network structure; $p(\mathbf{x})$ is independent of s , and thus can be removed from the evaluation of s ; $p(\mathbf{x}|s)$ denotes the marginal likelihood of the network structure, which can be obtained by marginalizing over the parameters:

$$p(\mathbf{x}|s) \propto \int_{\theta} p(\mathbf{x}|s, \theta)p(\theta|s)d\theta. \quad (6.4)$$

The optimal network structure is obtained by maximizing a posterior distribution criterion:

$$s^* = \underset{s}{\operatorname{argmax}} p(s|\mathbf{x}). \quad (6.5)$$

However, the optimal network structure s^* may not be acquired by directly maximizing posterior probability, because it is infeasible to enumerate all the possible network structures when the number of nodes is large, which is due to the fact that the number of network structures grows exponentially with the number of nodes [232]. So heuristic or stochastic algorithms are often employed to learn the optimal structure. In this study, we use Markov chain Monte Carlo (MCMC) simulation [234]. MCMC can generate a Markov chain of structures which would converge to the optimal structure in the equilibrium of iteration.

Here the MCMC algorithm is implemented using the Metropolis-Hastings method [235]. The algorithm of structure learning using Metropolis-Hastings method is described in Algorithm 6.1. In step 2, the probability of $Q(s_{new}|s_{old})$ can be approximated with $1/N_{s_{old}}$, where $N_{s_{old}}$ is the size of the neighborhood of s_{old} , that is, the number of possible directed acyclic structures that can be obtained from s_{old} by applying three elementary operations: adding, reversing and deleting an edge. In step 3, $\frac{p(s_{new}|\mathbf{x})}{p(s_{old}|\mathbf{x})} \times \frac{Q(s_{old}|s_{new})}{Q(s_{new}|s_{old})}$ is called Metropolis-Hasting acceptance ratio, where $\frac{Q(s_{old}|s_{new})}{Q(s_{new}|s_{old})}$ is named Hastings ratio, which can be estimated by $N_{s_{old}}/N_{s_{new}}$. The most important advantage of the Metropolis-Hasting method is that it cancels out the intractable denominator of the posterior probability by using the ratio. Under general conditions, the iteration of Metropolis-Hasting procedure produces a Markov chain whose distribution could converge to the true distribution of posterior probability [232].

Algorithm 6.1. Structure learning of GRN using DBN and MCMC

INPUT: Time-course gene expression data

Steps:

- 1: Initialization: obtain the initial network structure s_0 by calculating the mutual information among the gene expression data;
- 2: Given a network structure s_{old} , a new structure s_{new} is proposed with the probability of $Q(s_{new}|s_{old})$ by applying three elementary operations on the old structure: adding, reversing and deleting an edge;
- 3: Accept the new network structure with this probability

$$\min \left\{ 1, \frac{p(s_{new}|\mathbf{x})}{p(s_{old}|\mathbf{x})} \times \frac{Q(s_{old}|s_{new})}{Q(s_{new}|s_{old})} \right\}, \quad (6.6)$$

where $p(s|\mathbf{x})$ is the posterior probability which can be obtained by Eq. (6.3);

- 4: Repeat step 2 and step 3 until convergence of the structure or the maximum iteration times reached;

OUTPUT: network structure s in equilibrium.

After running the algorithm many times (e.g. ten thousands of times), the final structure is obtained by averaging all the output structures (i.e. the network structures in equilibrium). Let connectivity matrixes $(C_1, C_2, \dots, C_M)$ denote the output structures of GRNs: $C_k = \{c_{i,j}\}_{N \times N}$, where $k = 1, \dots, M$, and M is the number of running times, N is the number of genes. Let $P = \{p_{i,j}\}_{N \times N}$ be probability matrix of the final structure, where $p_{ij} \in [0, 1]$ denotes the probability of connectivity between the i th gene and the j th gene. Then $P = \frac{1}{M} \sum_{k=1}^M C_k$. After that, we use a threshold (here it is 0.5) to transform the probability matrix P to a connectivity matrix $C = \{c_{i,j}\}_{N \times N}$ where $c_{ij} = 1$ when $p_{ij} \geq 0.5$ and 0 otherwise.

6.2.2 Integration of Epigenetic Prior into GRN Inference

In this study, the epigenetic prior is constructed based on the histone modification profiles of genes. Other kinds of epigenetic data such as DNA methylation could also be used in the future.

The genome wide histone modification profiles could be represented as a 3-dimension matrix (or a tensor): $H = \{H_i\}_{i=1}^N$, where N is the number of genes, H_i is the histone profile of the i th gene, and $H_i = \{h_{l,f}^i\}_{L_i \times F}$, where $h_{l,f}^i$ is a positive float number which denotes the enrichment of the f th histone modification at the l th measured locus of the

i th gene; L_i and F are the number of locations measured for the i th gene and the number of histone modification types respectively. There are two types of locations for a gene: intragenic and intergenic regions, which are locations within the gene body and between genes respectively.

In [168], the average enrichment of each histone modification type across different locations is calculated for each gene, to obtain a vector of F positive float numbers, each of which denotes the level of this type of histone modification for this gene. The calculation of the HM vector for the i th gene is formulated as follows:

$$\mathbf{v}_i = \frac{1}{L_i} \left[\sum_{l=1}^{L_i} h_{l,1}^i, \sum_{l=1}^{L_i} h_{l,2}^i, \dots, \sum_{l=1}^{L_i} h_{l,F}^i \right], \quad (6.7)$$

where L_i and F are the number of locations measured for the i th gene and number of histone modification types respectively. However, taking average of HM enrichment across all locations could miss some important information related to the locations. We observed that histone modifications have different characteristics in intragenic and intergenic regions. Therefore we treat these two types of locations differently and calculate the average HM enrichment for each type separately (see Section 6.2.3 for details).

We construct the prior matrix $B = \{b_{i,j}\}_{N \times N}$ as follows: First calculate Pearson correlation coefficient $\rho(\mathbf{v}_i, \mathbf{v}_j)$ of HM vectors for each gene pair (i, j) where i and j are integers and $1 \leq i, j \leq N$. In a prior matrix, $b_{i,j}$ denotes the probability of the i th gene and the j th gene having regulatory relation. Hence, $b_{i,j} \in [0, 1]$. Since $\rho(\mathbf{v}_i, \mathbf{v}_j) \in [-1, 1]$, we set $b_{i,j} = |\rho(\mathbf{v}_i, \mathbf{v}_j)|$ which is the absolute value of $\rho(\mathbf{v}_i, \mathbf{v}_j)$, because a high correlation (no matter negative or positive correlation) hints high probability of having regulations between two genes (see Section 6.2.3). Here we ignore the sign of regulation. In other words, we use “having directed regulation relations” to represent both positive regulation (i.e. activation) and negative regulation (i.e. repression).

We use Gibbs distribution to model the prior probability $P(s)$ for DBN, as follows [167]:

$$P(s) = \frac{1}{Z} e^{-\beta E(s)}, \quad (6.8)$$

where β is a hyper-parameter, $Z = \sum_{s \in S} e^{-\beta E(s)}$, and S is the set of all possible GRNs. Z is a normalizing function which will be canceled out in MCMC. Following [162], the $E(s)$ in Eq. (6.8) is given by

$$E(s) = \sum_{i,j=1}^n |b_{i,j} - c_{i,j}|, \quad (6.9)$$

where $c_{i,j}$ is the element of connectivity matrix of the network structure s , $c_{i,j} = 1$ if the i th gene and the j th gene are connected in s , and 0 otherwise. The closer the prior

Algorithm 6.2. Structure learning of GRN inference with epigenetic prior

INPUT: Time-course gene expression data, epigenetic data (i.e. histone modification profiles here)

Steps:

- 1: Construct the prior matrix $\{b_{i,j}\}_{N \times N}$ as follows.
 - 1.1: Calculate HM vector from HM profiles using Eq. (6.7) for each gene;
 - 1.2: Compute Pearson correlation $\rho(\mathbf{v}_i, \mathbf{v}_j)$ between HMs vectors of each gene-pair (i, j) ;
 - 1.3: Set $b_{i,j} = |\rho(\mathbf{v}_i, \mathbf{v}_j)|$, i.e. the absolute value of correlation;
- 2: Generate the initial network structure s_0 by calculating the mutual information among the gene expression data;
- 3: **Repeat:**

Given a network structure s_{old} , a new structure s_{new} is proposed using the Metropolis-Hastings method;
- 4: Calculate the prior probabilities for s_{old} and s_{new} respectively using Gibbs distribution: $P(s_{old}) = \frac{1}{Z} e^{-\beta E(s_{old})}$, $P(s_{new}) = \frac{1}{Z} e^{-\beta E(s_{new})}$, where Z need not be calculated because it will be canceled out in the next step;
- 5: Accept the new network structure with this probability

$$\min \left\{ 1, \frac{p(\mathbf{x}|s_{new})}{p(\mathbf{x}|s_{old})} \times \frac{Q(s_{old}|s_{new})}{Q(s_{new}|s_{old})} \times \frac{p(s_{new})}{p(s_{old})} \right\}, \quad (6.10)$$

- 6: **until** convergence of the structure or the maximum iteration times reached

OUTPUT: network structure s in equilibrium.

network $\{b_{i,j}\}_{N \times N}$ is to $\{c_{i,j}\}_{N \times N}$, the smaller $E(s)$ is, and the larger prior probability $P(s)$ would be. So if the prior network can somehow reflect the benchmark network, the prior probability of structure s could lead it to the real GRN. In this way, the prior knowledge is encoded into the prior matrix $\{b_{i,j}\}_{N \times N}$ and used in the process of GRN inference.

Our algorithm of GRN inference with epigenetic prior integrated is summarized in Algorithm 6.2.

6.2.3 Epigenetic Prior Generation

We observed that the prior probabilities $b_{i,j}$ generated from intragenic regions show different characteristic from those of intergenic regions. The prior probabilities from HM profiles of intragenic regions (referred as “intragenic priors”) are overall higher than those

from intergenic regions (referred as “intergenic priors”). The high probabilities from intragenic regions could provide strong power of prior to affect the inference of GRN. On the other hand, the intergenic prior of a gene-pair (i, j) with regulatory relation would be more likely higher than those of random pairs, which however is not necessarily the case for intragenic priors, see Figure 6.2. Actually, the phenomenon that epigenetic data such as histone marks and DNA methylation show different characteristics in different locations has been reported [236]. The higher correlations of confirmed gene-pairs than random pairs can serve as prior knowledge for GRN inference.

Figure 6.2: Comparison of correlations of HM vectors between random gene-pairs and confirmed gene pairs in yeast.

Therefore, we combine the intragenic prior with intergenic prior to take advantage of both the high prior probabilities in intragenic regions and the discriminative power of the prior probabilities in intergenic regions. Since the prior probabilities of the intergenic prior are relatively low, which may jeopardize the performance of the combined prior, we transform the intergenic prior in the following way: We first randomly pick W (here $W = 5000$) elements $r_1, r_2, \dots, r_W$ by sampling with replacement from the matrix of intergenic prior and get a random distribution $RD = \{r_1, r_2, \dots, r_W\}$, then re-calculate each value e in the intergenic prior matrix as

$$e_{new} = \frac{|\{r_i | r_i \leq e, r_i \in RD\}|}{W}, \quad (6.11)$$

where the numerator is the number of elements in RD that are smaller than e . The transformation on the intergenic prior matrix can raise the overall values in the matrix of intergenic prior while enhance its discriminative power, see Figure 6.3.

Figure 6.3: Comparison of normalized correlations of HM vectors between random gene-pairs and confirmed gene-pairs in yeast.

After the transformation of the intergenic prior, we combine it with intragenic prior, which has high correlation values. Let G be the matrix of intragenic prior, and In be the matrix of transformed intergenic prior. Then the two priors are combined in the following way to obtain a new prior GIn :

$$GIn(i, j) = \frac{G(i, j) + In(i, j)}{2}, \quad (6.12)$$

where $G(i, j)$ and $In(i, j)$ are elements from G and In respectively, and $i, j = 1, 2, \dots, N$, N is the number of genes. In this way, the new prior GIn combines the advantages from intergenic prior and intragenic prior, which are the discriminative power to potentially recognize the true interactions among genes, and the strength to affect the inference of GRN, respectively.

As a result, we propose several priors (i.e. G , In , and GIn) as follows. We will compare their performance in GRN reconstruction using DBN.

- A : the prior used in [168]
- G : the prior generated from HM profiles in intragenic regions
- In : the transformed version of I (which is the prior generated from HM profiles in intergenic regions), using Eq. (6.11)
- GIn : the combination of G and In , using Eq. (6.12)

6.2.4 Synthetic Expression Data Generation

We use both real and synthetic gene expression to evaluate the performance of our method for GRN reconstruction. The synthetic data are simulated from a given benchmark network using the first-order multivariate vector autoregressive (MVAR) [181, 182] model. Here the benchmark network is a 9-yeast-gene network from [168]. Let $x = [x(t)]_{t=1}^T = [(x_i(t))_{i=1}^N]_{t=1}^T$ be the time-series gene expression data of a network. According to the first order MVAR model, the expression values of all genes at time point t are given by:

$$x(t) = x(t-1) \times S_{weight} + \varepsilon(t), \quad (6.13)$$

where the weight S_{weight} is the connectivity matrix of the given network multiplied by the random weights which are sampled from a uniform distribution on the interval $[-1, -0.5]$ and $[0.5, 1]$ such that there would be approximately equal numbers of positive and negative weights; $\varepsilon(t)$ is the Gaussian random noise at time t . The initial expression values $x(t)_{t=0}$ are sampled from the uniform distribution on the interval $[0, 1]$. Then the expression data in subsequent time points are generated using Eq. (6.13). The values of gene expression in the generated dataset are continuous numbers, which would be converted to three discrete values $\{-1, 0, 1\}$ by the discretization method “equal width interval binning” [237]. The three discrete values represent “down regulation”, “no regulation” and “up regulation” respectively.

6.3 Experiments and Results

We compare the performance of DBN combined with different priors introduced in Section 6.2.3 using both real data and synthetic data. The real data are gene expression data of yeast cell cycle (24 time points of *cdc-15* cell cycle arrest) from [201]. The synthetic data are generated by the MVAR model as described in Section 6.2.4. Here six synthetic datasets have been generated, which contain 10, 30, 50, 70, 100, 200 time points respectively. Each dataset consists of 20 time series and the final results take the average of the performance metrics. The epigenetic data used here are histone modification profiles of yeast, taken from [238]. We use *Precision*, *Recall* and *F-measure* as performance metrics. Please refer to Section 2.1.3 for their definitions. The benchmark network used here is the same as in [168]. It is a nine-gene regulatory network in yeast cell cycle, which was experimentally identified by [200].

The comparison of F-measures between the different priors is shown in Figure 6.4, where the error bars represent the standard deviation among the 20 datasets. As seen from Figure 6.4, the prior *GIn* has the highest performance among all the proposed priors in real data, and most of the time in synthetic data. Detailed comparisons are shown

in Table 6.1. According to the results, F-measure performance was improved averagely by 4% by intragenic prior (G) and 6% by combined prior (i.e. GIn) of intragenic and normalized intergenic prior for synthetic data. For real data only GIn prior provide improvement in F-measure (3%) when compared with other priors.

Figure 6.4: Performance (F-measure) comparison of different priors for GRN inference on both real data and synthetic data (10~200 time points).

To show the effectiveness of GIn prior more clearly, we depict boxplot of F-measures of two cases of GRN inference (i.e. without prior and with GIn prior), see Figure 6.5. The comparison experiments were performed on the real dataset and six simulated datasets with different time points. We carried out a paired t-test on the results of no-prior and the GIn prior, for each of the seven types of data (i.e. real data and six groups of synthetic data). The null hypothesis is that the pairwise difference between the results of no-prior and the GIn prior (of the 20 time series datasets in each group of data) has a mean not less than zero. The p-values of paired t-tests are shown in the legend of Figure 6.5. As shown in Figure 6.5, with the integration of the proposed GIn prior, the F-measures of GRN inference are significantly improved than the case without prior, in most datasets.

Table 6.1: DBN performance of different priors and different numbers of time points

Prior	Data	Precision	Sensitivity	F -measure
No prior	Real	0.21	0.13	0.16
	10T	0.29	0.18	0.22
	30T	0.48	0.28	0.35
	50T	0.68	0.37	0.47
	100T	0.91	0.42	0.57
A	Real	0.15	0.11	0.13
	10T	0.29	0.2	0.24
	30T	0.41	0.23	0.29
	50T	0.63	0.29	0.39
	100T	0.86	0.29	0.44
G	Real	0.17	0.18	0.17
	10T	0.28	0.23	0.25
	30T	0.49	0.39	0.43
	50T	0.68	0.43	0.52
	100T	0.85	0.43	0.55
In	Real	0.16	0.12	0.14
	10T	0.3	0.22	0.25
	30T	0.49	0.31	0.38
	50T	0.63	0.38	0.47
	100T	0.87	0.34	0.49
GIn	Real	0.26	0.15	0.19
	10T	0.32	0.24	0.27
	30T	0.51	0.37	0.42
	50T	0.66	0.43	0.51
	100T	0.87	0.44	0.58

For a close look at how epigenetic prior can improve GRN inference, we give an example in Figure 6.6, where one can look closely at the predicted networks by DBN with the *GIn* prior compared to the case without prior. Figure 6.6.a shows the benchmark network used in this study. From the comparison of Figure 6.6.b and Figure 6.6.c we can see, more FP and FN are obtained when the epigenetic prior is not incorporated in DBN. On the other hand, the *GIn* prior can bring more TP while keep the number of FP and FN under control.

Figure 6.5: Performance (F-measure) comparison between GRN inference without prior and with the *GIn* prior, from different datasets. The p-values of paired t-tests for the comparison on the seven groups of data (from left to right, and top to bottom) are $8.78\text{e-}04$, $2.32\text{e-}04$, 0.0019 , 0.0016 , 0.0043 , 0.0452 , 0.2737 , respectively.

The parameter β in Gibbs prior indicates the influence of prior evidence on DBN inference relative to the data. When $\beta \rightarrow 0$, prior probability given by Eq. (6.8) is flat and does not provide sufficient information about the network structure. When β is increasing it provide more information about the structure. Since β controls the weight of prior on GRN inference, a large β could be set if one is very confident about the prior. Then, how large value for β is appropriate? In previous experiments, we set $\beta = 1$. Here we study the effect of β on the performance of different priors, by setting β from 1 to 10 and compare F-measures of inferred networks in the ten experiments. Results show that the F-measure often increases as β is raised from 1 to 5. But from 6 to 10, the increment of β does not bring any obvious improvement. Figure 6.7 describes the variation of F-measure when β changes from 1 to 10, using benchmark network as prior. This extreme example shows that β does not need to be too large even when the prior matrix is very confident. The results shown in Figure 6.7 are from experiments on synthetic data with 30 time points. Experiments on other datasets including real data have similar results (not shown here).

6.6.a Benchmark network

6.6.b Inferred network by DBN without prior

6.6.c Inferred network by DBN with the GIn prior

Figure 6.6: Comparison of inferred networks by DBN without and with prior. Green line denotes true positive (TP), red line represents false positive (FP), and blue line means false negative (FN).

Figure 6.7: F-measure variation of GRN inference using benchmark as prior with β from 1 to 10.

6.4 Conclusions and Future Work

In this chapter, we integrated epigenetic priors into dynamic Bayesian network for GRN reconstruction. We proposed several new priors from the same epigenetic profile by treating the intragenic and intergenic regions separately and using the technique of random normalization. Experiments on both synthetic data and real data of yeast cell cycle showed that epigenetic prior can improve the performance of GRN reconstruction. Such improvement was obtained by increasing the number of true positives while decreasing the number of false negatives, because the higher correlations of HM vectors of confirmed gene-pairs than random pairs can serve as good prior knowledge for GRN inference.

There are a few directions to further improve our work here. Firstly, the performance of epigenetic priors proposed here was tested on just one benchmark network. It would be interesting to test them on other benchmark networks of yeast and other species. Secondly, we generated our priors based on different genomic regions (i.e. intragenic and intergenic). New priors could also be generated by selecting the locations of the probes or the types of HMs based on feature selection techniques. Finally, other epigenetic data especially DNA methylation are potential good priors for GRN inference.

Chapter 7

Conclusions and Future Work

Understanding gene regulation is of critical importance in elucidating the mechanisms of cellular processes and various phenotypes such as diseases. However, the regulation of gene expression is highly complex, with the intricate interactions among DNAs, RNAs, proteins and small molecules. This thesis focuses on the regulation of gene expression in mRNA transcription. In particular, our goal is to reconstruct the interactions between genes from gene expression data and to improve the inference by integrating multiple sources of data and knowledge. In this chapter, we conclude the thesis by first summarizing our work described in the previous chapters and then giving a few directions for future studies.

7.1 Conclusions

In this thesis, we study the regulation of gene expression on two basic and interacting layers: genetic and epigenetic layers. Our task is to infer the regulatory relationships among genes from gene expression data. As we know, gene regulation is a very complex process, with many factors involved. We aim to improve the performance of GRN inference by integrating knowledge as well as considering time delays in gene regulation. To investigate the relationship between epigenetic modifications and gene expression on a global scale, we propose a framework based on association rule mining to explore histone code in transcription. Finally, we integrate epigenetic data into the process of GRN inference to improve the performance.

First, we proposed a method (named SingCellNet) based on the framework of epigenetic landscape by synthesizing the data-driven and knowledge-driven modeling techniques. Our method adopts probabilistic Boolean network for network modeling, and

genetic algorithm as search strategy. Based on a technique of simulation-based inference, SingCellNet is able to accurately reconstruct the genetic interactions from single-cell gene expression data by integrating the structure of a cell lineage tree with the transcriptional patterns from single-cell data. One major advantage of SingCellNet is the ability to infer GRN from short time-series data with sparse time points. We compared SingCellNet with existing popular methods of GRN inference on the single-cell transcriptional data of mouse preimplantation development. Experimental results showed that SingCellNet outperformed the other methods in most cases. SingCellNet (after slight adaption) can also identify the operational interactions in a given network topology, corresponding to specific cell fate decisions. Our method is one of the first attempts to infer gene regulatory networks from single-cell transcriptional data, incorporating the knowledge from cell development (i.e. the “directionality” of development along branches of the cell lineage tree).

Then, we improved GRN inference by taking into account time delay, which is an inherent feature of gene regulation. Inferring time-delayed gene relationships is an even more challenging task. Here we tackled this problem by developing an approach based on cross-correlation and network deconvolution (ND). The ND algorithm is a newly proposed method of GRN inference by recovering direct relationships from the data with the mix of direct and indirect relationships, such as correlations. This method is promising for GRN reconstruction, showing good accuracy, robustness, flexibility of generalization and so on. However, the original ND algorithm did not consider time delays, and it cannot infer the directions of gene interactions either. We extended the power of ND by taking into account the time delays in gene regulation. We first determined the probable time lags of regulation for each pair of genes based on cross-correlation. Then we aligned the time samples based on the inferred time lags before calculating the correlations of gene expression profiles. After that, ND was applied to reconstruct the direct regulatory relationships between genes. In this way, we designed a method named time-delayed ND which can infer the directions of gene interactions. By taking into account the information of time delays among gene interactions, our method is able to infer GRNs more accurately than the original ND. We also compared our method with other approaches for learning time-delayed GRNs, by applying them on five real-life gene expression datasets of yeast. Our method (i.e. time-delayed ND) has significantly higher sensitivity in detecting genetic relationships while keeps good balance in overall performance (i.e. with high F-measures).

Apart from GRN inference, we have also looked at gene regulation on a global scale by exploring the relationships between combinatorial histone modifications and transcription state. Based on association rule mining, we proposed a framework to obtain interpretable rules which describe how combinations of histone modifications regulate gene expression. We estimated the statistical significance (measured by p-values) of these rules by Fisher’s

exact test, and further adjusted the p-values using permutation-based multiple testing. Then based on the adjusted p-values, we filtered out the non-significant rules. Our significant rules are validated by literature with experimental evidence. However, rules with multiple (combinatorial) HMs currently lack experimental support, probably due to some limitations in detecting such relationships experimentally. These rules with combinatorial HMs could guide further wet-lab experiments of exploring histone code. Furthermore, we built a model to predict gene expression using our rules, which outperformed a published Bayesian model. The results of our work indicate that histone modifications regulate gene expression on a large scale. Our findings provide new support for the hypothesis of “histone code”, and offer an insight into the study of epigenetic regulatory mechanisms in transcription as well as other biological processes.

Finally, we integrated epigenetic data into the reconstruction of GRNs to improve the performance of GRN inference. Epigenetic factors such as histone modifications are closely related with gene expression in that they would affect the accessibility of the *cis*-regulatory element on the DNA of genes and the binding of transcription factors to the genes. Therefore, histone modifications profiles can provide information about gene regulation and the integration of epigenetic data is promising for the improvement of GRN inference. We chose the DBN model to infer GRNs, for the strength of Bayesian network in the integration of prior knowledge. Furthermore, DBNs are superior to traditional Bayesian networks in that DBNs can learn temporal relationships (including feed-back loops) between genes from time-series gene expression data. We studied the underlying characteristics of epigenetic data (histone modifications profiles) in different locations of genes, and observed that the correlations of HMs data in intragenic regions were overall higher than those in intergenic regions, but HMs in intergenic regions had better discriminability in detecting gene interactions than intragenic regions. Based on this observation, we designed a prior matrix by combining the HMs correlations from intragenic and intergenic regions, and then integrated the prior information into Bayesian inference using Gibbs distribution (see Section 6.2.2 and 6.2.3 for details). The effectiveness of our method was evaluated on both simulated and real gene expression data. Experimental results showed that the performance of GRN inference was improved after the integration of epigenetic data. By incorporating the information from epigenetic data, our method is able to increase the number of true positives while keep the number of false positives under control. Our results indicate that epigenetic factors are closely involved in gene regulation and therefore epigenetic data can serve as a good source of prior knowledge for GRN inference.

7.2 Future Directions

In this thesis, we have investigated the mechanisms of gene regulation in the fields of GRN inference as well as histone code mining. We proposed a novel method by the syn-

thesis of data-driven and knowledge-driven models to achieve a more accurate modeling of gene regulation. We also improved the performance of GRN inference by integrating the information from time delays and epigenetic data. So far we have mainly focused on investigating gene regulation through a GRN and its interaction with epigenetic factors. The ultimate goal of gene regulation research is to understand how cell identity is maintained and transformed. To achieve this goal, we need to include other essential components in the cellular regulatory system such as signaling pathways and the regulation of cell cycle, etc. Therefore, aside from finishing the future work of our ongoing projects, we plan to extend our work in the following directions, to gain a further and deeper understanding of gene regulation.

7.2.1 Cell Cycle

Considering cell cycle is a critical step to model the dynamic interactions between genetic and epigenetic layers. Starting from the fertilized zygote, cells of an organism have been under tight control of cell cycles. How the regulation of gene expression in a cell maintains or changes during cell cycles is particularly interesting for elucidating the mechanisms of some time-evolving biological processes such as embryonic development.

In eukaryotes, the cell cycle includes interphase, the mitotic phase (including cytokinesis), and often a resting phase called Gap 0 (G_0) between them. The interphase consists of three stages, namely Gap 1 (G_1), Synthesis (S) and Gap 2 (G_2). The mitotic phase is composed by four stages, i.e. prophase, metaphase, anaphase and telophase. In interphase, the cell keeps growing, carries out DNA synthesis, and prepares for the cell division in mitosis. Since interphase accounts for most of the time (usually over 90%) in cell cycle, many activities based on gene interactions happen in this phase. Besides, the chromosomes are highly condense during the mitotic phase, making the DNA sequence much less accessible to transcription factors (hence few gene interactions occur). On the other hand, it is believed that the epigenetic configuration is established during mitosis (especially in telophase) [239]. Therefore, cell cycle provides an opportunity for the interactions between gene expression and epigenetic states, and including cell cycle is critical to understand how cell types are maintained while keeping the potential to change to other cell types.

Here, we propose to extend our current work to include more details in the interaction between genetic regulation and epigenetic regulation as follows. In our model, factors from genetic and epigenetic layers can interact through the progress of cell cycle. During the interphase, where chromatin is in loose state and it is easier for transcription, genetic regulation dominates this phase and proteins are generated in accord with the epigenetic states established in the previous cell cycle. In this phase, the GRN acts upon the epigenetic environment and controls the expression of genes. In mitosis (especially

telophase), on the other hand, the chromatin accessibility greatly diminishes and protein expression almost ceases. This phase provides a window for the reconfiguration of epigenetic profiles. Our goal is to elucidate the stability and plasticity of cell identities during cell differentiation, as well as cellular reprogramming.

7.2.2 Cellular Reprogramming

It is natural to extend the research on cell differentiation to cellular reprogramming, in that reprogramming is the inverse process of development and there might be some common mechanisms for the transformation of cell identities.

Reprogramming is a process that a differentiated cell such as a skin cell or B-cell is transformed backward to a pluripotent stem cell by some techniques such as somatic cell nuclear transfer [240, 241] or Yamanaka’s approach by overexpressing some defined factors (e.g. Oct4, Sox2, Klf4, and c-Myc) [242]. The pluripotent cells obtained by inducing the expression of defined factors are called induced pluripotent stem cells (iPSCs). The success in iPSC reprogramming is one of the most significant and exciting achievements in biology and medical research in the last decade. Because the production of iPSCs does not use embryonic materials, it does not involve ethical issue. Therefore, iPSC would be a very promising technology for developing patient-specific regenerative medicine.

However, the mechanisms behind reprogramming are still far from clear. For example, what are the epigenetic and genetic pathways that enable reprogramming? Why is the reprogramming efficiency generally so low, and what determines the efficiency of reprogramming? To elucidate the mechanisms of reprogramming, novel computational models are needed to simulate the process of reprogramming and analyze the interactions between epigenetic factors and gene expression at both transcriptional and translational levels.

Computational models have been built to understand the mechanism of reprogramming. Artyomov *et al.* [243] proposed a computational model to describe how cell identity stays or changes with the interactions between the epigenetic and genetic regulatory networks. They arranged cell states on a Cayley tree, and utilized both Potts model and Ising model to characterize interactions between modules (i.e. nodes in the tree). Hamiltonian functions were used to describe how the epigenetic state of a particular module (i.e. cell type) is established by the interactions among current epigenetic status of other modules and the protein atmosphere established in the previous stage. Artyomov’s model is able to simulate how cell types are maintained and transformed after some perturbations happen in the protein atmosphere or epigenetic environment, thus can capture the downstream effect of ectopic expression of reprogramming factors. However, Artyomov’s model does not take into account the expression and epigenetic marks of particular genes. This low resolution makes it difficult to track precisely what

would happen after the ectopic expression of reprogramming factors and how reprogramming is possible. Furthermore, it does not consider the fact that epigenetic marks have “memory” and thus cannot yet model the dynamic process of epigenetic effects precisely.

In 1957, Waddington [33] proposed a conceptional model of epigenetic landscape which characterizes the process of cellular decision-making during development as a ball (a metaphor of the cell) rolling down a hillside which contains many trajectories (see Section 2.2.1 for more details). With the development of modern technology in genomics and epigenomics, the idea of epigenetic landscape has found more applications in explaining phenomena of cellular activities [9, 244, 36].

In the future, we plan to apply the Waddington’s epigenetic landscape and propose a model to elucidate the underlying mechanisms of reprogramming. Our model would couple genetic regulation and epigenetic regulation through the framework of cell cycle. And the GRN and epigenetic regulators will be integrated into the landscape model, which could provide a high-resolution picture of the mechanisms of reprogramming, and provide valuable guidance to wet-lab experiments of producing iPSCs.

7.2.3 Inference of Large-Scale GRNs

Currently most of the existing methods of GRN inference could not be applicable to large-scale networks. Because of the complexity of gene regulation and the dearth of high-quality data, the accurate inference of GRNs of moderate size has been a challenging task. Furthermore, the time complexity of GRN inference often increases exponentially with the number of genes in the network [245]. Therefore, the inference of large-scale GRNs is very difficult to achieve with good performance. However, to infer large-scale GRNs is a critical step to study the mechanisms of gene regulation in higher organisms of which the components of regulatory circuits are largely unknown [246]. Furthermore, the reconstruction of large-scale GRNs is necessary for the modeling of cellular activities on a global scale and for understanding the whole biological system more comprehensively and realistically.

The methods of GRN inference based on simple techniques such as correlations and mutual information [51] could be applied to infer large-scale GRNs. However, the performance of these methods is yet to be improved because of the large numbers of false positives. Efforts have been made to improve the performance of these correlation-based methods for GRN inference. Margolin *et al.* [52] utilized the “data processing inequality” to eliminate most of the indirect interactions learned by mutual information. Feizi *et al.* [196] proposed an approach to extract the direct interactions between genes from the convolution of direct and indirect relationships of genes showing in the correlation of gene expression. A similar technique was developed by Barzel *et al.* [247] to silence the indirect dependencies of genes in the correlation based network. These methods are

applicable to the inference of large GRNs, due to the high efficiency and encouraging accuracy of their algorithms. Chowdhury *et al.* [248] proposed a method based on network decomposition to infer large-scale GRNs. Their method decomposes a GRN into two independent sub-networks which are inferred separately and then combined to obtain the final output network. The efficiency and accuracy of their method are enhanced by incorporating prior knowledge such as maximum in-degree. In [246], Kim *et al.* reviews both experimental and computational techniques to reconstruct large-scale GRNs, considering *cis* and/or *trans* factors. Combining small- and large-scale network analysis is suggested to better understand the regulatory circuits of gene expression.

The methods of GRN inference proposed in this thesis (e.g. SingCellNet) can potentially scale to the inference of large networks, by using faster languages for implementation and other techniques such as parallel computing. It would be very interesting to assess our methods on large-scale networks if related data including benchmark (for validation) are available.

7.2.4 Towards Whole-Cell Modeling

All molecular components within a cell are interconnected. Hence, any separation of cellular processes would lead to limited understanding or inaccurate prediction of a particular phenotype such as a disease. Transcriptional activities are closely connected with signaling pathways. If a GRN is analogous to an intricate electric circuit, then signaling would act as switches which activate or block the expression of genes. Furthermore, to understand the dynamic control of gene expression, we need to take into account the kinetics of transcription and its response to the concentration of transcription factors and so on. The concentration of transcription factors is largely determined by metabolic activities. Therefore, signaling pathways, GRNs (including epigenetic factors), and metabolic pathways are intrinsically connected with each other in a cell. And an integration of models for these processes is necessary for a more accurate characterization of a biological system, which is promising for *de novo* biological discoveries.

There have been efforts for the integrative modeling of whole-cell activities. In 1999, Tomita *et al.* [249] published a software named “E-CELL”, which is one of the pioneering projects in whole-cell modeling. E-CELL incorporated signaling, gene expression and metabolism into one model and defined “reaction rules” to simulate cell behaviours. The reaction rules are based on differential equations which describe the reactions during gene expression and cellular metabolism and so on. E-CELL was applied to simulate the activities of a hypothetical cell with 127 genes taken from *Mycoplasma genitalium*, a human urogenital parasite with a very small genome. In 2012, another whole-cell computational model was proposed by Karr *et al.* [250]. This model is able to capture diverse cellular processes by combining multiple mathematical models. Compared to the

time when E-CELL was developed, much more datasets have been available by 2012, and various pathways within cells are characterized in more details. Therefore, the model of Karr *et al.* can achieve a more comprehensive whole-cell modeling, by including all 525 genes in the genome of *Mycoplasma genitalium*. The simulation results of their model are overall consistent with experimental data. Furthermore, their model can predict previously unobserved biological functions and cellular behaviors.

Although existing projects of whole-cell modeling have achieved very encouraging results, many details in the regulatory mechanisms of cellular activities are still unclear. In particular, how cell fate decision is made based on the interactions of the molecular components? It is desirable to build a hierarchical control model to capture how cell types are maintained and transformed. For our future work, we plan to integrate signaling pathways into our GRN model with epigenetic factors, to form a hierarchical regulatory system for gene regulation. Further, metabolic pathways would be included to simulate the dynamic changes of gene expression with more details.

References

- [1] Qing Zhou, Hiram Chipperfield, Douglas A Melton, and Wing Hung Wong. A gene regulatory network in mouse embryonic stem cells. *Proceedings of the National Academy of Sciences*, 104(42):16438–16443, 2007.
- [2] Bing Li, Michael Carey, and Jerry L Workman. The role of chromatin during transcription. *Cell*, 128(4):707–719, 2007.
- [3] Arie B Brinkman, Thijs Roelofsen, Sebastiaan WC Pennings, Joost HA Martens, Thomas Jenuwein, and Hendrik G Stunnenberg. Histone modification patterns associated with the human X chromosome. *EMBO Reports*, 7(6):628–634, 2006.
- [4] Karl Pfeifer. Mechanisms of genomic imprinting. *The American Journal of Human Genetics*, 67(4):777–787, 2000.
- [5] Min Wu, Chee-Keong Kwoh, Teresa M Przytycka, Jing Li, and Jie Zheng. Epigenetic functions enriched in transcription factors binding to mouse recombination hotspots. *Proteome Science*, 10(Suppl 1):S11, 2012.
- [6] Reini F. Luco, Qun Pan, Kaoru Tominaga, Benjamin J. Blencowe, Olivia M. Pereira-Smith, and Tom Misteli. Regulation of alternative splicing by histone modifications. *Science Signalling*, 327(5968):996, 2010.
- [7] Manel Esteller. Cancer epigenomics: DNA methylomes and histone-modification maps. *Nature Reviews Genetics*, 8(4):286–298, 2007.
- [8] Conrad H Waddington. The epigenotype. *International Journal of Epidemiology*, 41(1):10–13, 2012.
- [9] Aaron D. Goldberg, C. David Allis, and Emily Bernstein. Epigenetics: A landscape takes shape. *Cell*, 128(4):635, 2007.
- [10] Alan Horsager. A short introduction to epigenetics. <https://www.episona.com/short-introduction-epigenetics/>. Accessed: Apr. 2015.

REFERENCES

- [11] Walter Doerfler. In pursuit of the first recognized epigenetic signal—DNA methylation: A 1976 to 2008 synopsis. *Epigenetics*, 3(3):125–133, 2008.
- [12] Wikipedia the free encyclopedia. Histone. <https://en.wikipedia.org/wiki/Histone>. Accessed: Apr. 2015.
- [13] HHMD: Human Histone Modification Database. <http://202.97.205.78/hhmd/histone.jsp>. Accessed: Aug. 2015.
- [14] Zhibin Wang, Chongzhi Zang, Jeffrey A. Rosenfeld, Dustin E. Schones, Artem Barski, Suresh Cuddapah, Kairong Cui, Tae-Young Roh, Weiqun Peng, Michael Q. Zhang, and Keji Zhao. Combinatorial patterns of histone acetylations and methylations in the human genome. *Nature Genetics*, 40(7):897–903, 2008.
- [15] Artem Barski, Suresh Cuddapah, Kairong Cui, Tae-Young Roh, Dustin E. Schones, Zhibin Wang, Gang Wei, Iouri Chepelev, and Keji Zhao. High-resolution profiling of histone methylations in the human genome. *Cell*, 129(4):823–37, 2007.
- [16] Brian D. Strahl and C. David Allis. The language of covalent histone modifications. *Nature*, 403(6765):41–45, 2000.
- [17] Thomas Jenuwein and C. David Allis. Translating the histone code. *Science Signalling*, 293(5532):1074, 2001.
- [18] Theodora Agaloti, Guoying Chen, and Dimitris Thanos. Deciphering the transcriptional histone acetylation code for a human gene. *Cell*, 111(3):381–392, 2002.
- [19] Shelley L. Berger. The complex language of chromatin regulation during transcription. *Nature*, 447(7143):407–12, 2007.
- [20] Jung-Shin Lee, Edwin Smith, and Ali Shilatifard. The language of histone crosstalk. *Cell*, 142(5):682–685, 2010.
- [21] Sui Huang and Stuart A Kauffman. Complex gene regulatory networks—From structure to biological observables: Cell fate determination. *Computational Complexity: Theory, Techniques, and Applications*, pages 527–560, 2012.
- [22] Mark Ptashne. On the use of the word epigenetic. *Current Biology*, 17(7):R233–R236, 2007.
- [23] Mark Ptashne. Faddish stuff: epigenetics and the inheritance of acquired characteristics. *The FASEB Journal*, 27(1):1–2, 2013.

REFERENCES

- [24] Mark Ptashne. Epigenetics: Core misconception. *Proceedings of the National Academy of Sciences*, 110(18):7101–7103, 2013.
- [25] David Crews and Andrea C. Gore. Chapter 26 - Transgenerational Epigenetics: Current controversies and debates. In Trygve Tollefsbol, editor, *Transgenerational Epigenetics*, pages 371 – 390. Academic Press, Oxford, 2014.
- [26] Graham McVicker, Bryce van de Geijn, Jacob F Degner, Carolyn E Cain, Nicholas E Banovich, Anil Raj, Noah Lewellen, Marsha Myrthil, Yoav Gilad, and Jonathan K Pritchard. Identification of genetic variants that affect histone modifications in human cells. *Science*, 342(6159):747–749, 2013.
- [27] Maya Kasowski, Sofia Kyriazopoulou-Panagiotopoulou, Fabian Grubert, Judith B Zaugg, Anshul Kundaje, Yuling Liu, Alan P Boyle, Qiangfeng Cliff Zhang, Fouad Zakharia, Damek V Spacek, et al. Extensive variation in chromatin states across humans. *Science*, 342(6159):750–752, 2013.
- [28] Helena Kilpinen, Sebastian M Waszak, Andreas R Gschwind, Sunil K Raghav, Robert M Witwicki, Andrea Orioli, Eugenia Migliavacca, Michaël Wiederkehr, Maria Gutierrez-Arcelus, Nikolaos I Panousis, et al. Coordinated effects of sequence variation on DNA binding, chromatin structure, and transcription. *Science*, 342(6159):744–747, 2013.
- [29] Heidi Ledford. Language: Disputed definitions. *Nature*, 455(7216):1023–1028, 2008.
- [30] Hong Yu, Shanshan Zhu, Bing Zhou, Huiling Xue, and Jing-Dong J. Han. Inferring causal relationships among different histone modifications and gene expression. *Genome Research*, 18(8):1314–1324, 2008.
- [31] Xiang Zhou, Carolyn E Cain, Marsha Myrthil, Noah Lewellen, Katelyn Michelini, Emily R Davenport, Matthew Stephens, Jonathan K Pritchard, and Yoav Gilad. Epigenetic modifications are associated with inter-species gene expression variation in primates. *Genome Biology*, 15(12):547, 2014.
- [32] Maria Gutierrez-Arcelus, Tuuli Lappalainen, Stephen B Montgomery, Alfonso Buil, Halit Ongen, Alisa Yurovsky, Julien Bryois, Thomas Giger, Luciana Romano, Alexandra Planchon, et al. Passive and active DNA methylation and the interplay with genetic variation in gene regulation. *eLife*, 2:e00523, 2013.
- [33] Conrad Hal Waddington. *The strategy of the genes: A discussion of some aspects of theoretical biology*. London: Allen and Unwin, 1957.

REFERENCES

- [34] Elena R Alvarez-Buylla, Eugenio Azpeitia, Rafael Barrio, Mariana Benítez, and Pablo Padilla-Longoria. From ABC genes to regulatory networks, epigenetic landscapes and flower morphogenesis: making biological sense of theoretical approaches. In *Seminars in Cell & Developmental Biology*, volume 21, pages 108–117. Elsevier, 2010.
- [35] Minsoo Choi, Jue Shi, Sung Hoon Jung, Xi Chen, and Kwang-Hyun Cho. Attractor landscape analysis reveals feedback loops in the p53 network that control the cellular response to DNA damage. *Science Signaling*, 5(251):ra83–ra83, 2012.
- [36] Sui Huang. The molecular and mathematical basis of Waddington’s epigenetic landscape: A framework for post-darwinian biology? *Bioessays*, 34(2):149–157, 2012.
- [37] Chunhe Li and Jin Wang. Quantifying cell fate decisions for differentiation and reprogramming of a human stem cell network: landscape and biological paths. *PLoS Computational Biology*, 9(8):e1003165, 2013.
- [38] Akira Watanabe, Yasuhiro Yamada, and Shinya Yamanaka. Epigenetic regulation in pluripotent stem cells: a key to breaking the epigenetic barrier. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 368(1609):20120292, 2013.
- [39] Sui Huang. Cell lineage determination in state space: A systems view brings flexibility to dogmatic canonical rules. *PLoS Biology*, 8(5):e1000380, 2010.
- [40] Michael Hecker, Sandro Lambeck, Susanne Toepfer, Eugene Van Someren, and Reinhard Guthke. Gene regulatory network inference: Data integration in dynamic models - A review. *Biosystems*, 96(1):86–103, 2009.
- [41] Thomas Schlitt and Alvis Brazma. Current approaches to gene regulatory network modelling. *BMC Bioinformatics*, 8(Suppl 6):S9, 2007.
- [42] Stuart Kauffman. Homeostasis and differentiation in random genetic control networks. *Nature*, 224(5215):177–8, 1969.
- [43] Shoudan Liang, Stefanie Fuhrman, and Roland Somogyi. REVEAL, a general reverse engineering algorithm for inference of genetic network architectures. *Pacific Symposium on Biocomputing*, pages 18–29, 1998.
- [44] Tatsuya Akutsu, Satoru Miyano, Satoru Kuhara, et al. Identification of genetic networks from a small number of gene expression patterns under the Boolean network model. In *Pacific Symposium on Biocomputing*, volume 4, pages 17–28, 1999.

REFERENCES

- [45] Tatsuya Akutsu, Satoru Miyano, and Satoru Kuhara. Algorithms for identifying Boolean networks and related biological networks based on matrix multiplication and fingerprint function. *Journal of Computational Biology*, 7(3-4):331–343, 2000.
- [46] Harri Lähdesmäki, Ilya Shmulevich, and Olli Yli-Harja. On learning gene regulatory networks under the Boolean network model. *Machine Learning*, 52(1-2):147–167, 2003.
- [47] Shawn Martin, Zhaoduo Zhang, Anthony Martino, and Jean-Loup Faulon. Boolean dynamics of genetic regulatory networks inferred from microarray time series data. *Bioinformatics*, 23(7):866–874, 2007.
- [48] Graham J. Hickman and T. Charlie Hodgman. Inference of gene regulatory networks using Boolean-network inference methods. *Journal of Bioinformatics and Computational Biology*, 7(06):1013–1029, 2009.
- [49] Stefan Bornholdt. Boolean network models of cellular regulation: prospects and limitations. *Journal of the Royal Society Interface*, 5(Suppl 1):S85–S94, 2008.
- [50] Ilya Shmulevich, Edward R. Dougherty, Seungchan Kim, and Wei Zhang. Probabilistic Boolean networks: a rule-based uncertainty model for gene regulatory networks. *Bioinformatics*, 18(2):261–74, 2002.
- [51] Atul J. Butte and Isaac S. Kohane. Mutual information relevance networks: functional genomic clustering using pairwise entropy measurements. In *Pacific Symposium on Biocomputing*, volume 5, pages 418–429, 2000.
- [52] Adam A. Margolin, Ilya Nemenman, Katia Basso, Chris Wiggins, Gustavo Stolovitzky, Riccardo D. Faveria, and Andrea Califano. ARACNE: An algorithm for the reconstruction of gene regulatory networks in a mammalian cellular context. *BMC Bioinformatics*, 7(Suppl 1):S7, 2006.
- [53] Pietro Zoppoli, Sandro Morganella, and Michele Ceccarelli. TimeDelay-ARACNE: Reverse engineering of gene networks from time-course data by an information theoretic approach. *BMC Bioinformatics*, 11(1):154, 2010.
- [54] Hidde De Jong. Modeling and simulation of genetic regulatory systems: A literature review. *Journal of Computational Biology*, 9(1):67–103, 2002.
- [55] Timothy S. Gardner, Diego di Bernardo, David Lorenz, and James J. Collins. Inferring genetic networks and identifying compound mode of action via expression profiling. *Science Signalling*, 301(5629):102, 2003.

REFERENCES

- [56] Shuhei Kimura, Kaori Ide, Aiko Kashihara, Makoto Kano, Mariko Hatakeyama, Ryoji Masui, Noriko Nakagawa, Shigeyuki Yokoyama, Seiki Kuramitsu, and Akihiko Konagaya. Inference of S-system models of genetic networks using a cooperative coevolutionary algorithm. *Bioinformatics*, 21(7):1154–1163, 2005.
- [57] Pablo Meyer, Thomas Cokelaer, Deepak Chandran, Kyung Hyuk Kim, Po-Ru Loh, George Tucker, Mark Lipson, Bonnie Berger, Clemens Kreutz, Andreas Raue, et al. Network topology and parameter estimation: from experimental design methods to gene regulatory network kinetics using a community based approach. *BMC Systems Biology*, 8(1):13, 2014.
- [58] David Heckerman. A tutorial on learning with Bayesian networks. *Innovations in Bayesian Networks*, pages 33–82, 2008.
- [59] Nir Friedman, Michal Linial, Iftach Nachman, and Dana Pe’er. Using Bayesian networks to analyze expression data. *Journal of Computational Biology*, 7(3-4):601–620, 2000.
- [60] Chris J. Needham, James R. Bradford, Andrew J. Bulpitt, and David R. Westhead. A primer on learning in Bayesian networks for computational biology. *PLoS Computational Biology*, 3(8):e129, 2007.
- [61] Kevin Murphy and Saira Mian. Modelling gene expression data using dynamic Bayesian networks. Technical report, Computer Science Division, University of California, Berkeley, CA, 1999.
- [62] Vitoantonio Bevilacqua, Giuseppe Mastronardi, Filippo Menolascina, Paolo Panarale, and Giuseppe Romanazzi. Bayesian gene regulatory network inference optimization by means of genetic algorithms. *Journal of Universal Computer Science*, 15(4):826–839, 2009.
- [63] Mark P. Styczynski and Gregory Stephanopoulos. Overview of computational methods for the inference of gene regulatory networks. *Computers & Chemical Engineering*, 29(3):519–534, 2005.
- [64] Julian Besag. Markov chain monte carlo methods for statistical inference. *Department of Statistics University of Washington, Seattle, USA*, 2004.
- [65] L. Jason Steggles, Richard Banks, Oliver Shaw, and Anil Wipat. Qualitatively modelling and analysing genetic regulatory networks: a Petri net approach. *Bioinformatics*, 23(3):336–343, 2007.

REFERENCES

- [66] L. Jason Steggles, Richard Banks, and Anil Wipat. Modelling and analysing genetic networks: From Boolean networks to Petri nets. In *Computational Methods in Systems Biology*, pages 127–141. Springer, 2006.
- [67] Daniel Marbach, James C. Costello, Robert Küffner, Nicole M Vega, Robert J. Prill, Diogo M. Camacho, Kyle R. Allison, Manolis Kellis, James J. Collins, Gustavo Stolovitzky, et al. Wisdom of crowds for robust gene network inference. *Nature Methods*, 9(8):796–804, 2012.
- [68] Alina Sîrbu, Heather J Ruskin, and Martin Crane. *Stages of gene regulatory network inference: the evolutionary algorithm role*. INTECH Open Access Publisher, 2011.
- [69] Alex Greenfield, Aviv Madar, Harry Ostrer, and Richard Bonneau. DREAM4: Combining genetic and dynamic information to identify biological networks and dynamical models. *PLoS One*, 5(10):e13397, 2010.
- [70] John Watkinson, Kuo-ching Liang, Xiadong Wang, Tian Zheng, and Dimitris Anastassiou. Inference of regulatory gene interactions from expression data using three-way mutual information. *Annals of the New York Academy of Sciences*, 1158(1):302–313, 2009.
- [71] Alexandre Irrthum, Louis Wehenkel, Pierre Geurts, et al. Inferring regulatory networks from expression data using tree-based methods. *PLoS One*, 5(9):e12776, 2010.
- [72] Fantine Mordelet and Jean-Philippe Vert. SIRENE: supervised inference of regulatory networks. *Bioinformatics*, 24(16):i76–i82, 2008.
- [73] Atul J. Butte and Isaac S. Kohane. Unsupervised knowledge discovery in medical databases using relevance networks. In *Proceedings of the AMIA Symposium*, page 711. American Medical Informatics Association, 1999.
- [74] 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, 2003.
- [75] Mingzhou Joe Song, Chris K. Lewis, Eric R. Lance, Elissa J. Chesler, Roumyana Kirova Yordanova, Michael A. Langston, Kerrie H. Lodowski, and Susan E. Bergeson. Reconstructing generalized logical networks of transcriptional regulation in mouse brain from temporal gene expression data. *EURASIP Journal on Bioinformatics and Systems Biology*, 2009:545176, 2009.

REFERENCES

- [76] Robert Küffner, Tobias Petri, Pegah Tavakkolkhah, Lukas Windhager, and Ralf Zimmer. Inferring gene regulatory networks by ANOVA. *Bioinformatics*, 28(10):1376–1382, 2012.
- [77] Ming Yuan and Yi Lin. Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 68(1):49–67, 2006.
- [78] Anne-Claire Haury, Fantine Mordelet, Paola Vera-Licona, and Jean-Philippe Vert. TIGRESS: trustful inference of gene regulation using stability selection. *BMC Systems Biology*, 6(1):145, 2012.
- [79] Edward R Morrissey, Miguel A Juárez, Katherine J Denby, and Nigel J Burroughs. Inferring the topology of a non-linear sparse gene regulatory network using fully Bayesian spline regression. Technical report, Working paper, 2010.
- [80] Néhémý Lim, Yasin Şenbabaoğlu, George Michailidis, and Florence dAlché Buc. OKVAR-Boost: a novel boosting algorithm to infer nonlinear dynamics and interactions in gene regulatory networks. *Bioinformatics*, 29(11):1416–1423, 2013.
- [81] DAK Maduranga, Jie Zheng, Piyushkumar A. Mundra, and Jagath C. Rajapakse. Inferring gene regulatory networks from time-series expressions using Random Forests ensemble. In *Pattern Recognition in Bioinformatics*, pages 13–22. Springer, 2013.
- [82] Zeeshan Gillani, Muhammad SH Akash, Matiur Rahaman, and Ming Chen. CompareSVM: Supervised, Support Vector Machine (SVM) inference of gene regularity networks. *BMC Bioinformatics*, 15(1):395, 2014.
- [83] Dirk Husmeier. Sensitivity and specificity of inferring genetic regulatory interactions from microarray experiments with dynamic Bayesian networks. *Bioinformatics*, 19(17):2271–2282, 2003.
- [84] Bartek Wilczyński and Norbert Dojer. BNFinder: exact and efficient method for learning Bayesian networks. *Bioinformatics*, 25(2):286–287, 2009.
- [85] Haifen Chen, DAK Maduranga, Piyushkumar A Mundra, and Jie Zheng. Integrating epigenetic prior in dynamic Bayesian network for gene regulatory network inference. In *IEEE Symposium on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB)*, pages 76–82. IEEE, 2013.

- [86] Nizamul Morshed, Madhu Chetty, and Nguyen Xuan Vinh. Simultaneous learning of instantaneous and time-delayed genetic interactions using novel information theoretic scoring technique. *BMC Systems Biology*, 6(1):62, 2012.
- [87] Yifeng Li and Alioune Ngom. The max-min high-order dynamic Bayesian network learning for identifying gene regulatory networks from time-series microarray data. In *IEEE Symposium on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB)*, pages 83–90. IEEE, 2013.
- [88] Peter Spirtes, Clark N Glymour, and Richard Scheines. *Causation, prediction, and search*, volume 81. MIT press, 2000.
- [89] Piyushkumar A. Mundra, Jie Zheng, Mahesan Niranjan, Roy E. Welsch, and Jagath C. Rajapakse. Inferring time-delayed gene regulatory networks using cross-correlation and sparse regression. In *Bioinformatics Research and Applications*, pages 64–75. Springer, Berlin Heidelberg, 2013.
- [90] Madhukar S Dasika, Anshuman Gupta, Costas D Maranas, and Jeffrey D Varner. A mixed integer linear programming (MILP) framework for inferring time delay in gene regulatory networks. In *Pacific Symposium on Biocomputing*, volume 9, pages 474–485. World Scientific, 2004.
- [91] Xia Li, Shaoqi Rao, Wei Jiang, Chuanxing Li, Yun Xiao, Zheng Guo, Qingpu Zhang, Lihong Wang, Lei Du, Jing Li, et al. Discovery of time-delayed gene regulatory networks based on temporal gene expression profiling. *BMC bioinformatics*, 7(1):26, 2006.
- [92] Hari Krishna Yalamanchili, Bin Yan, Mulin Jun Li, Jing Qin, Zhongying Zhao, Francis YL Chin, and Junwen Wang. DDGni: Dynamic delay gene-network inference from high-temporal data using gapped local alignment. *Bioinformatics*, 30(3):377–383, 2014.
- [93] Peng Li, Ping Gong, Haoni Li, Edward J Perkins, Nan Wang, and Chaoyang Zhang. Gene regulatory network inference and validation using relative change ratio analysis and time-delayed dynamic bayesian network. *EURASIP Journal on Bioinformatics and Systems Biology*, 2014(1):1–10, 2014.
- [94] Leung-Yau Lo, Kwong-Sak Leung, and Kahyun Lee. Inferring time-delayed causal gene network using time-series expression data. *IEEE/ACM Trans Comput Biol Bioinform*, 12(5):1169–82, 2015.

REFERENCES

- [95] Jin Wang, Kun Zhang, Li Xu, and Erkang Wang. Quantifying the Waddington landscape and biological paths for development and differentiation. *Proceedings of the National Academy of Sciences*, 108(20):8257–8262, 2011.
- [96] Sui Huang. Gene expression profiling, genetic networks, and cellular states: An integrating concept for tumorigenesis and drug discovery. *Journal of Molecular Medicine*, 77(6):469–480, 1999.
- [97] Sui Huang. Reprogramming cell fates: reconciling rarity with robustness. *Bioessays*, 31(5):546–560, 2009.
- [98] Steven H Strogatz. *Nonlinear dynamics and chaos: with applications to physics, biology, chemistry, and engineering*. Westview press, 2014.
- [99] Laurence Calzone, Laurent Tournier, Simon Fourquet, Denis Thieffry, Boris Zhivotovsky, Emmanuel Barillot, and Andrei Zinovyev. Mathematical modelling of cell-fate decision in response to death receptor engagement. *PLoS Computational Biology*, 6(3):e1000702, 2010.
- [100] Laurent Tournier and Madalena Chaves. Uncovering operational interactions in genetic networks using asynchronous Boolean dynamics. *Journal of Theoretical Biology*, 260(2):196–209, 2009.
- [101] Nicola Bonzanni, Abhishek Garg, K. Anton Feenstra, Judith Schutte, Sarah Kingston, Diego Miranda-Saavedra, Jaap Heringa, Ioannis Xenarios, and Berthold Gottgens. Hard-wired heterogeneity in blood stem cells revealed using a dynamic regulatory network model. *Bioinformatics*, 29(13):i80–8, 2013.
- [102] Huilei Xu, Yen-Sin Ang, Ana Sevilla, Ihor R Lemischka, and Avi Ma’ayan. Construction and validation of a regulatory network for pluripotency and self-renewal of mouse embryonic stem cells. *PLoS Computational Biology*, 10(8):e1003777, 2014.
- [103] Mircea Andreucut, Julianne D. Halley, David A. Winkler, and Sui Huang. A general model for binary cell fate decision gene circuits with degeneracy: indeterminacy and switch behavior in the absence of cooperativity. *PLoS One*, 6(5):e19358, 2011.
- [104] Jose Davila-Velderrain, Juan Carlos Martinez-Garcia, and Elena R Alvarez-Buylla. Epigenetic landscape models: the post-genomic era. *bioRxiv*, page 004192, 2014.
- [105] James E Ferrell. Bistability, bifurcations, and Waddington’s epigenetic landscape. *Current Biology*, 22(11):R458–R466, 2012.

REFERENCES

- [106] Sudin Bhattacharya, Qiang Zhang, and Melvin E Andersen. A deterministic map of Waddington’s epigenetic landscape for cell fate specification. *BMC Systems Biology*, 5(1):85, 2011.
- [107] Jin Wang, Li Xu, Erkang Wang, and Sui Huang. The potential landscape of genetic circuits imposes the arrow of time in stem cell differentiation. *Biophysical Journal*, 99(1):29–39, 2010.
- [108] Abhishek Garg, Kartik Mohanram, Giovanni De Micheli, and Ioannis Xenarios. Implicit methods for qualitative modeling of gene regulatory networks. In *Gene Regulatory Networks*, pages 397–443. Springer, 2012.
- [109] Fangting Li, Tao Long, Ying Lu, Qi Ouyang, and Chao Tang. The yeast cell-cycle network is robustly designed. *Proceedings of the National Academy of Sciences of the United States of America*, 101(14):4781–4786, 2004.
- [110] Ethan G Thompson and Timothy Galitski. Quantifying and analyzing the network basis of genetic complexity. *PLoS Computational Biology*, 8(7):e1002583, 2012.
- [111] Carlos Villarreal, Pablo Padilla-Longoria, and Elena R Alvarez-Buylla. General theory of genotype to phenotype mapping: derivation of epigenetic landscapes from N-node complex gene regulatory networks. *Physical Review Letters*, 109(11):118102, 2012.
- [112] Bo Han and Jin Wang. Quantifying robustness and dissipation cost of yeast cell cycle network: the funneled energy landscape perspectives. *Biophysical Journal*, 92(11):3755–3763, 2007.
- [113] Paola Oliveri and Davidson Eric H. Gene regulatory network controlling embryonic specification in the sea urchin. *Current Opinion in Genetics & Development*, 14(4):351–60, 2004.
- [114] Ben D. Macarthur, Avi Ma’ayan, and Ihor R. Lemischka. Systems biology of stem cell fate and cellular reprogramming. *Nature Reviews Molecular Cell Biology*, 10(10):672–81, 2009.
- [115] Sui Huang, Ingemar Ernberg, and Stuart Kauffman. Cancer attractors: A systems view of tumors from a gene network dynamics and developmental perspective. *Seminars in Cell & Developmental Biology*, 20(7):869–76, 2009.
- [116] Regina Samaga, Julio Saez-Rodriguez, Leonidas G. Alexopoulos, Peter K. Sorger, and Steffen Klamt. The logic of EGFR/ErbB signaling: theoretical properties and analysis of high-throughput data. *PLoS Computational Biology*, 5(8):e1000438, 2009.

REFERENCES

- [117] Min Wu, Ri-Qi Su, Xiaohui Li, Tom Ellis, Ying-Cheng Lai, and Xiao Wang. Engineering of regulated stochastic cell fate determination. *Proceedings of the National Academy of Sciences of the United States of America*, 110(26):10610–5, 2013.
- [118] Guoji Guo, Mikael Huss, Guo Qing Tong, Chaoyang Wang, Li Li Sun, Neil D. Clarke, and Paul Robson. Resolution of cell fate decisions revealed by single-cell gene expression analysis from zygote to blastocyst. *Developmental Cell*, 18(4):675–85, 2010.
- [119] Efrat Oron and Natalia Ivanova. Cell fate regulation in early mammalian development. *Physical Biology*, 9(4):045002, 2012.
- [120] Janet Rossant and Patrick P. Tam. Blastocyst lineage formation, early embryonic asymmetries and axis patterning in the mouse. *Development*, 136(5):701–13, 2009.
- [121] Philipp S Hoppe, Daniel L Coutu, and Timm Schroeder. Single-cell technologies sharpen up mammalian stem cell research. *Nature Cell Biology*, 16(10):919–927, 2014.
- [122] Anastasia A Samsonova, Mahesan Niranjana, Steven Russell, and Alvis Brazma. Prediction of gene expression in embryonic structures of *Drosophila melanogaster*. *PLoS Computational Biology*, 3(7):e144, 2007.
- [123] Fuchou Tang, Kaiqin Lao, and M. Azim Surani. Development and applications of single-cell transcriptome analysis. *Nature Methods*, 8(4 Suppl):S6–11, 2011.
- [124] Fuchou Tang, Catalin Barbacioru, Siqin Bao, Caroline Lee, Ellen Nordman, Xiaohui Wang, Kaiqin Lao, and M. Azim Surani. Tracing the derivation of embryonic stem cells from the inner cell mass by single-cell RNA-Seq analysis. *Cell Stem Cell*, 6(5):468–478, 2010.
- [125] Liying Yan, Mingyu Yang, Hongshan Guo, Lu Yang, Jun Wu, Rong Li, Ping Liu, Ying Lian, Xiaoying Zheng, Jie Yan, et al. Single-cell RNA-Seq profiling of human preimplantation embryos and embryonic stem cells. *Nature Structural & Molecular Biology*, 20(9):1131–1139, 2013.
- [126] Zhigang Xue, Kevin Huang, Chaochao Cai, Lingbo Cai, Chun-yan Jiang, Yun Feng, Zhenshan Liu, Qiao Zeng, Liming Cheng, Yi E Sun, et al. Genetic programs in human and mouse early embryos revealed by single-cell RNA sequencing. *Nature*, 500(7464):593–597, 2013.

REFERENCES

- [127] Fernando H Biase, Xiaoyi Cao, and Sheng Zhong. Cell fate inclination within 2-cell and 4-cell mouse embryos revealed by single-cell RNA sequencing. *Genome Research*, 24(11):1787–1796, 2014.
- [128] Ankur P. Parikh, Wei Wu, Ross E. Curtis, and Eric P. Xing. TREEGL: reverse engineering tree-evolving gene networks underlying developing biological lineages. *Bioinformatics*, 27(13):i196–204, 2011.
- [129] Tatsunori Hashimoto, Tommi Jaakkola, Richard Sherwood, Esteban O. Mazzoni, Hynek Wichterle, and David Gifford. Lineage-based identification of cellular states and expression programs. *Bioinformatics*, 28(12):i250–7, 2012.
- [130] Victoria Moignard, Steven Woodhouse, Laleh Haghverdi, Andrew J Lilly, Yosuke Tanaka, Adam C Wilkinson, Florian Buettner, Iain C Macaulay, Wajid Jawaaid, Evangelia Diamanti, et al. Decoding the regulatory network of early blood development from single-cell gene expression measurements. *Nature Biotechnology*, 2015.
- [131] Andrea Ocone, Laleh Haghverdi, Nikola S Mueller, and Fabian J Theis. Reconstructing gene regulatory dynamics from high-dimensional single-cell snapshot data. *Bioinformatics*, 31(12):i89–i96, 2015.
- [132] Karen Sachs, Omar Perez, Dana Pe’er, Douglas A Lauffenburger, and Garry P Nolan. Causal protein-signaling networks derived from multiparameter single-cell data. *Science*, 308(5721):523–529, 2005.
- [133] Riten Mitra, Peter Müller, Peng Qiu, and Yuan Ji. Bayesian hierarchical models for protein networks in single-cell mass cytometry. *Cancer Informatics*, 13(Suppl 4):79, 2014.
- [134] Rosa Karlic, Ho-Ryun Chung, Julia Lasserre, Kristian Vlahovicek, and Martin Vingron. Histone modification levels are predictive for gene expression. *Proceedings of the National Academy of Sciences*, 107(7):2926–2931, 2010.
- [135] Gabriel Cuellar-Partida, Fabian A Buske, Robert C McLeay, Tom Whittington, William Stafford Noble, and Timothy L Bailey. Epigenetic priors for identifying active transcription factor binding sites. *Bioinformatics*, 28(1):56–62, 2012.
- [136] Chengyang Wang, Rui Tian, Qian Zhao, Han Xu, Clifford A Meyer, Cheng Li, Yong Zhang, and X Shirley Liu. Computational inference of mRNA stability from histone modification and transcriptome profiles. *Nucleic Acids Research*, 40(14):6414–6423, 2012.

REFERENCES

- [137] Min Wu, Chee Keong Kwoh, Teresa M Przytycka, Jing Li, and Jie Zheng. Integration of genomic and epigenomic features to predict meiotic recombination hotspots in human and mouse. In *Proceedings of the ACM Conference on Bioinformatics, Computational Biology and Biomedicine*, pages 297–304. ACM, 2012.
- [138] Jianrong Wang, Victoria V. Lunyak, and I. King Jordan. Genome-wide prediction and analysis of human chromatin boundary elements. *Nucleic Acids Research*, 40(2):511–529, 2012.
- [139] Mathieu Gissot, Krystyna A Kelly, James W Ajioka, John M Greally, and Kami Kim. Epigenomic modifications predict active promoters and gene structure in *Toxoplasma gondii*. *PLoS Pathogens*, 3(6):e77, 2007.
- [140] Nathaniel D. Heintzman, Gary C. Hon, R. David Hawkins, Pouya Kheradpour, Alexander Stark, Lindsey F. Harp, Zhen Ye, Leonard K. Lee, Rhona K. Stuart, and Christina W. Ching. Histone modifications at human enhancers reflect global cell-type-specific gene expression. *Nature*, 459(7243):108–112, 2009.
- [141] Jianzhong Su, Yunfeng Qi, Shengqiang Liu, Xueting Wu, Jie Lv, Hongbo Liu, Ruijie Zhang, and Yan Zhang. Revealing epigenetic patterns in gene regulation through integrative analysis of epigenetic interaction network. *Molecular Biology Reports*, 39(2):1701–12, 2012.
- [142] Xiaojiang Xu, Stephen Hoang, Marty W. Mayo, and Stefan Bekiranov. Application of machine learning methods to histone methylation ChIP-Seq data reveals H4R3me2 globally represses gene expression. *BMC Bioinformatics*, 11:396, 2010.
- [143] Chao Cheng, Koon-Kiu Yan, Kevin Y. Yip, Joel Rozowsky, Roger Alexander, Chong Shou, and Mark Gerstein. A statistical framework for modeling gene expression using chromatin features and application to modENCODE datasets. *Genome Biology*, 12(2):R15, 2011.
- [144] Chao Cheng and Mark Gerstein. Modeling the relative relationship of transcription factor binding and histone modifications to gene expression levels in mouse embryonic stem cells. *Nucleic Acids Research*, 40(2):553–68, 2012.
- [145] Li Teng and Kai Tan. Finding combinatorial histone code by semi-supervised biclustering. *BMC Genomics*, 13:301, 2012.
- [146] Misook Ha, Danny W. Ng, Wen-Hsiung Li, and Z. Jeffrey Chen. Coordinated histone modifications are associated with gene expression variation within and between species. *Genome Research*, 21(4):590–8, 2011.

REFERENCES

- [147] Bartek Wilczynski, Ya-Hsin Liu, Zhen Xuan Yeo, and Eileen EM Furlong. Predicting spatial and temporal gene expression using an integrative model of transcription factor occupancy and chromatin state. *PLoS Computational Biology*, 8(12):e1002798, 2012.
- [148] Florian Markowetz, Klaas W Mulder, Edoardo M Airoidi, Ihor R Lemischka, and Olga G Troyanskaya. Mapping dynamic histone acetylation patterns to gene expression in Nanog-depleted murine embryonic stem cells. *PLoS Computational Biology*, 6(12):e1001034, 2010.
- [149] Thais G. do Rego, Helge G. Roider, Francisco A.T. de Carvalho, and Ivan G. Costa. Inferring epigenetic and transcriptional regulation during blood cell development with a mixture of sparse linear models. *Bioinformatics*, 28(18):2297–2303, 2012.
- [150] Robert C. McLeay, Tom Lesluyes, Gabriel Cuellar Partida, and Timothy L. Bailey. Genome-wide in silico prediction of gene expression. *Bioinformatics*, 28(21):2789–2796, 2012.
- [151] Ivan G. Costa, Helge G. Roider, Thais G. do Rego, and Francisco de AT de Carvalho. Predicting gene expression in T cell differentiation from histone modifications and transcription factor binding affinities by linear mixture models. *BMC Bioinformatics*, 12(Suppl 1):S29, 2011.
- [152] Xianjun Dong, Melissa C. Greven, Anshul Kundaje, Sarah Djebali, James B. Brown, Chao Cheng, Thomas R. Gingeras, Mark Gerstein, Roderic Guig, and Ewan Birney. Modeling gene expression using chromatin features in various cellular contexts. *Genome Biology*, 13:R53, 2012.
- [153] Inkyung Jung and Dongsup Kim. Histone modification profiles characterize function-specific gene regulation. *Journal of Theoretical Biology*, 2012.
- [154] Tina Bianco-Miotto, Karen Chiam, Grant Buchanan, Shalini Jindal, Tanya K Day, Mervyn Thomas, Marie A Pickering, Melissa A O’Loughlin, Natalie K Ryan, and Wendy A Raymond. Global levels of specific histone modifications and an epigenetic gene signature predict prostate cancer progression and development. *Cancer Epidemiology Biomarkers & Prevention*, 19(10):2611–2622, 2010.
- [155] Scott Christley, Qing Nie, and Xiaohui Xie. Incorporating existing network information into gene network inference. *PLoS One*, 4(8):e6799, 2009.
- [156] Olivier Gevaert, Steven Van Vooren, and Bart De Moor. A framework for elucidating regulatory networks based on prior information and expression data. *Annals of the New York Academy of Sciences*, 1115(1):240–248, 2007.

REFERENCES

- [157] Mika Gustafsson and Michael Hörnquist. Gene expression prediction by soft integration and the Elastic Net – Best performance of the DREAM3 gene expression challenge. *PLoS One*, 5(2):e9134, 2010.
- [158] Alex Greenfield, Christoph Hafemeister, and Richard Bonneau. Robust data-driven incorporation of prior knowledge into the inference of dynamic regulatory networks. *Bioinformatics*, 29(8):1060–1067, 2013.
- [159] Amira Djebbari and John Quackenbush. Seeded Bayesian Networks: constructing genetic networks from microarray data. *BMC Systems Biology*, 2(1):57, 2008.
- [160] Iti Chaturvedi and Jagath C Rajapakse. Fusion of gene regulatory and protein interaction networks using skip-chain models. In *Pattern Recognition in Bioinformatics*, pages 214–224. Springer, 2008.
- [161] Richard Bonneau, David J Reiss, Paul Shannon, Marc Facciotti, Leroy Hood, Nitin S Baliga, and Vesteinn Thorsson. The Inferelator: an algorithm for learning parsimonious regulatory networks from systems-biology data sets de novo. *Genome Biology*, 7(5):R36, 2006.
- [162] Dirk Husmeier and Adriano V Werhli. Bayesian integration of biological prior knowledge into the reconstruction of gene regulatory networks with Bayesian networks. In *Computational Systems Bioinformatics Conference*, volume 6, pages 85–95, 2007.
- [163] Baikang Pei and Dong-Guk Shin. Reconstruction of biological networks by incorporating prior knowledge into Bayesian network models. *Journal of Computational Biology*, 19(12):1324–1334, 2012.
- [164] Matthew E Studham, Andreas Tjärnberg, Torbjörn EM Nordling, Sven Nelander, and Erik LL Sonnhammer. Functional association networks as priors for gene regulatory network inference. *Bioinformatics*, 30(12):i130–i138, 2014.
- [165] Agung Julius, Michael Zavlanos, Stephen Boyd, and George J Pappas. Genetic network identification using convex programming. *Systems Biology, IET*, 3(3):155–166, 2009.
- [166] Ritwik K Layek, Aniruddha Datta, and Edward R Dougherty. From biological pathways to regulatory networks. *Molecular BioSystems*, 7(3):843–851, 2011.
- [167] Seiya Imoto, Tomoyuki Higuchi, Takao Goto, Kousuke Tashiro, Satoru Kuhara, and Satoru Miyano. Combining microarrays and biological knowledge for estimating gene networks via Bayesian networks. *Journal of Bioinformatics and Computational Biology*, 2(01):77–98, 2004.

REFERENCES

- [168] Jie Zheng, Iti Chaturvedi, and Jagath C. Rajapakse. Integration of epigenetic data in Bayesian network modeling of gene regulatory network. *Pattern Recognition in Bioinformatics*, pages 87–96, 2011.
- [169] Jinyan Li, Qian Liu, and Tao Zeng. Negative correlations in collaboration: Concepts and algorithms. In *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 463–472. ACM, 2010.
- [170] Chris Ding and Hanchuan Peng. Minimum redundancy feature selection from microarray gene expression data. *Journal of Bioinformatics and Computational Biology*, 3(02):185–205, 2005.
- [171] Hitoshi Niwa, Yayoi Toyooka, Daisuke Shimosato, Dan Strumpf, Kadue Takahashi, Rika Yagi, and Janet Rossant. Interaction between Oct3/4 and Cdx2 determines trophoctoderm differentiation. *Cell*, 123(5):917–929, 2005.
- [172] Stephen Frankenberg, François Gerbe, Sylvain Bessonard, Corinne Belville, Pierre Pouchin, Olivier Bardot, and Claire Chazaud. Primitive endoderm differentiates via a three-step mechanism involving Nanog and RTK signaling. *Developmental Cell*, 21(6):1005–1013, 2011.
- [173] Katsuyoshi Takaoka and Hiroshi Hamada. Cell fate decisions and axis determination in the early mouse embryo. *Development*, 139(1):3–14, 2012.
- [174] Amy Ralston and Janet Rossant. Genetic regulation of stem cell origins in the mouse embryo. *Clinical genetics*, 68(2):106–112, 2005.
- [175] Ranadip Pal, Ivan Ivanov, Aniruddha Datta, Michael L. Bittner, and Edward R. Dougherty. Generating Boolean networks with a prescribed attractor structure. *Bioinformatics*, 21(21):4021–4025, 2005.
- [176] Ariel Feiglin, Adar Hacohen, Avital Sarusi, Jasmin Fisher, Ron Unger, and Yanay Ofran. Static network structure can be used to model the phenotypic effects of perturbations in regulatory networks. *Bioinformatics*, 28(21):2811–2818, 2012.
- [177] John H. Holland. Genetic algorithms. *Scientific American*, 267(1):66–72, 1992.
- [178] Kenneth De Jong. Learning with genetic algorithms: An overview. *Machine Learning*, 3(2-3):121–138, 1988.
- [179] David Maxwell Chickering. The WinMine toolkit. *Redmond, WA: Microsoft*, 2002.

REFERENCES

- [180] Christoph Müssel, Martin Hopfensitz, and Hans A Kestler. BoolNet - An R package for generation, reconstruction and analysis of Boolean networks. *Bioinformatics*, 26(10):1378–1380, 2010.
- [181] André Fujita, João R Sato, Humberto M Garay-Malpartida, Rui Yamaguchi, Satoru Miyano, Mari C Sogayar, and Carlos E Ferreira. Modeling gene expression regulatory networks with the sparse vector autoregressive model. *BMC Systems Biology*, 1(1):39, 2007.
- [182] Jagath C. Rajapakse and Piyushkumar A. Mundra. Stability of building gene regulatory networks with sparse autoregressive models. *BMC Bioinformatics*, 12(Suppl 13):S17, 2011.
- [183] Noriyuki Nishioka, Ken-ichi Inoue, Kenjiro Adachi, Hiroshi Kiyonari, Mitsunori Ota, Amy Ralston, Norikazu Yabuta, Shino Hirahara, Robert O Stephenson, Narumi Ogonuki, et al. The Hippo signaling pathway components Lats and Yap pattern Tead4 activity to distinguish mouse trophectoderm from inner cell mass. *Developmental Cell*, 16(3):398–410, 2009.
- [184] Nicholas M. Luscombe, M. Madan Babu, Haiyuan Yu, Michael Snyder, Sarah A. Teichmann, and Mark Gerstein. Genomic analysis of regulatory network dynamics reveals large topological changes. *Nature*, 431(7006):308–312, 2004.
- [185] Thomas MacCarthy, Andrew Pomiankowski, and Robert Seymour. Using large-scale perturbations in gene network reconstruction. *BMC Bioinformatics*, 6(1):11, 2005.
- [186] David J. Rodda, Joon-Lin Chew, Leng-Hiong Lim, Yui-Han Loh, Bei Wang, Huck-Hui Ng, and Paul Robson. Transcriptional regulation of Nanog by Oct4 and Sox2. *Journal of Biological Chemistry*, 280(26):24731–24737, 2005.
- [187] Guangjin Pan and James A. Thomson. Nanog and transcriptional networks in embryonic stem cell pluripotency. *Cell Research*, 17(1):42–49, 2007.
- [188] Vasundhra Kashyap, Naira C. Rezende, Kymora B. Scotland, Sebastian M. Shaffer, Jenny Liao Persson, Lorraine J. Gudas, and Nigel P. Mongan. Regulation of stem cell pluripotency and differentiation involves a mutual regulatory circuit of the Nanog, Oct4, and Sox2 pluripotency transcription factors with polycomb repressive complexes and stem cell microRNAs. *Stem Cells and Development*, 18(7):1093–1108, 2009.

REFERENCES

- [189] Vijay Chickarmane and Carsten Peterson. A computational model for understanding stem cell, trophoctoderm and endoderm lineage determination. *PLoS One*, 3(10):e3478, 2008.
- [190] Fenghao Xu, Huiqin Li, and Tianru Jin. Cell type-specific autoregulation of the caudal-related homeobox gene *Cdx-2/3*. *Journal of Biological Chemistry*, 274(48):34310–34316, 1999.
- [191] Noriyuki Nishioka, Shinji Yamamoto, Hiroshi Kiyonari, Hiroko Sato, Atsushi Sawada, Mitsunori Ota, Kazuki Nakao, and Hiroshi Sasaki. *Tead4* is required for specification of trophoctoderm in pre-implantation mouse embryos. *Mechanisms of Development*, 125(3):270–283, 2008.
- [192] Yojiro Yamanaka, Fredrik Lanner, and Janet Rossant. FGF signal-dependent segregation of primitive endoderm and epiblast in the mouse blastocyst. *Development*, 137(5):715–724, 2010.
- [193] Yojiro Yamanaka and Amy Ralston. Early embryonic cell fate decisions in the mouse. In *The Cell Biology of Stem Cells*, pages 1–13. Springer, 2010.
- [194] Claire Chazaud, Yojiro Yamanaka, Tony Pawson, and Janet Rossant. Early lineage segregation between epiblast and primitive endoderm in mouse blastocysts through the Grb2-MAPK pathway. *Developmental Cell*, 10(5):615–624, 2006.
- [195] Xiaoning Qian, Noushin Ghaffari, Ivan Ivanov, and Edward R. Dougherty. State reduction for network intervention in probabilistic Boolean networks. *Bioinformatics*, 26(24):3098–3104, 2010.
- [196] Soheil Feizi, Daniel Marbach, Muriel Médard, and Manolis Kellis. Network deconvolution as a general method to distinguish direct dependencies in networks. *Nature Biotechnology*, 2013.
- [197] Sophocles J. Orfanidis. *Optimum Signal Processing. An Introduction*. Prentice-Hall, United States, 1996.
- [198] Matthew Rhudy, Brian Bucci, Jeffrey Vipperman, Jeffrey Allanach, and Bruce Abraham. Microphone array analysis methods using cross-correlations. In *ASME 2009 International Mechanical Engineering Congress and Exposition*, pages 281–288. American Society of Mechanical Engineers, 2009.
- [199] Irene Cantone, Lucia Marucci, Francesco Iorio, Maria Aurelia Ricci, Vincenzo Belcastro, Mukesh Bansal, Stefania Santini, Mario di Bernardo, Diego di Bernardo, and Maria Pia Cosma. A yeast synthetic network for in vivo assessment of reverse-engineering and modeling approaches. *Cell*, 137(1):172–181, 2009.

REFERENCES

- [200] Itamar Simon, John Barnett, Nancy Hannett, Christopher T Harbison, Nicola J Rinaldi, Thomas L Volkert, John J Wyrick, Julia Zeitlinger, David K Gifford, Tommi S Jaakkola, et al. Serial regulation of transcriptional regulators in the yeast cell cycle. *Cell*, 106(6):697–708, 2001.
- [201] Paul T. Spellman, Gavin Sherlock, Michael Q. Zhang, Vishwanath R. Iyer, Kirk Anders, Michael B. Eisen, Patrick O. Brown, David Botstein, and Bruce Futcher. Comprehensive identification of cell cycle-regulated genes of the yeast *Saccharomyces cerevisiae* by microarray hybridization. *Molecular Biology of the Cell*, 9(12):3273–3297, 1998.
- [202] Tata Pramila, Wei Wu, Shawna Miles, William Stafford Noble, and Linda L Breen. The forkhead transcription factor *hcm1* regulates chromosome segregation genes and fills the s-phase gap in the transcriptional circuitry of the cell cycle. *Genes & Development*, 20(16):2266–2278, 2006.
- [203] Ola ElBakry, M. Omair Ahmad, and M.N.S. Swamy. Inference of gene regulatory networks with variable time delay from time-series microarray data. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 10(3):671–687, 2013.
- [204] Bonita J. Brewer, Ewa Chlebowicz-Sledziowska, and Walton L. Fangman. Cell cycle phases in the unequal mother/daughter cell cycles of *saccharomyces cerevisiae*. *Molecular and cellular biology*, 4(11):2529–2531, 1984.
- [205] Wolfgang Fischle, Yanming Wang, and C. David Allis. Binary switches and modification cassettes in histone biology and beyond. *Nature*, 425(6957):475–479, 2003.
- [206] Hironobu Kimura, Masako Tada, Norio Nakatsuji, and Takashi Tada. Histone code modifications on pluripotential nuclei of reprogrammed somatic cells. *Molecular and Cellular Biology*, 24(13):5710–5720, 2004.
- [207] Raphael Margueron, Patrick Trojer, and Danny Reinberg. The key to development: interpreting the histone code? *Current opinion in genetics & development*, 15(2):163–176, 2005.
- [208] Gaurvika M. Nayyar, Joel G. Breman, Paul N. Newton, and James Herrington. Poor-quality antimalarial drugs in southeast Asia and sub-Saharan Africa. *The Lancet Infectious Diseases*, 12(6):488–96, 2012.
- [209] Malcolm J. Gardner, Neil Hall, Eula Fung, Owen White, Matthew Berriman, Richard W. Hyman, Jane M. Carlton, Arnab Pain, Karen E. Nelson, Sharen Bowman, Ian T. Paulsen, Keith James, Jonathan A. Eisen, Kim Rutherford,

REFERENCES

- Steven L. Salzberg, Alister Craig, Sue Kyes, Man-Suen Chan, Vishvanath Nene, Shamira J. Shallom, Bernard Suh, Jeremy Peterson, Sam Angiuoli, Mihaela Pertea, Jonathan Allen, Jeremy Selengut, Daniel Haft, Michael W. Mather, Akhil B. Vaidya, David M. Martin, Alan H. Fairlamb, Martin J. Fraunholz, David S. Roos, Stuart A. Ralph, Geoffrey I. McFadden, Leda M. Cummings, G. Mani Subramanian, Chris Mungall, J. Craig Venter, Daniel J. Carucci, Stephen L. Hoffman, Chris Newbold, Ronald W. Davis, Claire M. Fraser, and Bart Barrell. Genome sequence of the human malaria parasite *Plasmodium falciparum*. *Nature*, 419(6906):498–511, 2002.
- [210] Zbynek Bozdech, Manuel Llinás, Brian Lee Pulliam, Edith D. Wong, Jingchun Zhu, and Joseph L. DeRisi. The transcriptome of the intraerythrocytic developmental cycle of *Plasmodium falciparum*. *PLoS Biology*, 1(1):e5, 2003.
- [211] Balbir K. Chaal, Archana P. Gupta, Brigitta D. Wastuwidyaningtyas, Yen-Hoon Luah, and Zbynek Bozdech. Histone deacetylases play a major role in the transcriptional regulation of the *Plasmodium falciparum* life cycle. *PLoS Pathogens*, 6(1):e1000737, 2010.
- [212] Nria Rovira-Graells, Archana P. Gupta, Evarist Planet, Valerie M. Crowley, Sachel Mok, Llus Ribas de Pouplana, Peter R. Preiser, Zbynek Bozdech, and Alfred Corts. Transcriptional variation in the malaria parasite *Plasmodium falciparum*. *Genome Research*, 22(5):925–938, 2012.
- [213] Ferhat Ay, Evelien M Bunnik, Nelle Varoquaux, Jean-Philippe Vert, William Stafford Noble, and Karine G Le Roch. Multiple dimensions of epigenetic gene regulation in the malaria parasite *Plasmodium falciparum*. *BioEssays*, 37(2):182–194, 2015.
- [214] Archana P. Gupta, Wai Hoe Chin, Lei Zhu, Sachel Mok, Yen-Hoon Luah, Eng-How Lim, and Zbynek Bozdech. Dynamic epigenetic regulation of gene expression during the life cycle of malaria parasite *Plasmodium falciparum*. *PLoS Pathogens*, 9(2):e1003170, 2013.
- [215] Rakesh Agrawal, Tomasz Imieliski, and Arun Swami. Mining association rules between sets of items in large databases. In *ACM SIGMOD Record*, volume 22, pages 207–216. ACM, 1993.
- [216] Qingfeng Chen and Yi-Ping Phoebe Chen. Mining frequent patterns for AMP-activated protein kinase regulation on skeletal muscle. *BMC Bioinformatics*, 7:394, 2006.

REFERENCES

- [217] Xochitl C. Morgan, Shulin Ni, Daniel P. Miranker, and Vishwanath R. Iyer. Predicting combinatorial binding of transcription factors to regulatory elements in the human genome by association rule mining. *BMC Bioinformatics*, 8:445, 2007.
- [218] Francisco J. Lopez, Armando Blanco, Fernando Garcia, Carlos Cano, and Antonio Marin. Fuzzy association rules for biological data analysis: A case study on yeast. *BMC Bioinformatics*, 9:107, 2008.
- [219] Sotiris Kotsiantis and Dimitris Kanellopoulos. Association rules mining: A recent overview. *GESTS International Transactions on Computer Science and Engineering*, 32(1):71–82, 2006.
- [220] Qiankun Zhao and Sourav S. Bhowmick. Association rule mining: A survey. <https://www.lri.fr/~antoine/Courses/Master-ISI/Regle-association.pdf>, 2003.
- [221] Rakesh Agrawal and Ramakrishnan Srikant. Fast algorithms for mining association rules. In *Proceedings of the 20th International Conference on Very Large Data Base (VLDB)*, volume 1215, pages 487–499, 1994.
- [222] Mohammed Javeed Zaki. Scalable algorithms for association mining. *IEEE Transactions on Knowledge and Data Engineering*, 12(3):372–390, 2000.
- [223] Jiawei Han, Jian Pei, and Yiwen Yin. Mining frequent patterns without candidate generation. In *ACM SIGMOD Record*, volume 29, pages 1–12. ACM, 2000.
- [224] Takeaki Uno, Tatsuya Asai, Yuzo Uchida, and Hiroki Arimura. LCM: An efficient algorithm for enumerating frequent closed item sets. In *Proceedings of the IEEE ICDM Workshop of Frequent Itemset Mining Implementations (FIMI), Melbourne, Florida, USA*. Citeseer, 2003.
- [225] Ronald A. Fisher. On the interpretation of Chi-Square from contingency tables, and the calculation of P. *Journal of the Royal Statistical Society*, pages 87–94, 1922.
- [226] Geoffrey I. Webb. Discovering significant patterns. *Machine Learning*, 68(1):1–33, 2007.
- [227] Guimei Liu, Haojun Zhang, and Limsoon Wong. Controlling false positives in association rule mining. *Proceedings of the VLDB Endowment*, 5(2):145–156, 2011.
- [228] Bradley E. Bernstein, Emily L. Humphrey, Rachel L. Erlich, Robert Schneider, Peter Bouman, Jun S. Liu, Tony Kouzarides, and Stuart L. Schreiber. Methylation of histone H3 Lys 4 in coding regions of active genes. *Proceedings of the National Academy of Sciences*, 99(13):8695–700, 2002.

REFERENCES

- [229] Siavash K. Kurdistani, Saeed Tavazoie, and Michael Grunstein. Mapping global histone acetylation patterns to gene expression. *Cell*, 117(6):721–33, 2004.
- [230] David Maxwell Chickering. A transformational characterization of equivalent Bayesian network structures. In *Proceedings of the Eleventh conference on Uncertainty in artificial intelligence*, pages 87–98. Morgan Kaufmann Publishers Inc., 1995.
- [231] Daniel Lowd. The Libra Toolkit. <http://libra.cs.uoregon.edu/>. Accessed: Jan. 2013.
- [232] Dirk Husmeier. Sensitivity and specificity of inferring genetic regulatory interactions from microarray experiments with dynamic Bayesian networks. *Bioinformatics*, 19(17):2271–2282, 2003.
- [233] Nir Friedman, Kevin Murphy, and Stuart Russell. Learning the structure of dynamic probabilistic networks. In *Proceedings of the Fourteenth conference on Uncertainty in artificial intelligence*, pages 139–147. Morgan Kaufmann Publishers Inc., 1998.
- [234] W.K. Hastings. Monte Carlo sampling methods using Markov chains and their applications. *Biometrika*, 57(1):97–109, 1970.
- [235] Siddhartha Chib and Edward Greenberg. Understanding the metropolis-hastings algorithm. *The American Statistician*, 49(4):327–335, 1995.
- [236] Sonja Althammer and Eduardo Eyras. Predictive models of gene regulation from high-throughput epigenomics data. *Comparative and Functional Genomics*, 2012, 2012.
- [237] James Dougherty, Ron Kohavi, and Mehran Sahami. Supervised and unsupervised discretization of continuous features. In *Machine Learning: Proceedings of the twelfth international conference*, pages 194–202. Morgan Kaufmann Publishers, Inc., 1995.
- [238] Dmitry K. Pokholok, Christopher T. Harbison, Stuart Levine, Megan Cole, Nancy M. Hannett, Tong Ihn Lee, George W. Bell, Kimberly Walker, P. Alex Rolfe, Elizabeth Herbolzheimer, Julia Zeitlinger, Fran Lewitter, David K. Gifford, and Richard A. Young. Genome-wide map of nucleosome acetylation and methylation in yeast. *Cell*, 122(4):517–527, 2005.

REFERENCES

- [239] Dieter Egli, Garrett Birkhoff, and Kevin Eggan. Mediators of reprogramming: Transcription factors and transitions through mitosis. *Nature Reviews Molecular Cell Biology*, 9(7):505–516, 2008.
- [240] John B. Gurdon. The developmental capacity of nuclei taken from intestinal epithelium cells of feeding tadpoles. *Journal of Embryology and Experimental Morphology*, 10(4):622–640, 1962.
- [241] I. Wilmut, A.E. Schnieke, J. McWhir, A.J. Kind, and K.H. Campbell. Viable offspring derived from fetal and adult mammalian cells. *Nature*, 385:810–813, 1997.
- [242] Kazutoshi Takahashi and Shinya Yamanaka. Induction of pluripotent stem cells from mouse embryonic and adult fibroblast cultures by defined factors. *Cell*, 126(4):663–676, 2006.
- [243] Maxim N. Artyomov, Alexander Meissner, and Arup K. Chakraborty. A model for genetic and epigenetic regulatory networks identifies rare pathways for transcription factor induced pluripotency. *PLoS Computational Biology*, 6(5):e1000785, 2010.
- [244] Shinya Yamanaka. Elite and stochastic models for induced pluripotent stem cell generation. *Nature*, 460(7251):49–52, 2009.
- [245] Valdimir Filkov. Identifying gene regulatory networks from gene expression data. *Handbook of Computational Molecular Biology*, pages 27.1–27.29, 2005.
- [246] Harold D. Kim, Tal Shay, Erin K. O’Shea, and Aviv Regev. Transcriptional regulatory circuits: predicting numbers from alphabets. *Science*, 325(5939):429–432, 2009.
- [247] Baruch Barzel and Albert-László Barabási. Network link prediction by global silencing of indirect correlations. *Nature Biotechnology*, 31(8):720–725, 2013.
- [248] Ahsan Raja Chowdhury and Madhu Chetty. Network decomposition based large-scale reverse engineering of gene regulatory network. *Neurocomputing*, 160:213–227, 2015.
- [249] Masaru Tomita, Kenta Hashimoto, Koichi Takahashi, Thomas Simon Shimizu, Yuri Matsuzaki, Fumihiko Miyoshi, Kanako Saito, Sakura Tanida, Katsuyuki Yugi, J Craig Venter, et al. E-CELL: software environment for whole-cell simulation. *Bioinformatics*, 15(1):72–84, 1999.
- [250] Jonathan R. Karr, Jayodita C. Sanghvi, Derek N. Macklin, Miriam V. Gutschow, Jared M. Jacobs, Benjamin Bolival, Nacyra Assad-Garcia, John I. Glass, and Markus W. Covert. A whole-cell computational model predicts phenotype from genotype. *Cell*, 150(2):389–401, 2012.