UNIVERSITY OF CALIFORNIA, SAN DIEGO

Computational inference of transcriptional regulation in Eukaryotes

A dissertation submitted in partial satisfaction of the requirements for the degree
Doctor of Philosophy

in

Chemistry

by

Jie Liu

Committee in charge:

Professor Wei Wang, Chair
Professor J. Andrew McCammon, Co-Chair
Professor Katja Lindenberg
Professor Shankar Subramaniam
Professor Milton Saier
Professor Peter Wolynes

2010

The Dissertation of Jie Liu is approved, and it is acceptable in quality and form for publication on microfilm and electronically:

Co-Chair

Chair

University of California, San Diego

2010

TABLE OF CONTENTS

LIST OF ABBREVIATIONS

CDK	cyclin-dependent kinase
CDF	cumulative distribution function
ChIP	chromatin immunoprecipitation
CP	conditional probability
DAG	directed acyclic graph
DBN	Dynamic Bayesian network
EM	expectation maximization
GO	Gene Ontology
GRN	gene regulatory network
ICA	independent components analysis
JPD	joint probability distribution
MARS	multivariate adaptive regression splines
MET	methionine biosynthetic
MLE	maximum likelihood estimation
NCR	nitrogen catabolite repression
NGS	next-generation sequencing
NMF	non-negative matrix factorizations
NPV	negative predictive value
ODE	ordinary differential equation
PBN	Probabilistic Boolean Network
PCA	principal component analysis
PPI	protein-protein interaction
PPV	positive predictive value
PSFM	position-specific frequency matrix
PSWM	position specific weight matrix
SAM	S-adenosylmethionine
SOM	self-organizing map
SVD	singular value decomposition
TF	transcription factor
TFBS	TF binding site
TFPE	TF perturbation experiment

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGEMENTS

Seven years may be nothing more than a numeric symbol to many people, but it means a lot to me. From a layman to a well-trained “skilled researcher”, I have learned many things that are invaluable to my future career.

I would like to acknowledge Professor Wei Wang for his support as the chair of my committee. He not only served as my supervisor but also encouraged and challenged me throughout my entire academic program. I will always remember the numerous discussions with him, his dedication to research, and his encouragement when I was in difficulty.

I am also deeply grateful to Professor J. Andrew McCammon, who offered me this precious opportunity and financial support to pursue my PhD at UCSD. He gave me a lot of valuable suggestions in the process of my PhD study. It has been my honor to work with Professor McCammon.

I also want to express my gratitude to Dr. Ron Yu and Dr. Li Shen for their cooperation on my research.

Chapter 2, in full, is a reprint of the material as it appears in Identification of direct target genes using joint sequence and expression likelihood with application to DAF-16 Yu, Ron X.; Liu, Jie; True, Nick; Wang, Wei. PLoS One. 2008; 3(3):e1821. The dissertation author was a co-author of this paper.

Chapter 3, in full, is currently being prepared for submission for publication of the material. Liu, Jie; Wang, Wei. The dissertation author was the primary investigator and author of this material.

Chapter 4, in full, is currently being prepared for submission for publication of the material. Liu, Jie; Yu, Ron X.; Wang, Wei. The dissertation author was the primary investigator and author of this material.

VITA

1998	Bachelor of Biochemistry, Beijing Normal University
2005	Master of Physical Chemistry, University of California, San Diego
2003	Teaching Assistant, Department of Chemistry and Biochemistry University of California, San Diego
2004-2009	Research Assistant, Department of Chemistry and Biochemistry University of California, San Diego
2009	Teaching Assistant, Department of Chemistry and Biochemistry University of California, San Diego
2010	Doctor of Philosophy, University of California, San Diego

PUBLICATIONS

"Identification of direct target genes using joint sequence and expression likelihood with application to DAF-16" Yu RX, Liu J, True N, Wang W. PLoS One. 2008; 3(3):e1821

"GBNet: deciphering regulatory rules in the co-regulated genes using a Gibbs sampler enhanced Bayesian network approach" Shen L, Liu J, Wang W. BMC Bioinformatics. 2008; 9:395

"CompMODEM: Prediction of regulatory interactions between transcription factors and their targets" Liu J, Wang W. Manuscript in preparation

"ActivMiner: Simultaneous Inference of Transcription Factor Activity and Target Genes" Liu J, Yu RX., Wang W. Manuscript in preparation

ABSTRACT OF THE DISSERTATION

Computational inference of transcriptional regulation in Eukaryotes

by

Jie Liu

Doctor of Philosophy in Chemistry

University of California, San Diego, 2010

Professor Wei Wang, Chair
Professor J. Andrew McCammon, Co-Chair

Inference of transcriptional regulation, which includes discovering binding sites of a transcription factor (TF), identifying its direct target genes and detecting its dynamical activity, is an important step towards reconstructing transcription network. In this dissertation, I have developed three novel computational methods to tackle this task by integrating large-scale genomic data.

All three methods train a probabilistic model using sequence motif, gene expression, TF binding and conservation data. This probabilistic model provides an elegant way to reduce noise in individual data by integrating multiple sources of data. Mathematically, they maximize the joint likelihood of the observable data using expectation-maximization (EM) method. The hidden variables in the models represent

the identity of a gene (target or not in TRANSMODIS and CompMODEM) and the activity of a TF (in ActivMiner). The EM algorithm iteratively determines these hidden variables and the parameters in the models.

The three methods have different purposes. The first two methods called TRANSMODIS and CompMODEM aim to identify binding sites and direct target genes of TFs. TRANMODIS takes into account that target genes of a TF normally share similar sequence motifs in the TF binding regions and gene expression patterns under different conditions. If only a single gene expression or TF binding experiment is available, in addition to the sequence and expression information, CompMODEM considers conservation of TF binding sites in the model because functional regulatory sites tend to be evolutionarily constrained. Both TRANSMODIS and CompMODEM assume the TF of interest is active. When such information is not available, ActivMiner aims to simultaneously infer the dynamic activity of TFs and their regulatory targets.

These methods have been successfully applied to multiple species including human, worm and yeast. The studies presented in this dissertation lay the foundation of inferring gene regulatory network, which is a great challenge in the post-genome era. With the fast accumulation of genomic data, these methods will provide a set of useful tools to understand transcriptional mechanisms.

Chapter 1. Introduction

1.1. Introduction to gene regulatory networks

The central dogma of molecular biology describes that transcription and translation are the two main steps while a protein is synthesized in a living cell. Transcription is a sophisticated multi-step cascade, in which RNA polymerase II (Pol II) transcribes a DNA template into a messenger RNA in concert with a wide range of transcription initiation, elongation, capping, termination, and histone modifying factors. During the process of transcription, it is well known how a single strand of mRNA is produced from a double stranded DNA template, but how the regulated transcription machinery is recruited conditionally still remains unclear. This regulation controls when a particular transcription is triggered or prevented and how much a specific mRNA needs to be created in response to different environmental stimuli. It is one of the most important and complicated processes in transcription. In the study of transcriptional regulation, biologists define a gene regulatory network (GRN) as a collection of regulatory proteins and their closely associated genes across a genome, within which the regulatory proteins interact with each other and directly and/or indirectly with their target genes, and thereby govern the expression lever of the target genes according to the external environment. The transcription factors (TFs), which initiate and subsequently control the gene expression, are the most important regulatory proteins in regulatory networks or cascades. They normally contain one or more DNA-binding domains, which usually attach to either enhancer or promoter regions of DNA adjacent to the genes that

they regulate. These specific non-coding DNA binding sites, also called binding motifs or cis-regulatory elements, are highly evolutionarily conserved across different specials, which is an important and valuable character utilized by many biologists to identify these binding sites. After binding to DNA, the TFs perform their functions alone or with other TFs or other regulatory proteins, such as coactivators and methylases, by promoting (as an activator), or blocking (as a repressor) the recruitment of RNA polymerase to their target genes. There are a wide variety of mechanisms for the regulation of gene expression, such as (1) stabilizing or blocking the binding of RNA polymerase to DNA, (2) catalyzing the acetylation or deacetylation of histone proteins (The TF can either do so directly or recruit other proteins with this catalytic activity), and (3) recruiting coactivator or corepressor proteins to the TF DNA complex. The combinatorial mechanism of regulation among a relatively small number of TFs and other regulatory proteins is a prerequisite for the efficient and unique control of expression of every single gene in a huge whole genome. In addition to the basal transcription regulation, the gene regulatory networks involve in many other vital cellular processes, such as cell cycle, development, and intercellular signaling. Due to their important roles in these biological processes, the gene regulatory networks have been associated with some human diseases like diabetes and cancers. This clinical importance makes many TFs the potential direct or indirect drug targets. For instance, approximately 10% of currently prescribed drugs directly target the nuclear receptor class of TFs, including tamoxifen and bicalutamide for the treatment of breast and prostate cancer, respectively, and the various types of anti-inflammatory and anabolic.

In recent years, the realization of gene regulatory networks has become the major challenge and goal of the systems biology enterprise. Because the networks themselves are extremely complex and combinatorial, and the available data for the construction of networks are often inaccurate and defective, this job has not yet been completed flawlessly over the last few decades. Though absolutely difficult, it still has attracted close attention from many researchers. The accomplishment of the transcriptional regulation networks will help to unearth the fundamental principle of sophisticated cellular processes, which has so far been poorly understood but is obviously beneficial to clinical demand. This task is so arduous and promising that both experimental and computational biologists have been very enthusiastic to cooperate with each other to build up the gene regulatory networks by utilizing the complementary techniques. Although the underlying detailed mechanism of the transcriptional regulation has still remained unknown, strictly speaking, a huge amount of valuable experimental data has been accumulated acceleratedly by the biologists. Owing to the various and massive information, a number of computational approaches to construct gene regulatory networks has been systematically and rapidly developed to address this challenge in the last few decades.

All these approaches usually utilize one of the two major fundamental learning strategies, namely bottom-up and top-down. In the bottom-up approaches, the individual biochemical processes involved in the regulatory pathway are first specified in great detail. In contrast, the top-down approaches attempt to reverse engineer regulatory networks initially from high-throughput data sources. The bottom-up approaches have mainly been applied to small, well-defined biological model system, while the

complementary top-down approaches have been mostly used to investigate the high-throughput data sources in the entire genome. The top-down approaches have become increasingly popular and largely promoted by advances in high-throughput experimental technologies and rapid development in computational approaches, because the former makes it possible to measure the global response of a biological system to specific interventions and the latter is capable of systematically integrating and analyzing various high-throughput experimental data.

1.2. Omics data in the inference of gene regulatory networks

The inference of GRNs is greatly promoted by advances in high-throughput technologies, which have provided abundant different types of 'omics' data such as genomic data (DNA level), transcriptomic data (mRNA level), and proteomic data (protein level). In the following, the main characteristics and application of diverse omics data are outlined and their relative merits and demerits are discussed.

1.2.1. Microarray data

The most original yet still popular high-throughput data used to construct gene regulatory networks are the microarray data that describe the system's response to time-series, cell-specific, tissue-specific and/or perturbation experiments. [1] In the time-series experiments, the transcriptional profiles that are acquired under each of the examined sets of conditions correspond to different, sequential in time snapshots of the biological

process/system under investigation. In this case, it is of interest to identify the genes whose expression profile over time changes drastically due to the applied perturbation. Moreover, it would be of interest to compare the various timepoints with respect to the change in their transcriptional profile due to the applied perturbation, with the consideration that they are components of the same time-series. [2; 3] Cell-specific and tissue-specific gene expression is a fundamental aspect of multicellular biology, underlying the development, function, and maintenance of diverse cell types within an organism. Accounting for cell/tissue-specific expression is a precursor to any system-level understanding of metazoan organismal development and function. Microarray experiments analyzing cell/tissue-specific expression are able to discover cell/tissue-specific genes based on the difference in mRNA levels and generate networks of cell/tissue-specific functional interactions. [4] Different perturbation experiments can be designed depending on the techniques available and the system of interest, and include manipulations of environmental factors as well as interventions on the genetic, transcriptomic, proteomic or metabolic level. Environmental perturbation experiments include heat shock, chemical stresses, compound-treatments, and so on. [5; 6] DNA knock-out and over-expression experiments are two traditional genetic perturbation experiments. In addition, the RNAi knock-down experiment is one of the well-known perturbation experiments on the transcriptomic level. TF perturbation experiments (TFPEs) are one special class of microarrays, in which the only perturbation is deletion, mutation or over-expression of a TF. TFPEs including DNA knock-out or over-expression experiments and RNAi knock-down experiments are very important for the GRNs reconstruction because they focus on a concerned TF and provide the competent

evidence in the influence of this TF over their target genes. The relatively gene expression alteration measured by genome-wide DNA microarrays represents the systematically functional reaction to the corresponding perturbation. [7]

The analysis methods for microarray data are mainly divided into three categories: clusterings, model-based approaches and matrix decompositions. [8-14] Clustering approaches classify genes into distinct groups or organize them hierarchically according to their expression patterns across different time, treatments, and tissues. The genes in the same cluster with the similar gene expression pattern are assumed to be potentially functionally related or regulated by a common TF or a common set of TFs. This means that the genes sharing similar expression patterns tend to be the candidates of the target genes of a TF or TF complex. Most clustering methods do not attempt to model the underlying biology. A disadvantage of such methods is that they partition genes or experiments into mutually exclusive clusters, whereas in reality a gene or an experiment may belong to multiple biological processes. Clustering methods alone are imprecise, because they only indicate co-expression, but do not directly identify co-regulation. Model-based approaches first generate a probabilistic model that explains the interactions among biological entities participating in GRNs, and then train the parameters or latent variables of the model on the large microarray datasets. Because microarray data are not sufficient for inferring GRNs, model-based approaches usually integrate multiple sources of data with microarray data. Matrix decompositions consider the microarray data as a genes × arrays matrix and decompose it into components that have a desired property. Unlike clustering methods, matrix decompositions are able to assign each single gene different memberships in multiple groups.

1.2.2. Genome sequence data

Genome sequence data have been the prime genomic data used to build GRNs when the availability of complete genome sequences for all genes creates the opportunities for identifying cis-regulatory elements that control gene expression. The analysis of sequence data mainly focuses on the investigation of TF binding sites (TFBSs), because TFBS motifs occur in many regions of non-coding DNA sequence such as promoters, enhancers and silencers. A widely used strategy first clusters genes based on their expression profile across multiple conditions and then searches for over-represented DNA motifs in the regulatory regions of each gene cluster. [15] In addition, since cis-regulatory elements are functional and subjected to evolutionary selection, they evolve less rapidly than the surrounding non-coding regions within closely related organisms. Therefore evolutionarily conserved motifs of orthologous genes in related species are more likely to be true cis-regulatory elements. [16] However, the appearance of TFBSs doesn't necessarily indicate the physical binding between the regulatory proteins and DNA, which depends on the secondary and teritary structure besides the primary structure of the DNA.

1.2.3. Chromatin immunoprecipitation data

The main technique for assaying the actual protein-DNA binding in vivo is chromatin immunoprecipitation (ChIP). A precise map of binding sites for TFs, core transcriptional machinery and other DNA-binding proteins is vital for deciphering the

GRNs that underlie various biological processes. The combination of nucleosome positioning and dynamic modification of DNA and histones is essential in gene regulation and guides a cell's development and differentiation. Chromatin states can influence transcription directly by altering the packaging of DNA to allow or prevent access to DNA-binding proteins, or they can modify the nucleosome surface to enhance or impede recruitment of effecter protein complexes. ChIP experiments have become the indispensable tool for studying these mechanisms. In ChIP, antibodies are used to select specific proteins or nucleosomes, enriching DNA fragments bounded to these proteins or nucleosomes. The introduction of microarrays allowed the fragments obtained from ChIP to be identified by hybridization to a microarray (ChIP-chip) [17; 18], therefore enabling a genome-scale view of DNA-protein interactions. Owing to the rapid technological developments in next-generation sequencing (NGS) [19], chromatin immunoprecipitation followed by sequencing (ChIP-seq), one of the early applications of NGS, has become the novel promising method. In ChIP-seq, the DNA fragments of interest are sequenced directly instead of being hybridized on an array. ChIP-seq offers many advantages over ChIP-chip and therefore provides substantially improved data. First, its base pair resolution is the greatest improvement over ChIP-chip. Generally, the maximum resolution is around 30-100bp in ChIP-chip, while it can be a single nucleotide in ChIP-seq. Second, nucleic acid hybridization in ChIP-chip is complex and dependents on many factors, including the GC content, length, concentration and secondary structure of the target and probe sequences. Therefore, cross-hybridization between imperfectly matched sequences frequently occurs and contributes to the main noise in ChIP-chip. ChIP-seq does not suffer from the noise generated by the hybridization step in ChIP-chip. Third,

the intensity signal measured on arrays might not be linear over its entire range, and its dynamic range is limited below and above saturation points. In a recent study, distinct and biologically meaningful peaks seen in ChIP-seq were obscured when the same experiment was conducted with ChIP-chip. Fourth, in ChIP-seq the genome coverage is not limited by the repertoire of probe sequences fixed on the array. This is particularly important for the analysis of repetitive regions of the genome, which are typically masked out on arrays. Fifth, ChIP-seq needs less amount of ChIP DNA sample, requires less amplification than its array-based predecessor ChIP-chip. Finally, multiplexing is possible in ChIP-seq but not in ChIP-chip. [20]

With the progress of genome-wide location analysis for DNA-binding proteins provided by ChIP-chip and ChIP-seq technology, direct experimental evidence of TFBSs in regulatory regions has become available. Therefore, a large number of computational programs have been developed to predict the TFBSs by applying the ChIP data. [21; 22] However, the physical bound presence of a regulator to a regulatory region doesn't imply the functional happening, and, like expression data, is still noisy and limited to the particular physiological conditions of the experimental protocol used.

1.2.4. Proteomic data

Like the genomic and transcriptomic data, the proteomic data can also be used to reconstruct GRNs. Mass spectrometric protein identification and the yeast two-hybrid system have greatly promoted the study of protein-protein interactions. [23; 24] Proteins often form complexes with other proteins to achieve specific function and activity. The

regulatory proteins, which consist of TFs, coactivators, chromatin remodelers, histone acetylases, deacetylases, kinases, methylases, and so on, interact with each other and with DNA to form the transcription initiation complex, and stimulate or repress transcription according to the external environment. Therefore, protein-protein interactions provide valuable information about the combinatorial regulation, which is essential and important in GRNs. Increasing knowledge on the molecular mechanisms underlying gene regulation will eventually allow regulatory systems to be modeled on a fine level of granularity. However, the application of proteomic data in GRN research is usually difficult, because the structural variety of proteins and their functional interactions cause a high degree of complexity.

1.2.5. Genome Annotation data

Gene functional annotations have been recently used to promote the study of genes and their regulatory interactions. The Gene Ontology (GO) project [25] is a major bioinformatics initiative with the aim of standardizing the representation of gene and gene product attributes across species and databases. This project has developed three structurally controlled vocabularies (ontologies) that describe gene products in terms of their associated biological processes, cellular components and molecular functions in a species-independent manner. The functional annotations in the GO database are organized in a hierarchical way defining subsets of gene that share common biological functions in a GRN. But the fact that many annotations in Gene Ontology are quite incomplete limits the application of the annotation information.

1.2.6. Literature resource

A novel data resource of wealthy biological information found in the scientific literature has increasingly attracted eyeballs from the researchers. Many text mining tools have been developed to automatically extract interrelations between genes and proteins from literature with sufficient reliability and thus provide valuable information for GRN modeling. [26] Nevertheless, these tools have not yet been brought into wide use because of the complication and immaturity of the text mining algorithm.

1.3. Computational approaches for identifying gene modules

Various omics experimental technologies, like microarray, ChIP, next generation sequencing, and so on, provide massive valuable high-throughput data for elucidating how genes interact with each other and how a cell's regulatory network controls vast batteries of genes simultaneously according to different external environments. However, none of these omics data is perfect and sufficient to be utilized alone in the construction of GRNs. Firstly, each individual omics data resource is noisy and incomplete due to the inevitable measurement errors. For example, the main painful errors in most microarray and ChIP-chip experiments are caused by the hybridization, a necessary step in these experiments. Secondly, A GRN has a large and complex network structure. Therefore, a single experiment is usually unable to provide the thorough and direct evidence of what reactions (the edges) really happen among the genes (the nodes) in this network. For

instance, the ChIP data can only indicate the physical binding between a TF and its potential binding site, but cannot imply the functional binding that triggers the transcriptional regulation. On the other hand, the microarray experiments provide the proof of a TF's functional effect on both direct targets, whose regulatory regions are bound by the TF, and indirect targets, which is regulated by the targets of the TF. In other words, microarrays cannot distinguish direct and indirect targets of the TF.

To overcome imperfection and inaccuracy caused by an individual omics data set, the computational biologists have developed a lot of statistical algorithms to reliably infer GRNs by integrating the diverse and complementary biological resources. Since the information provided by each individual omics data set is incomplete, computational models utilizing single data set inevitably suffer from the trade-off between specificity and sensitivity. The primary benefit of integrating the complementary data sets is the improvement of both the specificity and sensitivity. For instance, in regard to identification of the direct targets of a TF, the traditional methods utilize expression and binding data separately. The methods using only expression data are unable to isolate the direct targets from the indirect ones because the change of expression level of indirect targets is insignificant and these methods filter erroneously the true direct targets with relatively small expression change due to the arbitrary cut-off. Similarly, the methods using only the binding information usually predict false targets, which have the binding sites only by chance, and miss the true targets because of their relatively weak binding to the TF. In contrast, the advanced methods integrate both expression and binding information to successfully identify the true direct targets, whose expression alters affected by the TF and whose binding sites exist. These methods are able to filter out

both the indirect targets without binding sites and the false targets with stochastic binding sites and non-expression. Furthermore, the real targets with either small expression or weak binding can be recognized with the help of the strong complementary binding and expression signals, respectively. Thereby, the algorithms integrating multiple types of omics data have been rapidly developed over recent years to address the inference of the GRNs. The early stage of their application is to identify the gene modules, the major approaches of which will be summarized in this section. The advanced stage is to infer the GRNs, which will be discussed in the next section.

A gene module is defined as a set of co-regulated genes, to which the same set of TFs binds directly. Most of the approaches to identify the gene modules focus on the discovery of the binding sites, the direct targets, the activities and the dependence of a TF and/or a set of co-regulating TFs. An incomplete list of these approaches includes k-means, self-organizing map (SOM), hierarchical clustering, linear regression, Gibbs sampling, expectation-maximization algorithm, singular value decomposition (SVD), independent components analysis (ICA).

1.3.1. Hierarchical Clustering

The primitive hierarchical clustering methods are unsupervised and mainly employ the multiple microarray data to indentify the co-regulated gene targets. The whole genome genes are organized hierarchically into distinct clusters according to similarity in patterns of gene expression across time, cells, tissues, treatments, and so on. A mathematical measurement of similarity is described by the correlative relationships

between two genes, such as the Euclidean distance, Pearson correlation coefficient, or Spearman rank correlation coefficient of the two n-dimensional vectors (genes) representing a series of n measurements. The genes in the same cluster are assumed to be potentially functionally related with or influenced by a common TF or a common set of co-TFs.

Eisen et al. [8] applied this method to the yeast and human microarray data and observed a strong tendency for the tightly clustered genes to share common roles in cellular processes. For example, they successfully discovered the extremely tight cluster containing eight histone genes, which were well known to be co-regulated and transcribed at a particular point in the cell cycle. Since the genes in the same cluster with the similar expression pattern have the similar biological function, the previously unannotated new genes can be assigned likely cellular functions based on the existing annotation of other co-group genes. Wu et al. [9] first assembled all genes in the entire yeast genome into functionally related groups by applying multiple clustering methods including hierarchical clustering. Then they predicted probable cellular functions for poorly characterized genes according to the annotation confidence values computed from the well characterized genes in the same cluster. And they were able to experimentally verify several of their predictions. Furthermore, the genes sharing the similar expression patterns and biological functions are likely to share the binding motif of a TF or a set of co-TFs, which can be identified by the motif discovery methods, such as MEME.

Hierarchical clustering methods usually divide the genes into mutually exclusive clusters, but in reality a gene may possess multiple biological functions and thus multiple cluster membership. Moreover, the co-expressed genes in the same cluster are not

necessarily co-regulated because their correlated expression patterns may be caused by indirect regulation or experimental measurement error. As a result, these methods alone are not imprecise.

1.3.2. Maximum likelihood estimation

Maximum likelihood estimation (MLE) fits a predefined model to observed data to determine the model's parameters that maximize the probability (likelihood) of the sample data. MLE is probably the most widely used estimation method for the identification of transcriptional modules. As a well-studied iterative optimization algorithm, expectation maximization (EM) is one of the most popular and powerful techniques for MLE, when the data are incomplete or the likelihood function involves latent variables. EM algorithm iteratively computes the maximum-likelihood estimates between an expectation (E) step and a maximization (M) step. Specifically, EM algorithm initializes the parameters of the predefined model with a reasonable guess, calculates the expectation of the log-likelihood evaluated using the current estimate for the latent variables (E step), updates the parameters that maximize the expected log-likelihood found on the previous E step and that will be used in the next E step (M step), and then alternates between performing an E step and an M step until the likelihood converges, i.e., reaching a local maxima. EM has the advantage of being simple, robust and easy to implement. However, EM is a hill-climbing approach, thus it can only be guaranteed to reach a local maxima. Whether EM will actually reach the global maxima completely depends on the initialization. If it starts at the right "hill", it will be able to find the global

maxima. When there are multiple local maximas, it is often hard to identify the right "hill". The commonly used strategy to solving this problem is to try many different initial values and choose the solution that has the highest converged likelihood value.

Beiley et al. [15] developed an algorithm called MEME, which extends the EM algorithm to identify the TF binding sites shared by a set of co-regulated genes. They successfully discovered both the CRP and Lex A binding sites from a set of sequences containing one or both sites. One of the best advantages of MEME is that it does not restrict exactly one appearance of a binding motif in each examined sequence. That is to say, every potential target gene may contain more than one copy of the binding motif or no copy. Another attractive advantage is its ability to discover multiple binding sites in the co-regulated genes. This is also satisfied with the biological reality that co-regulated genes are very likely under the combinatorial control of multiple TFs. Within a single motif, MEME, like most motif discovery methods, does not allow for insertions or deletions, which exist ubiquitously in biology. Therefore, MEME is limited to learning a restricted class of motifs. Furthermore, MEME is not suited to whole genome TFBS motif discovery because of the shortness and degeneracy of the motifs. That means the input sequences of MEME have to be carefully predefined with prior knowledge, which is usually little.

Wang et al. endeavored to identify the direct target genes of a TF from an entire genome by developing three algorithms (i.e. MODEM [10], TRANSMODIS [11], and CompMODEM), all EM-based algorithms with different integration of multiple sources of data. Among them, MODEM, integrating both sequence and expression data, is the predecessor of the other two. The inputs to MODEM are a public TFBS, the whole

genome promoter sequences and a single genome-wide microarray measurement related to a TF of interest, such as ChIP-chip or TFPE. MODEM is able to identify the direct target genes of a TF and refine the input consensus motif by outputting a position-specific frequency matrix (PSFM) that presents extra precise information of the binding motif. However, the inevitable notable noise in each array may cause MODEM to be trapped in local optima and thus reduces the quality of its performance. To address this problem, we developed another program called TRANSMODIS based on MODEM. TRANSMODIS takes multiple TFPE arrays, instead of a single one, as the input and assumes that the true direct targets are the genes containing the consensus motif of the TF of interest as well as exhibiting consistent expression changes in most of the TFPEs. Compared with MODEM, TRANSMODIS is less sensitive to noise in an individual experiment because of the consistency requirement on gene expression level across multiple experiments. TRANSMODIS accurately identified the direct targets of PHO4 in yeast and the targets of DAF-16 in worm. Besides TRANSMODIS, CompMODEM is another attempt to enhance the accuracy of MODEM by integrating phylogenetic conservation, as well as sequences and expression. By adding phylogenetic conservation information into MODEM, CompMODEM simultaneously reduces both false positives and false negatives.

The common purpose of the three methods above is to statically identify the direct true targets of a TF of interest. However, the activities of a TF are dynamic and thus the targets of the TF are alterable according to the activities of the TF under different experiment conditions. We developed another algorithm called ActivMiner to simultaneously infer the TF's activity in each experiment and target genes of the TF

corresponding to its activity. The target genes of a TF contain the binding sites of the TF in their promoters and their gene expression levels are coherent with the activity of the TF. Meanwhile, the activity of the TF is defined by the activation or repression of its targets. Since neither the label of the target gene nor activity of the TF is observed, ActivMiner iteratively infers the activity and target genes of the TF using EM algorithm within every cursorily pre-grouped cluster.

1.3.3. Matrix decomposing

In the mathematical discipline of linear algebra, a matrix decomposition is a factorization of a matrix into some canonical form. There are many different matrix decompositions; each finds use among a particular class of problems. Matrix decomposition methods have been introduced for discovering transcriptional modules mainly from microarray data. This kind of methods takes microarray data as a matrix containing a mixture of unknown signals that may correspond to specific biological sources. Unlike conventional clustering methods, these methods classify the genes by similarity in the expression of any chosen subset with tight correlated experiments, rather than by overall similarity in the expression on all conditions. In biology, this is reasonable because of the different combinatorial regulations between a TF and other TFs. In addition, they can also partition genes into mutually inclusive modules to reflect the fact that genes may have multiple functions or are active in multiple biological processes. A variety of matrix decomposition methods have been proposed for microarray data

analysis, including singular value decomposition (SVD), independent components analysis (ICA) and non-negative matrix factorizations (NMF).

SVD is known as a popular implement for principal component analysis (PCA) in statistics. When applying SVD to the microarrays, it is a linear transformation of the expression data from the genes × arrays space to the reduced eigengenes × eigenarrays space. In the latter space, the data are diagonalized such that each eigengene is expressed only in the corresponding eigenarray, with the corresponding eigenexpression level indicating its relative significance. The eigengenes and eigenarrays are unique, and therefore also data-driven, orthonormal superpositions of the genes and arrays, respectively. Alter et al. [12] presented the use of SVD in analyzing genome-wide expression data. They normalized the data by filtering out the eigengenes (and eigenarrays) that are inferred to represent noise or experimental artifacts, and thus made meaningful comparison of the expression of different genes across different arrays in different experiments. Additionally, they sorted the data according to the correlations of the gene (and arrays) with eigengenes (and eigenarrays) and gave a global picture of the dynamics of gene expression, in which individual gene or array appears to be classified into groups of similar regulation and function, or similar cellular state and biological phenotype, respectively. Afterwards, upon comparing similar experiments, where a regulator was activated or repressed, the expression pattern of one of the significant eigengenes may be correlated with those of this regulator and its targets. This eigengene, therefore, can be associated with the observed genome-wide effect of the regulator. The expression pattern of the corresponding eigenarray is correlated with those observed in samples, in which the regulator is activated or repressed. This eigenarray, therefore, can

be associated with these samples. Unlike the traditional clustering methods, the groups of genes (or arrays) are not defined by overall similarity in expression, but only by similarity in the expression of any chosen subset of eigengenes (or eigenarrays).

ICA is a powerful statistical and computational technique for revealing independent hidden factors that underlie sets of random variables, measurements, or signals. For the large observed multivariate data, ICA defines a generative model, where the data variables are assumed to be linear mixtures of some unknown latent variables. These latent variables (i.e. the independent components), assumed to be non-Gaussian and as mutually statistically independent as possible, can be linearly decomposed by ICA. ICA is very similar to PCA in that both methods project a data matrix into components in a different space. However, the goals of the two methods are different. PCA finds the uncorrelated components of maximum variance, and is ideal for compressing data into a lower-dimensional space by removing the least significant components. On the other hand, ICA finds the statistically independent components, and is ideal for separating mixed signals. Lee et al. [13] proposed applying ICA to decompose microarray data into independent gene expression patterns corresponding to putative biological processes, which can be characterized by the predominant functional annotations of genes within the component. They also grouped genes into mutually non-exclusive clusters with statistically significant functional coherence. Finally, they demonstrated that ICA outperformed other leading methods, such as principal component analysis, k-means clustering and the Plaid model, in constructing functionally coherent clusters on microarray datasets from yeast, worm and human.

NMF is a group of algorithms in multivariate analysis and linear algebra involving the decomposition of a nonnegative matrix V into two nonnegative matrices, W and H, via a multiplicative updates algorithm. In the context of a p×n gene expression matrix V consisting of observations on p genes from n samples, each column of W defines a metagene, and each column of H represents the metagene expression pattern of the corresponding sample. One characteristic of NMF is that, using dimensionality reduction, it is capable of identifying patterns that exist in only a subset of the experimental conditions, in which smaller sets of genes behave in a strongly correlated fashion. Such an approach might be particularly useful in identifying biological subsets of genes that function in concert in a relatively tightly regulated manner. It might also be an especially sensitive means for detecting functional genetic relationships. The most common application of NMF in computational biology has been in the area of molecular pattern discovery, especially for gene and protein expression microarray studies. This is an exploratory area characterized by lack of priori knowledge of the expected expression patterns for a given set of genes or any phenotype. However, NMF has proved to be a successful method in the elucidation of biologically meaningful classes. For instance, Kim et al. [14] applied NMF to cluster genes and predicted functional cellular relationships in yeast using gene expression data. Brunet et al. [27] applied it to elucidate cancer subtypes by decomposing leukemia and brain cancer data sets.

1.4. Computational approaches for inferring gene regulatory networks

Although identification of individual gene modules is the important but elementary step in studying GRNs, synthesis of GRNs is the ultimate goal in systems biology. To understand the nature of cellular function, it is necessary to study the behavior of genes in a holistic, rather than individual, manner because the expressions and activities of genes are not isolated or independent of each other. Inferring GRNs involves the selection of a network model and the inference of topology and functions of the network from data. A lot of computational approaches have been developed to build models for mimicking GRNs, covering from continuous modeling to discrete modeling. By treating concentrations of gene products as time-dependent variables, three kinds of computational models are proposed so far, namely continuous-time and continuous-variable models (e.g. differential equations), discrete-time and continuous-variable models (e.g. Bayesian networks) and discrete-time and discrete-variable models (e.g. Boolean networks).

1.4.1. Differential equation models

In systems biology, differential equations can relate changes in gene transcript concentration to each other and to an external perturbation. Thus, they have been widely used to model the dynamic behavior of GRNs in a more quantitative manner. Their flexibility allows describing complex relations among genes and their regulators. A modeling of the gene expression dynamics may apply ordinary differential equations

(ODEs): $\frac{dx}{dt} = f(x, p, u, t)$, where $x(t) = (x_1(t), \ldots, x_n(t))$ is the gene expression vector of the genes $1, \ldots, n$ at time t, f is the function that describes the rate of change of the state variables x_i in dependence on the model parameter set p, and the externally given perturbation signals u. To reverse-engineer a GRN using ODEs means to choose a functional form for f and then to estimate the unknown parameters p from the gene expression data and other measured signals x, u and t, using some optimization technique. In general, without constraints, an ODE has multiple solutions and is not uniquely identifiable from data. Thus, the identification of model structure and model parameters requires specifications of the function f and constraints representing prior knowledge, simplifications or approximations. Specifically, the function f can be linear or non-linear. In reality, regulatory processes are usually characterized by complex non-linear dynamics. However, many GRN inference approaches based on differential equations only consider linear models or very specific types of non-linear functions in order to solve the ODEs numerically, because many differential equations cannot be solved analytically.

Bansal et al. [28] developed TSNI algorithm to identify the gene network as well as the direct targets of the perturbations. Based on linear ODEs, TSNI is applied when gene expression data are dynamically (time-series) measured after a perturbation. For small networks (tens of genes), it is able to correctly infer the network structure. For large networks (hundreds of genes), its performance is best for predicting the direct targets of a perturbation.

It is known that transcriptional regulation cannot be explained by simple linear systems. The identification of non-linear models is not only limited by mathematical

difficulties in and computational efforts for solving ODE numerically and identifying parameters, but also mainly by the small sample size, which is usually insufficient to reliably identify non-linear interactions. Thus, the search space for non-linear model structure identification has to be stringently restricted. For this reason, inference of non-linear systems employ predefined functions that reflect prior knowledge available. Sakamoto et al. [29] developed an evolutionary method for identifying small-scale GRNs from the observed time series microarray data by defining the non-linear polynomial ODEs as a model of the network. They applied the method only to three target networks and successfully inferred the systems of differential equations.

1.4.2. Bayesian Networks

A Bayesian networks is a graphical way to represent a particular factorization of the joint probability distribution (JPD) of a set of random variables in a probabilistic model. They have been a popular choice as computational approaches applied to the problem of reverse-engineering GRNs from various data, especially microarrays. A Bayesian network is a representation of a joint probability distribution over a set of random variables. It consists of two components: a directed acyclic graph (DAG) and a family of conditional distributions for each variable in the graph. Together, these two components determine a unique JPD. The structure of a DAG is defined by two sets: nodes and directed edges. The nodes represent random variables in the Bayesian sense, which may be observable quantities, latent variables, unknown parameters or hypotheses and are drawn as circles labeled by the variables' names. If the variable represented by a node is observed, then the node is said to be an evidence node, otherwise it is said to be

hidden or latent. The edges represent direct causal dependencies between variables and are drawn by arrows between nodes. So a benefit of Bayesian networks is that they may be interpreted as a causal model that generated the data. In the model, an arrow from node A to node B indicates that a value taken by variable B depends on the value taken by variable A. Node A is then referred to as a parent of B and, similarly, B is referred to as the child of A. An extension of these genealogical terms is often used to define the sets of "descendants" -- the set of nodes that can be reached on a direct path from the node, or "ancestor" nodes -- the set of nodes from which the node can be reached on a direct path. The structure of the acyclic graph guarantees that there is no node that can be its own ancestor or its own descendent. Such a condition is of vital importance to the factorization of the joint probability of a collection of nodes. Although the arrows represent direct causal connection between the variables, the reasoning process can operate on Bayesian networks by propagating information in any direction. A Bayesian network reflects a simple conditional independence statement. Namely, each variable is conditional independent of its nondescendents in the graph given the state of its parents. More specifically, if a node has no ancestor, its local probability distribution is said to be unconditional, otherwise it is conditional. This property is used to significantly reduce the number of parameters that are required to characterize the JPD of the variables. This reduction provides an efficient way to compute the posterior probabilities given the evidence.

Bayesian networks reflect the stochastic nature of gene regulation and assume that gene expression values can be described by random variables, which follow probability distributions. As they represent regulatory relations by probability, Bayesian networks are

thought to model randomness and noise as inherent features of gene regulatory processes. This ability to handle incomplete noisy data as well as hidden variables and to avoid over-fitting a model to training data makes Bayesian networks attractive candidates for GRN modeling. Most importantly, Bayesian networks provide a very flexible framework for integrating various types of data and prior knowledge in the process of GRN inference to derive a suitable network structure. Methods for learning Bayesian networks usually contain three essential progresses, namely model selection, parameter fitting and fitness rating. Model selection is to define a DAG as candidate graph of relationships. Parameter fitting is to find the best conditional probabilities (CPs) for each node given a graph and experimental data. Fitness rating is to score each candidate model. The higher the score, the better the network model (the DAG and the learned CP distribution) fits to the data. The model with the highest score represents the GRN inference result. Thereby, the critical step is model selection. The naive approach is to simply enumerate all possible DAGs for the given number of nodes. Unfortunately, the number of DAGs grows super-exponentially as the number of nodes increases. Thus, the problem of finding an optimal network is NP-hard. Consequently, one has to choose between restricting to small gene networks and inferring suboptimal networks by heuristic search methods. Therefore, heuristics are always needed to efficiently learn a Bayesian network.

Bayesian networks are widely used for GRN reconstruction. Friedman et al. [30] applied this method to the S. cerevisiae cell-cycle measurements of Spellman et al. [2]. They demonstrated that Bayesian network has the advantage over the clustering approach in that it attempts to discover causal relationships and interactions between genes other than positive correlation, and finer intracluster structure from data. Hartemink et al. [31]

also proposed a technique for scoring models of genetic regulatory networks based on Bayesian networks to analyze yeast expression data. In their approach, Bayesian networks are used to describe relationships between variables in a GRN. Unlike other clustering approaches, a Bayesian network can describe arbitrary combinatorial control of gene expression and thus is not limited to pair-wise interactions between genes. Due to their probabilistic nature, Bayesian networks are robust in the face of both imperfect data and imperfect models. Moreover, Bayesian networks permit latent variables capturing unobserved factors and allow relationships at varying levels of refinement to be specified. Most importantly, the models are biologically interpretable and can be scored rigorously against observational data.

However, the conventional Bayesian network has some major limitations. First, several networks with the same undirected graph structure but different directions of some edges may represent the same JPD. Hence, relying on expression levels only, the origin and the target of an interaction become indistinguishable. Second, the acyclicity constraint eliminates feedback loops that are essential in GRNs. Third, it does not account for the dynamics of a gene regulatory system. Fortunately, these limitations may be overcome through generalizations like dynamical Bayesian networks (DBNs), which model the stochastic evolution of a set of random variables over time. In comparison with general Bayesian networks, discrete time is introduced and conditional distributions are related to the values of parent variables in the previous time point. DBNs separate input nodes from output nodes, i.e. each molecular entity is represented by a regulator node (representing the expression level at time t) as well as by a target node (representing the

expression level at time $t + \Delta t$). Therefore, DBNs are able to describe regulatory feedback mechanisms, because a feedback loop will not create a cycle in a DBN.

Perrin et al. [32] applied DBN to the experimental data relative to the S.O.S. DNA Repair network of the Escherichia coli bacterium. DBN appears to be able to extract the main regulations between the genes involved in this network. Dojer et al. [33] applied DBN to the realistic simulations data to infer a relative large scale GRN (10 genes) and showed that the quality of inferred networks dramatically improves when integrating perturbation experiments into time series microarrays.

Shen et al. [34] presented a Bayesian network approach called GBNet to infer GRNs, which focused on the combinatorial regulation of TFs by searching the cooperative DNA motifs in transcriptional regulation and the sequence constraints that these motifs may satisfy. We showed that GBNet outperformed the other available methods in the simulated and the yeast data. We also demonstrated the usefulness of GBNet on learning regulatory rules between YY1, a human TF, and its cofactors. Most of the rules learned by GBNet on YY1 and cofactors were supported by literature. In addition, a spacing constraint between YY1 and E2F was also supported by independent TF binding experiments.

1.4.3. Boolean Networks

The first Boolean networks were proposed by Stuart A. Kauffman in 1969 [35] to model GRNs at the coarse level. In a Boolean network, each gene, each input, and each output is represented by a node in a directed graph, where there is an arrow from one

node to another if and only if there is a causal link between the two nodes. The state of each node in the graph can be characterized as either ON (one) or OFF (zero). For a gene, ON corresponds to the gene being expressed; for inputs and outputs, ON corresponds to the substance being present. Time is viewed as proceeding in discrete steps. At each step, the new state of a node is a Boolean function of the prior states of the nodes with arrows pointing towards it. As a discrete dynamical network, continuous gene expression signals have to be transformed to binary data before inferring a Boolean network. The discretization can be performed by clustering and thresholding, using support vector regression. We can view Boolean networks as coarse simplifications of the differential equation models.

Although the continuous approaches, such as ODEs and Bayesian networks, have a more accurate physical representation of the system, Boolean networks may represent the only practical alternative for modeling large-scale genetic regulatory systems. One of the main objectives of Boolean network modeling is to study generic coarse-grained properties of large genetic networks and the logical interactions of genes, without knowing specific quantitative details. The biological basis for the development of Boolean networks as models of genetic regulatory networks lies in the fact that during regulation of functional states, the cell exhibits switch-like behavior, which is important for cells to move from one state to another in a normal cell growth process or in situations, when cells need to respond to external signals, many of which are detrimental. Recent study indicates that many realistic biological questions may be answered within the seemingly simplistic Boolean networks, which emphasizes fundamental, generic principles rather than quantitative biochemical details. Moreover, Boolean networks is

the only model system that is able to yield insights into the overall behavior of large genetic networks and allow the study of large high-throughout data in a global fashion.

Shmulevich et al. [36] addressed the problem of inferring the structure of GRNs using the Boolean network model with so-called Best-Fit Extension method. They showed that the problem of identifying the network structure using Boolean networks is polynomial-time solvable, implying its practical applicability to real data analysis.

Nevertheless, the most salient limitation of standard Boolean networks is their inherent two-state determinism in the complicated genetic network of higher-order eukaryotes. In higher organisms, it is more likely that the regularity of genetic function and interaction known to exist is not due to switch-like logical rules, but rather to the intrinsic self-organizing stability of the dynamical system, despite the existence of stochastic components in the cell. Therefore, the assumption of only one logical rule per gene may lead to incorrect conclusions when inferring these rules from gene expression measurements, as experimental data are typically noisy and the number of samples is small relative to that of parameters to be inferred. Probabilistic Boolean Networks (PBNs) is introduced to overcome the deterministic rigidity of Boolean networks. The basic idea of PBNs is to extend the Boolean network to accommodate more than one possible function for each node. PBNs share the appealing properties of Boolean networks, but are able to cope with uncertainty, in both the data and the model selection. In a PBN, just one Boolean function giving the next state of a variable is likely to be only partially accurate. In most situations, different Boolean functions may actually describe the transition, but these are outside the scope of the conventional Boolean model. Consequently, if we are uncertain which transition rule should be used, a PBN involving a set of possible Boolean

functions for each variable may be more suitable than a network, in which there is only a single function for each variable. PBN represents an interface between the absolute determinism of Boolean networks and the probabilistic nature of Bayesian networks, in that it incorporates rule-based uncertainty. This compromise is important because rule-based dependencies between genes are biologically meaningful, while mechanisms for handling uncertainty are conceptually and empirically necessary [37].

Shmulevich et al. [38] developed a PBN model for random gene perturbations to assess the effect of gene perturbations on long-run network behavior and to derive an explicit formula for the perturbation probabilities. Their result demonstrated that states of the network that are more 'easily reachable' from other states are more stable in the presence of gene perturbations.

1.5. Network Validation

Network validation after reconstruction of GRNs is necessary. However, the validation of GRNs may be a very difficult task because the presumptions that underlie the chosen modeling architecture and modeled components may oversimplify the true complexity in GRNs [39]. In addition, the available data is noisy and inadequate with respect to the data requirements for large-scale models. Furthermore, the inference result is not always unique, i.e. some model elements cannot be identified. As a general rule, the predicted models should be validated by data, information and observations that are not used for modeling. Usually, there are two main ways to validate GRNs, i.e. experimentally or computationally.

1.5.1. Experimental Validation

Scientific discovery is an iterative process of building models to explain experimental observations and validating models with new experiments. Therefore, experimental validation is the direct and reliable method to investigate the authenticity of biological meanings of inferred GRNs. Usually, experimental validation includes analyzing the network, choosing the best new experiments to test the GRN, conducting the experiments, and integrating the resulting data. The problem of choosing the best experiments to estimate a model has been a significant area of research in validation. Yeang et al. [40] describe an experimental validation and refinement method by utilizing the inferred GRNs as a criterion for choosing optimal knockout microarray experiments. If an intermediate gene knockout fails to affect downstream genes in a pathway, that pathway is removed from the model. Using this procedure, they evaluated 38 candidate regulatory networks in yeast and perform four high-priority gene knockout experiments. The refined networks supported previously unknown regulatory mechanisms downstream of SOK2 and SWI4.

1.5.2. Computational Validation

However, in most cases, the experimental validation is either unaffordable or infeasible, for instance, the knockout experiments of fatal TFs. Furthermore, besides the data already used for GRN modeling, there are still lots of other data sources not utilized and available for the validation, such as other type experiment data, annotations and

literatures. Therefore, computational validation has become the important and efficient validation method. Chaturvedi et al. [41] developed a technique to validate large-scale GRNs by comparing with corresponding protein-protein interaction (PPI) networks. The GRN were obtained with Bayesian networks while PPI networks were obtained from database of known PPI interactions. They looked for exact matches and then reduced networks by skipping one or more genes in GRN. They demonstrated their technique on expression profiles of differentially expressed genes in the S. cerevisiae cell cycle. They validated GRNs against a merged database of 53235 genes. The precisions of GRN obtained over all genes were from 0.82 to 0.95 in all the phases. Batt et al. [42] presented an approach for the validation of models of GRNs, which combines a method for qualitative modeling and simulation with techniques for model checking. The applicability of this model-validation approach was illustrated by the analysis of the complex regulatory network underlying the nutritional stress response of E.coli. They constructed a model of a part of this network, consisting of key proteins and their interactions involved in the carbon starvation response, and validated this model by the available experimental data in the literature.

1.6. Discussion

Figure 1 illustrates an overall flow diagram for inferring GRNs. Experimental biologists have designed high-throughput technologies of increasingly higher quality and provided more and more various valuable omics data. Meanwhile, these data have greatly facilitated the implementation of numerous sophisticated algorithms, which have been

proposed for reverse engineering of GRNs. The GRN, a functional network in living organisms at the gene level, involves a lot of extremely complicated biochemical progresses, which includes not just DNAs but other components such as proteins, metabolites, etc. The purpose of GRN is to represent the transcriptional cascades and the regulation rules underlying the gene expression. Understanding GRNs can provide new ideas for treating complex diseases and breakthroughs for designing new drugs. The first step to infer GRNs is to identify gene modules of a TF, including discovery of TFBS, prediction of TF targets and determination of TF activities. These gene modules have been further utilized to build the networks between TFs or TF and other co-regulatory proteins. Sometimes, this network reconstruction also employs the raw experiments data directly. All predicted networks required the validation from either experimental or computational evidence. The proved networks can be used in turn as the data resource to identify gene modules or establish GRNs. Moreover, the demand for high quality experimental data from the computational methodology drives speedy development of the high-throughput technologies, while the massive data sets have raised the need for effective and efficient analysis algorithms. Therefore, the whole process of inferring GRNs is an iterative benign cycle.

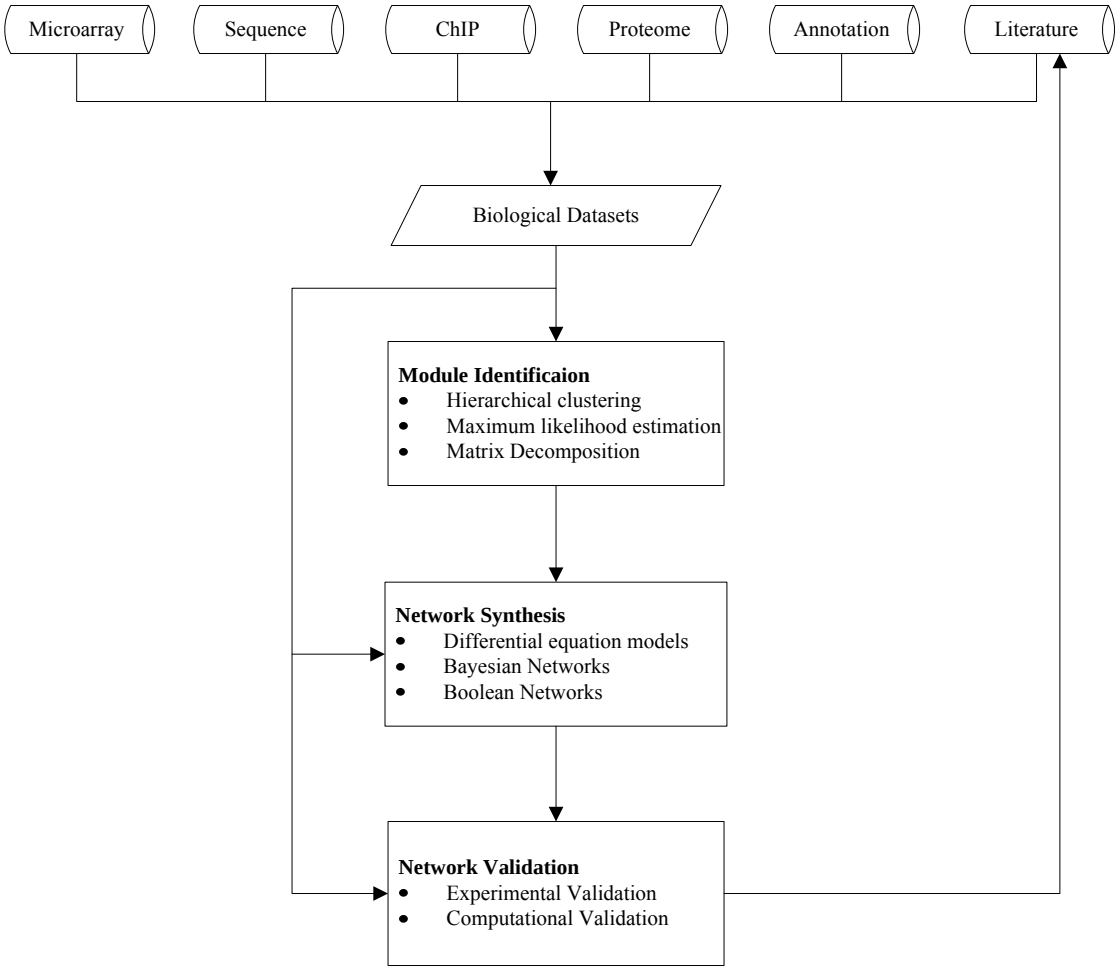

Figure 1. Flow chart for inferring GRNs.

Chapter 2. TRANSMODIS: Identification of Direct Target Genes Using Joint Sequence and Expression Likelihood with Application to DAF-16

2.1. Introduction

One of the major goals in the post-genome era is to establish a connectivity diagram of transcription network, which requires identification of direct targets of TFs. One commonly used approach to detect regulatory interactions between TFs and genes is ChIP-chip [17; 18], which is a binding assay. However binding of a TF to regulatory sequences does not necessarily imply regulation of gene expression. Furthermore, the applicability of ChIP-chip analysis is limited by the availability of antibody against a TF of interest. Therefore, ChIP-chip experiment is often complemented by functional assays using gene microarray.

To determine genes that are regulated by a specific TF, the TF is constitutively activated or inhibited such that the target genes of the TF should have significant expression changes in most of these experiments, which we call TFPEs [43]. In TFPEs, a combination of thresholds, e.g. the least amount of fold change considered to be significant and the minimum number of experiments in which the gene expression changes are required to be significant, need to be pre-specified. However the choice of threshold values tends to be arbitrary. Thresholds are usually hand-picked on a case-by-case basis, depending on the data set. More importantly, direct and indirect targets of the TF cannot be discriminated by expression alone.

In this paper, we present a probabilistic model called TRANSMODIS (TRANScription MOdule DIScovery) which integrates sequence and expression information in target identification. The parametric model can remove the arbitrariness commonly associated with the selection of thresholds for gene expression change. Consideration about the presence or absence of a binding motif in promoters can help distinguish direct from indirect targets. Many motif finding algorithms have been developed and the performance of motif finding algorithms has been steadily improving. We thus assume that the core binding motif of a TF of interest has been determined a priori and is provided as an input to TRANSMODIS. TRANSMODIS is not a motif finding algorithm, rather it focuses on determining direct targets of a TF.

Several computational methods had been developed previously to identify direct targets of TFs. MARSMotif [44; 45] fits splines to gene expressions and determines motifs and genes regulated by the motif simultaneously. Beyer et al. [46] applied a Bayesian method to integrate various types of information to generate a list of putative targets of TFs in yeast. Their approach was not designed to identify targets of a TF in multiple microarray experiments. ARACNe [47] is an approach for reconstructing regulatory Target Gene Identification networks from a large number of expression profiles. It first identifies statistically significant gene-gene coregulations, and then eliminates indirect relationships, which are thought to be the weakest interactions within three-gene loops. The idea is that the remaining edges in the network should have a high probability of representing either direct regulatory interactions or interactions realized by post-transcriptional modifications. ARACNe is a novel approach; however it does not make use of any sequence data and its inferred gene-gene interactions are non-directional.

Segal et al. [48; 49] built probabilistic models to search for genes showing similar expression patterns and also sharing common motif profiles. Their models were complex and the parameters of their models were learned iteratively via greedy search. Compared with the general scenario that Segal et al. were dealing with, TRANSMODIS handles a much simpler situation. As the core motif is given and the target genes of the TF of interest should show significant expression changes in most of the experiments, the search for optimal parameter values in TRANSMODIS is less likely to be trapped in local optima.

The intuition behind TRANSMODIS is that genes containing the consensus core motif of the TF as well as exhibiting consistent expression changes in all TFPEs are likely to be true direct targets. In TRANSMODIS, gene expressions are modeled by a two-component Gaussian mixture model and the binding site sequences are assumed to be generated from a multinomial distribution which is represented by a PSWM. By maximizing the joint likelihood of sequence and expression, TRANSMODIS identifies a set of genes that have consistent and highly elevated expressions and high scoring putative binding sites.

TRANSMODIS is a generalization of MODEM [10], a model we developed previously that is applicable only to a single gene expression microarray or ChIP-chip experiment. Compared with MODEM, TRANSMODIS is less sensitive to noise in individual experiments because of the consistency requirement on gene expression level across multiple experiments. TRANSMODIS also adds an additional step to score genes that do not contain a copy of the consensus binding motif in their promoter regions.

Because consensus binding motif is not known for every TF and sets of TFPEs are limited, a true genome-wide verification of TRANSMODIS is not yet practical. Thus we validated the performance of TRANSMODIS on Pho4p, a TF in budding yeast Saccharomyces cerevisiae. A comparison with previously reported target genes and the target genes selected by the original authors who did the perturbation experiments showed that TRANSMODIS is a promising method for direct target identification and is expected to yield a low false discovery rate (FDR) in general. On a larger scale, TRANSMODIS was applied to a set of ChIP-chip data [8] and evaluated against two other methods. Since no complete list of targets of any TF is known, the comparison was based on positive prediction value (PPV), which is the portion of true positives in all findings. TRANSMODIS demonstrated better performance than the two other methods on a majority of the 81 TFs tested. We then applied TRANSMODIS to identify immediate targets of DAF-16, which is a critical TF influencing the lifespan of nematode Caenorhabditis elegans.

2.2. Results

2.2.1. Validation of TRANSMODIS by simulation

We first validated TRANSMODIS on simulated data where the true targets were known. Each simulated data set consisted of 1000 genes and ten experiments. Out of the 1000 genes, ten were targets and the other 990 genes were non-targets. The expression values of non-target genes were identically and independently sampled from the standard normal distribution N(0,1). And those of targets were simulated from the normal

distribution with a mean of three and a variance of one N(3,1). To make the problem more challenging, within each experiment, ten non-target genes were randomly selected to have their expressions drawn from the N(3,1) distribution of target genes and five target genes were randomly selected to have their expressions reduced by half.

The consensus binding motif was chosen to be TGTTTAC. All target genes had this core binding motif present in their upstream sequences except for two of the ten target genes, which had binding motifs that differed from the consensus binding motif in two nucleotides, namely, TTTTAAC and AGTTTCC. The upstream sequences of all non-targets were simply generated from the uniform background. Each upstream sequence was 600-nucleotide long.

A total of ten simulated data sets were generated and analyzed. The results are listed in Table 1. TRANMODIS showed a clear advantage over MODEM on the simulated data sets. With most Target Gene Identification data sets (9 out of 10), TRANSMODIS identified the complete set of true targets except for the fifth simulated data set, where TRANSMODIS missed one true target. TRANSMODIS had no false positives in all cases. MODEM, on the other hand, failed to find any target genes by the majority voting rule. Note that when MODEM was applied to an individual array, it did identify a list of targets; however since most of the genes on the lists were false positives, no gene (including true targets) made half of the lists. The number of true targets on most lists was between zero and two. Thus the simulation study showed that the gain of using information from all arrays all at once by TRANSMODIS was substantial.

2.2.2. Validation of TRANSMODIS in Saccharomyces cerevisiae

To further validate the model, TRANSMODIS was applied to identify immediate targets of Pho4p, a TF in model organism Saccharomyces cerevisiae. Multiple perturbation microarray experiments were done for Pho4p. The PHO regulatory system is one of the most well studied pathways in Saccharomyces cerevisiae. In a low phosphate (Pi) concentration medium, the cyclin-dependent kinase (CDK) inhibitor Pho81p inactivates the Pho80p-Pho85p complex, leading to an accumulation of hypophosphorylated form of Pho4p in the nucleus and subsequent activation of phosphate responsive genes. In order to identify all genes involved in the phosphate response, Ogawa et al. [7] carried out eight microarray experiments, namely, low Pi vs. high Pi in WT (NBW7) exp 1, low Pi vs. high Pi in WT (NBW7) exp 2, low Pi vs. high Pi in WT (DBY7286), $PHO4^c$ vs. WT, pho80Δ vs. WT, pho85Δ vs. WT, $PHO81^c$ vs. WT exp 1 and $PHO81^c$ vs. WT exp 2. Pho4p was active in each of these experiments and up-regulated expressions of its target genes. Ogawa et al. considered a set of 20 genes that showed at least a two-fold increase of expression in at least five out of the eight experiments as Pho4p targets. In contrast to the somewhat arbitrary criterion used by Ogawa et al., TRANSMODIS provides a parametric model to remove this arbitrariness.

Using the known binding motif CACGTGG of Pho4p and the eight microarray experiments of Ogawa et al. as inputs, TRANSMODIS found 19 genes from the entire Saccharomyces cerevisiae genome (about 6000 genes) as Pho4p targets (Table 2 and Table 8). The 19-gene TRANSMODIS target list was nearly identical to the 20 genes identified by Ogawa et al. except for YER038C, which was dropped by TRANSMODIS.

The YER038C gene is unlikely to be PHO-regulated because it does not contain the consensus Pho4-binding motif or variants in its promoter.

There were nine genes reported to be PHO-regulated prior to the study of Ogawa et al. These nine genes were PHO11, PHO5, PHO89, PHO8, SPL2, PHO12, PHO86, PHO84 and PHO81 [7]. All of them except PHO81 were correctly identified as targets by both Ogawa et al. and TRANSMODIS. A heatmap of the expression profiles of PHO81 and its two homologs YPL110C and SPL2 is shown in Figure 2. The heatmap reveals that SPL2 had a consistently higher differential expression in all experiments (an average increase of 16-fold) than PHO81 and YPL110C (an average increase of 1.6-fold and 2-fold respectively) (p-value = 0.015 from two-sample t-test) (Figure 2). Indeed, both Ogawa et al. and TRANSMODIS identified SPL2 as a Pho4p target. Based on the gene expression data, the selection of SPL2 and the omission of PHO81 and YPL110C by TRANSMODIS are consistent with one's intuition.

TRANSMODIS is an extension to MODEM, which was developed for analyzing a single microarray experiment. To compare the performance of TRANSMODIS with that of MODEM, we applied MODEM in two different ways on this data set. The first approach was to calculate the average expression of each gene in all experiments and apply MODEM to this ''single'' array of averaged expressions. The second approach was to apply MODEM on all eight expression data separately and then select target genes using majority voting (Table 2). We have also listed the MODEM result on a single PHO4 mutation experiment $PHO4^{c}$ vs. WT, in which the Pho4p was constitutively active in Table 2.

One of the known targets, PHO81, was missed by all approaches because of the weak evidence in the expression data (Figure 2). The eight other earlier known targets were successfully identified by all approaches. Only PHO86 was missed when MODEM was run on the averaged expression profile of all arrays. It is not surprising that TRANSMODIS was more stringent than MODEM, identifying fewer targets than MODEM. The average number of target genes found by MODEM from an individual experiment of Ogawa et al. was 32. By requiring consistent up-regulation in all experiments, TRANSMODIS can filter out non-targets that would otherwise be erroneously identified from a single array analysis. At the same time, being less sensitive to random noise in individual experiments, TRANSMODIS can recover some of the true targets that would otherwise be missed by MODEM.

Different from MODEM, TRANSMODIS has an additional step of scoring promoter sequences that do not contain the consensus core motif (up to a certain number of allowed mismatches). Upon evaluation of such a gene without the core motif, if the probability of being a true target using the learned model parameters is greater than 0.5, TRANSMODIS will tag this gene as a target as well. For example, TRANSMODIS identified PHM7 as a Pho4p target; the putative binding site in PHM7 was found to be CAAGTGC, which differs from the consensus binding motif in two nucleotides and therefore was not evaluated by MODEM.

2.2.3. Comparative assessment of TRANSMODIS

There are only a limited number of multiple perturbation experiments publicly available for the same TFs. In order to assess the performance of TRANSMODIS on a genomic data set, we applied it to the ChIP-chip data of 204 TFs [50]. The ChIP-chip experiments were done under different conditions for a portion of the 204 TFs. There are 26 and 15 TFs for which ChIP-chip experiments were done under 3 and .3 conditions respectively. Since the TFs were not necessarily active under each of these conditions and the number of experiments was small, we could not blindly apply TRANSMODIS to experiments available for a TF. We therefore analyzed each ChIP-chip experiment separately and manually selected the experiment that satisfied the following two criteria: there is a significant motif identified by REDUCE [51] in the experiment and the enriched functions of the identified target genes are consistent with those of the TFs.

We compared the performance of TRANSMODIS with two other methods for identifying TF binding. The first one is a Bayesian method that integrates diverse information to predict TF binding in yeast [46] and the second one is an error model developed by Young and colleagues [17]. Since no complete list of targets for any TF is available, sensitivity and specificity cannot be calculated for any of these methods. Therefore, we computed PPV, the portion of true positives in the total predictions. The true positives were taken from three databases: TRANSFAC, SCPD and YPD. We compared the results of the three methods on 81 TFs that had at least one target gene known in the literature and on which the Bayesian method made predictions (Table 3).

On average, the Bayesian method had the most predictions while the error model had the least. The average PPV for the TRANSMODIS, the Bayesian method and the

error model were 8.58%, 6.57% and 6.32%. More specifically, TRANSMODIS performed better than the Bayesian method and the error model on 44 and 46 TFs respectively, and TRANSMODIS performed worse than the other two methods on 22 and 13 TFs respectively. The PPVs are small for all three methods, which is probably due to the fact that only a small set of conditions was tested in the ChIP-chip experiments. It also highlights the need to continuously improve target identification methods.

2.2.4. Identification of genes involved in ageing

Encouraged by the success of TRANSMODIS on finding direct targets of TFs in Saccharomyces cerevisiae, we applied it to tackle a more challenging problem, namely the identification of direct targets of DAF-16 in nematode Caenorhabditis elegans. DAF-16 is a TF playing critical roles in worm ageing. The mechanism of ageing remains to be an important and unsolved mystery. Whereas the normal lifespan of an adult worm is only two to three weeks, individuals carrying mutations that decrease insulin/insulin-like growth factor 1 (IGF-1) signaling can live twice as long [52]. Mutations in gene daf-2, which is predicted to encode an insulin/IGF receptor ortholog, together with a downstream TF, DAF-16, can increase lifespan significantly. DAF-2 negatively regulates the activity of DAF-16, a FOXO-family TF.

Identifying direct targets of DAF-16 can shed light on the functional mechanism of DAF-16 at influencing lifespan. Lee et al. [53] took a comparative genomics approach to identify orthologous genes containing the conserved DAF-16 binding sites in their promoter sequences and Oh et al. [54] used ChIP followed by cloning to search for direct

downstream targets of DAF-16. Lee et al. found that the expression of 7 genes were controlled by DAF-16 while Oh et al. chose to study 33 genes out of 103 candidates and 18 genes showed significant (either up or down) expression changes in a DAF-16 dependent manner. The results of these studies were useful but the number of direct targets identified was limited. To identify genes that are regulated by the DAF-2 pathway and investigate their roles in the ageing process, Murphy et al. [55] deduced the daf-2 and DAF-16 activity using RNAi and analyzed the resultant gene expression profiles using cDNA microarrays. First, genes with a minimum of fourfold expression change were selected by hierarchical clustering of 60 arrays (5 mutant arrays plus 55 time course arrays); in addition, genes showing highly consistent expressions, regardless of the amount of fold change, were also included. Then based upon the p-values obtained from SAM [56] and a visual inspection of genes for genes that were more overly expressed than the others, a top group of 58 genes was chosen to be further validated for their influence on lifespan.

The gene expression microarray experiments conducted by Murphy et al. were functional assays and had multiple time points. We re-analyzed the data using TRANSMODIS to automatically identify the direct targets of DAF-16 without arbitrary thresholds and human involvement. We pooled together the time course data, which consisted of an early adult time course (ten time points from 0-48 h of adulthood) and a longer time course (ten time points from 0-192 h of adulthood), on worms exposed to daf-2 RNAi and worms exposed to daf-16 and daf-2 RNAi. Arrays at 0h time point were left out of the analyses and we also discarded eight arrays with a high percentage of missing data. It left us with a set of twenty eight arrays. The numbers of daf-2(RNAi)

treatments and daf-2(RNAi);daf-16(RNAi) treatments were approximately equal (15 versus 13). We retrieved 1kb upstream sequence of the translational start site of each ORF from WormBase [57].

Using the twenty eight time course gene expression arrays, the upstream sequence data, and the binding motif TRTTTAC defined by Murphy et al., TRANSMODIS was run twice to the same data set with signs inverted in the second run, giving two classes of genes. Following the nomenclature defined in Murphy et al., class 1 genes are genes that were induced in daf-2(RNAi) animals but repressed in daf-2(RNAi);daf-16(RNAi) animals, and class 2 genes are the opposite genes which were repressed in daf-2(RNAi) animals but induced in daf-2(RNAi);daf-16(RNAi) animals. Class 1 and class 2 genes are candidate genes that extend and shorten worm lifespan respectively.

TRANSMODIS identified 39 class 1 genes and 150 class 2 genes (Figure 3 Table 9 and), compared with 263 class 1 genes and 251 class 2 genes that were found by Murphy et al. using hierarchical clustering. Twenty of the TRANSMODIS predictions are in common with the 58 genes in Murphy et al. Furthermore the two lists of class 1 genes share 34 genes and the two class 2 gene lists overlap with 44 genes. The amount of overlap is statistically significant. Hierarchical clustering by itself cannot distinguish between direct and indirect targets. That was why Murphy et al. used other criteria to prioritize their target list. TRANSMODIS provided a systematic and automatic target selection procedure that can be used in place of the original authors' method which needed human involvement.

There was no significant overlap between the targets found by TRANSMODIS and the two previous studies of Lee et al. and Oh et al.. The target genes identified by Lee

et al. and Oh et al. did not have consistent significant expression changes in the time course experiments of Murphy et al. It could be that those genes are regulated by DAF-16 transiently or only at a specific temporal stage. For example, the expression of ZK593.4 was significantly upregulated in the short time course experiments of daf-2 RNAi at 1, 3, 4, 6, 8 and 12 hour time points, but showed almost no change in the long time course experiments of daf-2 RNAi. In the double daf-2;daf-16 RNAi knock-down experiments, ZK593.4 had significant down-regulation only at the first three time points. Such a pattern was not unique to ZK593.4 and was observed for thousands of genes and hence it is hard, if not impossible, to pick out direct targets of DAF-16 exhibiting this particular pattern. The targets identified by TRANSMODIS could be complementary to the previous studies of Lee et al. and Oh et al..

The extended motifs (the core motif plus immediate flanking regions) of the TRANSMODIS targets are shown in Figure 4 and the extended motifs of the two classes differ significantly at the flanking regions. The class 1 genes seem to prefer GSGAGNNTRTTTACTBCANCG (the core motif is underlined) while the class 2 genes seem to prefer STCGACRTRTTTACAGNTSGS. It was suggested that DAF-16 can function both as an activator and a repressor [53; 55]. The direction of regulation by DAF-16 may depend on cooperation between DAF-16 and other TFs binding to the same promoter [53; 55]. Our finding suggests the possibility that the binding sites of the other TFs may partially overlap with that of DAF-16. We therefore hypothesize that the extended motifs of the two target classes are recognized by TFs that function side by side with DAF-16 in a competitive or cooperative manner. This hypothesis can be tested experimentally by using immobilized DNA segments to pull down the cofactors.

We searched for enriched motifs in the 1 kb upstream sequences of TRANSMODIS targets using MobyDick [58], a dictionary motif finding algorithm. The MobyDick algorithm found approximately 300 motifs in each class of targets. We clustered these motifs based on their similarities and evaluated the significance of their occurrences using bootstrap. Among the class 1 targets, AGTTCC, CTCCACC, CTGATAAG and CTTATCA were significantly enriched (p-values<0.01, unadjusted for multiple testing). The p-value of a motif was computed as the probability of observing the same or larger number of occurrences of that motif in a random set of genes, which was a bootstrap sample without replacement from the entire Caenorhabditis elegans genome. We took 10,000 bootstrap samples to compute the p-values. The motif CTTACTA matched the binding motif of GATA family of TFs documented in WormBase and was also identified as an enriched motif by Murphy et al.. Murphy et al. pointed out in their paper that the motif CTTATCA might be bound by a TF that cooperates with DAF-16. Among the TRANSMODIS class 2 genes, the following motifs were significantly enriched: AGATKAGR, CTGATAAG and CTTATCA. We then scanned the 2000 bp upstream region of translational start site of TRANSMODIS class 1 and class 2 homolog genes (the best BLAST matches) in human. The motif CTGATAAG was found to be enriched in the class 1 human genes as well (bootstrap p-value = 0.0061), which suggested that this motif may have functional roles. The other motifs had failed to make the 0.01 p-value cutoff. It is not clear at this point whether CTGATAAG is an extended reverse variant of the canonical GATA motif TGATAAG or a binding site for another TF. There are 11 GATA factors encoded in the Caenorhabditis elegans genome. The deviation of CTGATAAG from the canonical GATA motif implies that, if it is

indeed bound by a GATA factor, then only a subset of GATA factors specifically bind to this motif and cooperate with DAF-16 to regulate the class 1 genes. Since oxidoreductases are enriched in the class 1 genes (see below) and GATA factors MED-1 and MED-2 are known to be involved in oxidative stress response mediated by SKN-1 [59], MED-1 and MED-2 should be the first TFs to be investigated.

To understand the mechanism of DAF-16 at affecting lifespan, we examined enriched molecular functions for the two classes of target genes. On the Murphy et al. class 1 and class 2 genes, the GO term analysis showed that the class 1 genes were enriched for oxidoreductase activities and the class 2 genes were enriched for peptidase activities. The target genes selected by TRANSMODIS had significant overlap with the Muphy et al. genes for both classes. However while there were still many oxidoreductases among the TRANSMODIS class 1 genes, the TRANSMODIS class 2 genes were no longer enriched for peptidase activities. Therefore there were slight changes in the GO term analysis results between the two sets of class 2 genes.

Among the twenty TRANSMODIS class 1 genes that had gene ontology annotations, nearly half of them (9 out of 20) were oxidoreductases (the Bonferroni corrected p-value was about 10^{-4}). Numerous correlations between oxidative stress resistance and longevity have been described [60], consistent with the observation that daf-2 RNAi worms lived significantly longer than wild types. This observation also highlights the regulatory role of DAF-16 on oxidoreductases to extend lifespan. The nine oxidoreductases are C30G12.2, R09B5.6, C06B3.4, W06D12.3, C06B3.5, B0213.15, K12G11.3, F11A5.12 and K07C6.4. Murphy et al. had examined five of them, namely C06B3.4, B0213.15, K12G11.3, F11A5.12 and K07C6.4, on affecting animal lifespan

using RNAi. Knocking down the activities of all but B0213.15 extended lifespan, though not significantly. No significant biological processes or compartments were found, implying that the oxidoreductases are involved in many different processes. Combined with the functional study in, the GO term analysis suggested that the effects of oxidoreductases on ageing might be cooperative/collective and this is why mutations of their upstream regulators, e.g. DAF-2 and DAF-16, can significantly extend lifespan. TRANSMODIS identified 150 class 2 genes, involved in a diverse array of biological processes and functions. A significant portion of the genes (12 out of 63 annotated genes) are involved in macromolecule metabolism but the p-value was not significant at all. The most enriched biological processes were phosphate transport (13 out of 63 genes, p-value = 10^{-10}) and ion transport (15 out of 63 genes, p-value = 10^{-9}). The molecular functions of the class 2 genes with a p-value<0.01 were being structural constituents of cuticle (12 genes, p-value = 10^{-10}) and structural molecules (14 genes, p-value = 10^{-5}). These observations suggest possible functional roles of DAF-16 on affecting lifespan that have not yet been well studied.

2.3. Discussion

TRANSMODIS is a probabilistic model for predicting direct targets from binding motif, sequence data, expression data and ChIP-chip experiments. The probabilistic framework removes arbitrary cutoffs in target selection procedures and allows integration of data coming from various sources. Compared with other criteria for identifying targets,

TRANSMODIS is usually more stringent by requiring consistent and significant expression fold changes across all experiments.

The methodology was validated on a set of TFPEs perturbing the activity of Pho4p in Saccharomyces cerevisiae. TRANSMODIS had successfully recovered a majority of previously known direct targets, i.e. the nine genes that were reported to be PHO-regulated prior to the study of Ogawa et al. Because we do not know the total number of true targets of Pho4p, it is difficult at the current stage to give sensitivity and specificity analyses of TRANSMODIS. To assess the performance of TRANSMODIS, we applied TRANSMODIS and two other methods (a Bayesian method [46] and an error model [17]) on a set of 81 TFs in Saccharomyces cerevisiae. Using PPV as a measure of efficiency and accuracy, TRANSMODIS performed better than the Bayesian method and the error model on 44 and 46 TFs, and performed worse than the other two methods on 22 and 13 TFs, respectively.

Using simulated data sets, it was shown that TRANSMODIS could recover nearly every target gene every time and had few false positives; whereas MODEM, a previously developed method which is applicable to a single experiment, failed to find any target genes on the same data sets. Therefore, TRANSMODIS, though an extension of MODEM, was much more effective at identifying targets than MODEM when multiple arrays were available. If TRANSMODIS is fed a random motif, it can still make target predictions provided that the expression data is unaltered. This is due to the fact that true consensus binding motifs are usually short and degenerate, hence contributing less information than genomic expression data, especially when that data is combined from several experiments.

Some true targets can be missed by TRANSMODIS if the true targets had inconsistent induction in all experiments. The reason can be biological (e.g., transient regulation by the TF or combinatorial regulation of several TFs) or technical (e.g., systematic error or noise of microarray experiments). Nevertheless, the result of TRANSMODIS would be consistent with one's intuition given the data.

The usefulness of TRANSMODIS was demonstrated in the identification of immediate targets of DAF-16, a critical TF in Caenorhabditis elegans that regulates ageing. TFPE experiments are functional assays and are commonly used by researchers to identify targets of a TF, particularly in higher organisms. TRANSMODIS identified target genes that showed DAF-16 dependent expression changes, and expanded the list of known DAF-16 targets. An interesting finding of our analysis is that the flanking sequences of the core motif recognized by DAF-16 differ dramatically in the two classes of targets with opposite effects on lifespan. The observation may provide a clue to the TFs that cooperate with DAF-16 to specifically regulate the two classes of genes. We also found several putative binding motifs for the cofactors of DAF-16 in regulating lifespan. In particular, GATA factors may play important roles in regulating class 1 genes.

It is possible to obtain comparable results to TRANSMODIS by raising the cutoffs sometimes. However it is not clear how high the cutoffs should be set to in the absence of a guideline. If we require the induction ratio of target gene expression to be at least two-fold in at least six out of the eight Pho4p experiments done by Ogawa et al., the target list will then shorten to fewer than 17 genes. So in order to yield a comparable target list, we probably would like to stick with the selection rule of requiring a marked up-regulation in five experiments for targets. Depending on the specific choice of the

threshold, the final Pho4p target list is going to be of different length. For example, the target gene list consists of 20, 19 and 18 genes if the required cutoff is set to 2.1-fold, 2.2-fold, and 2.3-fold respectively. When the cutoff is raised from two-fold (the original threshold used by Ogawa et al.) to 2.1-fold, there is no change to the target list. When the cutoff is raised from 2.1-fold to 2.2-fold, gene YER038C/KRE29 gets dropped and the target list becomes identical to the TRANSMODIS target list. Further increasing the cutoff to 2.3-fold drops gene YOL084W/PHM7, which is likely to be a true Pho4p target. Therefore even though it is possible to produce comparable results to TRANSMODIS by changing the thresholds, it is unclear how to find these thresholds and any choice would be arbitrary without an appropriate justification.

TRANSMODIS assumes that (1) the TF of interest has activities in all experiments; and hence the true immediate targets of a TF of interest ought to have consistent and significant expression changes in most if not all microarray experiments, and (2) the promoters of direct targets contain good matches to the consensus binding motif. These assumptions do not always hold. For example, the promoters of targets may contain motifs that could be bound by the TF but are not because of a lack of cofactors or an inaccessible chromatin structure. Or there can be a situation where only a subset of direct targets was upregulated because the TF recognizes different motifs under different conditions. In these situations, TRANSMODIS is not able to recover the full set of targets but only a subset of them.

In order to use TRANSMODIS, one has to supply a consensus binding motif, which is not always known in advance, especially in higher eukaryotic organisms. However as more biological knowledge is accumulated and deposited into databases such

as TRANSFAC [61] and JASPAR [62], we believe that TRANSMODIS will find more applications in the future. A Java implementation of TRANSMODIS is available upon request. Or the users may choose to upload and analyze their microarray data at http://haedi.ucsd.edu/.

2.4. Materials and Methods

2.4.1. The parametric model and the EM algorithm

The parametric model of TRANSMODIS contains an expression and a sequence component. Target genes are assumed to differ from non-targets in both expression levels and patterns of extended motifs (i.e., the core motif along with immediate flanking regions). The expression of targets and non-targets is modeled by a two-component Gaussian mixture distribution, and the nucleotide frequencies at each position of the extended binding motif are assumed to be multinomial which is represented by a PSWM. Presumably, non-targets do not have binding sites and their sequences are drawn from a background PSWM.

The model is fitted to the sequence and expression data by maximizing the observed log-likelihood function, $LL_o = \log P(S, E; \theta)$, where S and E denote the sequence and expression data respectively, and θ denotes the collection of all model parameters. After expansion into a sum,

$$LL_0 = \sum_{i=1}^{N} \log \left[\left(\prod_{k=1}^{W} f_{a_k(i),k} \right) \left(\prod_{j=1}^{M} \frac{1}{\sqrt{2\pi}\sigma_1} e^{-\frac{(E_{ij}-\mu_1)^2}{2\sigma_1^2}} \right) \lambda + \left(\prod_{k=1}^{W} f^0_{a_k(i),k} \right) \left(\prod_{j=1}^{M} \frac{1}{\sqrt{2\pi}\sigma_0} e^{-\frac{(E_{ij}-\mu_0)^2}{2\sigma_0^2}} \right) (1-\lambda) \right],$$

where λ is the proportion of true targets among all genes; μ_1 and σ_1 denote the mean and the standard deviation of the target expression distribution; N and M denote the total number of genes and experiments respectively; f_{jk} and f^0_{jk} denote the (j,k)th entries of target and non-target PSWMs, or the probabilities of observing the jth alphabet at position k in the target and non-target sequence; A is the size of the sequence alphabet (e.g., A=4 for DNA sequences); W is the length of extended motifs; and $a_k(i)$ returns the alphabet of the ith sequence at position k. When M=1, i.e. there is a single expression profile, TRANSMODIS reduces to MODEM and the observed log-likelihood function shown above becomes that of the mixture model in MODEM.

Direct maximization of LL_0 is a formidable task. Therefore an EM algorithm was derived to compute MLEs. The EM algorithm is a generic iterative algorithm for parameter estimation by maximum likelihood principle. Each iteration of an EM algorithm involves two steps: an E-step where the latent variables are imputed by their expectations and an M-step where the complete log-likelihood function is maximized. In our model, the latent variables are the membership variables. Define $Z = (Z_1, \ldots, Z_N)$, where $Z_i = \begin{cases} 1, & \text{if gene } i \text{ is a true target} \\ 0, & \text{otherwise.} \end{cases}$

The complete log-likelihood LL_c is given by $\log P(S,E,Z;\theta) = \sum_{i=1}^{N} \log P(S_i, E_{i\cdot}, Z_i;\theta)$, where $E_{i\cdot}$ denotes the expression profile of gene i, i.e., the i[th] row of expression matrix E. Because Z_i is a binary random variable, we have

$$\log P(S_i, E_{i\cdot}, Z_i;\theta) = \log[P(S_i, E_{i\cdot} \mid Z_i;\theta)P(Z_i;\theta)]$$

The M-step involves the maximization of function $Q(\theta,\hat{\theta})$, which is, by definition, $E_{Z|S,E;\hat{\theta}}(LL_c)$:

$Q(\theta,\hat{\theta}) = E_{Z|S,E;\hat{\theta}}(LL_c)$ where $\hat{r}_i = E_{Z_i|S_i,E_{i\cdot};\hat{\theta}}(Z_i)$. In words, $\hat{r}_i$ is the probability of gene i being a target given the observed data, S_i and $E_{i\cdot}$, under the current estimate $\hat{\theta}$.

The function $Q(\theta,\hat{\theta})$ can be re-written as a sum of three parts:

$Q(\theta,\hat{\theta}) =$

The three addends are the log-likelihoods from the mixing proportion, the sequence data and the expression data respectively. After taking the partial derivatives of $Q(\theta,\hat{\theta})$ with respect to each unknown model parameter and setting them to zero, we obtain the following updating formulas:

$$\lambda = \frac{\sum_{i=1}^{N} \hat{r}_i}{N} \tag{1}$$

$$\mu_1 = \frac{\sum_{i=1}^{N}\sum_{j=1}^{M}(\hat{r}_i E_{ij})}{M\sum_{i=1}^{N}\hat{r}_i} \tag{2}$$

$$\sigma_1^2 = \frac{\sum_{i=1}^{N}\sum_{j=1}^{M}[\hat{r}_i(E_{ij}-\mu_1)^2]}{M\sum_{i=1}^{N}\hat{r}_i} \tag{3}$$

$$f_{jk} = \frac{\sum_{i=1}^{N}\hat{r}_i I(a_k(i)=j)}{\sum_{i=1}^{N}\hat{r}_i}, \tag{4}$$

$$f_{jk}^{0} = \frac{\sum_{i=1}^{N}(1-\hat{r}_i) I(a_k(i)=j)}{\sum_{i=1}^{N}(1-\hat{r}_i)}, \tag{5}$$

where $I(\cdot)$ is the indicator function.

Computation of r_i's

By definition, $\hat{r}_i = E_{Z_i|S_i,E_{i\cdot};\hat{\theta}}(Z_i) = \Pr(Z_i = 1 \mid S_i, E_{i\cdot}; \hat{\theta})$. Using Bayes' theorem, we have

$$\Pr(Z_i = 1 \mid S_i, E_{i\cdot}; \hat{\theta}) = \frac{\Pr(S_i, E_{i\cdot} \mid Z_i = 1; \hat{\theta})\Pr(Z_i = 1; \hat{\theta})}{\Pr(S_i, E_{i\cdot} \mid Z_i = 1; \hat{\theta})\Pr(Z_i = 1; \hat{\theta}) + \Pr(S_i, E_{i\cdot} \mid Z_i = 0; \hat{\theta})\Pr(Z_i = 0; \hat{\theta})} \tag{6}$$

Even though the formula in (6) is correct analytically, $P(S_i, E_{i\cdot} \mid Z_i = 1; \hat{\theta})$ and $P(S_i, E_{i\cdot} \mid Z_i = 0; \hat{\theta})$ are typically very small quantities such that a direct computation would cause numerical underflow. Therefore, TRANSMODIS computes the probabilities $\hat{r}_i$s, by the following equivalent equation to avoid underflow:

$$\hat{r}_i = \frac{\hat{\lambda}}{\hat{\lambda} + (1-\hat{\lambda})\exp[\log P(S_i, E_{i\cdot} \mid Z_i = 0; \hat{\theta}) - \log P(S_i, E_{i\cdot} \mid Z_i = 1; \hat{\theta})]} \tag{7}$$

Initialization of parameters

Each experiment profile is standardized to have mean zero and standard deviation one. This is a necessary data-adjustment step to correct array bias that arises from variation in the technology rather than variation in the biology. The parameters for the baseline Gaussian component, μ_0 and σ_0 , are set to zero and one respectively throughout the iterations.

Initially, nucleotide frequencies at each position (i.e., entries in the PSWMs) for targets and non-targets are both set to the overall observed frequencies in all sequences. The PSWMs start to diverge during subsequent iterations once they are updated.

The default initial values for the other parameters in the model, λ and σ_1, are chosen to be 0.2, 2 and 0.5 respectively.

Convergence criterion

Convergence is considered to be achieved if the difference between two consecutive iterations of each parameter is less than a prescribed threshold. In particular, the set of convergence criteria used was: $|\lambda^{(k)} - \lambda^{(k-1)}| < 0.01$ and $|f_{jk}^{0(k)} - f_{jk}^{0(k-1)}| < 0.02$ for all $1 \le j \le A$ and $1 \le k \le W$.

Safe-guarding

When fitting a two component Gaussian mixture model to expression data, one needs to be cautious not to misinterpret conditional likelihood ratios. Denote fitted target

and non-target distributions by $N(\mu_1, \sigma_1^2)$ and $N(\mu_0, \sigma_0^2)$ respectively. Without loss of generality, suppose that $\mu_1 > \mu_0$, then there are two scenarios where a direct interpretation of the conditional likelihood ratios can be misleading: When $\sigma_1 > \sigma_0$, for a very negative expression value e, the ratio of probabilities of observing e from the non-target distribution over the target distribution (i.e. $\Pr(e \mid e \sim N(\mu_0, \sigma_0^2)) / \Pr(e \mid e \sim N(\mu_1, \sigma_1^2))$) can be less than one, implying that the gene with an expression level of e is more likely to be a target than a non-target. However intuition tells us that actually the opposite is true, that is, negative expressions are more likely to be observed from non-target genes than from target genes (Figure 5). Similarly, when $\sigma_1 < \sigma_0$, for a very positive expression e, the probability ratio $\Pr(e \mid e \sim N(\mu_0, \sigma_0^2)) / \Pr(e \mid e \sim N(\mu_1, \sigma_1^2))$ can be much greater than one, leading to the wrong interpretation that the gene is a non-target (Figure 5).

The underlying cause in both cases is an unequal variance between the target and non-target distributions. The implication is that using a two-component Gaussian mixture model to describe real expression data is inadequate. Nonetheless the normal mixture model is an analytically simple yet powerful parametric model to summarize expression data, under which MLEs can be computed via an EM algorithm. One simple remedy for the problem is to constrain the two variances to be equal, i.e., $\sigma_1 = \sigma_0$. However we opted for a second solution: whenever an expression $e < \mu_0$ is observed, we required the ratio $\Pr(e \mid e \sim N(\mu_0, \sigma_0^2)) / \Pr(e \mid e \sim N(\mu_1, \sigma_1^2))$ to be bounded below by $\Pr(e \mid e \sim N(\mu_0, \sigma_0^2)) / \Pr(e \mid e \sim N(\mu_1, \sigma_0^2))$ (i.e. the probability ratio after substituting σ_1

for σ_0 as if the target distribution has variance σ_0^2). And similarly when $e > \mu_1$, the conditional likelihood ratio $\Pr(e \mid e \sim N(\mu_0, \sigma_0^2)) / \Pr(e \mid e \sim N(\mu_1, \sigma_1^2))$ is restrained from exceeding $\Pr(e \mid e \sim N(\mu_0, \sigma_0^2)) / \Pr(e \mid e \sim N(\mu_1, \sigma_0^2))$. We call our approach safe-guarding because it guards against a conditional likelihood ratio falling into an undesirable range.

Outlier detection and removal

The EM algorithm is sensitive to outliers. For example, imagine a non-target gene having all expression values close to μ_0 except for one value which is very large (μ_1) due to some experimental error. This single spurious measurement can cause a large deviation in the computed probability and thus make the non-target gene falsely identified as a target gene by the EM algorithm. Because such outliers are detrimental to the analysis, they are searched for by TRANSMODIS and removed once found.

Under our expression model, the distribution of expression measurements of any given gene, whether a target or not, is Gaussian. Therefore the largest expression value of each gene is examined by comparing it with the rest of the gene's expression measurements to see if it is probable to obtain an extreme value as large as the observed maximum. More precisely, let $E_{iJ} = \max_{j=1\ldots M} E_{ij}$ be the maximal expression value observed for gene i. A Gaussian density function is then fitted to the remaining $(M-1)$ values: $\hat{\mu} = \sum_{j \neq J} E_{ij} / (M-1)$ and $\hat{\sigma}^2 = \sum_{j \neq J} (E_{ij} - \hat{\mu})^2 / (M-2)$ (if $M \geq 3$). Assuming that all of the M expression values were drawn from this fitted Gaussian distribution, $N(\hat{\mu}, \hat{\sigma}^2)$, then the probability of observing a maximum order statistics as large as E_{iJ} is given by

$$1-[\Phi((E_{iJ}-\hat{\mu})/\hat{\sigma})]^{M}, \qquad (8)$$

where function Φ is the cumulative distribution function (CDF) of a standard normal. The entry E_{iJ} is removed (i.e., treated as if it were missing) if the probability in (8) is less than α, a user-specified threshold (by default, $\alpha = 0.05$).

The outlier detection and removal scheme described above is consistent with our parametric expression model and can remove up to one outlier per gene.

A robust updating formula for σ_1

Under our expression model, all target genes have the same average expression level. However in reality, this expression model is too simple to hold up. More likely, target genes have different baseline levels of expression. In other words, some target genes might be consistently more over-expressed than others even though all target genes are over-expressed in all experiments. Therefore it is sensible to subtract the baseline expression level of each gene when estimating σ_1, the standard deviation of gene expressions of a target,

$$\sigma_1^2 = \frac{\sum_{i=1}^{N}\sum_{j=1}^{M}[\hat{r}_i(E_{ij}-\overline{E}_{i\cdot})^2]}{(M-1)\sum_{i=1}^{N}\hat{r}_i}, \qquad (9)$$

where $\overline{E}_{i\cdot} = \sum_{j=1}^{M} E_{ij}/M$ is the average expression of gene i. If the true underlying generative model is genuinely a two-component Gaussian mixture model, then replacing Equation (3) by Equation (9) has a negligible effect on the EM algorithm as both

formulas give unbiased estimates of σ_1. However when the true underlying model deviates away from a two-component Gaussian mixture model, e.g., target genes do display different baseline levels of expression, then Equation (9) produces a smaller estimate of σ_1 than Equation (3). This downward bias has a beneficiary effect in controlling false positive rate because the probability of mistaking a non-target gene (assuming whose expression is below μ_1) as a target steadily decreases as σ_1^2 shrinks. Consequently, by having fewer non-target genes falsely classified as targets, the estimated value of μ_1 is less likely to shift unduly downward during subsequent iterations.

To verify that the substitution of Equation (3) by Equation (9) has minimal effect on target gene selection when the true generative model is indeed a two-component Gaussian mixture model, we carried out a simulation study, in which the targets' expression distribution was one of the nine normal distributions (μ_1 = 1, 2 or 5, and σ_1^2 = 0.5, 1 or 2, a total of nine combinations) while the non-targets' expressions were all simulated from $N(0,1)$. For each target distribution, a total of 100 data sets were generated. Each simulated data set consists of 100 target genes and 900 non-target genes. To remove information from sequence data, all simulated genes had identical promoter sequences. Simulation results showed that by using either formula to update σ_1, the resultant target lists were always nearly identical (Table 4). We also compared the sensitivity and specificity of two target lists and found that using Equation (3) resulted in a higher sensitivity while using Equation (9) resulted in a higher specificity, but the differences were small and negligible (Figure 6 and Figure 7). Because robustness was

given a higher priority over sensitivity, TRANSMODIS updates its σ_1 estimate by using Equation (9).

Dealing with missing expression values

It is common for microarray data to have missing entries. Many methods would require the missing entries to be imputed first; however imputation is optional with TRANSMODIS. In the presence of missing data, TRANSMODIS derives an EM algorithm that maximizes the likelihood on the available expression data entries only.

2.4.2. The program

The inputs to TRANSMODIS are: (1) the 59 upstream sequences of all genes in the genome; (2) multiple genome-wide microarray measurements, such as TFPEs or ChIP-chip experiments or a combination of both. The parametric framework allows ChIP-chip experiments to be incorporated into the model just as any other microarray experiments as long as the TF is activated under the ChIP-chip experimental conditions; and (3) the core DNA motif recognized by the TF, typically six to eight bases long. The core motif could have been known a priori or be identified by a motif finding algorithm. The TRANSMODIS program consists of two steps. In the first step, the parametric model of TRANSMODIS is fitted to genes containing matches to the input core motif in their promoters to obtain MLEs via an EM algorithm. The matches do not have to be perfect matches; it is still considered a match if the nucleotide subsequence differs from the core motif in only one base pair. The reverse complement of the input core motif is also

scanned for. If a promoter has multiple matched copies of the input core motif, all copies are extracted and aligned to create an initialization of the PSWM of the target genes. Then during iterations of the EM algorithm, the copy with the highest score according to the current estimate of the target PSWM is chosen as the putative TF binding site.

In the second step of the TRANSMODIS analysis, genes that do not contain copies of the core motif (i.e. genes that were not used for the estimation of model parameters in the first step) have their promoters scanned for the core motif on both strands. If the probability of being a target is computed to be greater than that of being a non-target, the gene will be brought into the target list. No model parameters are estimated or modified during this step. The sole purpose of this second step is to catch potential true targets that lack a copy of the consensus binding motif and therefore would otherwise be overlooked if this step was not taken.

The output of TRANSMODIS are (1) two PSWMs, one for target genes and the other for non-targets. The weight matrices go beyond the core motif and cover the immediate flanking regions beside the core motif; and (2) the probability of being a true target for each gene. By default, genes are identified as targets if the probabilities are greater than 0.5.

TRANSMODIS is computationally efficient and converges fast. The running times on the Pho4p and Daf16p data sets were 2 minutes and 31 seconds and 8 minutes and 53 seconds respectively on a 2.4 GHz single processor computer with 512 KB of cache memory.

2.4.3. GO term analysis

GO Term Finder [25] was used for the gene ontology analyses. The analyses were run on the annotation file submitted on March 21, 2006 for Saccharomyces cerevisiae and the annotation file submitted on March 20, 2006 for Caenorhabditis elegans. Bonferroni correction was used to adjust p-values for multiple testing.

Acknowledgments

Chapter 2, in full, is a reprint of the material as it appears in Identification of direct target genes using joint sequence and expression likelihood with application to DAF-16 Yu, Ron X.; Liu, Jie; True, Nick; Wang, Wei. PLoS One. 2008; 3(3):e1821. The dissertation author was a co-author of this paper.

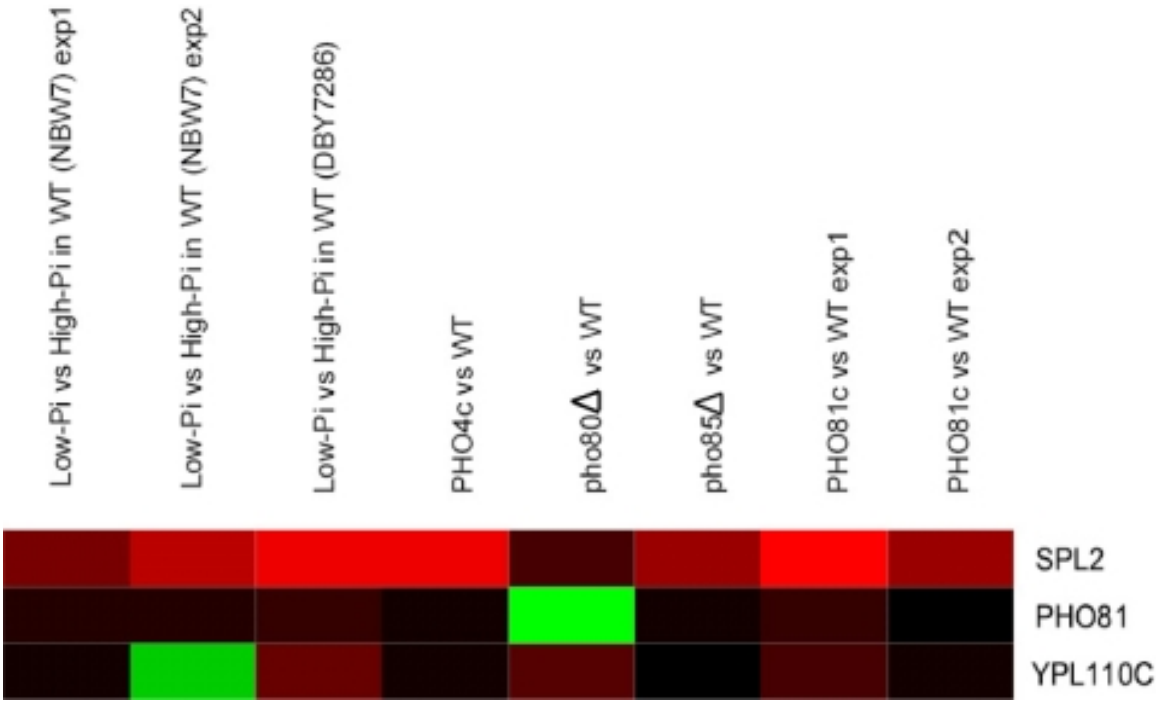

Figure 2. Comparison between the expression profiles of PHO81 and its two homologs SPL2 and YPL110C in the eight TFPE experiments of Pho4p.

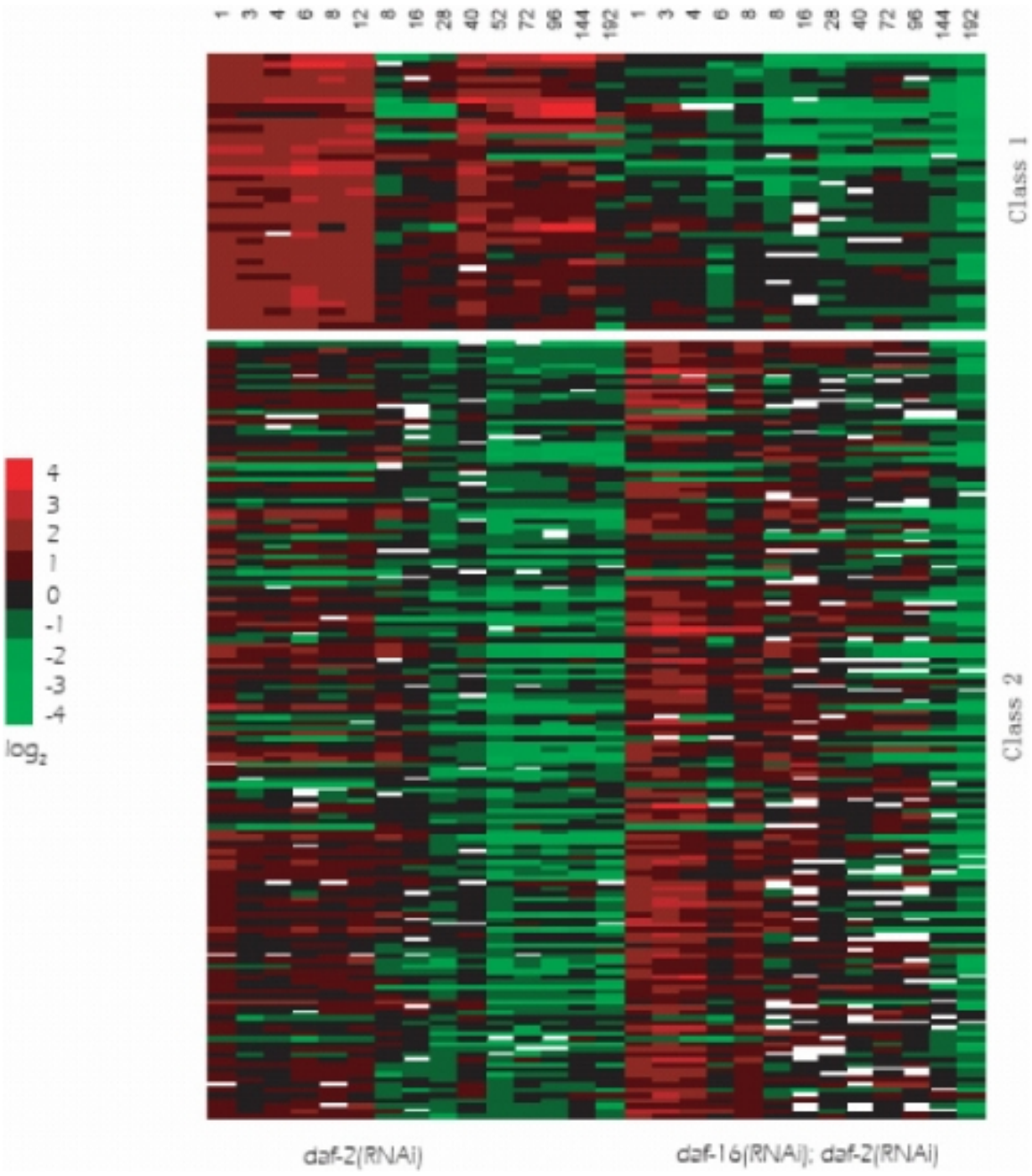

Figure 3. Expression profiles of class 1 and class 2 direct targets of DAF-16 in Caenorhabditis elegans identified by TRANSMODIS.

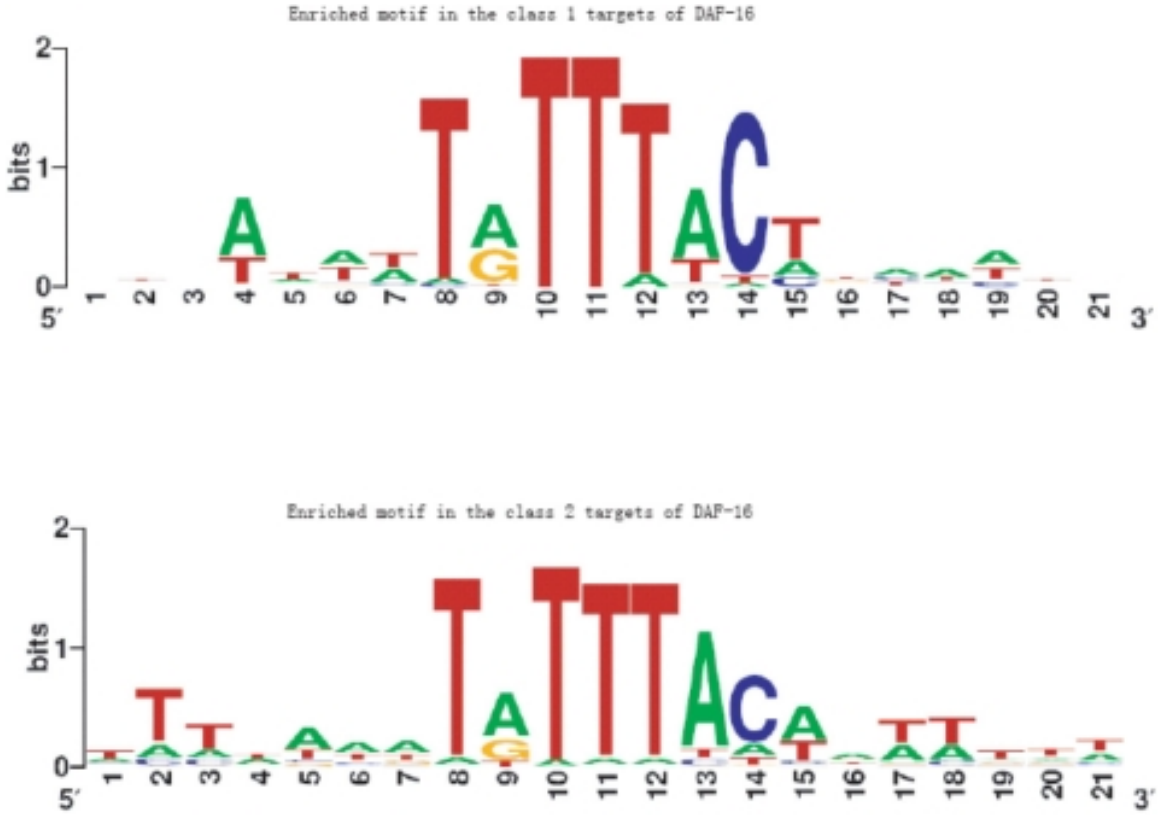

Figure 4. Enriched motifs in the class 1 and class 2 target genes of DAF-16.
The x axis is the position and the y axis is the log2 ratio between the target and non-target weight matrices.

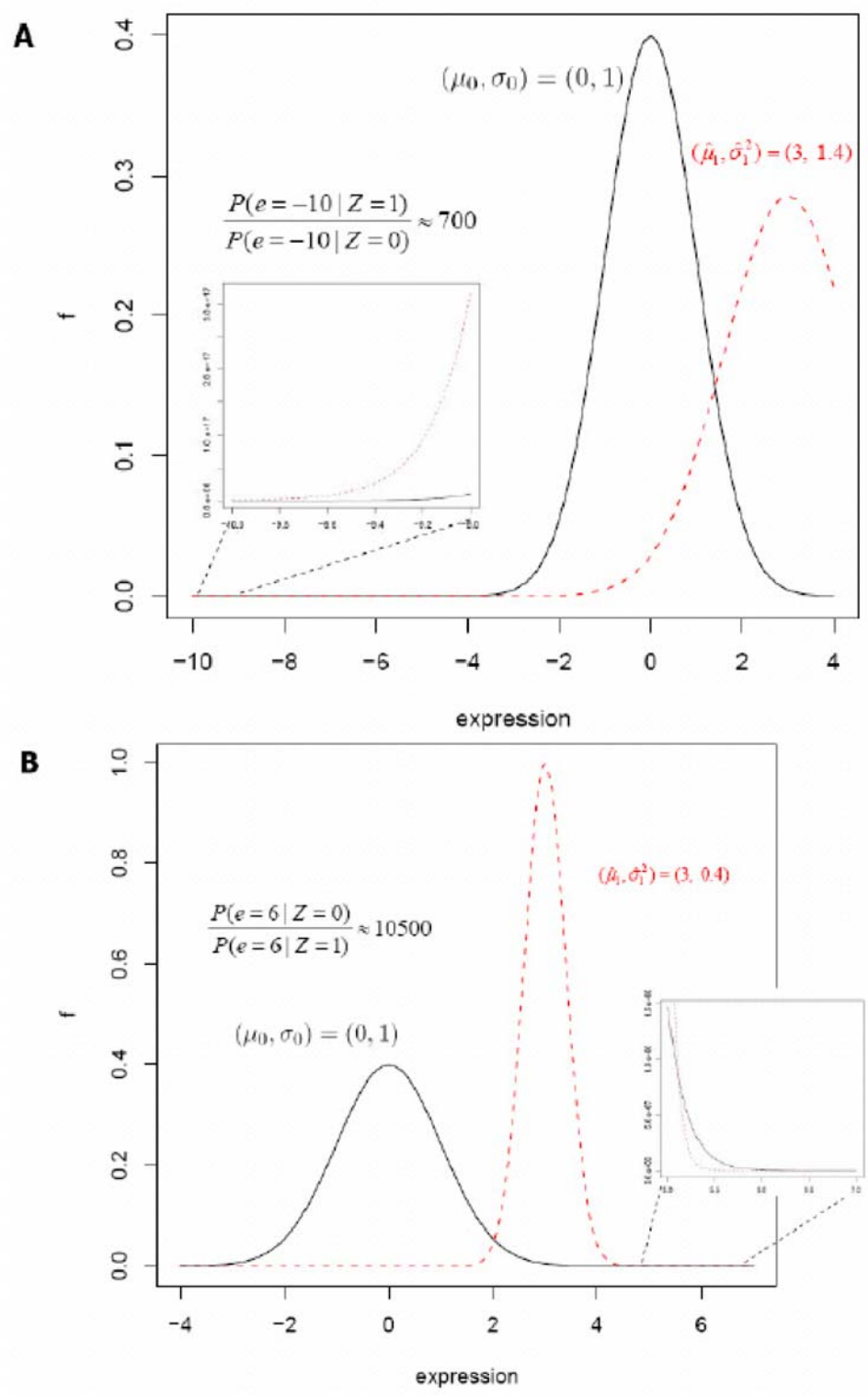

Figure 5. Illustration of drawing invalid conclusions due to unequal variances.

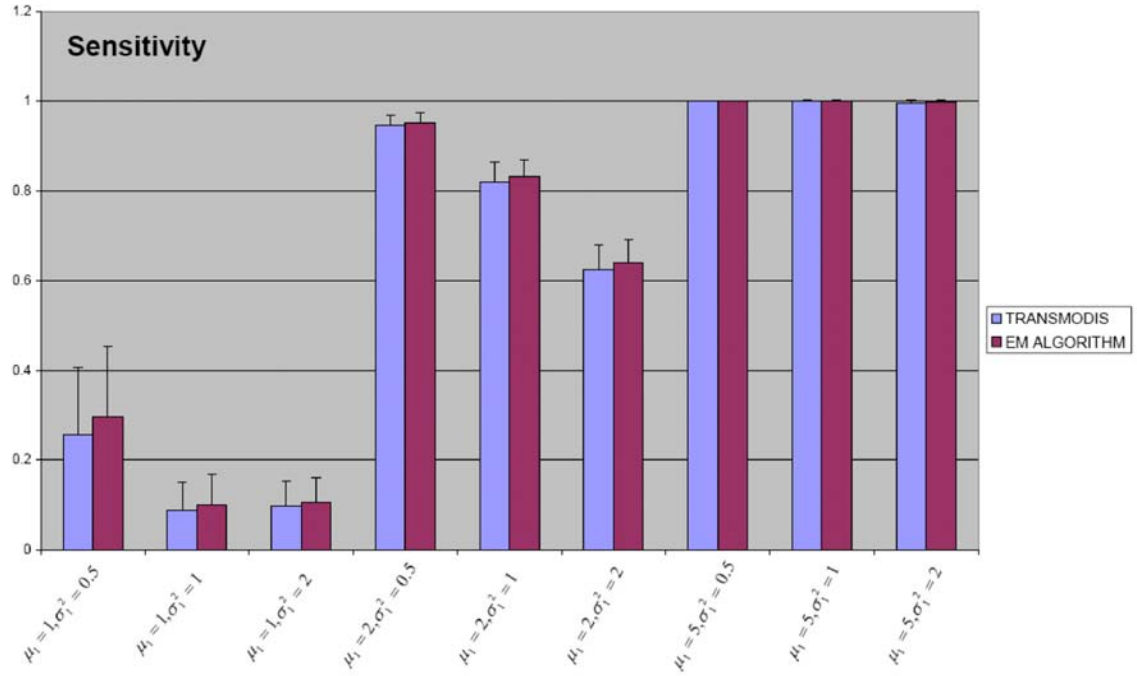

Figure 6. Comparison of sensitivity between the two updating formulas for the standard deviation of target distribution.

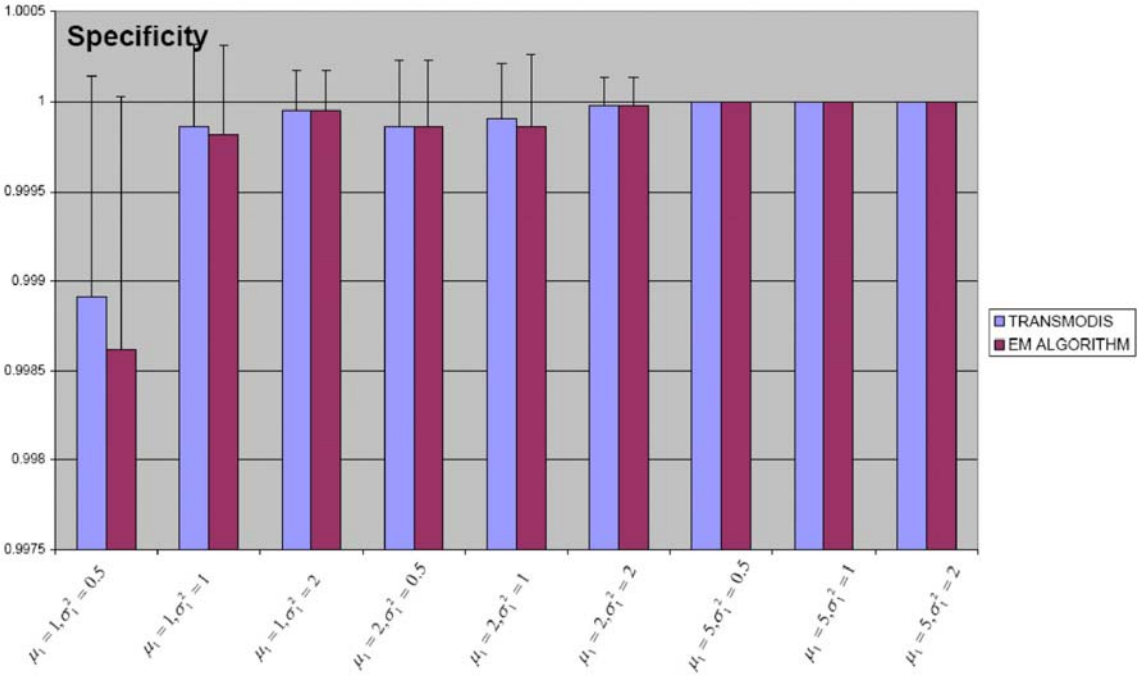

Figure 7. Comparison of specificity between the two updating formulas for the standard deviation of target distribution.
Even though the one standard error bar is drawn above one, no actual specificity was ever greater than one.

Table 1. TRANSMODIS and MODEM results on ten simulated data sets.

Simulated data set	#1	#2	#3	#4	#5	#6	#7	#8	#9	#10
TRANMODIS	10/10*	10/10	10/10	10/10	9/9	10/10	10/10	10/10	10/10	10/10
MODEM on array 1	1/33	0/57	2/59	1/40	2/26	0/24	0/55	2/67	4/72	0/55
MODEM on array 2	0/31	1/22	1/27	2/32	0/48	2/41	0/38	2/46	2/37	0/19
MODEM on array 3	0/37	2/45	1/35	0/36	0/19	2/38	3/41	0/28	0/23	0/58
MODEM on array 4	2/50	0/38	2/26	0/38	1/34	0/54	2/47	0/23	2/41	0/50
MODEM on array 5	2/43	0/17	0/38	1/32	3/30	3/50	1/73	1/63	0/29	1/80
MODEM on array 6	1/28	1/29	1/40	2/30	1/71	1/29	0/38	1/26	3/46	2/42
MODEM on array 7	1/33	0/40	0/38	0/53	3/36	0/45	2/35	6/41	0/33	2/34
MODEM on array 8	3/32	0/58	1/39	0/29	2/32	0/36	0/56	0/30	2/50	0/31
MODEM on array 9	0/22	1/45	1/94	1/25	0/52	0/45	3/69	0/30	0/19	0/26
MODEM on array 10	1/26	1/43	0/43	2/32	0/58	0/32	1/33	1/32	1/61	0/57
MODEM (majority voting)	0/0	0/0	0/0	0/0	0/0	0/0	0/0	0/0	0/0	0/0

*The ratio A/B indicates that the method predicted a total of B genes as direct targets and out of these B genes, A genes were true targets.

Table 2. Target genes selected using different approaches.

Gene	ORF	Ogawa *et al.*	TRANSMODIS	MODEM (average expression profile)	MODEM (individual arrays; majority rule)	MODEM ($PHO4^c$ vs. WT)
PHO11*	**YAR071W**	√	√	√	√	√
PHO5*	**YBR093C**	√	√	√	√	√
PHO89*	**YBR296C**	√	√	√	√	√
PHM6	YDR281C	√	√	√	√	√
PPN1	YDR452W	√	√		√	√
PHO8*	**YDR481C**	√	√	√	√	√
PHM8	YER037W	√	√		√	
HIS1	YER055C	√	√			
HOR2	YER062C	√	√	√	√	
VTC1	YER072W	√	√		√	√
VTC2	YFL004W	√	√	√		√
SPL2*	**YHR136C**	√	√	√	√	√
PHO12*	**YHR215W**	√	√	√	√	√
VTC4	YJL012C	√	√	√	√	√
PHO86*	**YJL117W**	√	√		√	√
PHO84*	**YML123C**	√	√	√	√	√
PHM7	YOL084W	√	√			
CTF19	YPL018W	√	√	√	√	√
VTC3	YPL019C	√	√	√	√	√
KRE29	YER038C	√				
SWC3	YAL011W					√
YAR069C	YAR069C					√
YAR070C	YAR070C			√	√	√
KRE2	YDR483W			√		√
MNN1	YER001W			√		
ARO9	YHR137W			√	√	√
REC107	YJR021C					√
YJR039W	YJR039W					√
NUP85	YJR042W					√
PTK2	YJR059W			√		
CDA1	YLR307W					√
YLR402W	YLR402W					√
YML089C	YML089C					√
YMR291W	YMR291W			√		
YPL110C	YPL110C			√		
CTF4	YPR135W					√
PHO81*	**YGR233C**					

*The nine genes that were previously reported to be under PHO regulation prior to the study of Ogawa *et al.*

Table 3. Comparison between TRANSMODIS and two other methods for target gene identification one the set of ChIP-chip data by Harbison et al.

TF	Known targets	Total number of predictions			Number of predictions known to be true			PPV		
		TRANSMODIS	Bayesian	Error model	TRANSMODIS	Bayesian	Error model	TRANSMODIS	Bayesian	Error model
ABF1	30	240	176	267	9	5	5	0.038	0.028	0.019
ACE2	8	85	335	92	2	2	2	0.024	0.006	0.022
ADR1	10	189	20	35	1	0	0	0.005	0	0
ARG80	8	16	7	16	3	2	3	0.188	0.286	0.188
ARG81	8	17	20	28	3	4	4	0.176	0.200	0.143
ARO80	2	12	32	27	2	2	2	0.167	0.063	0.074
ASH1	1	21	10	0	0	0	0	0	0	NA
BAS1	13	41	147	41	8	10	8	0.195	0.068	0.195
CBF1	11	86	252	281	3	7	5	0.035	0.028	0.018
CIN5	1	117	169	153	0	0	0	0	0	0
CUP9	2	35	6	21	1	1	1	0.029	0.167	0.048
DAL80	22	49	8	13	0	0	0	0	0	0
DAL81	10	114	79	96	7	5	7	0.061	0.063	0.073
DAL82	8	54	93	59	6	8	6	0.111	0.086	0.102
FKH1	1	167	116	142	0	0	0	0	0	0
FKH2	2	121	353	122	2	2	2	0.017	0.006	0.016
FZF1	1	35	5	17	0	0	0	0	0	0
GAT1	4	124	41	27	3	1	1	0.024	0.024	0.037
GCN4	57	68	169	75	23	32	22	0.338	0.189	0.293
GCR1	20	42	55	15	0	5	2	0	0.091	0.133
GCR2	9	47	43	56	4	5	4	0.085	0.116	0.071
GLN3	31	118	141	68	16	16	11	0.136	0.113	0.162
HAC1	5	10	56	15	1	3	1	0.100	0.054	0.067
HAL9	1	33	15	28	0	0	0	0	0	0
HAP1	14	149	189	151	10	9	10	0.067	0.048	0.066
HAP2	30	23	54	21	2	2	2	0.087	0.037	0.095
HAP3	27	10	19	30	1	2	2	0.100	0.105	0.067
HAP4	27	74	170	77	7	9	7	0.095	0.053	0.091
HAP5	25	13	24	12	1	0	0	0.077	0	0
HSF1	16	71	122	102	12	12	13	0.169	0.098	0.127
IME1	15	20	1	0	0	0	0	0	0	NA
INO2	20	33	62	48	5	10	7	0.152	0.161	0.146
INO4	18	31	64	37	9	13	9	0.290	0.203	0.243
IXR1	1	9	2	28	0	0	0	0	0	0
LEU3	7	19	61	24	6	6	4	0.316	0.098	0.167
MAC1	8	8	47	18	3	4	4	0.375	0.085	0.222
MBP1	38	121	394	61	15	25	8	0.124	0.063	0.131
MCM1	32	92	240	107	18	20	16	0.196	0.083	0.150
MET28	1	20	1	17	0	0	0	0	0	0
MET4	9	25	76	28	4	5	1	0.160	0.066	0.036
MIG1	29	10	67	22	1	8	2	0.100	0.119	0.091
MOT3	4	22	11	8	0	0	0	0	0	0
MSN1	1	114	1	5	0	0	0	0	0	0
MSN2	36	154	199	47	11	17	4	0.071	0.085	0.085
MSN4	33	115	163	71	8	13	4	0.070	0.080	0.056
PDR1	15	323	108	8	4	4	0	0.012	0.037	0.000
PDR3	9	8	39	21	1	2	1	0.125	0.051	0.048
PHO2	19	33	2	33	1	0	1	0.030	0	0.030
PHO4	24	72	82	31	4	8	7	0.056	0.098	0.226
PPR1	4	15	24	28	0	2	0	0	0.083	0
PUT3	2	14	66	90	1	2	0	0.071	0.030	0
RAP1	35	291	196	0	17	13	0	0.058	0.066	N/A
RCS1	11	39	183	261	7	10	0	0.179	0.055	0
REB1	21	278	313	0	4	4	0	0.014	0.013	N/A
RFX1	5	12	57	25	2	4	2	0.167	0.070	0.080
RGT1	6	9	1	0	1	1	0	0.111	1.000	N/A
RIM101	4	115	27	7	0	0	0	0	0	0
RME1	2	29	66	40	1	1	0	0.034	0.015	0
ROX1	13	104	94	6	1	2	0	0.010	0.021	0
RPH1	1	25	68	8	0	1	0	0	0.015	0
RPN4	7	144	212	101	4	7	4	0.028	0.033	0.040
RTG3	5	26	47	37	4	4	4	0.154	0.085	0.108
SIP4	2	9	69	21	1	2	1	0.111	0.029	0.048
SKN7	21	187	201	190	8	6	6	0.043	0.030	0.032
STE12	78	60	567	63	24	34	25	0.400	0.060	0.397
STP1	1	60	117	72	1	1	0	0.017	0.009	0
SUM1	2	81	110	60	1	0	1	0.012	0	0.017
SUT1	1	95	73	69	0	0	0	0	0	0
SWI4	14	105	271	161	5	6	4	0.048	0.022	0.025
SWI5	11	46	203	120	3	7	5	0.065	0.034	0.042
SWI6	44	118	430	158	10	19	10	0.085	0.044	0.063
TEC1	44	62	46	43	3	0	0	0.048	0	0
THI2	8	34	67	47	5	8	7	0.147	0.119	0.149
UGA3	3	9	42	32	2	2	0	0.222	0.048	0.000
UME6	40	286	239	134	18	18	10	0.063	0.075	0.075
XBP1	5	65	50	77	1	1	1	0.015	0.020	0.013
YAP1	39	25	314	72	5	11	7	0.200	0.035	0.097
YAP6	1	15	242	60	1	0	1	0.067	0	0.017
YHP1	1	42	9	20	0	0	0	0	0	0
YRR1	4	66	3	23	0	0	0	0	0	0
ZAP1	12	22	62	22	4	9	4	0.182	0.145	0.182
Average	14.4	72.8	111.3	58.6	4.3	5.6	3.5	0.086	0.066	0.063

The cutoff of the error model is set to 0.001, as suggested by the original authors

Table 4. Overall agreement between the two strategies of updating sigma-1.
(calculated in terms of Cohen's kappa)

	$\mu_1 = 1$	2	5
$\sigma_1^2 = 0.5$	0.9221±0.0940	0.9970±0.0046	1.000±0.0000
1	0.9118±0.1010	0.9917±0.0095	0.9999±0.0008
2	0.9559±0.0636	0.9876±0.0137	0.9998±0.0010

Chapter 3. CompMODEM: Prediction of regulatory interactions between transcription factors and their targets

3.1. Introduction

Identification of target genes directly regulated by a TF is essential for deciphering gene regulatory networks and understanding cooperative mechanisms between TFs. Extensive experimental and computational methods have been developed to tackle this problem. Traditionally, the targets of a TF have been determined by experimental technologies in vivo.

For example, chromatin immunoprecipitation with microarray hybridization (ChIP-chip) [17; 50; 63] or sequencing (ChIP-seq) [64] is a widely used technology for detection of TF's binding locations in the genome. Briefly for this method, a protein of interest with chromatin in a cell lysate is temporarily bonded, the chromatin-protein complexes are then sheared and DNA fragments associated with the protein are selectively immunoprecipitated, then the associated DNA fragments are purified and sequence are determined. These enriched DNA sequences are supposed to be the binding sites of the protein in vivo. However, observed DNA binding in the regulatory region alone is not always sufficient to indicate the occurrence of true interaction between a TF and a potential target gene. Even if binding physically happens, the event may not be biologically relevant, or the observed binding may relate to some cellular function other than gene regulation. Moreover, interaction mapping projects, like ChIP or microarray experiments, are difficult to complete because a cell's pattern of interactions is strongly

dependent on variables such as the cell type, secondary structures of protein and DNA, combination of cofactors, genetic background, stage of development, time after stimulus, or specific environmental or biological condition. On the other hand, many true binding events may be missed by ChIP because the relevant conditions have not yet been examined.

Another widely used experiment for this purpose is the TFPE [65]. In a TFPE, the activity of a particular TF is perturbed by mutating, deleting or overexpressing the TF itself or other TFs that regulate this TF, and thus the TF's target genes should have significant expression changes compared with the reference state in the microarray experiment. However, TFPEs only demonstrate the functional character of a TF, while DNA-protein location experiments only display the binding character. In addition, TFPEs are unable to distinguish direct targets from indirect targets, which might be controlled by the targets of the TF and also present expression variance. Furthermore, the cellular conditions, under which TFPEs are performed, may not exhaust all scenarios for the regulatory interactions between a TF and its target genes. Namely, different cooperative cofactors of the TF may be activated under the cellular conditions and therefore the targets of the TF may vary upon conditions. Therefore, neither DNA-protein location experiment nor TFPE is sufficient to identify the direct targets of a TF of interest.

Computational methods have been developed to infer regulatory interactions between TFs and target genes by integrating information from multiple data resources [10; 11; 46; 66]. Among these studies, Beyer et al. [46] focused on identifying targets of TFs by combining various data, such as ChIP-chip and protein interaction data, into a probabilistic model. They showed that the performance of their method is better than the

error model used to determine target genes only from ChIP-chip experiments. Wang et al. [10] developed an EM-based method MODEM to identify the target genes from ChIP-chip/TFPE, genome promoter sequences and predefined core binding motifs.

I have implemented a novel method called CompMODEM (Comparative MODEM) by extending the MODEM method to incorporate additional phylogenetic conservation information. The intuition of this method is the following: (1) a true direct target of a TF is likely to contain the binding motif recognized by the TF, which is the special short consensus DNA segment with several mismatches and can be detected by ChIP-chip experiments or by examining the promoter sequence of each gene; (2) the gene expression of the true target should have significant either increase or decrease if the TF is mutated/deleted/overexpressed in a TFPE; (3) functional segments, such as TF binding sites, are likely to be conserved during evolution and the level of conservation of orthologous DNA sequences across species is thus informative in identification of TF targets. CompMODEM has been developed as a probabilistic model to integrate these available data resources.

I have applied the algorithm to analyze 514 ChIP-chip [50; 63; 67] and 221 TFPEs [1; 6; 7] for 150 TFs [68-70] of the budding yeast Saccharomyces cerevisiae.

3.2. Methods

CompMODEM is the extension of the MODEM algorithm. The intuition of CompMODEM is the following: a target gene of a TF should contain the binding motif recognized by the TF; the motif should be conserved across species; the gene should have

significant gene expression in TFPE or show strong ratio in ChIP-chip experiments. CompMODEM uses a probabilistic model to integrate the information to identify direct targets of TFs. The inputs to CompMODEM include the following. (A) The gene expression from TFPE or fluorescence ratio from ChIP-chip experiments. I use $\log_2$(ratio) in the calculations. (B) The core motif recognized by a TF. The core motif usually is 6 to 8 bp long and can be obtained from literature or computational predictions. (C) The promoter sequences. I take up to 600bp of the 5' UTR regions in the Saccharomyces cerevisiae genome. To reduce the searching space, I only consider promoters that contained the binding motif (with mismatches allowed) of the TF under consideration. In this study, I allow up to one mismatch to the consensus motif. I also include 7bps at both ends of the consensus motif, called the extended motif, to consider the preferred nucleotides in the flanking regions. (D) The multiple alignments of promoter sequences across four Saccharomyces sensu stricto species, e.g., S. cerevisiae, S. paradoxus, S. mikatae, and S. bayanus. The outputs of the algorithms are: (A) A refined PSFM that includes both the core motif and the flanking regions (the extended motif). (B) The probability of being a true target for each gene. (C) Classification of each gene to be a target or non-target.

The goal of CompMODEM is to maximize the joint likelihood $L(Z;\theta \mid S,E,C)$ of the observed data: the DNA segments (extended motifs) in the promoter regions $S=(S_{nm}:n=1,\ldots,N;m=1,\ldots,M)$, where N is the total number of the extended motifs in S. cerevisiae and M is the number of the fungal genomes under consideration (M=4 in this study) and S_{nm} is the extended motif of the n-th gene in the m-th species;

$E = \left(E_n : n = 1,\ldots,N\right)$, where E_n is the n-th gene's log(ratio) value in a TFPE or ChIP-chip experiment; $C = \left(C_n : n = 1,\ldots,N\right)$, where C_n is the conservation score for the n-th gene; $Z = \left(Z_n : n = 1,\ldots,N\right)$ is a hidden variable and the value of Z_n reflects whether the nth extended motif is a target or not:

$Z_n = \begin{cases} 1, & \text{if gene } n \text{ is a target gene} \\ 0, & \text{otherwise} \end{cases}$ represents all the parameters of the probabilistic model (see below for details).

I assume (A) the distributions of S, E and C are independent given θ and Z; (B) the sequences (extended motifs) are independent from each other. Using Bayes rule, the joint likelihood can be expanded as:

$$\begin{aligned} L\left(Z;\theta \mid S,E,C\right) &= \prod_n P\left(S_{n1},E_n,C_n \mid Z_n;\theta\right) \\ &= \alpha \prod_n \sum_{Z_n \in \{1,0\}} P\left(S_{n1},E_n,C_n \mid Z_n;\theta\right) P(Z_n;\theta) \end{aligned}$$

where S_{n1} is the n-th extended motif in the template species S. cerevisiae and α is a constant that can be ignored when maximizing the likelihood.

$P\left(Z_n;\theta\right) = \begin{cases} \lambda, & \text{if } Z_n = 1 \\ 1-\lambda, & \text{if } Z_n = 0 \end{cases}$, where λ is a prior parameter of the percentage of true targets among all the extended motifs under consideration.

$$P(S_{n1},E_n,C_n \mid Z_n;\theta)=\begin{cases} P(S_{n1} \mid Z_n;\theta)P(E_n \mid Z_n;\theta), & \text{if no alignment} \\ P(S_{n1} \mid Z_n;\theta)P(E_n \mid Z_n;\theta)P(C_n \mid Z_n;\theta), & \text{otherwise} \end{cases},$$

$$P(S_{n1} \mid Z_n;\theta)=\begin{cases} \prod_l f^1_{l,S_{n1}(l)}, & \text{if } Z_n=1 \\ \prod_l f^0_{l,S_{n1}(l)}, & \text{if } Z_n=0 \end{cases},$$

$$P(E_n \mid Z_n;\theta)=\begin{cases} \frac{1}{\sqrt{2\pi}\sigma_E^1} e^{-\frac{(E_n-\mu_E^1)^2}{2(\sigma_E^1)^2}}, & \text{if } Z_n=1 \\ \frac{1}{\sqrt{2\pi}\sigma_E^0} e^{-\frac{(E_n-\mu_E^0)^2}{2(\sigma_E^0)^2}}, & \text{if } Z_n=0 \end{cases},$$

$$P(C_n \mid Z_n;\theta)=\begin{cases} \frac{1}{\sqrt{2\pi}\sigma_E^1} e^{-\frac{(C_n-\mu_C^1)^2}{2(\sigma_C^1)^2}}, & \text{if } Z_n=1 \\ \frac{1}{\sqrt{2\pi}\sigma_C^0} e^{-\frac{(C_n-\mu_C^0)^2}{2(\sigma_C^0)^2}}, & \text{if } Z_n=0 \end{cases},$$

where f^1 and f^0 are the PSFMs for the targets and non-targets, respectively. $f_{l,S_{nl}(l)}$ or $f^0_{l,S_{nl}(l)}$ denotes the occurrence frequency of nucleotide $S_{nl}(l)$ at position l ($l=1,\ldots,L$) in the corresponding matrix. L is the total length of the extended motifs. I use the occurrence frequency of each nucleotide in the S. cerevisiae genome as pseudocounts in calculating $f_{l,S_{nl}(l)}$ and $f^0_{l,S_{nl}(l)}$. The distributions of expression and conservation scores are assumed to be Gaussian distributions. μ and σ are the mean and standard deviation of the Gaussian distributions, respectively. The superscripts of 1 and 0 represent targets and non-targets, respectively, and the subscripts of E and C indicate expression and conservation values, respectively. When considering conservation, the alphabet includes a, c, g, and t as well as gap in the multiple sequence alignments, denoted as "-". The length of the extended motifs in all species is the same as that in S. cerevisiae. If there is

no alignment for n-th gene of m-th species, S_{nm} is allowed to be a missing value. The conservation score for the n-th extended motif C_n is computed as

$$C_n = \frac{\sum_{m=2}^{M}\left(I(S_{nm} \text{ not missing}) \sum_{l\in\{\text{core motif}\}} \log\left(f^{1}_{l,S_{nm}(l)} \right) \right)}{\sum_{m=2}^{M} I(S_{nm} \text{ not missing})}$$, where $I(\cdot)$ is indicator function. If $\sum_{m=2}^{M} I(S_{nm} \text{ not missing}) = 0$ is assigned to be a missing value. The conservation score measures how similar the aligned sequences in the other fungal species are to the core motif in S. cerevisiae. If all the aligned sequences are exactly same, the conservation score is zero; otherwise, it is a negative value. If a substantial amount of conservation data was missing, the likelihood function has considerable local maxima, which would cause bad performance of EM algorithm. For the purpose of overcoming this problem, CompMODEM is performed without conservation information first to achieve a more accurate initial parameter $\theta^{(0)}$ and then integrates conservation to refine the target gene list.

The log-likelihood is $\log L\left(Z_n;\theta \mid S,E,C\right) = \sum_n \log P\left(S_{n1},E_n,C_n \mid Z_n;\theta\right)$. An EM algorithm was used to estimate the model parameters iteratively. In the E step, the expected log-likelihood was computed using the current parameters $\theta^{(k)}$. In the M step, the parameters are updated as following by maximizing the expected log-likelihood calculated in the E step.

$$\lambda = \frac{\sum_n r_n}{N},$$

$$f_{lb}^1 = \frac{\sum_n r_n \cdot I(S_{n1}(l) = b)}{\sum_n r_n}, \qquad f_{lb}^0 = \frac{\sum_n (1 - r_n) \cdot I(S_{n1}(l) = b)}{\sum_n (1 - r_n)},$$

$$\mu_E^1 = \frac{\sum_n r_n E_n}{\sum_n r_n}, \qquad \sigma_E^1 = \left(\frac{\sum_n r_n \left(E_n - \mu_E^1 \right)^2}{\sum_n r_n} \right)^{1/2},$$

$$\mu_C^1 = \frac{\sum_n I(C_n \text{ not missing}) r_n C_n}{\sum_n I(C_n \text{ not missing}) r_n}, \qquad \sigma_C^1 = \left(\frac{\sum_n I(C_n \text{ not missing}) r_n \left(C_n - \mu_C^1 \right)^2}{\sum_n I(C_n \text{ not missing}) r_n} \right)^{1/2},$$

$$\mu_C^0 = \frac{\sum_n I(C_n \text{ not missing})(1 - r_n) C_n}{\sum_n I(C_n \text{ not missing})(1 - r_n)}, \quad \sigma_C^0 = \left(\frac{\sum_n I(C_n \text{ not missing})(1 - r_n) \left(C_n - \mu_C^1 \right)^2}{\sum_n I(C_n \text{ not missing})(1 - r_n)} \right)^{1/2}.$$

The EM stopped when the convergence criterion $\left| \lambda^{(k+1)} - \lambda^{(k)} \right| \le 10^{-6}$ was satisfied. The expression data are normalized such that $\mu_E^0 = 0$ and $\sigma_E^0 = 1$, which are fixed as the non-target parameters. The initial value $\lambda^{(0)}$ was set to 0.

The probability of being a target is:

$$r_n = \frac{P\left(S_{n1}, E_n, C_n \mid Z_n = 1; \theta \right)}{P\left(S_{n1}, E_n, C_n \mid Z_n = 1; \theta \right) + P\left(S_{n1}, E_n, C_n \mid Z_n = 0; \theta \right)}.$$

All the extended motifs are then classified into target or non-target category:

$$\begin{cases} \text{target}, & \text{if } r_n > 0.5 \\ \text{non-target}, & \text{otherwise} \end{cases}.$$

3.3. Results

To establish a comprehensive list of direct targets of TFs in Saccharomyces cerevisiae, I have applied CompMODEM to 514 ChIP-chip [50; 63; 67] and 221 TFPEs [1; 6; 7] for 150 TFs, among which 89 TFs have known target lists from the databases SCPD [68], YPD [69] and TRANSFAC [70]. I have also collected 279 possible binding motifs of the 150 TFs from databases and literatures and refined them using CompMODEM.

GO analysis [25] has been performed to output the significant biological processes among the targets of each TF. The top 10 enriched biological processes of the target genes have been manually examined, and then the best binding site and target list for each TF have been chosen. The TFs and their targets in the majority of transcription modules are consistent in terms of the biological processes.

The purpose of CompMODEM is to improve the accuracy of MODEM by accounting for conservation information as well as binding and expression. Another set of 148 modules with the same 148 TFs has been again constructed using MODEM. Among the 148 constructed modules, only 89 have known target lists. The comparison between the two sets of these 89 modules shows that CompMODEM has the advantage of reducing both false positive and false negative rates as expected. First, many non-target genes, which are mistakenly identified by MODEM because of their binding sites and/or expression levels, are filtered out due to their bad conservation across multiple species, resulting in a remarkable reduction of the false positive rates. 58 out of 89 transcription modules constructed using CompMODEM have fewer false positives than their counterparts constructed using MODEM. Moreover, a considerable number of true

target genes, which are unable to be recognized by MODEM because of their weak binding and/or small expression ratio, are identified successfully by CompMODEM only because their strong conservation information can compensate for their poor binding and/or expression information. 48 out of the 89 modules have higher positive predictive values (PPVs) in CompMODEM than in MODEM, while only 19 out of the 89 modules show the opposite trend. This indicates that CompMODEM has the capability of augmenting true positives while not increasing false positives. Furthermore, because of the informative conservation, CompMODEM exhibits higher specificity, sensitivity, negative predictive values (NPVs) and accuracy than MODEM does (Table 5).

In addition, CompMODEM also performs more accurately than another traditional approach, ChIP-chip [17]. The ChIP-chip targets are only determined by p-values of enrichment ratios with an arbitrary threshold 0.001 [17]. A comparison has been drawn between 68 modules constructed using CompMODEM and ChIP-chip, respectively. 40 out of these 68 ChIP-chip transcription modules determine fewer target genes than the corresponding CompMODEM modules due to the stringent p-value cutoff of the ChIP-chip approach. Consequently, CompMODEM identifies more true positives as well as more false positives than ChIP-chip approach. This explains why CompMODEM achieves higher sensitivity and NPVs while lower specificity and accuracy than ChIP-chip method does. But beyond our expectation, CompMODEM acquires higher PPVs than ChIP-chip approach, revealing that, in CompMODEM, the increase in the number of false positives is not as fast as that in the number of true positives (Table 5).

Furthermore, I have compared the overall performance of these three classification methods using ROC curve. Figure 8 shows that the ROC curve of CompMODEM is closer to the upper left corner than the ones of MODEM and ChIP, indicating that CompMODEM has better overall performance than the other two methods have.

3.4. Discussion

Although the various data resources for constructing transcription modules have been dramatically accumulated and well applied recently, none of them is capable of identifying alone a TF's target genes with high sensitivity and specificity. Thus, a joint probabilistic model for transcription module discovery, CompMODEM, has been developed with integration of a variety of information such as the sequence, expression and conservation.

Compared with the traditional clustering methods based on multiple microarray experiments, CompMODEM requires only one single experimental array such as ChIP-chip or TFPE. This makes CompMODEM promising in that numerous single experimental data are available. Moreover, the correlation of the gene expressions within each cluster does not imply the genes are under the control of a specific TF due to the complex combinatory regulations. CompMODEM takes the advantage of binding information to distinguish the direct targets from the indirect ones. Another advantage of CompMODEM is that it reduces false positives at no expense of increasing false negatives, which is favorable for biologists to test the predictions at least cost. This

desirable performance benefits from the sufficient resources, which provide the evidences of both binding activity and functional activity. Also worth noting, the conservation information, as an extra constrain, results in the less sensitivity to experimental noise.

The first of the two limitations of CompMODEM is that the prior knowledge of the TF binding site is necessary. However, the consensus binding motifs of all TFs are not completely known. This problem could be solved using the approaches for motif finding like REDUCE [71], MDscan [72]. Though the putative binding motifs from the motif finding methods are not as accurate as the canonical ones, CompMODEM is able to refine the input binding motifs. The second limitation is that CompMODEM encounters serious local-maximum problems when dealing with many missing conservation data. This can be overcome when the ortholog sequence database become complete.

Acknowledgements

Chapter 3, in full, is currently being prepared for submission for publication of the material. Liu, Jie; Wang, Wei. The dissertation author was the primary investigator and author of this material.

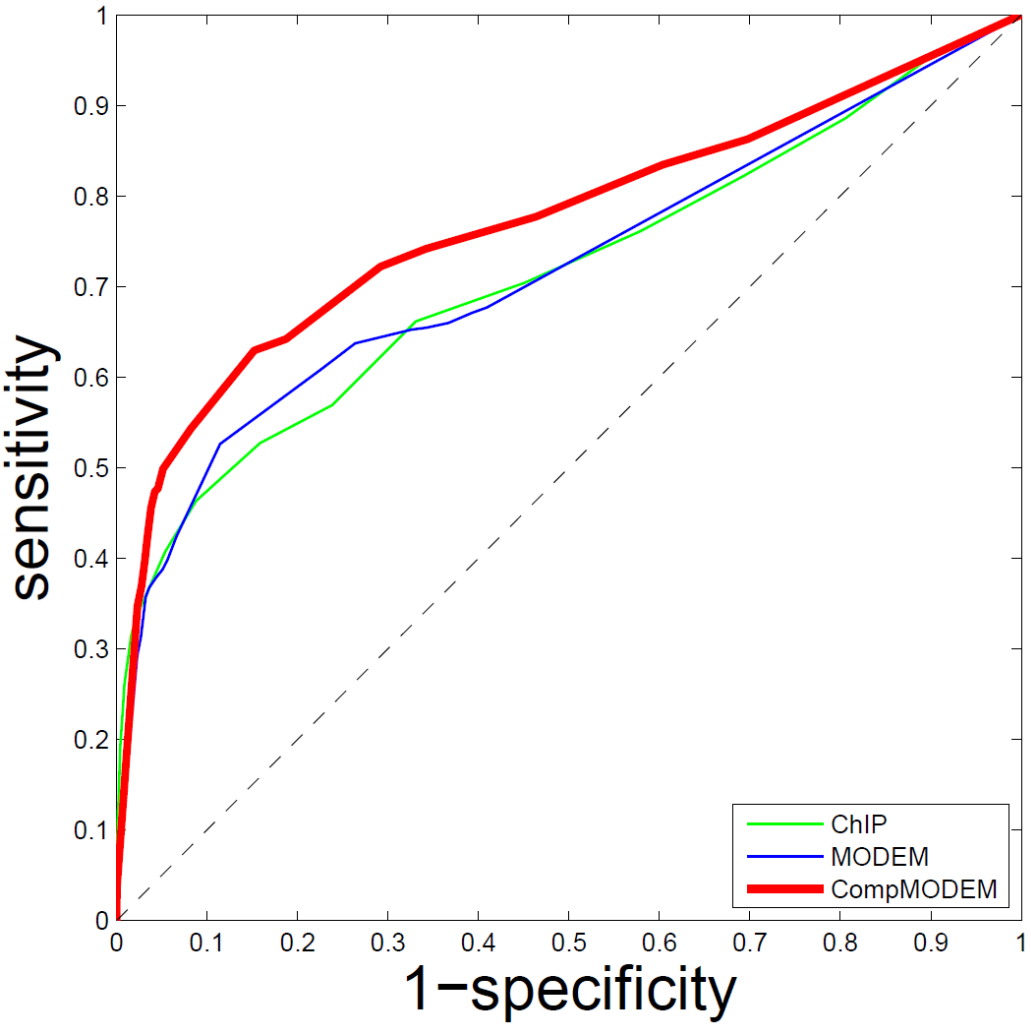

Figure 8. ROC curve.

Table 5. Compare CompMODEM with MODEM and ChIP-chip.

Method	Sensitivity		Specificity		PPV		NPV		Accuracy	
	Worse	Better	Worse	Better	Worse	Better	Worse	Better	Worse	Better
MODEM (89)	14	26	26	58	19	48	14	26	26	59
ChIP-chip (68)	11	27	45	21	11	36	11	27	45	21

Chapter 4. ActivMiner: Simultaneous Inference of Transcription Factor Activity and Target Genes

4.1. Introduction

Accurate reconstruction of GRN is necessary to understand how organisms, especially for metazoans, have the capability of controlling the highly specific expression of an individual gene while utilizing only a limited number of the TFs. Most recent studies have built the GRN statically by integrating various types of information. [10; 11; 46; 66] However, GRN is dynamic in the sense that it has different realization at different stages of development or in response to different environmental cues. Namely, TFs are activated or inactivated under a particular cellular condition. Consequently the regulatory links between the TFs and their target genes are present or absent and the transcriptional network undergoes condition-specific rewiring. Therefore, inference of the dynamic realization of biological networks still remains a challenge. [73-76]

Efforts have been made recently to identify activities of TFs by comparing the gene expression profiles between the condition of interest and a TF perturbation (mutation, overexpression or deletion of a TF) or chromatin immunoprecipitation with microarray (ChIP-chip) experiment. [43; 77; 78] These methods either were not designed to identify target genes under a particular condition [43; 78] or took target genes of a TF directly from ChIP-chip experiments[77]. Linear regression methods were also developed to identify active DNA motifs recognized by TFs in a single gene expression experiment [51; 79], which implies the activation of the TF. Given the motifs, to identify their direct

targets is not trivial. Das et al. [44; 45] fit a multivariate adaptive regression splines (MARS) model of known binding motifs to gene expression and the gene activation threshold in the model was used to determine direct targets of the active motifs.

All methods mentioned above have been applied to microarray experiments individually. They did not use any information of the coherent gene expression patterns for co-regulated genes in a time series or multiple cellular condition experiments. Segal et al. [49] simultaneously inferred the regulatory motifs and genes in a module using a graphical model. This model is hard to use in practice and the greedy searching algorithm is prone to be trapped in local optima. We present here a probabilistic model called ActivMiner that simultaneously infers the activity and target genes of a TF in an individual experiment while also considering the co-expression of genes across multiple experiments. The target genes of a TF are those containing the binding sites of the TF in their promoters and the gene expression levels being coherent with the activity of the TF. The activity of the TF is defined by the activation or repression of its targets. Since neither the label of the target gene nor the activity of the TF is observed, ActivMiner iteratively infers the activity and target genes of the TF using EM. Currently, this model considers only TFs individually and its output can be used to learn constraints of combinatorial regulation between TFs in the future.

In this study, ActivMiner is applied to temporal expression profiling of cell cycle [2] and dilution [5] experiments for Saccharomyces cerevisiae and identified active TFs. Some of the results reinforced prior biological knowledge and others were new hypotheses that provide new insights and await experimental validation.

4.2. Methods

4.2.1 The ActivMiner model

The computational task is to simultaneously infer the activities of TFs and their target genes. It is reasonable to assume that, when a TF is active, the majority of its target genes are either induced or repressed, and when the TF is inactive, the expression levels of the majority of its target genes are indistinguishable from the non-targets. Therefore, there are two sets of latent variables: $Z_1,\ldots,Z_N$, where N is the total number of genes, and $X_1,\ldots,X_M$, where M is the number of experiments:

$$Z_i = \begin{cases} 1, & \text{if gene } i \text{ is a target gene} \\ 0, & \text{otherwise} \end{cases}$$

The observed data are promoter sequences and expression profiles, denoted by S and E respectively. Our model assumption is that targets and non-targets differ in their patterns of extended motifs and the distributions of their expressions, where the extended motif is defined as the core consensus binding motif plus flanking regions on both sides. More explicitly, the distribution of nucleotides at each position of the extended motif is assumed to be multinomial and the positions are independent from each other. The resultant product multinomial distribution is represented by a position specific weight matrix (PSWM). There are two PSWMs in our model, one for targets and the other for non-targets. Expressions of targets and non-targets in the j^{th} experiment are considered to be drawn from two Gaussian distributions, $N(\mu_j,\sigma_j^2)$ and $N(\mu_0,\sigma_0^2)$ if the TF is active in the experiment. If the TF is inactive in experiment j, targets' expressions are assumed to be drawn from the background distribution, i.e., $N(\mu_0,\sigma_0^2)$, as well. Without loss of

generality, we assume that $\mu_j > \mu_0$ because if the mean of target expressions is in fact below the background mean, we could simply negate the logarithm of expression ratios. Parameters μ_0 and σ_0 for the background expression distribution are fixed to the values calculated from all genes in the genome.

Notations

S ---Sequence data

E ---Expression data

S_i ---Promoter sequence of gene i

E_{ij} ---Gene i's expression in microarray experiment j

Z ---Hidden variables. $Z_i = 1$ if gene i is a target gene, and $= 0$ otherwise

X ---Hidden variables. $X_j = 1$ if the TF is active in experiment j, and $= 0$ otherwise

θ ---Collection of all model parameters

$\hat{\theta}$ ---Current estimate of θ ---Proportion of target genes in the entire genome

π ---Proportion of experiments in which the TF is active

pi ---3.14159

μ_j ---Mean expression level of target genes in the j^{th} experiment

σ_j^2 ---Variance of target gene expressions in the j^{th} experiment

f_{jk} ---Frequency of observing nucleotide j at the k^{th} position in the binding motif PSWM

f_{jk}^0 ---The (j,k) entry in the background PSWM

N ---Total number of genes

M ---Total number of experiments

W ---Length of extended motif

A ---Size of sequence alphabet ($=4$ for DNA sequences)

$a_k(S_i)$ ---Nucleotide at the kth position in sequence S_i ---Indicator function

The observed log-likelihood of our model is

$$L(\theta \mid S,E,Z,X) = \iint_{Z,X} P(S,E,Z,X;\theta)dXdZ$$
$$= \iint_{Z,X} P(S,E \mid Z,X;\theta)P(Z;\theta)P(X;\theta)dXdZ$$
$$= \iint_{Z,X} P(S \mid Z;\theta)P(E \mid Z,X;\theta)P(Z;\theta)P(X;\theta)dXdZ$$

Direct optimization of the observed log-likelihood is difficult; therefore the maximization is carried out iteratively via the EM algorithm that maximizes a conditional expectation of the complete log-likelihood at each iteration. Assuming the independence between S and E, we can write the complete log-likelihood as:

$$l_c = \log P(S,E,Z,X;\theta) = \log P(S \mid Z;\theta)P(E \mid Z,X;\theta)P(Z;\theta)P(X;\theta)$$

E-step:

$$Q(\hat{\theta};\theta) = \mathrm{E}_{Z|S,E,C,\theta} \log L(\theta;S,E,Z,X)$$
$$= \sum_{i=1}^{N} \log \sum_{Z_i \in \{1,0\}} P(S_i \mid Z_i;\theta)P(E_i \mid Z_i,X_i;\theta)P(Z_i;\theta)P(X_i;\theta) \quad .$$
$$= \sum_{i=1}^{N} \log \left(\begin{array}{l} \lambda \left(\prod_{k=1}^{W} f_{a_k(i),k} \right) \left(\prod_{j=1}^{M} \left(\frac{1}{\sqrt{2\pi}\sigma_j} e^{-\frac{(E_{ij}-\mu_j)^2}{2(\sigma_j)^2}} \pi + \frac{1}{\sqrt{2\pi}\sigma^0} e^{-\frac{(E_{ij}-\mu^0)^2}{2(\sigma^0)^2}} (1-\pi) \right) \right) + \\ (1-\lambda) \left(\prod_{k=1}^{W} f^0_{a_k(i),k} \right) \left(\prod_{j=1}^{M} \frac{1}{\sqrt{2\pi}\sigma^0} e^{-\frac{(E_{ij}-\mu^0)^2}{2(\sigma^0)^2}} \right) \end{array} \right)$$

M-step:

$\theta^{\text{new}} = \underset{\theta}{\text{argmax}}\, Q(\hat{\theta};\theta)$.

By taking partial derivatives of $Q(\hat{\theta};\theta)$ with respect to unknown parameters in θ and setting the partial derivatives to zero, we obtain a set of formulas for updating the model parameters:

$$\lambda = \left(\sum_{i=1}^{N} \hat{r}_i \right) / N \tag{1}$$

$$\pi = \left(\sum_{j=1}^{M} \hat{q}_j \right) / M \tag{2}$$

$$\mu_j = \frac{\sum_i \hat{r}_i E_{ij}}{\sum_i \hat{r}_i} \tag{3}$$

$$\sigma_j = \sqrt{\frac{\sum_i \hat{r}_i (E_{ij} - \mu_j)^2}{\sum_i \hat{r}_i}} \tag{4}$$

$$f_{jk} = \frac{\sum_{i=1}^{N} \hat{r}_i I(a_k(i) = j)}{\sum_{i=1}^{N} \hat{r}_i}, \tag{5}$$

$$f_{jk}^{0} = \frac{\sum_{i=1}^{N} (1 - \hat{r}_i) I(a_k(i) = j)}{\sum_{i=1}^{N} (1 - \hat{r}_i)}, , \tag{6}$$

where $\hat{r}_i = E_{Z,X|S,E;\hat{\theta}}(Z_i)$ is the probability of gene i being a true target under current parameter estimates; $\hat{q}_j = E_{Z,X|S,E;\hat{\theta}}(X_j)$ is the probability of the TF being active in

experiment j under $\hat{\theta}$. Then we have: $\hat{r}_i \approx \frac{P(S_i \mid Z_i = 1;\hat{\theta})P(E_{i\cdot} \mid Z_i = 1;\hat{\theta})P(Z_i = 1;\hat{\theta})}{\sum_{Z_i=0,1} P(S_i \mid Z_i;\hat{\theta})P(E_{i\cdot} \mid Z_i;\hat{\theta})P(Z_i;\hat{\theta})}$, and

$\hat{q}_j \approx \frac{P(S, E_{\cdot j} \mid X_j = 1;\hat{\theta})P(X_j = 1;\hat{\theta})}{\sum_{X_j=0,1} P(S, E_{\cdot j} \mid X_j;\hat{\theta})P(X_j;\hat{\theta})}$.

4.2.2. Flow of the algorithm

Given a TF and its DNA sequence consensus binding motif, genes that contain the consensus binding motif are input into tightClust [80], a clustering algorithm for identifying tightly co-expressed genes. The program can find up to a pre-specified number of clusters, 20 in this study. The tight clusters are then scanned for enrichment of known targets. The known targets were taken from the SCPD [68], YPD [69] and TRANSFAC [70] databases as well as the targets identified using CompMODEM algorithm. Enriched clusters are fitted by a parametric model using the ActivMiner algorithm described above. Upon convergence, non-targets are removed from the clusters and targets are retained. To decide which genes are targets and which genes are non-targets, the following rule is used: if $P(Z_i = 1;\hat{\theta}^{mle}) > 0.5$ where $\hat{\theta}^{mle}$ denotes the MLE, then gene i is classified as a target; otherwise, gene i is considered as a non-target. Similarly, for a TF to be regarded as being active in experiment j, the probability $P(X_j = 1;\hat{\theta}^{mle})$ is required to be greater than 0.5.

Since tight clusters were formed from genes with the consensus binding motif only, each resultant enriched tight cluster is then relaxed to incorporate genes lacking the consensus binding site but exhibiting highly similar expression patterns with genes

already in the cluster. The relaxed clusters of genes are input to the EM algorithm once again to finalize the lists of target genes and the profile of activities.

Each TF may correspond with multiple clusters that capture the various subgroups of the TF's target genes. Two clusters are merged together if and only if the amount of overlap is over eighty percent.

4.2.3. Tight clustering

tightClust [80] is a resampling-based method for finding tight and stable clusters. The intuition is that the tightest and most stable cluster of genes will be grouped together under repeated sampling. The method proceeds in a sequential manner where the most stable and the tightest cluster is removed first. Unlike most other clustering algorithm, e.g., K-means or hierarchical clustering, tightClust does not have to assign all genes into clusters. This feature of tightClust suits our purpose well, because we are not interested in forming clusters of all genes in the genome, but rather subsets of genes which are co-regulated by the same set of TFs.

To find more meaningful seed clusters that are more likely to be mediated by a given TF, only genes that contain the putative binding motif (consensus motif with one motif in this study) of the TF in their promoters were input to tightClust. In addition, if there were more than 2500 genes having a putative binding site, top 2500 genes with the most variance of gene expression or the largest ChIP-chip ratios were selected. A favorable by-product was a sizable reduction in the execution time of the tightClust procedure.

4.2.4. Relaxation of tight clusters

When tightClust was run, we have only considered genes with putative binding sites. To consider the degeneracy of TF binding motifs, we relaxed each enriched tight cluster to include genes containing no binding motif but exhibiting similar expression patterns. Four criteria were used to select additional genes:

$$cor(G,\bar{E}) \geq \min_{1\leq i\leq n} cor(E_{i.},\bar{E})$$
$$cor(G',\bar{E}') \geq \min_{1\leq i\leq n} cor(E_{i.}',\bar{E}')$$
$$cor(G'',\bar{E}'') \geq \min_{1\leq i\leq n} cor(E_{i.}'',\bar{E}'')\ ,$$
$$d(G,\bar{E}) \geq \max_{1\leq i\leq n} d(E_{i.},\bar{E})$$

where $E_{i.}$ denotes the expression profile of the ith gene within the cluster; $\bar{E}$ is the average expression profile of the cluster; and G represents the expression profile of the new gene under investigation. The first and second order derivatives of gene expression profiles are denoted by $\bar{E}'$ (or G') and $\bar{E}''$ (or G''), respectively. The first three criteria are based on the correlation, and the last criterion imposes constraint on the Euclidean distance.

The first criterion requires that the correlation between the expression profiles of the new gene and the cluster average should be at least as large as the smallest correlation between the expression profile of a gene already in the cluster and the cluster average. The second and third criteria are analogous to the first criterion but impose requirements on the first and second order derivatives. Because the expression signatures were

measured at discrete time points, the derivatives were estimated by using the finite difference formulas.

4.2.5. Mergence of clusters

For each TF, multiple target gene clusters could be found presumably representing distinct subgroups under combinatorial regulation. If there exist two clusters of the same TF sharing more than eighty percent of the target genes, these two clusters are merged. And if a target gene appears in more than one clusters regulated by the same TF, that gene is assigned to the cluster in which it has the highest probability of being a true target. (Figure 9)

4.3. Results

4.3.1. Cell cycle

First, ActivMiner was applied to detect both the targets and activities of the yeast TFs， which play important roles in the cell cycle using the microarray timecourse data sets [2]. These cell cycle microarray data are taken from wild-type Saccharomyces cerevisiae cultures synchronized by α-factor arrest, and arrest of a cdc15 temperature-sensitive mutant and elutriation, respectively. The reason to use these data as input is that many of these TFs have been well studied, making it easy to examine the accuracy of our new method. Recent studies [81; 82] have clearly recovered the interaction of the ten yeast cell cycle transcriptional regulators that can be classified into three groups. In brief,

Group I, including Mbp1, Swi4, Swi6 and Stb1, regulates the late G1 genes. Group II contains four TFs (i.e. Mcm1, Fkh1, Fkh2, and Ndd1), which control the G2/M genes' regulation. Group III consists of Mcm1, Swi5 and Ace3 and regulates the M/G1 genes' regulation.

The results I obtained (Figure 10 and Table 6) are well consistent with the previous literature. In Saccharomyces cerevisiae, gene expression in the late G1 phase is activated by two transcription regulatory complex, SBF and MBF. SBF contains Swi4 and Swi6 proteins and activates the transcription of G1 cyclin genes, cell wall biosynthesis genes, and the HO gene. MBF is composed of Mbp1 and Swi6 and activates the transcription of genes required for DNA synthesis. In addition, Stb1 is another late G1 gene regulator in cell cycle with a role in regulation of MBF-specific transcription at Start [83; 84]. ActivMiner shows that Mbp1, Swi4 and Swi6 share most targets genes as well as the activity pattern. For example, one of Mbp1 subgroups, which consists of 32 targets in total (Mbp1 group1 in Table 6), shares 18 targets with STB1, 19 targets with Swi4_1 (Swi4 group1) and 26 targets with Swi6_1 (Swi6 group1). Mbp1 shares the most common target with Swi6 because they form the MBF complex. As the G2/M activators, Mcm1, Ndd1, Fkh1 and Fkh2 work together, resulting in the same activity pattern and the statistically significant overlaps between Mcm1 (group1 and group2) and the other Group II factors. Similarly, in Group III, Mcm1, Ace2 and Swi5 also share activity pattern and target genes because all these three TFs are M/G1 transcriptional regulators.

Since the TFs in Group I, II and III activate their targets in different cell cycle phases, the activations of these three groups vary in sequential time points. Compared with TFs in Group I, those in Group II shift their activations afterwards and those in

Group III shift the activations slightly forwards. This is well consistent with the known knowledge. One interesting but unexpected result is that the TFs in Group III have both common target genes and activity patterns with Group I TFs. Table 6 shows two significant overlaps. One is between Mbp1 group2 genes and the genes of Mcm1 and Swi5. The other is between Mcm1 group3 genes and the genes of Mbp1, Swi4 and Swi6. It can be understood that there is no clearly boundary between M/G1 and late G1 phases. The unexpected results suggest that Group I TFs may have redundant function with Group III TFs in G1 stage.

Furthermore, ActivMiner is able to detect three different expression patterns for Mcm1, two identical to the other Group II TFs and the third one identical to the other Group III TFs. This observation is consistent with our knowledge that Mcm1 is the common TF in Group II and III. Unlike traditional gene regulatory module identification methods, ActivMiner allows one single TF to have multiple modules, because a TF can have multiple functions through the combinatorial regulations with different TFs according to various conditions. This character makes ActivMiner promising and helpful for further GRNs reconstruction. The results of Cell cycle prove that ActivMiner is able to discover most well known TFs and build the correct relationship among them.

The results also display the obvious advantage of ActivMiner over the traditional clustering methods. Although the latter methods can simply assign the genes to 5 stage groups (i.e. M/G1, G1, S, G2, M) [2], they are unable to discover which TF(s) regulate the genes in each group. ActivMiner also classifies the whole genome genes into groups and identifies simultaneously the TF for each group. In addition, it is known that a gene can be the direct target of different TFs with combinatorial interactions. The traditional

clustering methods usually assign each gene one single membership of a group exclusively, but ActivMiner allows each gene to own multiple memberships, which is biologically meaningful.

4.3.2. Dilution

ActivMiner is also applied to the dilution data [5]. In this dilution experiment, yeast cultures grew in chemostats under 36 different continuous culture conditions, namely, six different limiting nutrients each at six different dilution rates. This chemostat growth was limited by one of the following nutrients: glucose (G), ammonium (N), phosphate (P), sulfate (S), leucine (L), or uracil (U). The authors of the experiment applied traditional hierarchical clustering method to analyze the mRNA abundance data obtained from the 36 chemostat cultures and identified different nutrient-specific groups (e.g., G1-G4, P, S, and N). ActivMiner identifies the targets of the well known TFs that play roles in these nutrient-specific groups.

The Nitrogen catabolite repression (NCR) system is used by the cell to control the synthesis of proteins capable of handling poor sources of nitrogen. NCR-sensitive genes are not activated when rich sources are available; whereas they get expressed when only poor sources are left. Gln3, Gat1, Deh1 and Dal80 are four of the GATA gene family and are known TFs regulating NCR via their binding to the GATA sequences upstream of NCR-sensitive genes. In the presence of rich nitrogen sources, Gln3p and Gat1p are sequestered in the cytoplasm and can activate neither NCR-sensitive genes nor Deh1 and Dal80. The consequence of the low concentration of Gln3p in the nucleus is a low-level

expression of Deh1, Dal80 and NCR-sensitive genes. However, when only poor nitrogen sources are available, Gln3p and Gat1p are released into the nucleus. The former activates Gat1 and the two proteins together activate NCR-sensitive genes. After a delay (due to the time taken for transcription and translation), Dal80p and Deh1p are expressed and competitively inhibit these same genes. [85-89] In ActivMiner, I study only Gln3, Gat1 and Dal80 because Deh1 has no known target list. The target genes of Gln3 and Gat1 represent high expression levels in the condition of ammonium dilution, while the expressions of targets of Dal80 are not activated. These two results are agreeable with literature study.

Met4 is one of the transcriptional activators controlling the sulfur metabolic network in Saccharomyces cerevisiae. The Met4 transcriptional system is a simple model system to study the combinatorial control of transcriptional regulation. [90-93] Met4 is recruited to promoter DNA by one of two distinct sets of cofactors that bind different elements in methionine biosynthetic (MET) gene regulatory regions, either Met28-Cbf1 complex or Met28-Met31/32 complex. ActivMiner results display that Cbf1, Met4, Met28, Met31 and Met32 share most of their targets, proving the combinatorial regulation among the five TFs. In addition, the activity patterns of these TFs are very similar and display the activities in the sulfur dilution conditions. Furthermore, Met4 activates the transcription of a battery of MET genes when methionine levels are low, but it is not active when abundant methionine is available. High methionine leads to increased intracellular S-adenosylmethionine (SAM), which triggers the inactivation of Met4 by stimulating its polyubiquitination. [94-96] My results show that all the targets of

Cbf1, Met4, Met28, Met31 and Met32 found by ActivMiner are highly expressed in the six cultures with the limiting sulfate, consistent with our knowledge.

Adr1 controls expression of genes involved in ethanol utilization only after the diauxic transition. During the diauxic shift, when yeast cells deplete the glucose in the medium, the flow of metabolites changes dramatically to adapt to the use of alternative energy and carbon sources, primarily ethanol produced during fermentation. Mig1 is another TF involved in glucose repression. The yeast homolog of the AMP-activated protein kinase, Snf1, promotes Adr1 and Mig1's chromatin binding in the absence of glucose, and the protein phosphatase complex, Glc7/Reg1, represses their binding in the presence of glucose. [97-99] ActivMiner has correctly identified the positive activity of Adr1 and Mig1 in the six cultures all limited by glucose.

Pho4 is a well known TF controlling phosphate metabolic process. When the P_i concentration in the medium is low, the Pho81 protein inhibits the Pho80-Pho85 kinase activity, which in its active state catalyzes a hyperphosphorylation of Pho4. The hypophosphorylated form of Pho4 is preferentially localized to the nucleus, where together with Pho2, it activates target gene transcription. Alternatively, when the P_i concentrations are high, the Pho80-Pho85 kinase phosphorylates Pho4. In addition to having a lower affinity for Pho2 and the nuclear import protein Pse1/Kap121, phosphorylated Pho4 is a preferred substrate of the nuclear export protein Msn5, resulting in extranuclear localization. Phosphorylated Pho4 is thus unable to activate target gene expression [7]. The targets of Pho4 identified by ActivMiner show the high expression level in the six cultures all limited by phosphate. Interestingly, tightClust could not classify any well known target of Pho4 into any cluster at the beginning, but 10 out of 24

Pho4 known targets [11] were identified successfully by applying ActivMiner to these clusters created by tightClust. This makes ActivMiner promising in that it is capable of retrieving the true targets missed by the traditional clustering method.

In summary, ActivMiner can successfully identify the activity of most TFs in different dilution conditions and indirectly displays the combinatorial relations between co-regulators (Figure 11).

4.4. Discussion

ActivMiner is a probabilistic framework for the simultaneous inference of the activity profile and target genes of a TF, making it promising for reconstruction of the whole dynamic GRN in metazoans. For this simultaneous inference, tightClust is used to produce initial clusters of co-expressed genes and then ActivMiner is applied to refine these co-expressed clusters using EM algorithm. As a new approach, ActivMiner is able to give more accurate prediction on transcriptional regulation than the traditional methods, such as ChIP-chip experiments and clustering methods, because this approach integrates direct and indirect evidences of transcriptional regulation, namely the binding information and expression information, respectively, while the traditional methods usually used them separately. For instance, ChIP-chip experiments only take the binding information into consideration, making many non-targets with the binding sites be identified falsely. However, ActivMiner examined the expression lever of all candidate genes with the consensus binding motif to exclude those non-target genes only physically bound but not regulated by the TF. And this is why the ChIP-chip method always

identifies more targets and thus more false positive than ActivMiner does. In addition, ActivMiner scans all genes without the consensus binding sites to successfully retrieve the missing target genes containing weak binding sites by the aid of the significant similarity of expression changing pattern between the missing targets and the pre-identified targets. Therefore, ActivMiner has significant higher specificity than the ChIP-chip method does, but is only slightly less sensitivity than ChIP-chip method, which actually sacrifices its specificity to gain a good sensitivity. Especially in some cases, ActivMiner have both higher specificity and higher sensitivity than ChIP-chip method, indicating that ActivMiner has the capability of removing false positives without losing too many true targets by integrating expression information into binding information.

Furthermore, ActivMiner also gains better performance than traditional clustering method for the same reason. As far as we know, although the genes regulated by the same TF have the similar expression patterns under specific conditions, the genes with the similar expression patterns do not necessarily infer that they are expressed directly by the same TF. Generally, most previous clustering methods, such as hierarchical clustering and tightClust, are unable to explicitly present which TFs directly control the transcriptional regulation in each set of co-expressed genes. And the members in each co-expression set created by a clustering method can be controlled by a single TF or multiple co-regulated TFs. By integrating binding information, ActivMiner not only filters out some indirect targets without binding sites, but also identifies which TF regulates the set of co-expressed genes. Table 7 displays the comparison results, showing that ActivMiner outperforms ChIP-chip and tightClust. ChIP-chip method only gives higher sensitivity

and NPV but lower specificity, PPV and accuracy than ActivMiner does. In addition, tightClust has both lower sensitivity and lower specificity than ActivMiner has.

In ActivMiner, I combine two traditional methods, clustering and EM, commonly used to identify the gene modules. On one hand, Clustering methods only group the genes co-expressed but not co-regulated. EM algorithm is able to integrate binding information that can distinguish co-regulation from co-expression. Moreover, clustering methods are always faced with the selection of a cutoff, which cannot be too loose or too stringent. But in ActivMiner, there is no need to be very careful about the cutoff in tightClust, because EM algorithm will refine each cluster. On the other hand, EM algorithm can be easily trapped in a local optima because of an arbitrary initialization. The performance of clustering before EM provides a reasonable initialization and avoids the local optima.

In addition to identification of targets and activities of a TF, ActivMiner is able to demonstrate the combinatorial interactions between co-regulation TFs. By sharing the target genes and similar activity patterns, some well-known TFs represent the combinatorial regulations which are consistent with our knowledge. Furthermore, ActivMiner allows a single TF to have multiple modules, which is biologically reasonable. Some TFs have different target lists because of their distinct combinatorial regulation with other TFs. However, ActivMiner is not designed to discover the mechanisms of combinatorial regulation; the rules of regulatory interactions between co-TFs remain unclean. The problem might be solved using another program called GBNet [34] which our group has developed to reveal the combinatorial regulation rules.

Acknowledgements

Chapter 4, in full, is currently being prepared for submission for publication of the material. Liu, Jie; Yu, Ron X.; Wang, Wei. The dissertation author was the primary investigator and author of this material.

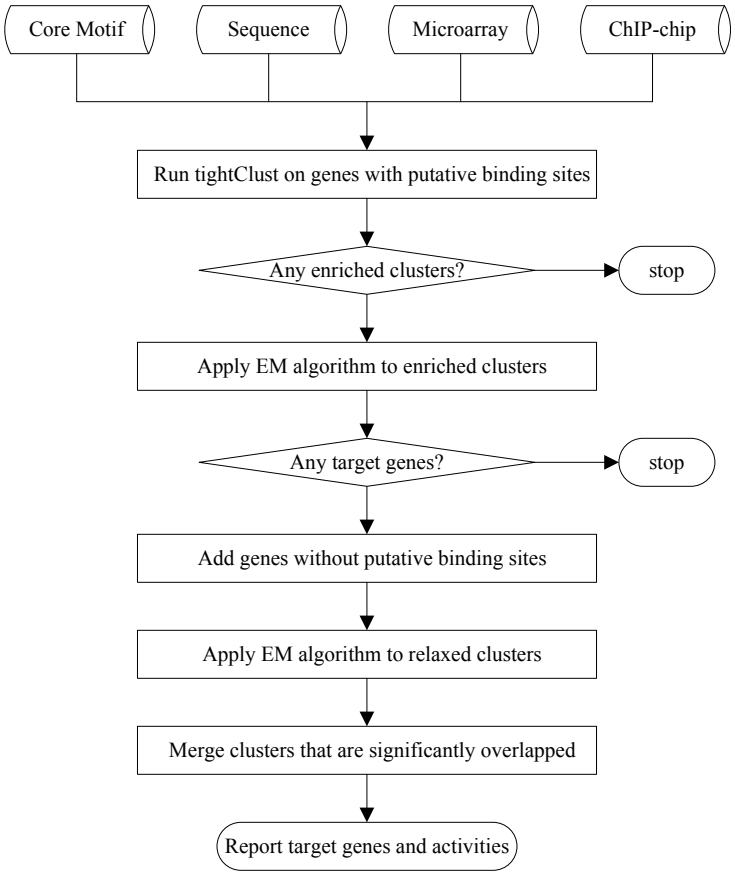

Figure 9. Flow chart for ActivMiner.

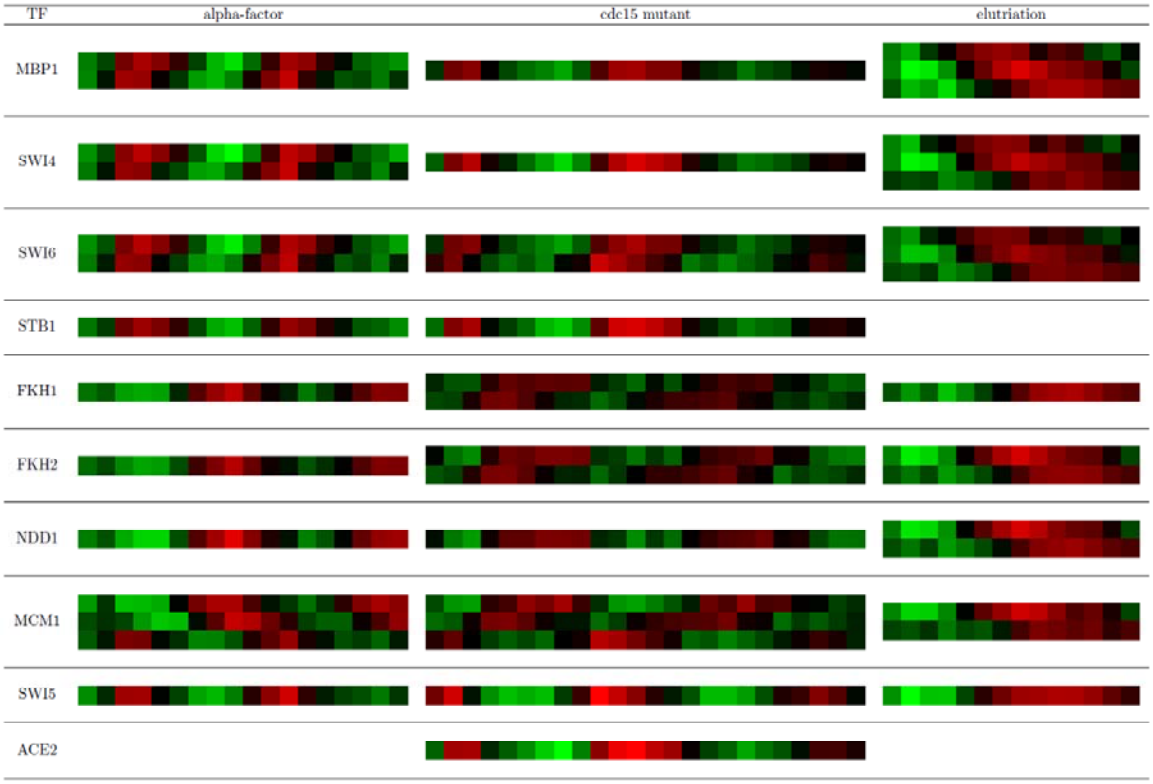

Figure 10. The activities of ten well known TFs in cell cycle.
Red and green colors represent up-and down-regulation, respectively. The brightness of the color is proportional to the ratio.

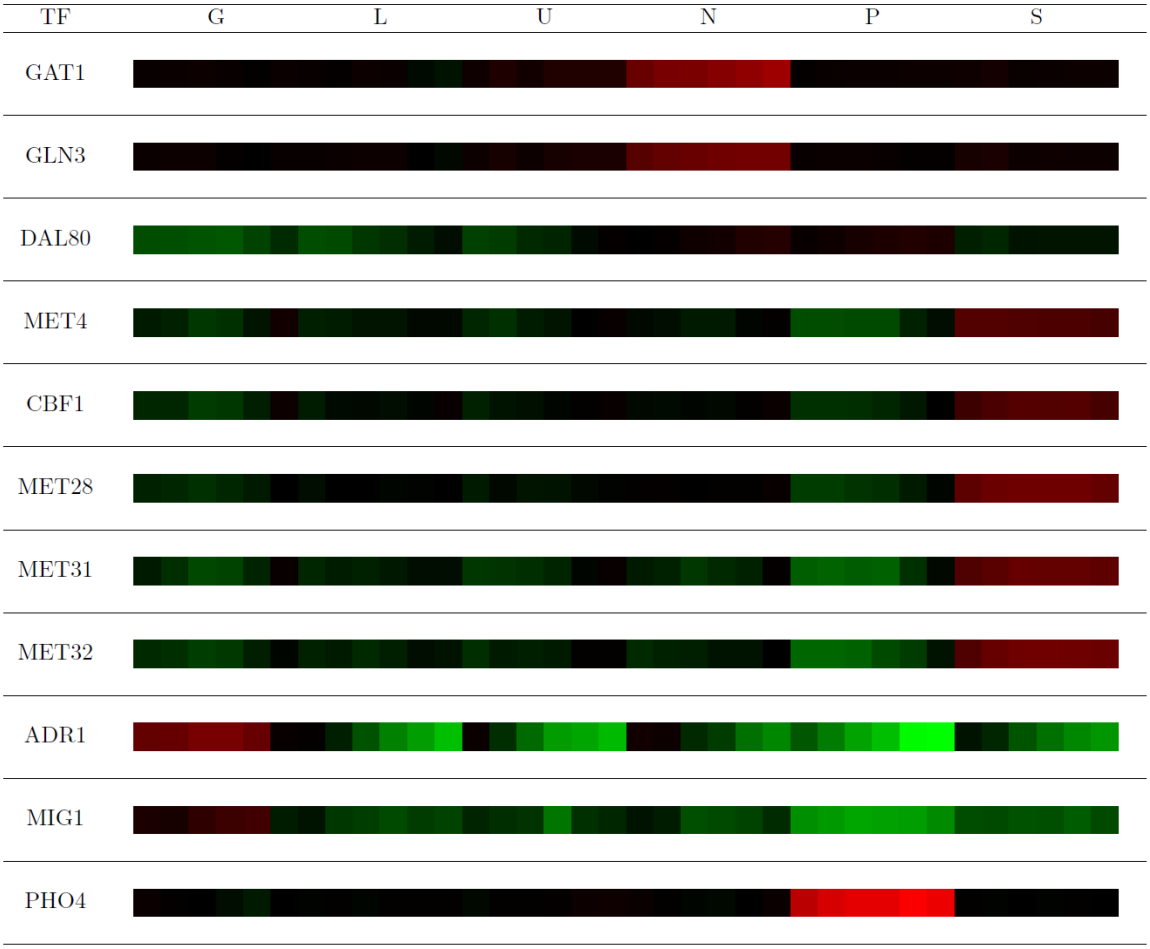

Figure 11. The activities of well known TFs in dilution experiments.

Table 6. The overlap between subgroups in alpha-factor arrest experiment.

Mbp1 Subgroup Overlap				Mcm1 Subgroup Overlap				
		Group1	Group2			Group1	Group2	Group3
		32	32			21	11	29
STB1	22	18	1	FKH1	24	8	8	0
SWI4_1	30	19	3	FKH2	30	11	9	0
SWI6_1	32	26	1	NDD1	18	8	8	0
SWI4_2	24	0	19	SWI5	16	0	0	13
SWI6_2	38	0	30	MBP1	32	0	0	23
MCM1	29	0	23	SWI4	24	0	0	21
SWI5	16	0	13	SWI6	38	0	0	28

Table 7. Compare ActivMiner with ChIP-chip and tightClust.

Method	Sensitivity		Specificity		PPV		NPV		Accuracy	
	Worse	Better	Worse	Better	Worse	Better	Worse	Better	Worse	Better
ChIP-chip (32)	17	9	6	26	7	20	16	9	5	27
tightClust (96)	4	40	6	69	1	49	4	40	3	74

Chapter 5. Connections between our methods

Our group has endeavored to identify the direct target genes of a TF from an entire genome. I have developed three algorithms (i.e. MODEM [10], TRANSMODIS [11], and CompMODEM), which are all EM-based algorithms with different integration of multiple sources of data. Among them, MODEM, integrating both sequence and expression data, is the predecessor of the other two. The inputs to MODEM are a public TFBS, the whole genome promoter sequences and a single genome-wide microarray measurement related to an interested TF, such as ChIP-chip or TFPE. The underlying idea of MODEM is that the true direct targets of a TF not only contain the binding sites of the TF in its promoter sequence, but also change the level of enrichment in a ChIP-chip experiment or expression in a TFPE. MODEM utilizes diverse types of data, which results in higher sensitivity and specificity than the previous traditional methods based merely on one of the binding site, enrichment and expression information. Besides accurately identifying the direct target genes of a TF, MODEM refines the input consensus motif by outputting a position-specific frequency matrix (PSFM) that presents extra precise information of the binding motif. In addition, MODEM broadens its usage by inputting one single ChIP-chip or TFPE array instead of a large set of experimental arrays, because the ChIP-chip experiments or TFPEs related with a special TF provide small number of arrays, usually replicates, which failed to be used in the old pattern recognition method. However, the inevitable notable noise in each array may trap MODEM in local optima and thus reduce the quality of its performance.

To address this problem, I developed another program named TRANSMODIS based on MODEM. Firstly, TRANSMODIS takes multiple TFPE arrays instead of a single array as the input and assumes that the true direct targets are the genes containing the consensus motif of the TF of interest as well as exhibiting consistent expression changes in most of the TFPEs. Compared with MODEM, TRANSMODIS is less sensitive to noise in individual experiments because of the consistency requirement on gene expression level across multiple experiments. Secondly, TRANSMODIS performs the search for optimal initial parameter values, which makes EM algorithm avoid being trapped in local optima. Finally, using the refined PSFM, the by-product of EM algorithm, TRANSMODIS scored the genes that do not contain a copy of the consensus binding motif in their promoter sequences and retrieve the omitted true targets due to the absence of the consensus binding motif. These three improvements increase the performance of TRANSMODIS to accurately identify the direct targets of PHO4 in yeast and the targets of DAF-16 in worm.

Like TRANSMODIS, CompMODEM is another attempt to enhance the accuracy of MODEM by integrating phylogenetic conservation, as well as sequences and expression. By joining phylogenetic conservation information into MODEM, CompMODEM simultaneously reduces both false positives and false negatives. On one hand, some false positives chosen by MODEM with random binding motifs and casual enrichment or expression changes can be filtered by CompMODEM due to the poor conservation across the close phylogenetic specials. On the other hand, the missing true targets having weak binding sites and small enrichment or expression changes are

retrieved with the help of the complement of strong conservation information in CompMODEM.

The common purpose of the above three methods is to statically identify the direct true targets of a TF of interest. However, the activities of a TF are dynamic and thus the targets of the TF are alterable according to the activities of the TF under different experiment conditions. I developed another algorithm called ActivMiner to simultaneously infer the TF's activity in each experiment and target genes of the TF corresponding to its activity. The target genes of a TF are those containing the binding sites of the TF in their promoters and the gene expression levels, coherent with the TF's activity. Meanwhile, the activity of the TF is defined by the activation or repression of its targets. Since neither the label of the target gene nor activity of the TF is observed, ActivMiner iteratively infers the activity and target genes of the TF using EM within every cursorily pre-grouped cluster.

Theoretically, ActivMiner has the capability to give more accurate prediction on transcriptional regulation than the traditional methods, such as ChIP-chip experiments and clustering methods, because this approach integrates direct and indirect evidences of transcriptional regulation, namely the binding information and expression information, respectively, which are usually used separately by previous methods. Moreover, ActivMiner combined two traditional methods, clustering and EM, which are usually individually used to identify gene modules. The combination of these two methods overcomes the limits of each method. Traditional clustering method is quite dependent on the cutoff/threshold, which cannot be too loose or too stringent. A loose cutoff results in low specificity, while a tight one results in low sensitivity. By combining EM algorithm,

clustering method may start with a stringent cutoff and then automatically loosen it by maximizing the likelihood. On the other hand, EM algorithm usually is sensitive to the initialization, which is always pre-defined based on an unknowledgeable guess. The benefit of performing clustering before EM is to provide a better starting point and to avoid being trapped in local optima. Thus, the results are more accurate by combining two methods than utilizing one single method.

The most promising benefit of ActivMiner is that it successfully identifies the subgroups of targets of a TF corresponding to different activities of the TF. The subgroups imply this single TF is able to play multiples roles through combinatorial regulation with other cofactors under various conditions. Combinatorial transcriptional regulation is an important means of achieving highly specific expression of individual genes using small groups of TFs. These groups of TFs integrate signals from different pathways to fine-tune the cellular response at the transcriptional level. The complexity of transcriptional regulation in higher species suggests that combinatorial regulation is of particular importance for metazoans. Therefore, ActivMiner provides the valuable information for creating the relations between TFs, which is one of the main purposes in GRNs inferring.

The connection of the above programs I developed is shown in Figure 12. Which program is suitable to use depends on the accessible data sets. For example, if multiple TFPEs are available, TRANSMODIS is more favorable than MODEM. If reliable phylogenetic information is obtainable, CompMODEM is the first choice. However, MODEM is still useful because of its least dependence of experimental data. On the other hand, ActivMiner is most promising due to its powerful and multiple abilities.

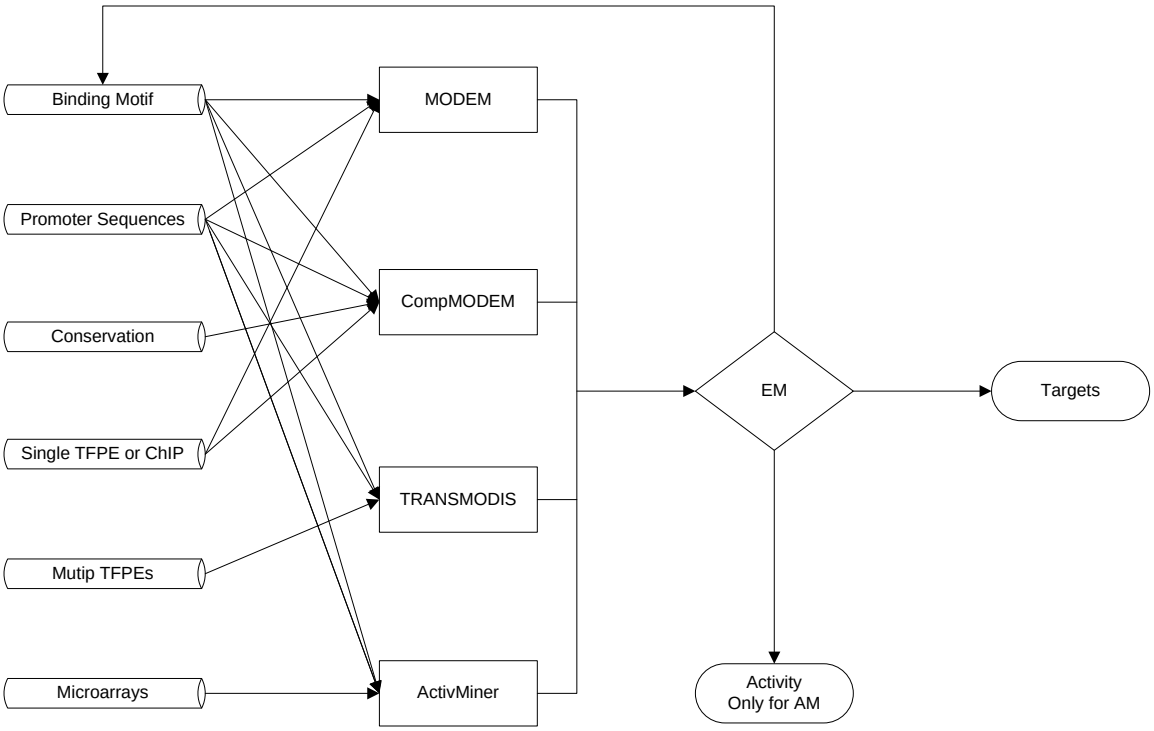

Figure 12. The connections between our methods.

Chapter 6. Further work

Heterogeneous large-scale datasets capturing diverse aspects of the biology of a cell are accumulating at a rapid pace, and efforts to integrate them into a coherent view of cell regulation are intensifying. Over the past few decades, the rapid development of high-throughput genomic technologies (i.e., next-generation sequencing, microarrays, ChIP) has catalyzed the accumulation of various wealthy genome-wide omics data (genomics, transcriptomics, and proteomics). These omics data provide the multi-angle views of the complicate GRNs on a genome-wide scale. For instance, genome sequence data represent the phylogenetic conservation, one of the characters of a TF binding motif, in both "single species, multiple genes" and "single gene, multiple species" investigations [100]. Transcriptome data measured by genome-wide DNA microarrays indicate the direct and indirect biological influence over target genes when a TF is perturbed. ChIP-chip and ChIP-seq data, which belong to proteome data representing protein-DNA interactions, proves the physical binding between a TF and its targets. Another proteome data about the protein-protein interaction aid the discovery of combinatorial regulation between TFs and their co-workers. Many computational methods and mathematical techniques have been proposed to infer GRNs by utilize and analyzing these massive data sets. For example, Lawrence et al. [101] developed a Gibbs sampling method to identify binding motif from genome sequence data. Bailey et al. [15] also found regulatory motifs from genome sequence data by using an EM algorithm. Blanchette et al. [102] implemented Phylogenetic Footprinting for the discovery of regulatory elements in a set of orthologous regulatory sequences from multiple species. Clustering, classification and

visualization methods are originally applied to gene expression data sets for reconstruction of GRNs mainly based on the similarity of expression patterns. To handle the noisy and latent variable in microarray data and to discover the interactions between genes in GRNs, many advanced approaches have been recently developed, such as Boolean networks [35; 103] and Bayesian networks [104; 105]. Johnson et al. [106] developed a method called MAT to identify the binding sites and regulatory targets of a TF from ChIP-chip data. Although individual omics data are beneficial for the reconstruction of GRNs, none is sufficient enough to accurately reverse engineer the complicated GRNs, because each omics datasets are routinely generated to study different aspects of biological systems. For example, even that a gene contains the binding motif of a TF doesn't imply this TF is able to bind to the gene because the binding between a TF and its target genes requires special secondary structures of both the TF and targets and participation of other regulatory proteins. Even the occurrence of the binding between a TF and a DNA in ChIP experiments cannot guarantee the TF playing its regulatory role. On the other hand, the variances of gene expression in TFPEs can provide the evidence of the influence of a TF over its targets, but cannot prove whether this influence is the direct effect from the TF or the indirect effect from the TF's targets.

In the post-genome era, biological researchers have had growing awareness of the need to move beyond the one gene or one protein approach and take a holistic view during all phases of research, including data collection, information processing, interpretation, knowledge acquisition, domain discovery, hypothesis generation and subsequent experimental design. [107] Furthermore, they have realized the necessity of

integrating complementary information churned out by multiple omics technologies to obtain a coherent view of the underlying biology. Besides, different methods have been combined to infer the GRNs. For example, Narayanan et al. [108] developed an algorithm JointCluster that identified sets of genes clustered well in multiple networks of interest, such as coexpression networks summarizing correlations among the expression profiles of genes and physical networks describing protein-protein and protein-DNA interactions among genes or gene-products. These identified clusters, which were derived from multiple genomic datasets and diverse reference classes, agreed with known biology of yeast under the growth conditions, and enabled functional predictions for the uncharacterized genes. Seok et al. [109] developed an algorithm that provides improved performance in the prediction of transcriptional regulatory relationships by supplementing the analysis of microarray data with a new method of integrating information from an existing knowledge base, such as the Yeast Proteome Database. Similarly, Djebbari et al. [110] developed Seeded Bayesian Networks to infer biologically relevant pathways from microarray data as well as prior information of the literature and/or protein-protein interaction data. They demonstrated that the use of seeds derived from the biomedical literature or high-throughput protein-protein interaction data, or the combination, provided improvement over a standard Bayesian Network analysis, allowing networks involving dynamic processes to be deduced from the static snapshots of biological systems that represent the most common source of microarray data. The use of various data resources greatly improves the ability of Bayesian Network analysis to learn gene interaction networks from gene expression data. Furthermore, MacIsaac et al. [16] demonstrated an example of combining two complementary computational strategies,

PhyloCon and CONVERGE, for conservation-based motif discovery. PhyloCon and CONVERGE are both designed to find evolutionarily conserved motifs among a set of genes that are believed to be co-regulated. These two programs use different inputs, search algorithms and scoring statistics. PhyloCon [100] begins with unaligned sequences and generates many local alignments from each orthologous group. The local alignments are assembled using a greedy algorithm to identify patterns that are both conserved in orthologous genes and present in many of the co-regulated promoters. PhyloCon scores sequences by measuring the relative likelihood that a sequence would emerge from the motif model and the background sequence model. By contrast, CONVERGE [50] is an EM-based algorithm for searching over pre-computed, static alignments and discovering specificities. CONVERGE motifs are scored by comparing the frequency of matching sequences in the bound and not-bound genes using a hypergeometric distribution. The motifs discovered by PhyloCon and CONVERGE are often complementary. The authors combined these motifs and then expanded the map of yeast regulatory sites, which revealed an elaborate and complex view of the yeast genetic regulatory network.

Knowing the tendency and advantage of utilizing different information resources instead of a single type of data, I have developed MODEM, TRANSMODIS and CompMODEM to identify the targets of a TF from multiple available data, including ChIP-chip or TFPE data, canonical or pre-identified core binding motifs, genome promoter sequence and phylogenetic conservation information. These complimentary data provide insights into different aspects of cell regulatory system and improve both specificity and sensitivity of gene module identification methods. In ActivMiner, we have combined clustering method and EM algorithm to identify a TF's activities and targets

simultaneously. Either Clustering method or EM algorithm alone is able to identify the activities or/and targets of a TF, but the combination of these two methods can improve the identification performance. First, the main disadvantage of clustering method is the arbitrary choose of a cutoff/threshold, which might be too loose or too stringent. Combined with EM algorithm, clustering method may start with a stringent cutoff and then loosen it by maximizing the likelihood. On the other hand, EM algorithm usually is sensitive to the starting point, which is always initialized based on a guess. The benefit of performing clustering before EM is to provide a better starting point and to avoid being trapped in local optima. Thus, the combination of two methods yields more accurate results than one single method.

Neither any of the experimental data is sufficient, nor is any of computational approaches perfect. However, it can be reasonably expected that at least some of the problems mentioned above will be considerably relieved in the near future. The emergence of new experimental techniques, along with the development of databases and other infrastructural provisions giving access to published and unpublished experimental data, is promising to relieve the data bottleneck. Together with the continuing increase of computer power, this might allow hitherto impractical approaches of modeling and simulation to be tried. [111] For example, with the development of the technology of NGS, ChIP-seq experiments already are able to reduce the noise and increase the resolution in comparison with ChIP-chip experiments. Also, two novel computational approaches for modeling gene regulatory networks, PBN and DBN, have drawn the most interest in the field of systems biology, [112] because these methods have several advantages over traditional clustering methods. Firstly, the methods are designed to

handle the missing values and latent variables. Secondly, they are capable of discovering causal relationships and interactions between genes other than positive correlation. Thirdly, they can be easily implemented to integrate multiple data resources. Finally, these efforts might result in computer-supported modeling environments that integrate a variety of experimental and computational tools to assist the biologists in unraveling the structure and functioning of GRNs.

In addition, GRN reconstruction has quickly generated a large number of hypotheses with regard to pathways and networks that remain speculative without experimental verification. This may be attributable to the many unrealistic assumptions and/or simplifications made in the methodological development. Neither have the potential and values of reverse engineering t been well appreciated yet. Indeed, interdisciplinary and concerted studies are urgently needed in the area of reverse engineering of GRNs. Biologists, physicists, engineers, and mathematicians need collaborate closely together to develop novel reverse engineering approaches based on more biologically realistic assumptions and to generate more experimentally testable hypotheses for systems biology. Only through committed interdisciplinary cooperation can the promise of reverse engineering be realized; this could greatly facilitate a holistic and quantitative understanding of biological systems. [113]

APPENDIX

Table 8. Pho4p target genes identified by TRANSMODIS.

Gene ORF/Name	p*	Extended motif	A†	B†	C†	D†	E†	F†	G†	H†
YAR071W/PHO11	1	gcgttcacacgtgggtttaaa	4.00	3.01	3.71	4.29	4.25	3.55	4.86	4.50
YBR093C/PHO5	1	gcactcacacgtgggactagc	3.08	1.25	3.09	2.82	4.29	3.08	4.07	4.62
YBR296C/PHO89	1	aatgcagcacgtgggagacaa	2.01	2.26	5.07	5.26	2.62	3.12	4.92	2.79
YDR281C/PHM6	1	tcgctgacacgtgggaggtgg	1.37	0.87	3.40	3.00	3.04	-.29	3.62	0.97
YDR452W/PPN1	1	aaattaggacgtggtttatag	2.60	1.25	0.78	1.76	1.95	1.14	2.29	1.98
YDR481C/PHO8	1	atcgctgcacgtggcccgacg	1.71	0.87	1.80	1.55	1.90	1.32	2.17	2.01
YER037W/PHM8	1	tgtgaagcacgttgctgcccc	0.54	0.16	1.97	0.33	1.53	1.31	2.20	1.97
YER055C/HIS1	0.995	ggtgactcacttggaagcttt	1.35	0.57	0.75	1.28	1.66	0.79	1.55	1.32
YER062C/HOR2	1	tttacgtcacgtgggaggccc	1.91	0.58	1.59	1.02	1.26	1.36	1.35	1.54
YER072W/VTC1	1	tccgagacacgtgctaatatc	3.17	2.43	3.14	2.49	2.51	2.52	3.21	2.04
YFL004W/VTC2	0.999	caagcagcacgtgggtttttt	1.28	0.77	1.39	1.60	2.03	0.23	1.71	1.58
YHR136C/SPL2	1	agcggagcacgtgggaaaaga	2.45	3.72	4.65	4.61	1.52	3.01	5.25	3.09
YHR215W/PHO12	1	gcgttcacacgtgggtttaaa	3.89	3.08	4.20	4.16	4.71	3.25	5.23	3.73
YJL012C/VTC4	1	tcatccgcacgtggctgcaca	2.22	2.10	2.89	3.30	2.80	1.90	3.07	3.09
YJL117W/PHO86	1	gcgcccgcacgtgctctttat	1.40	0.89	1.48	1.36	2.08	0.95	1.78	1.35
YML123C/PHO84	1	acacgtccacgtggaactatt	3.30	5.09	5.34	5.49	3.72	3.53	5.37	2.78
YPL018W/CTF19	1	gagggcccacgtggcttaata	-.12	1.86	1.92	1.86	1.77	1.63	2.10	0.51
YPL019C/VTC3	1	gagggcccacgtggcttaata	3.02	3.47	3.94	4.30	4.04	2.25	4.09	2.70
YOL084W/PHM7	1	atgtgcgcaagtgcttagaaa	1.35	2.33	2.31	0.12	1.06	1.29	0.99	1.16

* p denotes the probability of being a target gene

† The set of microarray experiments are: A. Low-Pi vs High-Pi in WT (NBW7) exp1; B. Low-Pi vs High-Pi in WT (NBW7) exp2; C. Low-Pi vs High-Pi in WT (DBY7286); D. PHO4c vs WT; E. pho80 vs WT; F. pho85 vs WT; G. PHO81c vs WT exp1; and H. PHO81c vs WT exp2.

Table 9. Class 1 ageing genes identified by TRANSMODIS.

Gene ORF	Gene Name	P*	Extended motif	Deviation contrast in log expression level comparing daf-2(RNAi) experiments to mixed timecourse data	Deviation contrast in log expression level comparing daf-16(RNAi):daf-2(RNAi) experiments to mixed timecourse data
T22G5.7	spp-12	1	actatcctgtttacttccaga	1.63	-1.66
T20G5.7	dod-6	1	tgaaaaatatttacttaacat	2.13	-1.36
C24B9.9	dod-3	1	gtgataatgtttaccccgcgg	1.16	-0.64
F28D1.5	thn-2	1	aagatttttttttccaaaaaa	1.54	-0.54
F48D6.4	f48d6.4	1	gttagtttattaacttagttt	1.12	-0.85
F28D1.3	thn-1	1	gtgggtttgtttacagtcctt	1.42	-0.61
Y40B10A.6	y40b10a.6	1	taattattatttactgagtaa	2.16	-1.95

Table 9. Class 1 ageing genes identified by TRANSMODIS. (continued)

ZK384.1	zk384.1	1	tcaacaatgtttgcaactccg	1.06	-1.75
T25C12.2	spp-9	1	aaaaaagtatttacccaaaag	0.45	-2.14
PDB1.1	pdb1.1	1	ctttcattatttactctattc	1.30	-0.62
ZK355.3	zk355.3	1	cacaaaaaatttacttcttgt	1.84	-1.06
K12G11.3	sodh-1	1	cccaaaatgttttctgaacaa	-0.02	-1.39
C02A12.4	lys-7	1	tttatactgtttacttcagtg	1.72	-1.49
R09B5.6	hacd-1	1	ccttttttgttaaccactttt	1.19	-0.80
C55B7.4	acdh-1	1	ctgaaaatgtttatttcttga	0.05	-1.23
K11G9.6	mtl-1	1	tgctggctgtttaccacttca	1.61	-2.82
C54F6.14	ftn-1	1	gggttcttgtttacagaaaca	1.91	-0.96
ZK384.2	zk384.2	1	ggtatgatattttctgaaatt	0.90	-0.69
K07C6.4	cyp-35b1	1	acaaatttatttactaaaatc	1.32	-0.63
B0213.15	cyp-34a9	1	tttataatttttacatttatt	1.24	-0.44
C54D10.1	cdr-2	1	ttaaaactatttaaattcaaa	1.21	-0.53
F11A5.12	stdh-2	1	cagatattattttcttcattc	1.28	-0.35
W06D12.3	fat-5	1	ttttgtttatttacttaatta	1.32	-0.45
T02B5.1	t02b5.1	1	tcatttttatttacatgtact	1.49	-0.12
ZK384.3	zk384.3	1	gaaattatattttctattcca	1.16	-0.15
C08E8.4	c08e8.4	1	gtgaccttgtttactgcctcc	1.34	-0.27
C06B3.4	stdh-1	1	caaaatatatttacagacagt	1.27	-0.35
B0286.3	b0286.3	1	atcattatattttcaaatttt	1.37	-0.19
E01A2.8	e01a2.8	1	ggaaatatgtttactgtaaaa	0.91	-0.79
C50F7.2	clx-1	1	cctcctttatttacattgacc	1.30	-0.16
M02D8.4	m02d8.4	1	cgttgtgtgtttactttattg	1.21	-0.37
C17G10.5	lys-8	1	atgataatgttttccgaaatt	1.07	-0.34
F49A5.6	thn-4	1	atgttgatattttctttcttg	1.20	-0.24
M01H9.3	m01h9.3	0.998	gtttgtctgtttccttcaaag	1.39	-0.06
C52D10.1	c52d10.1	0.992	ataattttatttattgttttt	1.22	-0.26
VZK822L.1	fat-6	0.975	tttatattattttctagaagc	1.20	-0.23
C06B3.5	c06b3.5	0.97	gcgagaatatttacttttttа	1.10	-0.22
C05E4.9	gei-7	0.898	atgtaattgtttactcaactt	1.14	-0.36
C30G12.2	c30g12.2	0.776	gaaaattcattaactgaaaca	0.91	-0.31

Table 10. Class 2 ageing genes identified by TRANSMODIS.

Gene ORF	Gene Name	P*	Extended motif	Deviation contrast in log expression level comparing daf-2(RNAi) experiments to mixed timecourse data	Deviation contrast in log expression level comparing daf-16(RNAi):daf-2(RNAi) experiments to mixed timecourse data
C32H11.9	c32h11.9	1	cttgtgatattcacaaagttt	-1.35	0.47

Table 10. Class 2 ageing genes identified by TRANSMODIS. (continued)

K04E7.2	opt-2	1	ctcgttatgtttactgtgtgt	-0.63	0.82
ZC513.11	str-138	1	ataaaaatatttcctaattta	-0.42	0.30
ZC404.5	srh-28	1	cttctggtatttacaacatta	-0.71	-0.35
ZK6.7	zk6.7	1	gtctgtatgtttacttttggt	-0.40	-0.20
Y38H6C.5	y38h6c.5	1	ttcattttatatactaaattc	-0.71	0.49
T02B11.5	srj-38	1	gttttttgttttctgaaaat	-0.96	-0.65
C32H11.4	c32h11.4	1	agtcacatatttacaaagttc	-0.97	0.76
ZK6.11	zk6.11	1	tttgcgacatttacagtttta	-1.56	0.82
W05B2.6	col-92	1	tattgtttctttactatgttt	-0.42	-0.62
T25C12.3	t25c12.3	1	ttgggaatgtttacttgttgc	-0.59	0.72
K09D9.2	cyp-35a3	1	tatgatatatttacagccccc	-0.95	0.33
T10B5.4	t10b5.4	1	ttccaagtattgacattttcc	-0.68	-0.45
B0554.6	dod-20	1	tccaacttatgtacattaacg	-0.55	0.91
C46E10.2	c46e10.2	1	gtctgcctatttacaagccag	-1.37	-0.45
C32H11.10	dod-21	1	gtatgctttttactgtcaat	-1.99	0.18
Y38H6C.20	y38h6c.20	1	ctttctttgttttctattttt	-1.33	-0.62
F55G11.7	f55g11.7	1	aatgacgaatttacaaatttt	-0.94	0.52
C32H11.12	dod-24	1	atttagatattaactaaagat	-1.39	0.45
K08D9.6	k08d9.6	1	ttcaaattattttcaagttac	-1.95	-1.66
F54B11.4	f54b11.4	1	tataaaatgtttagttaaaga	0.06	1.00
F28B4.3	f28b4.3	1	tttacgatatttagtttttt	-0.53	0.62
W05B2.1	col-94	1	gcggaagtgtttacgatcggt	-0.33	-0.72
F35E12.5	f35e12.5	1	ctttatatttttattattggt	-0.64	0.69
B0207.10	b0207.10	1	ttgaatttatttataattttt	-0.76	-0.37
F55G11.5	dod-22	1	cttaaaatgtttacaggtgga	-1.83	0.68
C31A11.5	c31a11.5	1	tacgatatattttcaattatt	-0.61	0.38
F15E11.12	f15e11.12	1	ctaaaaatatttacttgcctg	-0.51	-0.64
T24B8.5	t24b8.5	1	tttaaattgttttcatacttt	-0.67	0.20
W05B2.5	col-93	1	ttatgcatgtttaaacatttc	-0.51	-0.50
F15E11.1	f15e11.1	1	tttaaaatggttaccgtatca	-1.00	-0.31
F56A4.2	f56a4.2	1	gataatatttttacataaata	-1.39	0.32
F11G11.11	col-20	1	attgaattgtatacttttttt	-0.59	-1.05
H28G03.3	h28g03.3	1	aatttttttatttactaatctt	-1.66	-0.95
F57F4.4	f57f4.4	1	aataaaatttttactttactg	-0.82	0.45
T11F9.2	tag-140	1	cttcatatgtttaaaatttttt	-0.96	-0.76
K10D11.1	dod-17	1	gcaaaattatttacacgtgtt	-1.20	0.50
F46C8.6	dpy-7	1	tctgaaatgtgtacagttgca	-0.47	-0.14
F28H7.3	f28h7.3	1	tttcccatatttacatctcga	-0.87	0.28

Table 10. Class 2 ageing genes identified by TRANSMODIS. (continued)

ZK1037.4	nhr-246	1	tcaataatgtttacaaaaatc	-1.25	-0.93
F22D6.10	col-60	1	tatcaagtttttacacaatca	-0.65	-0.22
F49E12.2	dod-23	1	caattcaaatttacagaaaat	-0.90	0.89
T28H11.2	srm-1	1	tctgaaatatttaaaggtatt	-1.14	-1.20
F55G11.8	f55g11.8	1	ttctacgtattttcactcttt	-0.99	0.28
T03D3.1	ugt-53	1	tactatgtgtttacacaaaaa	-0.55	0.81
F44C4.3	cpr-4	1	tattctttttttacaacttca	-0.29	0.72
T05E12.3	t05e12.3	1	cattttctttttacaaaaaat	-0.23	1.00
F56G4.3	f56g4.3	1	aaaaaaatattaaccgttttt	-0.45	0.61
F32A5.3	f32a5.3	1	ctttcatgatttacaggtttt	-0.51	0.76
F49F1.1	f49f1.1	1	gttactttatttaaaaaattt	-0.29	0.66
T20F10.4	t20f10.4	1	atcttggtatttacaattatt	-0.73	0.13
M18.1	col-129	1	ctttgaatatttacaatttga	-0.26	-0.41
C32H11.13	c32h11.13	1	atttagatattaactaaagat	-0.23	0.66
C15A11.5	col-7	1	ttatcaatatttattaattgc	-0.41	-0.96
C53B4.5	col-119	1	ttttttatatttgcttatcaa	-0.33	-0.69
F53A9.8	f53a9.8	1	tataaaatattaactgaagat	-0.18	0.90
F11H8.3	col-8	1	ttagttttatttatttgttga	-0.15	-0.37
C34F6.2	col-178	1	ttttcagtaattacggtagaa	-0.14	-0.26
R04E5.10	ifd-1	1	atataaatattttctattaaa	-0.01	1.11
C01H6.1	col-61	1	atctaaatatttacaaacaat	-0.29	0.59
F55D12.6	f55d12.6	1	ttcaatttatttatgctttct	-0.52	0.11
W02D3.7	lbp-5	1	tgtgtcctatttactatatat	-0.24	0.80
F55G11.2	f55g11.2	1	tttttcgtgttttcaatttct	-0.30	0.78
C31G12.4	c31g12.4	1	attaatttatttaacgtacta	-0.84	-1.09
C07B5.5	nuc-1	1	tttgctatgtttactaaaatg	-0.30	0.57
D2023.7	col-158	1	tttaaactgttcacagatatt	-0.15	-0.29
ZK757.1	zk757.1	1	tttaccatatttacctctttt	-0.09	0.53
R13H4.3	r13h4.3	1	ctagcatttttttacaaagtta	-0.09	0.88
F07C3.1	ptd-2	1	gatttcctgtttaaaattgtt	-0.54	-0.53
F29C12.1	pqn-32	1	gttctagtatctacaaaatta	0.15	1.01
F15H10.1	col-12	1	ttttcagtatttgctattgac	-0.15	-0.42
F10G8.3	npp-17	1	gttataatatttataatcaaa	-0.53	-0.20
F08G5.6	f08g5.6	1	atataaatatttaccatgtca	-0.15	1.17
F40F4.6	f40f4.6	1	attttcttatttaacgacttt	-0.25	0.69
C25D7.12	c25d7.12	1	cttgttttatttataatcggt	-0.28	0.19
C17F3.3	c17f3.3	1	tttaaaaaatttacacaccaa	-0.17	0.65
T10B9.2	cyp-13a5	1	accgacatatttaccaaggcc	-0.63	-0.93

Table 10. Class 2 ageing genes identified by TRANSMODIS. (continued)

F23B2.12	pcp-2	1	tgataactgtttagagatgtt	-0.01	0.78
F35E12.9	f35e12.9	1	tgtgtaatattgacaaaaatt	-0.11	0.83
C05E11.5	amt-4	1	ctttaagaatttacacctcac	-0.09	0.80
F57B1.4	col-160	1	acggaactatttactgaaaac	-0.11	-0.67
K06H6.5	k06h6.5	1	ttctgtatatttaagattttt	-0.29	-0.20
F54F11.2	f54f11.2	1	ctaaaaatatttgccaaatac	-0.03	0.80
Y62H9A.4	y62h9a.4	1	ttctgattgtttaaacttta	-0.14	0.60
T11F9.9	col-157	1	ttggatatgtttcctataaaa	-0.08	-0.39
C08F11.8	ugt-22	1	gtgaaaatttttactgtttct	-0.04	0.59
F11D11.8	f11d11.8	1	tttagaatgtttaaggaaaat	-0.29	0.09
C54G4.2	c54g4.2	1	cttctcatatttataaaatga	-1.02	-1.52
W01A11.4	lec-10	1	ctttttgtatttccaaaatga	-0.24	0.38
T25B9.7	ugt-54	1	ataaaattatttacagaaata	-0.46	-0.34
C14A6.1	clec-48	1	atttaatttttacacgatta	-0.03	0.35
F55C7.2	f55c7.2	1	attttcatattttcagacgaa	0.29	1.24
Y106G6D.3	y106g6d.3	1	aactacctgtttactgtagtc	-0.11	0.75
C43D7.5	sdz-6	1	gaaagcctgtttacggatgga	0.32	1.14
ZC416.6	zc416.6	1	agaaaagtatatacaaatcca	-0.14	0.87
F59B2.13	f59b2.13	1	ctctttttatttacatttaaa	-0.31	0.04
F10D11.6	f10d11.6	1	atcatcatatttccattgtcc	-0.33	0.17
T09F5.7	t09f5.7	1	atccaattctttacatttggt	-0.10	0.05
DY3.5	pqn-26	1	tctaaaatgtttaaaatttgt	0.10	0.99
C55B6.5	c55b6.5	1	ataagcatatttattgataga	0.20	1.33
W05E10.4	tre-3	1	tatgaaatatttattgatatc	-0.03	0.60
C15A11.6	col-62	0.999	attaaagtatttaaaaaattt	-0.29	-0.70
B0379.2	b0379.2	0.999	ttataaatatttacgcaatat	-0.26	0.45
Y41E3.2	dpy-4	0.999	ataaaaaaatttactgtttct	-0.40	-1.05
F56H9.1	srx-113	0.999	attccaatatttatttctgtt	0.25	1.13
F16C3.1	f16c3.1	0.999	ttttcggtgtttacacgtctt	-0.50	0.04
F22A3.6	f22a3.6	0.999	gtttgattgtttactgttttg	-0.23	0.04
F38B6.5	col-172	0.999	cctcacttatttacatttctt	0.15	0.97
C46H11.6	c46h11.6	0.999	ttttatatgtttatacttttа	0.03	0.77
F56F4.7	f56f4.7	0.999	ataaaaacgtttacaaacctt	0.04	1.04
B0495.4	nhx-2	0.999	attaatatatatacttttttc	0.01	0.80
F33D11.6	f33d11.6	0.998	ttctagttatttacactttgg	-0.41	-0.16
C52D10.9	skr-8	0.998	aaaaaactgtttaaaatttat	-0.15	0.52
F26B1.4	col-58	0.997	tttcagatatctacttttga	-0.37	-0.65
C24H10.3	c24h10.3	0.996	attagaatatttattgaactc	-0.29	0.18

Table 10. Class 2 ageing genes identified by TRANSMODIS. (continued)

F52C9.5	f52c9.5	0.996	ttattcgtttttactttgtg	-0.35	-0.21
K12G11.4	sodh-2	0.996	cttgaaatgtttagaattctc	-0.31	-0.34
D1007.2	col-52	0.996	ttttaagtgtttcaagttttt	-0.25	0.30
C31H2.2	dpy-8	0.994	ttctgtatgtttattattttt	0.01	0.25
ZK1290.6	zk1290.6	0.993	tttaatctattttctcaagga	-0.03	0.67
R09A8.4	col-182	0.989	tctttttgtttaaatttta	-0.03	0.39
C15A11.1	col-35	0.988	atttacatatttaatcttttc	-0.27	-0.71
F56G4.2	pes-2	0.987	atcattgtatttaccgtatcg	-0.31	0.44
C24G7.1	c24g7.1	0.98	acttacttacttacagtagtt	0.05	0.85
K12H4.7	k12h4.7	0.97	gtttaattgtttactggaact	-0.15	0.25
T22G5.2	lbp-7	0.97	ctgcaatttttttacaaaaaat	-0.07	0.62
E03A3.8	e03a3.8	0.968	tcgtaaatatttacttatttt	-0.26	0.21
Y39D8A.1	y39d8a.1	0.965	gatattttatttacagtaccc	-0.01	0.81
F56D5.1	col-121	0.957	aataaaatttttactatttta	0.11	0.86
C08B11.4	nrf-6	0.946	aattatataattacagtactc	0.19	0.96
C09G5.8	c09g5.8	0.944	aattttatatttccttttttc	-0.13	0.43
B0393.7	b0393.7	0.941	attcaattaattacaattcat	-0.12	0.50
C05A9.1	pgp-5	0.94	tttaccgtgtttacttaataa	0.46	1.23
C46H11.1	c46h11.1	0.938	ttagttatgtatacaaatact	0.06	0.73
C55B7.4	acdh-1	0.914	ctgaaaatgtttatttcttga	0.05	-1.23
R09B5.2	cnc-1	0.912	tatgatttgttttcatttaat	0.16	0.82
R10E11.7	srxa-10	0.91	caagagctatttacacttctg	-0.39	0.12
W01B11.2	sulp-6	0.832	atttacatattttcaaataaa	0.12	0.75
C10A4.7	c10a4.7	0.801	actcaactatttagttttgac	-0.02	0.69
W06D4.2	w06d4.2	0.763	ttccatgtgttaacaataaat	0.17	0.72
C25A1.15	c25a1.15	0.754	ctgaaattatttaaaatttaa	-0.03	0.40
F55F8.7	f55f8.7	0.736	tttcaaatatttccaaaaatt	0.20	1.00
F55D12.4	unc-55	0.724	ctgatcataattacaatattt	-0.12	0.50
B0379.6	b0379.6	0.677	tatttttgtttaccacacac	-0.22	0.41
F46B3.7	f46b3.7	0.661	aatgatcaatttacagatgca	0.03	0.60
C46H11.7	c46h11.7	0.647	ctaaaactgtatacatttttt	-0.31	0.21
T28F2.4	t28f2.4	0.617	ttttttttattaacataagta	0.05	0.93
C42C1.7	c42c1.7	0.534	ttttaaatgtttaaaaatttt	0.07	0.72
M04C9.4	m04c9.4	0.528	gtttaaatattttcaattcga	0.17	0.96
T16G12.1	t16g12.1	0.507	ataataatatttaattaatta	0.00	0.56

Table 11. Compare CompMODEM with MODEM.

TF	Sensitivity		Specificity		PPV		NPV		Accuracy	
	Old	New	Old	New	Old	New	Old	New	Old	New
ABF1	0.1667	0.3333	0.9449	0.9395	0.0135	0.0243	0.996	0.9968	0.9414	0.9368
ACE2	0.25	0.25	0.9875	0.9913	0.0235	0.0333	0.9991	0.9991	0.9866	0.9904
ADR1	0.4	0.4	0.9907	0.9926	0.0606	0.0755	0.9991	0.9991	0.9898	0.9917
ARG80	0.5	0.5	0.9983	0.9965	0.2667	0.1481	0.9994	0.9994	0.9977	0.9959
ARG81	0.375	0.375	0.9979	0.9982	0.1765	0.2	0.9992	0.9992	0.9971	0.9974
ARO80	1	1	0.9983	0.9974	0.1538	0.1053	1	1	0.9983	0.9974
ASH1	0	0	0.9842	0.9881	0	0	0.9998	0.9998	0.9841	0.988
AZF1	1	1	0.9042	0.9365	0.0016	0.0024	1	1	0.9042	0.9365
BAS1	0.5385	0.5385	0.9955	0.9967	0.1892	0.2414	0.9991	0.9991	0.9946	0.9958
CBF1	0.1818	0.0909	0.9902	0.9959	0.0299	0.0357	0.9986	0.9985	0.9889	0.9944
CHA4	0	1	0.9956	0.9949	0	0.0286	0.9998	1	0.9955	0.9949
CIN5	0	0	0.9743	0.9782	0	0	0.9998	0.9998	0.9742	0.9781
CRZ1	0	0	0.9959	0.9982	0	0	0.9994	0.9994	0.9953	0.9976
DAL80	0	0	0.9876	0.9917	0	0	0.9967	0.9967	0.9844	0.9884
DAL81	0.7	0.6	0.9839	0.986	0.0614	0.0606	0.9995	0.9994	0.9835	0.9854
DAL82	0.75	0.75	0.9914	0.9907	0.0952	0.0882	0.9997	0.9997	0.9911	0.9904
ECM22	0.5	0.5	0.9884	0.9911	0.0128	0.0167	0.9998	0.9998	0.9883	0.991
FKH1	0	1	0.9749	0.9775	0	0.0066	0.9998	1	0.9748	0.9775
FKH2	1	1	0.9758	0.9775	0.0123	0.0132	1	1	0.9758	0.9775
GAT1	0.75	0.25	0.9818	0.9872	0.0242	0.0116	0.9998	0.9995	0.9817	0.9868
GCN4	0.4035	0.4737	0.9932	0.9912	0.3382	0.3176	0.9948	0.9954	0.9881	0.9868
GCR1	0.15	0.15	0.9958	0.9959	0.0968	0.1	0.9974	0.9974	0.9932	0.9934
GCR2	0.4444	0.4444	0.9935	0.9949	0.0851	0.1053	0.9992	0.9992	0.9928	0.9941
GLN3	0.5161	0.4516	0.9846	0.9891	0.1356	0.1628	0.9977	0.9974	0.9824	0.9866
GZF3	0	0	0.9863	0.9896	0	0	0.9998	0.9998	0.9862	0.9895
HAC1	0	0.4	0.9989	0.9922	0	0.037	0.9992	0.9995	0.9982	0.9917
HAL9	0	0	0.995	0.9941	0	0	0.9998	0.9998	0.9949	0.994
HAP1	0.7143	0.7143	0.9791	0.9821	0.0671	0.0775	0.9994	0.9994	0.9785	0.9815
HAP2	0.0667	0.1333	0.9943	0.9931	0.05	0.08	0.9958	0.9961	0.9901	0.9892
HAP3	0.037	0.037	0.9973	0.9968	0.0526	0.0455	0.9961	0.9961	0.9934	0.9929
HAP4	0.2593	0.2963	0.9899	0.991	0.0946	0.1176	0.997	0.9971	0.9869	0.9881
HAP5	0.08	0.12	0.9931	0.9935	0.0417	0.0652	0.9965	0.9967	0.9896	0.9902
HSF1	0.75	0.75	0.9911	0.9919	0.169	0.1818	0.9994	0.9994	0.9905	0.9913
IME1	0	0	0.9979	0.9994	0	0	0.9977	0.9977	0.9956	0.9971
INO2	0.4	0.5	0.9932	0.9932	0.1509	0.1818	0.9982	0.9985	0.9914	0.9917
INO4	0.5556	0.5556	0.9868	0.9907	0.102	0.1389	0.9988	0.9988	0.9856	0.9895
IXR1	0	0	0.9901	0.9901	0	0	0.9998	0.9998	0.9899	0.9899
LEU3	0.8571	0.8571	0.9961	0.9961	0.1875	0.1875	0.9998	0.9998	0.9959	0.9959
MAC1	0.375	0.375	0.9992	0.9992	0.375	0.375	0.9992	0.9992	0.9985	0.9985
MATA1	0	0	0.9977	0.9977	0	0	0.9998	0.9998	0.9976	0.9976

Table 11. Compare CompMODEM with MODEM. (continued)

MBP1	0.3684	0.3158	0.9772	0.986	0.0848	0.1143	0.9963	0.996	0.9737	0.9821
MCM1	0.5625	0.625	0.9857	0.9787	0.1593	0.1242	0.9979	0.9982	0.9836	0.977
MET28	0	1	0.8636	0.9632	0	0.0041	0.9998	1	0.8635	0.9632
MET4	0.4444	0.6667	0.9968	0.9949	0.16	0.15	0.9992	0.9995	0.9961	0.9944
MIG1	0.069	0.1379	0.9916	0.9796	0.0345	0.0288	0.9959	0.9962	0.9875	0.976
MOT3	0	0	0.9884	0.9949	0	0	0.9994	0.9994	0.9878	0.9943
MSN1	0	0	0.9922	0.9947	0	0	0.9998	0.9998	0.992	0.9946
MSN2	0.3056	0.2778	0.986	0.9881	0.1058	0.1124	0.9962	0.996	0.9823	0.9842
MSN4	0.3636	0.3939	0.9858	0.9888	0.1132	0.1494	0.9968	0.997	0.9827	0.9859
NDT80	0	1	0.9243	0.986	0	0.0211	0.9997	1	0.924	0.986
OAF1	0	0	0.9958	0.9953	0	0	0.9964	0.9964	0.9922	0.9917
PDR1	0	0.2667	0.9932	0.991	0	0.0625	0.9977	0.9983	0.991	0.9893
PDR3	0.1111	0.1111	0.9989	0.9968	0.125	0.0455	0.9988	0.9988	0.9977	0.9956
PHO2	0.0526	0	0.9708	0.9973	0.0051	0	0.9972	0.9971	0.9682	0.9944
PHO4	0.5833	0.625	0.9971	0.9976	0.4242	0.4839	0.9985	0.9986	0.9956	0.9962
PPR1	0	0.5	0.8998	0.9985	0	0.1667	0.9993	0.9997	0.8992	0.9982
PUT3	0	1	0.9757	0.9979	0	0.125	0.9997	1	0.9754	0.9979
RAP1	0.4857	0.5143	0.9586	0.9647	0.0584	0.0714	0.9972	0.9973	0.9562	0.9623
RCS1	0.6364	0.5455	0.9952	0.9973	0.1795	0.25	0.9994	0.9992	0.9946	0.9965
REB1	0.1905	0.3333	0.9309	0.9068	0.0086	0.0112	0.9973	0.9977	0.9285	0.905
RFX1	0.4	0.4	0.9949	0.994	0.0556	0.0476	0.9995	0.9995	0.9944	0.9935
RGT1	0.1667	0.5	0.9988	0.995	0.1111	0.0833	0.9992	0.9995	0.998	0.9946
RIM101	0	0	0.9914	0.9932	0	0	0.9994	0.9994	0.9908	0.9926
RME1	0.5	0.5	0.9958	0.9892	0.0345	0.0137	0.9998	0.9998	0.9956	0.989
ROX1	0.0769	0.0769	0.9845	0.9893	0.0096	0.0139	0.9982	0.9982	0.9827	0.9875
RPH1	0	0	0.9854	0.9908	0	0	0.9998	0.9998	0.9853	0.9907
RPN4	0.5714	0.5714	0.979	0.9806	0.0278	0.0301	0.9995	0.9995	0.9785	0.9802
RTG1	0.6667	0.6667	0.992	0.9934	0.0702	0.0833	0.9997	0.9997	0.9917	0.9931
RTG3	0.8	0.8	0.9856	0.9896	0.04	0.0548	0.9998	0.9998	0.9854	0.9895
SIP4	0.5	0.5	0.9988	0.997	0.1111	0.0476	0.9998	0.9998	0.9986	0.9968
SKN7	0.381	0.2381	0.973	0.9801	0.0428	0.0365	0.998	0.9975	0.9712	0.9778
SKO1	0	0	0.9934	0.9979	0	0	0.9994	0.9994	0.9928	0.9973
STE12	0.4487	0.4231	0.9878	0.9892	0.3043	0.3173	0.9934	0.9931	0.9815	0.9826
STP1	0	1	0.9979	0.9964	0	0.04	0.9998	1	0.9977	0.9964
SUM1	0.5	1	0.9863	0.9836	0.0109	0.018	0.9998	1	0.9862	0.9836
SUT1	0	0	0.9814	0.9883	0	0	0.9998	0.9998	0.9812	0.9881
SWI4	0.3571	0.2857	0.985	0.9874	0.0476	0.0455	0.9986	0.9985	0.9836	0.9859
SWI5	0.3636	0.2727	0.9919	0.9946	0.069	0.0769	0.9989	0.9988	0.9908	0.9934
SWI6	0.3409	0.3182	0.9775	0.9825	0.0915	0.1077	0.9955	0.9954	0.9733	0.9781
TEC1	0.0682	0.0909	0.9872	0.9894	0.0341	0.0541	0.9938	0.9939	0.9811	0.9835
THI2	0.625	0.625	0.9956	0.9958	0.1471	0.1515	0.9995	0.9995	0.9952	0.9953
UGA3	0	0	0.9967	0.9826	0	0	0.9995	0.9995	0.9962	0.9821

Table 11. Compare CompMODEM with MODEM. (continued)

UME6	0.45	0.35	0.9595	0.9752	0.0629	0.0787	0.9965	0.996	0.9565	0.9715
XBP1	0	0	0.9839	0.9821	0	0	0.9992	0.9992	0.9832	0.9814
YAP1	0.1538	0.2564	0.9941	0.995	0.1333	0.2326	0.995	0.9956	0.9892	0.9907
YAP6	1	1	0.9979	0.9976	0.0667	0.0588	1	1	0.9979	0.9976
YHP1	0	0	0.9901	0.9931	0	0	0.9998	0.9998	0.9899	0.9929
YRR1	0	0	0.9901	0.9928	0	0	0.9994	0.9994	0.9895	0.9922
ZAP1	0.3333	0.3333	0.995	0.9967	0.1081	0.1538	0.9988	0.9988	0.9938	0.9955
	14	26	26	58	19	48	14	26	26	59

Table 12. Compare CompMODEM with ChIP-chip.

TF	Sensitivity		Specificity		PPV		NPV		Accuracy	
	Old	New	Old	New	Old	New	Old	New	Old	New
ABF1	0.1667	0.3333	0.9605	0.9395	0.0187	0.0243	0.9961	0.9968	0.9569	0.9368
ACE2	0.25	0.25	0.9865	0.9913	0.0217	0.0333	0.9991	0.9991	0.9856	0.9904
ARG81	0.5	0.375	0.9964	0.9982	0.1429	0.2	0.9994	0.9992	0.9958	0.9974
ARO80	1	1	0.9953	0.9974	0.0606	0.1053	1	1	0.9953	0.9974
ASH1	0	0	0.9923	0.9881	0	0	0.9998	0.9998	0.9922	0.988
AZF1	0	1	0.9988	0.9365	0	0.0024	0.9998	1	0.9986	0.9365
BAS1	0.6154	0.5385	0.9946	0.9967	0.1818	0.2414	0.9992	0.9991	0.9938	0.9958
CBF1	0.0909	0.0909	0.9934	0.9959	0.0222	0.0357	0.9985	0.9985	0.9919	0.9944
CHA4	0	1	0.9989	0.9949	0	0.0286	0.9998	1	0.9988	0.9949
CIN5	0	0	0.9805	0.9782	0	0	0.9998	0.9998	0.9803	0.9781
DAL80	0	0	0.994	0.9917	0	0	0.9967	0.9967	0.9907	0.9884
DAL81	0.7	0.6	0.9866	0.986	0.0729	0.0606	0.9995	0.9994	0.9862	0.9854
DAL82	0.75	0.75	0.9925	0.9907	0.1071	0.0882	0.9997	0.9997	0.9922	0.9904
FKH1	0	1	0.9787	0.9775	0	0.0066	0.9998	1	0.9785	0.9775
GAT1	0.25	0.25	0.9961	0.9872	0.037	0.0116	0.9995	0.9995	0.9956	0.9868
GCN4	0.386	0.4737	0.992	0.9912	0.2933	0.3176	0.9947	0.9954	0.9868	0.9868
GCR2	0.4444	0.4444	0.9922	0.9949	0.0714	0.1053	0.9992	0.9992	0.9914	0.9941
GLN3	0.3548	0.4516	0.9914	0.9891	0.1618	0.1628	0.997	0.9974	0.9884	0.9866
GZF3	0	0	0.998	0.9896	0	0	0.9998	0.9998	0.9979	0.9895
HAL9	0	0	0.9958	0.9941	0	0	0.9998	0.9998	0.9956	0.994
HAP1	0.7143	0.7143	0.9788	0.9821	0.0662	0.0775	0.9994	0.9994	0.9782	0.9815
HAP2	0.1	0.1333	0.9941	0.9931	0.0714	0.08	0.9959	0.9961	0.9901	0.9892
HAP3	0.0741	0.037	0.9968	0.9968	0.087	0.0455	0.9962	0.9961	0.9931	0.9929
HAP4	0.2593	0.2963	0.9894	0.991	0.0909	0.1176	0.997	0.9971	0.9865	0.9881
HSF1	0.8125	0.75	0.9866	0.9919	0.1275	0.1818	0.9995	0.9994	0.9862	0.9913
IME1	0	0	0.9989	0.9994	0	0	0.9977	0.9977	0.9967	0.9971
INO2	0.35	0.5	0.9938	0.9932	0.1458	0.1818	0.998	0.9985	0.9919	0.9917
IXR1	0	0	0.9958	0.9901	0	0	0.9998	0.9998	0.9956	0.9899
LEU3	0.5714	0.8571	0.9965	0.9961	0.1481	0.1875	0.9995	0.9998	0.9961	0.9959

Table 12. Compare CompMODEM with ChIP-chip. (continued)

MAC1	0.5	0.375	0.9979	0.9992	0.2222	0.375	0.9994	0.9992	0.9973	0.9985
MATA1	0	0	0.9986	0.9977	0	0	0.9998	0.9998	0.9985	0.9976
MBP1	0.2368	0.3158	0.9844	0.986	0.0804	0.1143	0.9956	0.996	0.9802	0.9821
MET28	0	1	1	0.9632	NA	0.0041	0.9998	1	0.9998	0.9632
MET4	0.1111	0.6667	0.9959	0.9949	0.0357	0.15	0.9988	0.9995	0.9947	0.9944
MIG1	0	0.1379	1	0.9796	NA	0.0288	0.9956	0.9962	0.9956	0.976
MOT3	0	0	0.9992	0.9949	0	0	0.9994	0.9994	0.9986	0.9943
PDR1	0	0.2667	0.9988	0.991	0	0.0625	0.9977	0.9983	0.9965	0.9893
PDR3	0.1111	0.1111	0.997	0.9968	0.0476	0.0455	0.9988	0.9988	0.9958	0.9956
PHO2	0	0	0.9998	0.9973	0	0	0.9971	0.9971	0.997	0.9944
PUT3	1	1	0.9986	0.9979	0.1818	0.125	1	1	0.9986	0.9979
RAP1	0	0.5143	1	0.9647	NA	0.0714	0.9947	0.9973	0.9947	0.9623
RCS1	0	0.5455	0.9607	0.9973	0	0.25	0.9983	0.9992	0.9592	0.9965
REB1	0	0.3333	1	0.9068	NA	0.0112	0.9968	0.9977	0.9968	0.905
RFX1	0.6	0.4	0.9956	0.994	0.0938	0.0476	0.9997	0.9995	0.9953	0.9935
RGT1	0	0.5	1	0.995	NA	0.0833	0.9991	0.9995	0.9991	0.9946
RME1	0	0.5	0.994	0.9892	0	0.0137	0.9997	0.9998	0.9937	0.989
ROX1	0	0.0769	0.9991	0.9893	0	0.0139	0.998	0.9982	0.9971	0.9875
RPH1	0	0	0.9998	0.9908	0	0	0.9998	0.9998	0.9997	0.9907
RPN4	0.5714	0.5714	0.9854	0.9806	0.0396	0.0301	0.9995	0.9995	0.985	0.9802
RTG3	0.8	0.8	0.9925	0.9896	0.0741	0.0548	0.9998	0.9998	0.9923	0.9895
SIP4	0.5	0.5	0.997	0.997	0.0476	0.0476	0.9998	0.9998	0.9968	0.9968
SKN7	0.2857	0.2381	0.9723	0.9801	0.0316	0.0365	0.9977	0.9975	0.9701	0.9778
SKO1	0	0	0.9973	0.9979	0	0	0.9994	0.9994	0.9967	0.9973
STE12	0.3846	0.4231	0.9856	0.9892	0.24	0.3173	0.9927	0.9931	0.9785	0.9826
STP1	0	1	0.9974	0.9964	0	0.04	0.9998	1	0.9973	0.9964
SUM1	0.5	1	0.9907	0.9836	0.0159	0.018	0.9998	1	0.9905	0.9836
SUT1	0	0	0.9896	0.9883	0	0	0.9998	0.9998	0.9895	0.9881
SWI4	0.2857	0.2857	0.9764	0.9874	0.0248	0.0455	0.9985	0.9985	0.9749	0.9859
SWI5	0.4545	0.2727	0.986	0.9946	0.051	0.0769	0.9991	0.9988	0.9851	0.9934
TEC1	0	0.0909	0.9935	0.9894	0	0.0541	0.9934	0.9939	0.9869	0.9835
THI2	0.875	0.625	0.994	0.9958	0.1489	0.1515	0.9998	0.9995	0.9938	0.9953
UGA3	0	0	0.9992	0.9826	0	0	0.9995	0.9995	0.9988	0.9821
UME6	0.25	0.35	0.9813	0.9752	0.0746	0.0787	0.9954	0.996	0.9769	0.9715
XBP1	0.2	0	0.9886	0.9821	0.013	0	0.9994	0.9992	0.988	0.9814
YAP1	0.1795	0.2564	0.9902	0.995	0.0972	0.2326	0.9951	0.9956	0.9854	0.9907
YAP6	1	1	0.9911	0.9976	0.0167	0.0588	1	1	0.9911	0.9976
YRR1	0	0	0.9965	0.9928	0	0	0.9994	0.9994	0.9959	0.9922
ZAP1	0.3333	0.3333	0.9973	0.9967	0.1818	0.1538	0.9988	0.9988	0.9961	0.9955
	11	27	45	21	11	36	11	27	45	21

Table 13. Target list in cell cycle.

	alpha-factor arrest	cdc15 mutant	elutriation
YAL029C	FKH2		
YAL040C	MCM1_1		
YAR007C	MBP1_2,STB1,SWI4_2,SWI6_2	ACE2,MBP1,STB1,SWI4,SWI6_1	SWI6_2
YAR018C	FKH2,MCM1_1,NDD1		FKH2_2,NDD1_2,SWI6_3
YAR071W	MCM1_1		
YBL002W			FKH2_1,MBP1_2,MCM1_1,NDD1_1,SWI4_2,SWI5,SWI6_2
YBL032W			FKH2_2
YBL035C			SWI6_1
YBL097W		FKH1_2	
YBL111C	MCM1_3,SWI4_1,SWI6_1		
YBL112C	MBP1_1,MCM1_3,SWI5,SWI6_1		
YBR008C		FKH1_2,FKH2_2	
YBR009C			FKH2_1,MBP1_2,MCM1_1,NDD1_1,SWI4_2,SWI6_2
YBR010W			FKH2_1,MCM1_1,NDD1_1,SWI4_2,SWI6_2
YBR038W			FKH1,MBP1_3,MCM1_2,SWI5
YBR054W	MCM1_1		
YBR070C	MBP1_2		SWI6_1
YBR071W	MBP1_1,SWI4_1,SWI6_1		
YBR073W			SWI4_1
YBR078W			FKH2_1,MBP1_2,MCM1_1,NDD1_1,SWI4_2,SWI5,SWI6_2
YBR088C	MBP1_2,STB1,SWI4_2,SWI6_2		MBP1_1,SWI4_1,SWI6_1
YBR089W			MBP1_1,SWI6_1
YBR092C	FKH2,MCM1_1		
YBR093C	MCM1_1		
YBR162C			FKH1,MBP1_3,MCM1_2,SWI5
YBR202W			MCM1_2
YBR222C			SWI6_3
YBR243C		FKH1_2,FKH2_2	
YCL022C	MBP1_1,MCM1_3,SWI5,SWI6_1		
YCL024W	MBP1_1,SWI4_2,SWI6_1	ACE2,SWI4,SWI6_1	
YCL060C		MBP1	
YCL063W	FKH1,FKH2		
YCL064C	FKH1,FKH2		
YCR005C			SWI4_3
YCR065W	MBP1_2,SWI4_2,SWI6_2	MBP1,SWI4	SWI4_1,SWI6_1
YDL003W	MBP1_2,STB1,SWI4_2,SWI6_2	ACE2,MBP1,STB1,SWI4,SWI6_1	MBP1_1,SWI4_1,SWI6_1
YDL018C	MBP1_2	MBP1	
YDL037C		FKH2_1,NDD1	
YDL055C		SWI4	SWI4_2
YDL101C		MBP1,SWI6_1	
YDL127W		MCM1_3,SWI5,SWI6_2	
YDL163W	MBP1_2,STB1,SWI4_2,SWI6_2		
YDL164C	MBP1_2,STB1,SWI4_2,SWI6_2		SWI6_1
YDL215C			SWI4_3,SWI6_3
YDR097C	MBP1_2,STB1,SWI4_2,SWI6_2	ACE2,MBP1,SWI4	MBP1_1,SWI6_1
YDR113C		SWI6_1	SWI6_1
YDR146C		FKH1_1,FKH2_1,MCM1_1,NDD1	FKH1,FKH2_2,NDD1_2,SWI6_3
YDR150W	FKH2		
YDR224C		SWI4	FKH2_1,MBP1_2,MCM1_1,NDD1_1,SWI4_2,SWI5,SWI6_2
YDR225W		SWI4	FKH2_1,MBP1_2,MCM1_1,NDD1_1,SWI4_2,SWI5,SWI6_2
YDR309C			MBP1_1,SWI4_1,SWI6_1
YDR353W		MBP1,SWI6_1	
YDR367W		FKH1_1,FKH2_1,NDD1	

Table 13. Target list in cell cycle. (continued)

YDR446W		FKH2_1,NDD1	
YDR451C		FKH1_2	FKH2_1,MBP1_2,MCM1_1,NDD1_1,SWI4_2,SWI5,SWI6_2
YDR452W			MBP1_3,SWI4_3,SWI6_3
YDR481C		SWI6_1	
YDR507C	MBP1_2,SWI6_2		SWI4_1,SWI6_1
YDR528W		MBP1	
YDR545W	MBP1_1,MCM1_3,SWI4_1,SWI5,SWI6_1	MCM1_3	
YEL017W		FKH1_2,FKH2_2	
YEL040W	MCM1_3,SWI4_1,SWI6_1		
YEL075C	MBP1_1,MCM1_3,SWI4_1,SWI6_1	MCM1_3,SWI6_2	
YEL076C	MBP1_1,MCM1_3,SWI4_1,SWI6_1		
YEL076C-A		MCM1_3,SWI6_2	
YEL077C	MBP1_1,MCM1_3,SWI4_1,SWI5,SWI6_1	SWI6_2	
YER001W		SWI4	
YER070W	MBP1_2,STB1,SWI4_2,SWI6_2		MBP1_1,SWI4_1,SWI6_1
YER095W	MBP1_2,STB1,SWI6_2	MBP1,SWI6_1	SWI6_1
YER110C		FKH1_1,FKH2_1,NDD1	
YER111C	MBP1_1,STB1,SWI4_2,SWI6_2		
YER124C		ACE2	
YER189W	MBP1_1,MCM1_3,SWI4_1,SWI5,SWI6_1	MCM1_3	
YER190W		MCM1_3,SWI6_2	
YFL037W		FKH1_2	FKH2_1,MCM1_1,NDD1_1,SWI4_2,SWI5,SWI6_2
YFL064C	MBP1_1,MCM1_3,SWI4_1,SWI5,SWI6_1		
YFL066C	MBP1_1,MCM1_3,SWI4_1,SWI6_1		
YFL067W	SWI4_1,SWI5		
YFL068W	MCM1_3		
YGL008C	MCM1_1	FKH2_1,MCM1_1,NDD1	MCM1_2
YGL021W	FKH1,FKH2,MCM1_2,NDD1	MCM1_1	FKH1,FKH2_2,MCM1_2,NDD1_2
YGL038C	MCM1_3,SWI4_1,SWI6_1	SWI6_1	
YGL101W		FKH1_2	
YGL116W	FKH1,FKH2,MCM1_1		FKH1,FKH2_2,NDD1_2,SWI4_3
YGL139W		FKH1_1	
YGL163C		MBP1	
YGL192W		FKH2_2	
YGL225W			MBP1_1,SWI4_1,SWI6_1
YGR014W	SWI4_2,SWI6_2		
YGR041W		ACE2	
YGR044C		ACE2,SWI5	
YGR086C		SWI6_2	
YGR092W			FKH2_2,MCM1_2,NDD1_2,SWI4_3
YGR108W	FKH1,FKH2,MCM1_2,NDD1	MCM1_1	FKH1,MBP1_3,MCM1_2,SWI5
YGR109C		ACE2,MBP1,SWI4,SWI6_1	MBP1_1
YGR138C	FKH1,FKH2,MCM1_1,NDD1	FKH2_2	SWI4_3,SWI6_3
YGR151C	MBP1_2,SWI6_2		
YGR152C		SWI4,SWI6_1	
YGR153W		SWI4	
YGR177C		FKH1_1,FKH2_1,NDD1	
YGR189C	MBP1_2,SWI4_2,SWI6_2	ACE2,MBP1,SWI4,SWI6_1	MCM1_2,SWI4_2,SWI5,SWI6_2
YGR221C		SWI4	SWI4_1,SWI6_1
YGR296W	MBP1_1,MCM1_3,SWI4_1,SWI5,SWI6_1	SWI5,SWI6_2	
YHL028W		MCM1_1	
YHL049C	MBP1_1,SWI4_1,SWI6_1	MCM1_3	
YHL050C	MBP1_1,MCM1_3,SWI4_1,SWI6_1		
YHR005C	MCM1_1		

Table 13. Target list in cell cycle. (continued)

YHR023W	FKH1,FKH2,MCM1_2,NDD1		SWI4_3
YHR061C		FKH2_2,MBP1	SWI6_2
YHR106W		SWI6_1	
YHR143W		ACE2,STB1,SWI4	
YHR149C	SWI6_1		MBP1_1,SWI4_1,SWI6_1
YHR151C	FKH2		
YHR215W	MCM1_1		
YHR218W	MBP1_1,MCM1_3,SWI6_1	MCM1_3,SWI6_2	
YIL026C		MBP1,SWI6_1	
YIL066C		ACE2,MBP1,STB1,SWI4,SWI6_1	MBP1_1,SWI6_1
YIL123W		FKH1_2,FKH2_2,MCM1_2	FKH2_1,MBP1_2,MCM1_1,NDD1_1,SWI4_2,SWI5,SWI6_2
YIL129C		MCM1_2	
YIL131C		MCM1_2	
YIL140W		ACE2,MBP1,STB1,SWI4,SWI6_1	SWI4_1,SWI6_1
YIL141W		STB1,SWI4	MBP1_1,SWI6_1
YIL158W	FKH1,FKH2,MCM1_2,NDD1		FKH1,MBP1_3,NDD1_2
YIL177C		MCM1_3,SWI6_2	
YJL051W		FKH2_1,MCM1_1	FKH2_2,NDD1_2,SWI6_3
YJL073W	MBP1_2,SWI6_2		
YJL074C	MBP1_2,STB1,SWI6_2	MBP1,SWI6_1	
YJL078C		ACE2,MBP1,STB1,SWI4,SWI6_1	
YJL079C		MCM1_1	
YJL092W			SWI6_2
YJL115W	MBP1_1		
YJL118W		FKH1_2,FKH2_2,MCM1_2	
YJL119C		FKH2_2	
YJL121C		FKH1_1,FKH2_1,NDD1	
YJL134W		FKH1_2	
YJL158C			MBP1_3,MCM1_2,SWI4_2,SWI6_2
YJL173C	MBP1_2,SWI6_2	MBP1,SWI6_1	SWI6_1
YJL187C		MBP1,SWI4	SWI6_1
YJL194W		SWI5	
YJL196C			SWI6_3
YJL225C	MBP1_1,MCM1_3,SWI4_1,SWI5,SWI6_1	MCM1_3,SWI6_2	
YJR001W		MCM1_2	
YJR030C		MBP1,SWI6_1	
YJR092W	FKH1,FKH2,MCM1_2,NDD1	FKH2_1,NDD1	
YKL008C		SWI4	
YKL045W	SWI4_2	MBP1,SWI6_1	
YKL069W	FKH1	FKH1_2,FKH2_2	
YKL096W		MCM1_2	SWI4_3
YKL096W-A	FKH2	FKH1_1,FKH2_1,NDD1	FKH1,FKH2_2,MBP1_3,NDD1_2,SWI4_3,SWI6_3
YKL113C	MBP1_2,STB1,SWI4_2,SWI6_2		
YKL116C	SWI6_1	MCM1_3,SWI5,SWI6_2	SWI6_3
YKL164C		SWI5	
YKL185W		SWI5	
YKR010C		FKH1_2	
YKR013W	SWI4_2	ACE2,SWI6_1	MBP1_1,SWI4_1,SWI6_1
YKR077W		SWI5	
YLL066C		SWI6_2	
YLR032W			SWI6_1
YLR049C		MCM1_3,SWI5,SWI6_2	SWI5
YLR079W	SWI5	SWI5	
YLR098C	FKH1,FKH2,NDD1		

Table 13. Target list in cell cycle. (continued)

YLR099C		FKH1_2	
YLR103C	MBP1_1,SWI6_1	MBP1	
YLR110C			MBP1_3
YLR121C		ACE2,SWI4	
YLR131C	FKH2,MCM1_2		FKH1,FKH2_2,NDD1_2
YLR183C	MBP1_2,STB1,SWI4_2,SWI6_2	ACE2,MBP1,STB1	SWI4_1,SWI6_1
YLR190W	FKH1,FKH2,MCM1_1,NDD1	FKH2_1,NDD1	FKH2_2,NDD1_2,SWI4_3,SWI6_3
YLR194C		SWI5	
YLR210W		FKH2_2	
YLR212C		MBP1,SWI6_1	
YLR249W			SWI4_1
YLR300W		MBP1,SWI6_1	MBP1_3,MCM1_2,SWI4_2,SWI5,SWI6_2
YLR342W			SWI4_2
YLR353W		FKH2_2	
YLR372W			MBP1_1,SWI4_1,SWI6_1
YLR382C		MBP1	
YLR383W		MBP1	MBP1_1
YLR389C		MCM1_2	MCM1_2
YLR455W		FKH1_2,FKH2_2,MCM1_2	
YLR462W	MBP1_1,MCM1_3,SWI4_1,SWI6_1	MCM1_3,SWI6_2	
YLR463C	MCM1_3,SWI4_1,SWI5,SWI6_1	MCM1_3	
YLR464W	MBP1_1,MCM1_3,SWI4_1,SWI6_1	SWI6_2	
YLR465C	MBP1_1,SWI5,SWI6_1		
YLR466W	MBP1_1,MCM1_3,SWI4_1,SWI6_1		
YLR467W	MBP1_1,MCM1_3,SWI4_1,SWI5,SWI6_1	MCM1_3,SWI6_2	
YML021C		MBP1	
YML027W	MBP1_2,STB1,SWI4_2,SWI6_2	ACE2,MBP1,STB1,SWI4,SWI6_1	MBP1_1,SWI4_1,SWI6_1
YML033W		MCM1_1	
YML034W	MCM1_2	MCM1_1	SWI4_3
YML048W		FKH1_1,NDD1	
YML052W		FKH2_1,MCM1_1,NDD1	
YML058W		MCM1_1	SWI4_3
YML119W	FKH1,FKH2,NDD1	MCM1_1	
YMR001C	FKH1,FKH2,MCM1_2,NDD1	MCM1_1	FKH1,FKH2_2,NDD1_2
YMR002W		MCM1_1	
YMR003W		MCM1_2	
YMR011W	MBP1_2		
YMR031C	FKH2,MCM1_1	MCM1_1	
YMR032W	FKH1,FKH2,MCM1_1,NDD1	MCM1_1	
YMR078C			MBP1_1,SWI6_1
YMR144W			SWI6_2
YMR163C		FKH1_2	
YMR179W	STB1,SWI4_2,SWI6_2	MBP1,SWI6_1	
YMR183C	FKH1,FKH2		
YMR199W			MBP1_2,SWI4_2,SWI5,SWI6_2
YMR215W			FKH2_1,MCM1_1,NDD1_1,SWI4_2
YMR305C			FKH2_1,MBP1_2,NDD1_1,SWI4_2,SWI6_2
YMR306W	SWI4_2,SWI6_2		
YMR307W		SWI6_1	FKH2_1,SWI4_2,SWI6_2
YNL030W			FKH2_1,MBP1_2,MCM1_1,NDD1_1,SWI4_2,SWI6_2
YNL032W		SWI6_2	SWI6_2
YNL057W	FKH1,FKH2,MCM1_1,NDD1		
YNL058C	FKH1,FKH2,MCM1_1,NDD1	NDD1	FKH1,FKH2_2,NDD1_2,SWI4_3
YNL225C		SWI6_1	

Table 13. Target list in cell cycle. (continued)

YNL231C	MBP1_2		
YNL233W	MCM1_3,SWI6_1		
YNL262W	MBP1_2,SWI6_2	MBP1,SWI6_1	SWI6_1
YNL278W			SWI6_2
YNL283C		FKH2_2	FKH2_1,MBP1_2,NDD1_1,SWI4_2,SWI6_2
YNL289W			SWI6_2
YNL300W	STB1,SWI4_2,SWI6_2		SWI6_1
YNL312W	MBP1_2,STB1,SWI4_2,SWI6_2		
YNL313C	MBP1_2		
YNL328C		SWI5	
YNL339C	MBP1_1,MCM1_3,SWI5,SWI6_1	MCM1_3,SWI5,SWI6_2	MCM1_1
YNR009W			SWI4_1,SWI6_1
YOL007C		ACE2,MBP1,STB1,SWI4,SWI6_1	MBP1_1,SWI4_1,SWI6_1
YOL017W	MBP1_1,MCM1_3,SWI6_1	MBP1	
YOL019W		MBP1,SWI6_1	
YOL030W	FKH1	FKH1_1	
YOL090W	MBP1_2,STB1,SWI4_2,SWI6_2	ACE2,MBP1,STB1,SWI4,SWI6_1	SWI6_1
YOL113W		SWI6_1	SWI6_2
YOL158C	MCM1_1		
YOR025W	MCM1_2	FKH2_1,MCM1_1,NDD1	
YOR058C		MCM1_1	
YOR066W	MCM1_1		
YOR073W		FKH1_2	
YOR074C	STB1,SWI4_2		SWI4_1,SWI6_1
YOR100C			SWI6_3
YOR246C	FKH2		
YOR247W			FKH2_1,MBP1_2,MCM1_1,NDD1_1,SWI4_2,SWI6_2
YOR256C		MCM1_1	
YOR298W		FKH1_1,MCM1_1	
YOR313C			FKH1,FKH2_2,MBP1_3,NDD1_2,SWI5,SWI6_3
YOR315W	FKH1,FKH2,MCM1_2,NDD1		FKH1,FKH2_2,MBP1_3,MCM1_2,NDD1_2,SWI5,SWI6_3
YOR316C			SWI4_3,SWI6_3
YOR323C		FKH2_2	
YOR324C		FKH2_2,MCM1_2	
YOR325W		FKH1_2,FKH2_2	
YOR326W		MCM1_2	MCM1_1
YPL014W	SWI6_1		
YPL057C	MBP1_2,STB1,SWI4_2,SWI6_2		
YPL111W		MCM1_2	
YPL116W		FKH1_2,MCM1_2	
YPL124W		SWI6_1	
YPL127C			FKH2_1,MBP1_2,MCM1_1,NDD1_1,SWI4_2,SWI6_2
YPL128C		FKH1_2,FKH2_2,MCM1_2	
YPL141C	FKH1,FKH2,MCM1_2,NDD1		FKH1,FKH2_2,SWI4_3,SWI6_3
YPL153C	MBP1_2,STB1,SWI4_2,SWI6_2	MBP1,SWI6_1	
YPL221W	MBP1_2,SWI6_2		
YPL242C	FKH1,FKH2,MCM1_1,NDD1		FKH1,FKH2_2,MBP1_3,NDD1_2,SWI4_3,SWI6_3
YPL255W	MBP1_2,SWI4_2,SWI6_2		SWI6_1
YPL256C	SWI4_2,SWI6_2	ACE2,MBP1,STB1,SWI6_1	SWI4_1,SWI6_1
YPL267W	MBP1_1,SWI4_2,SWI6_1		
YPL283C	MBP1_1,MCM1_3,SWI4_1,SWI5,SWI6_1		
YPR001W			SWI4_3
YPR006C			FKH1,FKH2_2,NDD1_2
YPR104C	FKH1		

Table 13. Target list in cell cycle. (continued)

YPR119W		FKH1_1,FKH2_1,MCM1_1,NDD1	FKH1,FKH2_2,MCM1_2,NDD1_2,SWI5
YPR120C	MBP1_1,SWI6_1	MBP1,SWI6_1	
YPR135W	MBP1_2,STB1,SWI4_2,SWI6_2		
YPR149W	MCM1_1	NDD1	
YPR156C	FKH1,FKH2,MCM1_1,NDD1		FKH1,FKH2_2,MBP1_3,MCM1_2,NDD1_2,SWI4_3,SWI6_3
YPR174C		MBP1,SWI6_1	
YPR175W	MBP1_2,STB1,SWI4_2	MBP1,SWI6_1	
YPR202W	MBP1_1,MCM1_3,SWI4_1,SWI5,SWI6_1		MCM1_1
YPR203W	MBP1_1,MCM1_3,SWI4_1,SWI6_1	MCM1_3,SWI6_2	
YPR204W	MBP1_1,MCM1_3,SWI6_1	MCM1_3,SWI6_2	

Table 14. Target list in dilution.

YAL012W	MET28,MET32
YAL054C	ADR1
YAR071W	PHO4
YBL015W	ADR1
YBR043C	GLN3
YBR139W	GLN3
YBR208C	GLN3
YBR213W	CBF1
YBR230C	ADR1
YBR293W	CBF1,MET31,MET32
YBR298C	ADR1
YCR010C	ADR1
YCR030C	DAL80
YCR094W	DAL80
YDL059C	MET28,MET32
YDL170W	GLN3
YDL171C	GLN3
YDL185W	GLN3
YDL208W	DAL80
YDL210W	GLN3
YDL231C	GLN3
YDL238C	GAT1,GLN3
YDR023W	DAL80
YDR090C	GLN3
YDR178W	ADR1
YDR242W	GAT1,GLN3
YDR253C	CBF1,MET28,MET32
YDR254W	MET28
YDR256C	ADR1
YDR399W	DAL80
YDR481C	PHO4

Table 14. Target list in dilution. (continued)

YDR502C	CBF1
YDR528W	DAL80
YEL064C	GLN3
YEL072W	MET28
YER015W	ADR1
YER037W	PHO4
YER070W	DAL80
YER091C	MET28,MET32
YER092W	MET28,MET32
YFL056C	CBF1,MET28
YFR017C	ADR1
YFR030W	MET28,MET31,MET32,MET4
YGL065C	DAL80
YGL127C	CBF1,MET28,MET31
YGL184C	MET31,MET32
YGL202W	GAT1
YGL205W	ADR1
YGR154C	MET28
YGR155W	CBF1,MET28
YGR221C	DAL80
YHL016C	GLN3
YHL032C	ADR1
YHL036W	CBF1,MET28,MET32
YHR018C	GLN3
YHR028C	GLN3
YHR037W	DAL80,GLN3
YHR039C	DAL80
YHR112C	CBF1
YHR136C	PHO4
YHR140W	GLN3
YHR176W	MET28
YHR202W	GLN3
YHR215W	PHO4
YIL071C	GAT1
YIL074C	CBF1,MET28
YIL146C	GAT1
YIL155C	ADR1
YIL160C	ADR1
YIL167W	GLN3
YIL168W	GLN3
YIR017C	CBF1,MET28
YIR018W	MET28

Table 14. Target list in dilution. (continued)

YIR027C	DAL80,GAT1,GLN3
YIR028W	GAT1,GLN3
YIR029W	GAT1,GLN3
YIR031C	DAL80,GAT1,GLN3
YIR032C	GAT1,GLN3
YJL010C	DAL80
YJL012C	PHO4
YJL060W	CBF1,MET28
YJL101C	MET28
YJL110C	DAL80
YJL117W	PHO4
YJL122W	DAL80
YJL172W	GAT1,GLN3
YJR010W	MET28,MET31,MET32
YJR127C	GLN3
YJR137C	CBF1,MET28,MET4
YJR138W	GLN3
YJR139C	MET28,MET4
YJR152W	GLN3
YKL103C	GAT1,GLN3
YKL148C	ADR1
YKR034W	DAL80,GAT1,GLN3
YKR068C	MET28
YKR069W	CBF1,MET28,MET31,MET4
YLL028W	DAL80
YLR063W	DAL80
YLR068W	DAL80
YLR092W	MET28,MET32
YLR155C	GLN3
YLR157C	GLN3
YLR158C	GLN3
YLR160C	GLN3
YLR164W	GAT1,GLN3
YLR220W	GLN3
YLR257W	GLN3
YLR276C	DAL80
YLR303W	MET28
YLR364W	MET32
YLR409C	DAL80
YLR436C	GAT1,GLN3
YML018C	MET28
YML089C	ADR1

Table 14. Target list in dilution. (continued)

YML106W	GLN3
YML123C	PHO4
YMR009W	CBF1
YMR081C	ADR1
YMR088C	GLN3
YMR170C	GLN3
YMR280C	ADR1
YMR301C	MET31,MET4
YNL101W	GLN3
YNL142W	GAT1
YNL191W	CBF1
YNL221C	MET28
YNL256W	MET28
YNL257C	MET28
YNL277W	MET28
YNR028W	DAL80
YOL007C	DAL80
YOL019W	GAT1,GLN3
YOL064C	MET31
YOL108C	GLN3
YOL128C	GAT1,GLN3
YOR003W	GLN3
YOR180C	ADR1
YOR270C	DAL80
YOR317W	GLN3
YOR341W	DAL80
YOR374W	ADR1
YPL018W	PHO4
YPL019C	PHO4
YPL024W	GAT1
YPL054W	GLN3
YPL134C	ADR1
YPL201C	ADR1
YPL227C	DAL80
YPL250C	MET28,MET32
YPL262W	ADR1
YPL267W	DAL80
YPR030W	ADR1
YPR035W	GLN3
YPR113W	DAL80
YPR167C	CBF1,MET4

REFERENCES

[1] Hughes TR, Marton MJ, Jones AR, Roberts CJ, Stoughton R, Armour CD, Bennett HA, Coffey E, Dai H, He YD, Kidd MJ, King AM, Meyer MR, Slade D, Lum PY, Stepaniants SB, Shoemaker DD, Gachotte D, Chakraburtty K, Simon J, Bard M, Friend SH . Functional discovery via a compendium of expression profiles. Cell 2000; 102: 109-26.

[2] Spellman PT, Sherlock G, Zhang MQ, Iyer VR, Anders K, Eisen MB, Brown PO, Botstein D, Futcher B . Comprehensive identification of cell cycle-regulated genes of the yeast Saccharomyces cerevisiae by microarray hybridization. Mol Biol Cell 1998; 9: 3273-97.

[3] Chu S, DeRisi J, Eisen M, Mulholland J, Botstein D, Brown PO, Herskowitz I . The transcriptional program of sporulation in budding yeast. Science (80-) 1998; 282: 699-705.

[4] Roy Choudhury D, Small C, Wang Y, Mueller PR, Rebel VI, Griswold MD, McCarrey JR . Microarray-Based Analysis of Cell-Cycle Gene ExpressionDuring Spermatogenesis in the Mouse. Biol Reprod 2010; : .

[5] Brauer MJ, Huttenhower C, Airoldi EM, Rosenstein R, Matese JC, Gresham D, Boer VM, Troyanskaya OG, Botstein D . Coordination of growth rate, cell cycle, stress response, and metabolic activity in yeast. Mol Biol Cell 2008; 19: 352-67.

[6] Gasch AP, Spellman PT, Kao CM, Carmel-Harel O, Eisen MB, Storz G, Botstein D, Brown PO . Genomic expression programs in the response of yeast cells to environmental changes. Mol Biol Cell 2000; 11: 4241-57.

[7] Ogawa N, DeRisi J, Brown PO . New components of a system for phosphate accumulation and polyphosphate metabolism in Saccharomyces cerevisiae revealed by genomic expression analysis. Mol Biol Cell 2000; 11: 4309-21.

[8] Eisen MB, Spellman PT, Brown PO, Botstein D . Cluster analysis and display of genome-wide expression patterns. Proc Natl Acad Sci U S A 1998; 95: 14863-8.

[9] Wu LF, Hughes TR, Davierwala AP, Robinson MD, Stoughton R, Altschuler SJ . Large-scale prediction of Saccharomyces cerevisiae gene function using overlapping transcriptional clusters. Nat Genet 2002; 31: 255-65.

[10] Wang W, Cherry JM, Nochomovitz Y, Jolly E, Botstein D, Li H . Inference of combinatorial regulation in yeast transcriptional networks: a case study of sporulation. Proc Natl Acad Sci U S A 2005; 102: 1998-2003.

[11] Yu RX, Liu J, True N, Wang W . Identification of direct target genes using joint sequence and expression likelihood with application to DAF-16. PLoS ONE 2008; 3: e1821.

[12] Alter O, Brown PO, Botstein D . Singular value decomposition for genome-wide expression data processing and modeling. Proc Natl Acad Sci U S A 2000; 97: 10101-6.

[13] Lee S, Batzoglou S . Application of independent component analysis to microarrays. Genome Biol 2003; 4: R76.

[14] Kim PM, Tidor B . Subsystem identification through dimensionality reduction of large-scale gene expression data. Genome Res 2003; 13: 1706-18.

[15] Bailey TL, Williams N, Misleh C, Li WW . MEME: discovering and analyzing DNA and protein sequence motifs. Nucleic Acids Res 2006; 34: W369-73.

[16] MacIsaac KD, Wang T, Gordon DB, Gifford DK, Stormo GD, Fraenkel E . An improved map of conserved regulatory sites for Saccharomyces cerevisiae. BMC Bioinformatics 2006; 7: 113.

[17] Ren B, Robert F, Wyrick JJ, Aparicio O, Jennings EG, Simon I, Zeitlinger J, Schreiber J, Hannett N, Kanin E, Volkert TL, Wilson CJ, Bell SP, Young RA . Genome-wide location and function of DNA binding proteins. Science (80-) 2000; 290: 2306-9.

[18] Iyer VR, Horak CE, Scafe CS, Botstein D, Snyder M, Brown PO . Genomic binding sites of the yeast cell-cycle transcription factors SBF and MBF. Nature 2001; 409: 533-8.

[19] Metzker ML . Sequencing technologies - the next generation. Nat Rev Genet 2010; 11: 31-46.

[20] Park PJ . ChIP-seq: advantages and challenges of a maturing technology. Nat Rev Genet 2009; 10: 669-80.

[21] Hon G, Ren B, Wang W . ChromaSig: a probabilistic approach to finding common chromatin signatures in the human genome. PLoS Comput Biol 2008; 4: e1000201.

[22] Won K, Ren B, Wang W . Genome-wide prediction of transcription factor binding sites using an integrated model. Genome Biol 2010; 11: R7.

[23] Tucker CL, Gera JF, Uetz P . Towards an understanding of complex protein networks. Trends Cell Biol 2001; 11: 102-6.

[24] Zhu H, Snyder M . Protein arrays and microarrays. Curr Opin Chem Biol 2001; 5: 40-5.

[25] Boyle EI, Weng S, Gollub J, Jin H, Botstein D, Cherry JM, Sherlock G . GO::TermFinder--open source software for accessing Gene Ontology information and finding significantly enriched Gene Ontology terms associated with a list of genes. Bioinformatics 2004; 20: 3710-5.

[26] Song Y, Chen S . Text mining biomedical literature for constructing gene regulatory networks. Interdiscip Sci 2009; 1: 179-86.

[27] Brunet J, Tamayo P, Golub TR, Mesirov JP . Metagenes and molecular pattern discovery using matrix factorization. Proc Natl Acad Sci U S A 2004; 101: 4164-9.

[28] Bansal M, Della Gatta G, di Bernardo D . Inference of gene regulatory networks and compound mode of action from time course gene expression profiles. Bioinformatics 2006; 22: 815-22.

[29] Sakamoto E, Iba H . Inferring a System of Differential Equations for a Gene Regulatory Network by

using Genetic Programming. IEEE Press 2001; 0: pp. 720–726.

[30] Friedman N, Linial M, Nachman I, Pe'er D . Using Bayesian networks to analyze expression data. J Comput Biol 2000; 7: 601-20.

[31] Hartemink AJ, Gifford DK, Jaakkola TS, Young RA . Using graphical models and genomic expression data to statistically validate models of genetic regulatory networks. Pac Symp Biocomput 2001; : 422-33.

[32] Perrin B, Ralaivola L, Mazurie A, Bottani S, Mallet J, d'Alché-Buc F . Gene networks inference using dynamic Bayesian networks. Bioinformatics 2003; 19 Suppl 2: ii138-48.

[33] Dojer N, Gambin A, Mizera A, Wilczyński B, Tiuryn J . Applying dynamic Bayesian networks to perturbed gene expression data. BMC Bioinformatics 2006; 7: 249.

[34] Shen L, Liu J, Wang W . GBNet: deciphering regulatory rules in the co-regulated genes using a Gibbs sampler enhanced Bayesian network approach. BMC Bioinformatics 2008; 9: 395.

[35] Kauffman SA . Metabolic stability and epigenesis in randomly constructed genetic nets. J Theor Biol 1969; 22: 437-67.

[36] Shmulevich I, YliHarja O, Astola J . Inference of genetic regulatory networks under the best-fit extension paradigm. In Proceedings of the IEEE 2001; June 3-6: .

[37] Shmulevich I, Dougherty ER, Kim S, Zhang W . Probabilistic Boolean Networks: a rule-based uncertainty model for gene regulatory networks. Bioinformatics 2002; 18: 261-74.

[38] Shmulevich I, Dougherty ER, Zhang W . Gene perturbation and intervention in probabilistic Boolean networks. Bioinformatics 2002; 18: 1319-31.

[39] Hecker M, Lambeck S, Toepfer S, van Someren E, Guthke R . Gene regulatory network inference: data integration in dynamic models-a review. Biosystems 2009; 96: 86-103.

[40] Yeang C, Mak HC, McCuine S, Workman C, Jaakkola T, Ideker T . Validation and refinement of gene-regulatory pathways on a network of physical interactions. Genome Biol 2005; 6: R62.

[41] Chaturvedi I, Sakharkar MK, Rajapakse JC . Validation of Gene Regulatory Networks from Protein-Protein Interaction Data: Application to Cell-Cycle Regulation

. PRIB 2007; LNBI 4774: pp.300-310.

[42] Batt G, Ropers D, de Jong H, Geiselmann J, Mateescu R, Page M, Schneider D . Validation of qualitative models of genetic regulatory networks by model checking: analysis of the nutritional stress response in Escherichia coli. Bioinformatics 2005; 21 Suppl 1: i19-28.

[43] Wang W, Cherry JM, Botstein D, Li H . A systematic approach to reconstructing transcription networks in Saccharomycescerevisiae. Proc Natl Acad Sci U S A 2002; 99: 16893-8.

[44] Das D, Banerjee N, Zhang MQ . Interacting models of cooperative gene regulation. Proc Natl Acad Sci U S A 2004; 101: 16234-9.

[45] Das D, Nahlé Z, Zhang MQ . Adaptively inferring human transcriptional subnetworks. Mol Syst Biol 2006; 2: 2006.0029.

[46] Beyer A, Workman C, Hollunder J, Radke D, Möller U, Wilhelm T, Ideker T . Integrated assessment and prediction of transcription factor binding. PLoS Comput Biol 2006; 2: e70.

[47] Basso K, Margolin AA, Stolovitzky G, Klein U, Dalla-Favera R, Califano A . Reverse engineering of regulatory networks in human B cells. Nat Genet 2005; 37: 382-90.

[48] Segal E, Taskar B, Gasch A, Friedman N, Koller D . Rich probabilistic models for gene expression. Bioinformatics 2001; 17 Suppl 1: S243-52.

[49] Segal E, Yelensky R, Koller D . Genome-wide discovery of transcriptional modules from DNA sequence and gene expression. Bioinformatics 2003; 19 Suppl 1: i273-82.

[50] Harbison CT, Gordon DB, Lee TI, Rinaldi NJ, Macisaac KD, Danford TW, Hannett NM, Tagne J, Reynolds DB, Yoo J, Jennings EG, Zeitlinger J, Pokholok DK, Kellis M, Rolfe PA, Takusagawa KT, Lander ES, Gifford DK, Fraenkel E, Young RA . Transcriptional regulatory code of a eukaryotic genome. Nature 2004; 431: 99-104.

[51] Bussemaker HJ, Li H, Siggia ED . Regulatory element detection using correlation with expression. Nat Genet 2001; 27: 167-71.

[52] Kenyon C, Chang J, Gensch E, Rudner A, Tabtiang R . A C. elegans mutant that lives twice as long as wild type. Nature 1993; 366: 461-4.

[53] Lee SS, Kennedy S, Tolonen AC, Ruvkun G . DAF-16 target genes that control C. elegans life-span and metabolism. Science (80-) 2003; 300: 644-7.

[54] Oh SW, Mukhopadhyay A, Dixit BL, Raha T, Green MR, Tissenbaum HA . Identification of direct DAF-16 targets controlling longevity, metabolism and diapause by chromatin immunoprecipitation. Nat Genet 2006; 38: 251-7.

[55] Murphy CT, McCarroll SA, Bargmann CI, Fraser A, Kamath RS, Ahringer J, Li H, Kenyon C . Genes that act downstream of DAF-16 to influence the lifespan of Caenorhabditis elegans. Nature 2003; 424: 277-83.

[56] Tusher VG, Tibshirani R, Chu G . Significance analysis of microarrays applied to the ionizing radiation response. Proc Natl Acad Sci U S A 2001; 98: 5116-21.

[57] Schwarz EM, Antoshechkin I, Bastiani C, Bieri T, Blasiar D, Canaran P, Chan J, Chen N, Chen WJ, Davis P, Fiedler TJ, Girard L, Harris TW, Kenny EE, Kishore R, Lawson D, Lee R, Müller H, Nakamura C, Ozersky P, Petcherski A, Rogers A, Spooner W, Tuli MA, Van Auken K, Wang D, Durbin R, Spieth J, Stein LD, Sternberg PW . WormBase: better software, richer content. Nucleic Acids Res 2006; 34: D475-8.

[58] Bussemaker HJ, Li H, Siggia ED . Building a dictionary for genomes: identification of presumptive regulatory sites by statistical analysis. Proc Natl Acad Sci U S A 2000; 97: 10096-100.

[59] An JH, Blackwell TK . SKN-1 links C. elegans mesendodermal specification to a conserved oxidative stress response. Genes Dev 2003; 17: 1882-93.

[60] Honda Y, Honda S . Oxidative stress and life span determination in the nematode Caenorhabditis elegans. Ann N Y Acad Sci 2002; 959: 466-74.

[61] Wingender E . Compilation of transcription regulating proteins. Nucleic Acids Res 1988; 16: 1879-902.

[62] Sandelin A, Alkema W, Engström P, Wasserman WW, Lenhard B . JASPAR: an open-access database for eukaryotic transcription factor binding profiles. Nucleic Acids Res 2004; 32: D91-4.

[63] Lee TI, Rinaldi NJ, Robert F, Odom DT, Bar-Joseph Z, Gerber GK, Hannett NM, Harbison CT, Thompson CM, Simon I, Zeitlinger J, Jennings EG, Murray HL, Gordon DB, Ren B, Wyrick JJ, Tagne J, Volkert TL, Fraenkel E, Gifford DK, Young RA . Transcriptional regulatory networks in Saccharomyces cerevisiae. Science (80-) 2002; 298: 799-804.

[64] Johnson DS, Mortazavi A, Myers RM, Wold B . Genome-wide mapping of in vivo protein-DNA interactions. Science (80-) 2007; 316: 1497-502.

[65] Holstege FC, Young RA . Transcriptional regulation: contending with complexity. Proc Natl Acad Sci U S A 1999; 96: 2-4.

[66] Tachibana C, Yoo JY, Tagne J, Kacherovsky N, Lee TI, Young ET . Combined global localization analysis and transcriptome data identify genes that are directly coregulated by Adr1 and Cat8. Mol Cell Biol 2005; 25: 2138-46.

[67] Workman CT, Mak HC, McCuine S, Tagne J, Agarwal M, Ozier O, Begley TJ, Samson LD, Ideker T . A systems approach to mapping DNA damage response pathways. Science (80-) 2006; 312: 1054-9.

[68] Zhu J, Zhang MQ . SCPD: a promoter database of the yeast Saccharomyces cerevisiae. Bioinformatics 1999; 15: 607-11.

[69] Costanzo MC, Crawford ME, Hirschman JE, Kranz JE, Olsen P, Robertson LS, Skrzypek MS, Braun BR, Hopkins KL, Kondu P, Lengieza C, Lew-Smith JE, Tillberg M, Garrels JI . YPD, PombePD and WormPD: model organism volumes of the BioKnowledge library, an integrated resource for protein information. Nucleic Acids Res 2001; 29: 75-9.

[70] Matys V, Fricke E, Geffers R, Gössling E, Haubrock M, Hehl R, Hornischer K, Karas D, Kel AE, Kel-Margoulis OV, Kloos D, Land S, Lewicki-Potapov B, Michael H, Münch R, Reuter I, Rotert S, Saxel H, Scheer M, Thiele S, Wingender E . TRANSFAC: transcriptional regulation, from patterns to profiles. Nucleic Acids Res 2003; 31: 374-8.

[71] Roven C, Bussemaker HJ . REDUCE: An online tool for inferring cis-regulatory elements and transcriptional module activities from microarray data. Nucleic Acids Res 2003; 31: 3487-90.

[72] Liu XS, Brutlag DL, Liu JS . An algorithm for finding protein-DNA binding sites with applications to chromatin-immunoprecipitation microarray experiments. Nat Biotechnol 2002; 20: 835-9.

[73] Li H, Wang W . Dissecting the transcription networks of a cell using computational genomics. Curr Opin Genet Dev 2003; 13: 611-6.

[74] Chua G, Robinson MD, Morris Q, Hughes TR . Transcriptional networks: reverse-engineering gene regulation on a global scale. Curr Opin Microbiol 2004; 7: 638-46.

[75] Blais A, Dynlacht BD . Constructing transcriptional regulatory networks. Genes Dev 2005; 19: 1499-511.

[76] Siggia ED . Computational methods for transcriptional regulation. Curr Opin Genet Dev 2005; 15: 214-21.

[77] Liao JC, Boscolo R, Yang Y, Tran LM, Sabatti C, Roychowdhury VP . Network component analysis: reconstruction of regulatory signals in biological systems. Proc Natl Acad Sci U S A 2003; 100: 15522-7.

[78] Gao F, Foat BC, Bussemaker HJ . Defining transcriptional networks through integrative modeling of mRNA expression and transcription factor binding data. BMC Bioinformatics 2004; 5: 31.

[79] Conlon EM, Liu XS, Lieb JD, Liu JS . Integrating regulatory motif discovery and genome-wide expression analysis. Proc Natl Acad Sci U S A 2003; 100: 3339-44.

[80] Tseng GC, Wong WH . Tight clustering: a resampling-based approach for identifying stable and tight patterns in data. Biometrics 2005; 61: 10-6.

[81] Simon I, Barnett J, Hannett N, Harbison CT, Rinaldi NJ, Volkert TL, Wyrick JJ, Zeitlinger J, Gifford DK, Jaakkola TS, Young RA . Serial regulation of transcriptional regulators in the yeast cell cycle. Cell 2001; 106: 697-708.

[82] Yang Y, Suen J, Brynildsen MP, Galbraith SJ, Liao JC . Inferring yeast cell cycle regulators and interactions using transcription factor activities. BMC Genomics 2005; 6: 90.

[83] Ho Y, Costanzo M, Moore L, Kobayashi R, Andrews BJ . Regulation of transcription at the Saccharomyces cerevisiae start transition by Stb1, a Swi6-binding protein. Mol Cell Biol 1999; 19: 5267-78.

[84] Costanzo M, Schub O, Andrews B . G1 transcription factors are differentially regulated in Saccharomyces cerevisiae by the Swi6-binding protein Stb1. Mol Cell Biol 2003; 23: 5064-77.

[85] Mazurie A, Bottani S, Vergassola M . An evolutionary and functional assessment of regulatory network motifs. Genome Biol 2005; 6: R35.

[86] Coffman JA, Rai R, Cunningham T, Svetlov V, Cooper TG . Gat1p, a GATA family protein whose production is sensitive to nitrogen catabolite repression, participates in transcriptional activation of nitrogen-catabolic genes in Saccharomyces cerevisiae. Mol Cell Biol 1996; 16: 847-58.

[87] Beck T, Hall MN . The TOR signalling pathway controls nuclear localization of nutrient-regulated transcription factors. Nature 1999; 402: 689-92.

[88] Hofman-Bang J . Nitrogen catabolite repression in Saccharomyces cerevisiae. Mol Biotechnol 1999; 12: 35-73.

[89] Boczko EM, Cooper TG, Gedeon T, Mischaikow K, Murdock DG, Pratap S, Wells KS . Structure theorems and the dynamics of nitrogen catabolite repression in yeast. Proc Natl Acad Sci U S A 2005; 102: 5647-52.

[90] Kuras L, Barbey R, Thomas D . Assembly of a bZIP-bHLH transcription activation complex: formation of the yeast Cbf1-Met4-Met28 complex is regulated through Met28 stimulation of Cbf1 DNA binding. EMBO J 1997; 16: 2441-51.

[91] Blaiseau PL, Thomas D . Multiple transcriptional activation complexes tether the yeast activator Met4 to DNA. EMBO J 1998; 17: 6327-36.

[92] Menant A, Baudouin-Cornu P, Peyraud C, Tyers M, Thomas D . Determinants of the ubiquitin-mediated degradation of the Met4 transcription factor. J Biol Chem 2006; 281: 11744-54.

[93] Lee TA, Jorgensen P, Bognar AL, Peyraud C, Thomas D, Tyers M . Dissection of combinatorial control by the Met4 transcriptional complex. Mol Biol Cell 2010; 21: 456-69.

[94] Thomas D, Surdin-Kerjan Y . Metabolism of sulfur amino acids in Saccharomyces cerevisiae. Microbiol Mol Biol Rev 1997; 61: 503-32.

[95] Hochstrasser M . Ubiquitin signalling: what's in a chain?. Nat Cell Biol 2004; 6: 571-2.

[96] Chandrasekaran S, Skowyra D . The emerging regulatory potential of SCFMet30 -mediated polyubiquitination and proteolysis of the Met4 transcriptional activator. Cell Div 2008; 3: 11.

[97] Young ET, Dombek KM, Tachibana C, Ideker T . Multiple pathways are co-regulated by the protein kinase Snf1 and the transcription factors Adr1 and Cat8. J Biol Chem 2003; 278: 26146-58.

[98] Young ET, Kacherovsky N, Van Riper K . Snf1 protein kinase regulates Adr1 binding to chromatin but not transcription activation. J Biol Chem 2002; 277: 38095-103.

[99] Schüller H . Transcriptional control of nonfermentative metabolism in the yeast Saccharomyces cerevisiae. Curr Genet 2003; 43: 139-60.

[100] Wang T, Stormo GD . Combining phylogenetic data with co-regulated genes to identify regulatory motifs. Bioinformatics 2003; 19: 2369-80.

[101] Lawrence CE, Altschul SF, Boguski MS, Liu JS, Neuwald AF, Wootton JC . Detecting subtle sequence signals: a Gibbs sampling strategy for multiple alignment. Science (80-) 1993; 262: 208-14.

[102] Blanchette M, Tompa M . Discovery of regulatory elements by a computational method for phylogenetic footprinting. Genome Res 2002; 12: 739-48.

[103] Akutsu T, Miyano S, Kuhara S . Identification of genetic networks from a small number of gene expression patterns under the Boolean network model. Pac Symp Biocomput 1999; : 17-28.

[104] Imoto S, Goto T, Miyano S . Estimation of genetic networks and functional structures between genes by using Bayesian networks and nonparametric regression. Pac Symp Biocomput 2002; : 175-86.

[105] Beal MJ, Falciani F, Ghahramani Z, Rangel C, Wild DL . A Bayesian approach to reconstructing genetic regulatory networks with hidden factors. Bioinformatics 2005; 21: 349-56.

[106] Johnson WE, Li W, Meyer CA, Gottardo R, Carroll JS, Brown M, Liu XS . Model-based analysis of tiling-arrays for ChIP-chip. Proc Natl Acad Sci U S A 2006; 103: 12457-62.

[107] Lee W, Tzou W . Computational methods for discovering gene networks from expression data. Brief Bioinform 2009; 10: 408-23.

[108] Narayanan M, Vetta A, Schadt EE, Zhu J . Simultaneous clustering of multiple gene expression and physical interaction datasets. PLoS Comput Biol 2010; 6: e1000742.

[109] Seok J, Kaushal A, Davis RW, Xiao W . Knowledge-based analysis of microarrays for the discovery of transcriptional regulation relationships. BMC Bioinformatics 2010; 11 Suppl 1: S8.

[110] Devarajan K . Nonnegative matrix factorization: an analytical and interpretive tool in computational biology. PLoS Comput Biol 2008; 4: e1000029.

[111] de Jong H . Modeling and simulation of genetic regulatory systems: a literature review. J Comput Biol 2002; 9: 67-103.

[112] Li P, Zhang C, Perkins EJ, Gong P, Deng Y . Comparison of probabilistic Boolean network and dynamic Bayesian network approaches for inferring gene regulatory networks. BMC Bioinformatics 2007; 8 Suppl 7: S13.

[113] He F, Balling R, Zeng A . Reverse engineering and verification of gene networks: principles, assumptions, and limitations of present methods and future perspectives. J Biotechnol 2009; 144: 190-203.