Genomic Methods for Studying the Post-Translational Regulation of Transcription Factors

Logan J. Everett
University of Pennsylvania, loganje@mail.med.upenn.edu

Follow this and additional works at: http://repository.upenn.edu/edissertations

Part of the Bioinformatics Commons, Biotechnology Commons, Computational Biology Commons, Genetics Commons, Genomics Commons, Molecular Biology Commons, Molecular Genetics Commons, and the Systems Biology Commons

Recommended Citation
http://repository.upenn.edu/edissertations/456

This paper is posted at ScholarlyCommons. http://repository.upenn.edu/edissertations/456
For more information, please contact libraryrepository@pobox.upenn.edu.
Genomic Methods for Studying the Post-Translational Regulation of Transcription Factors

Abstract
The spatiotemporal coordination of gene expression is a fundamental process in cellular biology. Gene expression is regulated, in large part, by sequence-specific transcription factors that bind to DNA regions in the proximity of each target gene. Transcription factor activity and specificity are, in turn, regulated post-translationally by protein-modifying enzymes. High-throughput methods exist to probe specific steps of this process, such as protein-protein and protein-DNA interactions, but few computational tools exist to integrate this information in a principled, model-oriented manner. In this work, I develop several computational tools for studying the functional implications of transcription factor modification. I establish the first publicly accessible database for known and predicted regulatory circuits that encompass modifying enzymes, transcription factors, and transcriptional targets. I also develop a model-based method for integrating heterogeneous genomic and proteomic data for the inference of modification-dependent transcriptional regulatory networks. The model-based method is thoroughly validated as a reliable and accurate computational genomic tool. Additionally, I propose and demonstrate fundamental improvements to computational proteomic methods for identifying modified protein forms. In summary, this work contributes critical methodological advances to the field of regulatory network inference.

Degree Type
Dissertation

Degree Name
Doctor of Philosophy (PhD)

Graduate Group
Genomics & Computational Biology

First Advisor
Sridhar Hannenhalli

Second Advisor
Stephen R. Master

Keywords
Network Inference, Proteomics, Regulatory Genomics, Computational Biology, Bioinformatics, Molecular Biology

Subject Categories
Bioinformatics | Biotechnology | Computational Biology | Genetics | Genomics | Molecular Biology | Molecular Genetics | Systems Biology

This dissertation is available at ScholarlyCommons: http://repository.upenn.edu/edissertations/456
GENOMIC METHODS FOR STUDYING
THE POST-TRANSLATIONAL REGULATION
OF TRANSCRIPTION FACTORS

Logan J. Everett

A DISSERTATION
in
Genomics and Computational Biology

Presented to the Faculties of the University of Pennsylvania
in Partial Fulfillment of the Requirements for the
Degree of Doctor of Philosophy

2010

Co-Supervisor of Dissertation

Sridhar Hannenhalli, Ph.D.
Associate Professor, Genetics

Co-Supervisor of Dissertation

Stephen R. Master, M.D., Ph.D.
Assistant Professor, Pathology
and Laboratory Medicine

Graduate Group Chairperson

Maja Busan, Ph.D.
Professor, Genetics

DISSERTATION COMMITTEE
Klaus H. Kaestner, Ph.D.,
Thomas and Evelyn Suor Butterworth Professor, Genetics
Thomas R. Kadesch, Ph.D., Professor, Genetics
Shane T. Jensen, Ph.D., Associate Professor, Statistics
Gregory E. Gonye, Ph.D., Assistant Professor,
Pathology, Anatomy, and Cell Biology (Thomas Jefferson University)
GENOMIC METHODS FOR STUDYING THE POST-TRANSLATIONAL REGULATION OF TRANSCRIPTION FACTORS

COPYRIGHT

2010

Logan James Everett
Acknowledgements

I would first and foremost like to thank my advisors, Dr. Sridhar Hannenhalli and Dr. Stephen Master, under whom I was given both the guidance and freedom necessary to grow as a researcher. The work presented here is a mere fraction of the training I’ve received from them. I am grateful to my thesis committee for the generous amount of time and brain power they gave to these projects. Many thanks to Dr. Klaus Kaestner, Dr. Tom Kadesch, Dr. Greg Gonye, and Dr. Shane Jensen for their constant encouragement. I would also like to thank the members of the Hannenhalli lab, especially Larry Singh, Matt Hansen, and Mugdha Khaladkar for making room 1412 a fun and enlightening place to be. In the Master lab, Charlene Bierl and Lin-Sheng Li have taught me countless wisdoms about science at the bench.

With respect to specific projects presented in this work, I would also like to thank Geetu Tuteja for providing support and additional data for the GLITR application, Gary Chen for correspondences regarding the COGRIM application, and Antony Vo for help in developing the PTM-Switchboard web interface. Additional thanks to Shane Jensen for major contributions to the MoNsTEr project, to Greg Gonye for enthusiastic and extensive feedback on manuscripts, and to Matt Hansen for making the Mimosa algorithm a reality.

Many thanks to my friends and family for their constant love and support. I am infinitely indebted to my parents for starting my education early, putting me through college, and always fostering my interests. And finally, to Ali, it would take hundreds more pages to explain all you’ve done for me. I owe you big time when your turn comes around.
ABSTRACT

GENOMIC METHODS FOR STUDYING THE POST-TRANSLATIONAL REGULATION OF TRANSCRIPTION FACTORS

Logan J. Everett
Sridhar Hannenhalli, Ph.D. Stephen R. Master, M.D., Ph.D.

The spatiotemporal coordination of gene expression is a fundamental process in cellular biology. Gene expression is regulated, in large part, by sequence-specific transcription factors that bind to DNA regions in the proximity of each target gene. Transcription factor activity and specificity are, in turn, regulated post-translationally by protein-modifying enzymes. High-throughput methods exist to probe specific steps of this process, such as protein-protein and protein-DNA interactions, but few computational tools exist to integrate this information in a principled, model-oriented manner. In this work, I develop several computational tools for studying the functional implications of transcription factor modification. I establish the first publicly accessible database for known and predicted regulatory circuits that encompass modifying enzymes, transcription factors, and transcriptional targets. I also develop a model-based method for integrating heterogeneous genomic and proteomic data for the inference of modification-dependent transcriptional regulatory networks. The model-based method is thoroughly validated as a reliable and accurate computational genomic tool. Additionally, I propose and demonstrate fundamental improvements to computational proteomic methods for identifying modified protein forms. In summary, this work contributes critical methodological advances to the field of regulatory network inference.
Contents

1 Post-Translation Regulation of Transcription Factor Proteins 1

1.1 Introduction ................................................. 1

1.2 Gene Regulation by Transcription Factors .................. 2

   Inference of TF Binding Sites ................................. 4

   Inference of TF-Dependent Gene Expression .................. 6

   Construction of Integrative Network Models .................. 8

1.3 Signaling Pathways and Post-Translational Modifications .... 10

1.4 PTM-Dependent Transcriptional Regulation ................. 12

   Genome-Wide Inference of TF-Modifiers .................... 15

   Limitations of Existing Methods ............................. 19

1.5 Overview of Thesis Work ................................... 21

2 PTM-Switchboard: a Database of MFG Triplets 22

2.1 Introduction ................................................. 22

2.2 Database Overview ......................................... 24

2.3 Supporting Evidence ....................................... 27

2.4 Guidelines for Literature Curation .......................... 29

2.5 Case Study: The FUS3/STE12/FUS1 Triplet .................. 31

2.6 Discussion .................................................. 32
3 MoNsTER: An Integrative Model of Transcriptional Regulation via TF-Modifying Enzymes

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1 Introduction</td>
<td>35</td>
</tr>
<tr>
<td>3.2 The Integrative Network Model</td>
<td>36</td>
</tr>
<tr>
<td>Primary Model Equations</td>
<td>36</td>
</tr>
<tr>
<td>Model Estimation Method</td>
<td>41</td>
</tr>
<tr>
<td>3.3 Data Sources and Pre-Processing</td>
<td>45</td>
</tr>
<tr>
<td>Simulated Data</td>
<td>46</td>
</tr>
<tr>
<td>Yeast Network Input Data</td>
<td>48</td>
</tr>
<tr>
<td>3.4 Analysis of Simulated Data</td>
<td>52</td>
</tr>
<tr>
<td>Simulated Sources of Noise</td>
<td>53</td>
</tr>
<tr>
<td>3.5 Analysis of Yeast MSN2/4 Network</td>
<td>56</td>
</tr>
<tr>
<td>3.6 Discussion</td>
<td>62</td>
</tr>
<tr>
<td>Development of the Heuristic Model-Fitting Framework</td>
<td>62</td>
</tr>
<tr>
<td>Interpreting the TF-Modifier Influence Parameter</td>
<td>63</td>
</tr>
<tr>
<td>Additional Data Available for Yeast</td>
<td>64</td>
</tr>
</tbody>
</table>

4 Analysis of the STAT1-Mediated Network in Human B Cells

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.1 Introduction</td>
<td>66</td>
</tr>
<tr>
<td>4.2 Materials and Methods</td>
<td>67</td>
</tr>
<tr>
<td>Human B Cell Expression Compendium</td>
<td>67</td>
</tr>
<tr>
<td>STAT1-Target Gene Priors from ChIP-seq</td>
<td>69</td>
</tr>
<tr>
<td>STAT1-Target Gene Priors from PWMs</td>
<td>70</td>
</tr>
<tr>
<td>Modifier-STAT1 Priors from STRING</td>
<td>71</td>
</tr>
<tr>
<td>Application of MoNsTER to Human Data</td>
<td>72</td>
</tr>
<tr>
<td>Human B Cell Expression Permutation</td>
<td>73</td>
</tr>
</tbody>
</table>
Annotation Enrichment Analysis ........................................... 74

4.3 MoNsTEr Predicts Three Distinct Network Models ................. 75
4.4 Analysis of the Primary STAT1-Mediated Network Model .......... 80
4.5 Discussion ............................................................... 88

Biological Insights from the Primary Model ............................. 88
Influence of Input Data Sources on Model Parameters ................. 90
Alternate STAT1-Mediated Network Models ............................. 92

5 Reliable Identification of PTMs Using Mass Spectrometry 103

5.1 Introduction .............................................................. 103
Multi-Stage Search Strategies .............................................. 105
False Discovery Rate in Computational Proteomics ......................... 107
FDR Estimation in Multi-Stage Search Strategies ......................... 108

5.2 Materials and Methods .................................................. 109
Preparation of Control Solution ............................................ 109
LC-MS/MS Protocol ......................................................... 109
Preparation of Search Databases ........................................... 110
Spectrum File Conversion and Database Searches ......................... 111
FDR Analysis ................................................................. 112

5.3 Results ................................................................. 114
Control Sample Validation Bias in Multi-Stage Search Strategies .... 114
Decoy-Based FDR Bias in the Refinement Stage ......................... 118
Artificial Point Mutations for Control Sample Validation .......... 121
An Improved Strategy for Unbiased FDR Estimates ...................... 122
FDR Bias is Independent of Search Parameters ....................... 124

5.4 Discussion ............................................................... 129
List of Tables

1.1 PTM Types Relevant to TF Regulation ........................................ 13
2.1 MFG Triplet Schema for PTM-Switchboard ................................. 25
2.2 Modifiers and TFs Included in PTM-Switchboard .......................... 26
2.3 PTM Schema for PTM-Switchboard ........................................... 26
2.4 PTM-Switchboard Evidence Codes ............................................. 28
3.1 MoNsTEr Input Data Notation ................................................... 39
3.2 MoNsTEr Model Parameter Notation ........................................... 42
3.3 Model Inputs for MoNsTEr Application to Yeast ........................... 49
3.4 Kinases and Target Genes Included in Yeast Network Model .......... 50
3.5 Known Kinases Regulating MSN2/4 .......................................... 59
3.6 Significance of Yeast Application AUC Values ............................. 60
4.1 Model Inputs for MoNsTEr Application in Human B Cells ............ 68
4.2 Properties of STAT1 Network Models ....................................... 77
4.3 Known STAT1 Target Genes ..................................................... 78
4.4 Annotations Enriched for Target Genes (Primary Model) .............. 82
4.5 Modifiers Predicted by Primary Network Model ........................... 84
4.6 Annotations Enriched for Modifiers (Primary Model) ................... 85
4.7 Annotation Enriched for Target Genes (Network Model II) .......... 96
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.8 Annotation Enriched for Target Genes (Network Model III)</td>
<td>100</td>
</tr>
<tr>
<td>5.1 Replicate Data for Proteomics Analysis</td>
<td>110</td>
</tr>
<tr>
<td>5.2 FDR Table, Semi-Tryptic Analysis</td>
<td>120</td>
</tr>
<tr>
<td>5.3 FDR Table, Point Mutation Analysis</td>
<td>123</td>
</tr>
<tr>
<td>5.4 FDR Table, Semi-Tryptic Analysis with Re-Rev Correction</td>
<td>124</td>
</tr>
<tr>
<td>5.5 FDR Table, Point Mutation Analysis with Re-Rev Correction</td>
<td>125</td>
</tr>
</tbody>
</table>
List of Figures

1.1 CREB Regulation via PTMs ........................................ 14
1.2 Example MFG Triplet .............................................. 16

3.1 Conceptual Diagram of Network Model .............................. 37
3.2 Accuracy of MoNsTER Using Simulated Data ....................... 54
3.3 Hierarchical Clustering of Yeast Models ............................ 57
3.4 ROC Curves for Target Gene Selection in Yeast Network .......... 58

4.1 PWMs for STAT1-Gene priors ...................................... 71
4.2 Hierarchical Clustering of STAT1 Models .......................... 76
4.3 Distributions of STAT1 Network Properties ........................ 79
4.4 Network Edge Properties of Predicted STAT1 Target Genes ......... 81
4.5 STAT1 Modifiers Predicted by MoNsTER Primary Network Model .. 83
4.6 Primary STAT1 Network Model vs. Permutated Expression Controls .. 87
4.7 Modifier-Gene Functional Associations in Primary Model .......... 88
4.8 Input Data vs. TF-Modifier Interaction Parameters ................ 93

5.1 Overlap of Single-Stage and Multi-Stage Search Results .......... 116
5.2 Distributions of Semi-Tryptic Spectrum Match Properties ........... 117
5.3 Analysis Workflows to Analyze FDR Estimates .................... 119
5.4 Control vs. Decoy FDR by Expect, Semi-Tryptic Searches .......... 120
Chapter 1

Post-Translation Regulation of Transcription Factor Proteins

1.1 Introduction

Changes in gene expression underlie many fundamental biological processes, including development, metabolism, and response to environmental stress. Attempts to understand these processes in terms of individual genes have been limited due to the inherent complexity of cellular biology. Many genes interact, resulting in functions that are largely dependent on the expression levels of other genes. Thus, multi-genic expression programs must be coordinately regulated to achieve the intended cellular state, and disruption of these programs is a major component of human disease [12, 44].

The characterization of cellular behavior at the systems level—rather than at the level of individual genes—is now a viable approach for the study of such complexity, due to advances in high-throughput technology and the completion of genome sequences [44, 113]. A prerequisite for any quantitative study of a cellular system is
a model of the underlying system structure, i.e., the direct physical and functional interactions between sets of genes, typically represented as a network [113, 125]. In particular, the interactions between DNA-binding proteins and the gene transcript levels they regulate are of particular interest in the study of gene expression [12, 112, 199]. There is also a cellular sub-network of proteins and other molecules involved in relaying, processing, and responding to signals from the cellular environment, and many of these signals lead to coordinated changes in gene expression programs [12]. A major challenge in the field of network biology is the overlay of these two classes of networks, creating a comprehensive model of signaling and coordinated transcriptional response in the nucleus [112]. Network models of these regulatory interactions provide a foundation for understanding how cells coordinate multigenic responses, and hold the potential to decipher systems-level behavior in the cell [44, 125].

In this chapter, I review the basic biology of transcription and cell signaling, and highlight recent methodological advances in the field of network inference. In subsequent chapters, I present novel computational methods and related resources for unraveling the complex regulatory network operating at the interface between cell signaling and transcriptional regulation.

### 1.2 Gene Regulation by Transcription Factors

Transcription is the process of copying information from genomic DNA to RNA, and is a critical step in the expression of all protein-coding genes. The precise transcript level is tightly regulated for most genes in both space (e.g., different tissues) and time (e.g., developmental programs). In a single organism, the same genome gives rise to a variety of specialized cell types and states, as a result of differential gene expression
regulated at the level of transcription initiation [12, 44]. Many functionally related genes, including members of a pathway, biological process, or a protein complex, tend to have similar spatiotemporal expression patterns [186, 192]. In eukaryotes, transcription of protein-coding genes is performed by RNA Polymerase II (RNAPII), and the rate at which RNAPII is recruited and activated at each transcription start site (TSS) is a major determinant of gene expression levels [103, 115].

Transcription by RNAPII occurs in several regulated steps [60]. First, the TSS region (also called the core promoter) must be cleared of other proteins, such as histones, that would otherwise obscure RNAPII from binding to the DNA. RNAPII and additional proteins can then assemble at the TSS, forming the pre-initiation complex (PIC). The PIC proceeds with transcription for 5–6 bases before several protein components must be released, a step referred to as promoter escape, or transcription is aborted. After successful promoter escape, elongation of the nascent transcript can then proceed, but is often paused at an upstream region proximal to the TSS. Chemical changes to the RNAPII protein are typically required to escape pausing and transcribe the rest of the gene (productive elongation). Thus, genes can be silenced by blocking this core transcriptional machinery or, more commonly, genes can be strongly induced by guiding RNAPII to specific genomic loci and promoting the initiation and elongation steps of transcription [60, 158].

Sequence-Specific Transcription Factors (TFs) are proteins which bind to specific DNA sequences and influence the rate of transcription at particular target genes, thereby coordinating the expression of multiple genes to organize a coherent cellular state. Many genes are directly regulated via TF binding in the region of DNA immediately upstream from the TSS - called the proximal promoter [48, 78]. In higher eukaryotes, many TFs also bind distal regulatory regions, e.g., enhancers and silencers, although these regions are especially challenging to annotate and link
to specific target genes on a genomic scale [78, 103, 190]. TFs typically function in combinatorial programs, with many TFs cooperating to regulate the transcription of each target gene, and these programs can precisely regulate the quantity of mRNA transcribed. The strength of physical interactions between a TF, its DNA binding site, and its related protein partners (co-factors) at each regulatory region can create differential gene regulatory effects. These interactions, and the protein level of each TF, can be fine-tuned by the cell to produce distinct cellular states and appropriate responses to intrinsic and extrinsic cues, e.g., stress, nutrient availability, and hormonal signals [103].

The study of gene-specific transcriptional regulation is typically built on two basic components: (i) identification of specific TF binding sites in annotated genomic regions, e.g., proximal promoters, and (ii) observation of TF-dependent changes in gene expression [12]. For several decades, these components have been studied in targeted (low-throughput) experiments designed to elucidate the details of regulation at a few gene promoters at a time [112]. For the purposes of this work, I will focus on briefly reviewing more recent methods applicable to genome-wide inference of transcriptional regulatory networks.

Inference of TF Binding Sites

A critical mechanism by which many TFs accomplish specificity for their target genes is by preferential binding to specific DNA sequences, or motifs, present in the promoters of target genes [78]. Hannenhalli and Levy have shown that functionally related genes have similar sequence motifs in their proximal promoters, supporting the idea that these DNA regions contain binding sites for specific TFs responsible for coordinating multigenic biological processes [79].

For a review of targeted methods, see refs. [48, 103]
TF-DNA binding events can be detected in vivo using a method called Chromatin Immunoprecipitation (ChIP). In brief, the complex of proteins bound to DNA—chromatin—is cross-linked in place and complexes containing a specific TF are isolated through antibody binding. The cross-linking is then reversed and all proteins are digested, leaving only the DNA from the sites where the TF of interest was bound. This technique can be coupled to high-throughput methods for characterizing the bound DNA by either hybridization to oligonucleotide microarrays (ChIP-chip) [77] or high-throughput sequencing (ChIP-seq) [151]. For example, ChIP-chip has been used extensively to characterize TF binding sites in the yeast Saccharomyces cerevisiae [81, 123]. Recent advances in these methods have allowed for the in vivo detection of intergenic TF binding sites in higher eukaryotes, but these binding sites are particularly difficult to link to specific target genes due to the lack of annotations for distal regulatory elements, and are typically ignored by current genome-scale network inference methods.

Binding sites can also be predicted based on the sequence-specificity of the TF. The most common model for representing this sequence-specificity is the Positional Weight Matrix (PWM) [185], although other models exist [78]. PWMs provide a probabilistic representation of TF binding sites in terms of the relative preference for all four bases at each position [78]. These models can be trained from in vivo binding sites identified through ChIP-chip/seq [81], or by in vitro binding to libraries of sequences, such as the Systematic Evolution of Ligands by Exponential Enrichment (SELEX) method [193]. More recently, DNA microarrays have been used to quantitatively probe the in vitro specificity of TFs [142]. Databases of PWMs

2Note that additional genomic annotations of distal regulatory elements, such as those sought by the ENCODE project [49], can be readily incorporated into the methods discussed in this work whenever the annotated element can be linked to specific target genes. Proximal promoters simply represent the easiest form of this problem, because they can generally be assumed to influence the nearest gene.
based on these experimental methods have been compiled, such as TRANSFAC [209] and JASPAR [168]. Once trained, a PWM model can be used to scan proximal promoter regions for likely binding sites, thereby inferring potential target genes for each TF. However, TF binding sites are typically short and degenerate, and even well-trained models are prone to high false positive rates. Therefore, PWMs should be regarded with caution when applied to a large number of proximal promoter sequences, and are generally not suitable for scanning the entire genome for distal regulatory elements [78].

**Inference of TF-Dependent Gene Expression**

A complementary approach to inferring transcriptional networks is the inference of TF-dependent expression patterns, typically measured in high-throughput by expression microarrays [172]. The most direct approach is to measure the differential expression of genes when a particular TF is deleted or otherwise perturbed. For example, Hu et al. experimentally determined the set of genes differentially expressed in *S. cerevisiae* when each TF was systematically knocked down [90]. Differentially expressed genes in these conditions typically include both direct and indirect (downstream) targets of the TF of interest. Hu et al. dealt with this problem by analyzing the overall transcriptional network inferred from the knock-out compendium and removing connections that appeared to be indirect, i.e., dependencies that are better explained by one or more intermediate TFs. However, this requires a compendium of all TF knock-outs in order to reliably detect indirect dependencies [90]. Cells can also compensate for individual TF perturbations through redundant mechanisms or feedback loops, further complicating the analysis of these experiments [68, 90].

Another method for characterizing transcriptional regulation is the analysis of co-
expression across multiple experimental conditions. Typical metrics of coexpression include *correlation coefficients* \([44, 54, 186]\), and *mutual information* \([20, 40, 184]\). Many compilations of expression data now exist, such as the *Gene Expression Omnibus* (GEO) \([9]\), and measures of coexpression typically gain power as the number of expression samples increases \([44]\). Stuart *et al.* showed that coexpression patterns are strongly conserved through evolution, and often correspond to co-regulated modules of genes with related functions \([44, 186]\). Basic patterns of coexpression can arise from a number of different regulatory relationships. One gene in a coexpressed pair may regulate the other, or both genes may be co-regulated by some other TF, and in both cases this regulation can be direct or indirect.

Despite the limitations noted above, coexpression networks have been used in many applications to successfully infer the structure of the underlying transcriptional network. Magwene *et al.* developed the *First-Order Conditional Independence* (FOCI) algorithm to filter an expression correlation network to the edges that are independent of other edges, and likely to represent either direct regulatory interactions or directly co-regulated genes \([133]\). Similarly, Margolin *et al.* developed the *Algorithm for the Reconstruction of Gene Regulatory Networks* (ARACNE), which first infers network edges based on mutual information, then filters indirect edges by removing the weakest edge from each fully connected set of three genes in the network, based on the *data-processing inequality* theory \([134]\). By focusing on the network edges predicted by ARACNE surrounding TFs, Basso *et al.* accurately predicted direct regulatory interactions, and modeled a transcriptional network in B cells and related cancers \([10]\). Faith *et al.* developed the *Context Likelihood of Relatedness* (CLR) algorithm, which is also based on mutual information, but filters out indirect edges by applying local thresholds to each gene in the network. Faith *et al.* used the CLR algorithm to accurately model the transcriptional regulatory net-
work underlying a large compendium of \textit{Escherichia coli} expression data \cite{55}. Thus, while simple coexpression metrics may primarily identify modules of co-regulated or functionally related genes, more advanced algorithms can mine collections of expression data for evidence of direct regulatory interactions between TFs and target genes.

\textbf{Construction of Integrative Network Models}

Given genome-wide data related to both TF binding sites and gene expression data, a major goal in computational biology is the construction of a network model that accurately describes the “circuitry” of TFs and their target genes \cite{12, 44, 112, 125, 199}. The “gold standard” for identification of a direct TF-Gene network edge is evidence for both TF binding in an annotated gene regulatory region and a TF-dependent change in gene expression\textsuperscript{3} \cite{190}. Proximal promoter regions can be automatically annotated genome-wide based on known gene transcripts, and therefore are the primary focus of current methods. Thus, one common approach is to calculate the strict overlap of genes near TF binding sites and genes that are differentially expressed in a TF perturbation experiment, although this often yields surprisingly low overlap, even in lower eukaryotes where distal regulatory elements are less common \cite{68, 90}. It is possible that some TF binding sites do not regulate proximal genes, and as previously noted, TF perturbation experiments are prone to indirect and compensatory effects. However, a substantial part of this problem is likely to originate from noise in the individual experiments \cite{68, 90}, which is of even greater concern when attempting to combine noisier prediction methods, such as PWMs and coexpression

\textsuperscript{3}Promoter “bashing”, i.e., the direct manipulation of the regulatory sequences bound by particular factors to study their influence on putative target genes, could be considered the “platinum standard” for TF-Gene network edges, especially with respect to distal regulatory elements, but such methods currently are not feasible to do in high-throughput and therefore are outside the scope of genome-scale network inference.
networks. Therefore, more elegant solutions to integrating expression and binding data have been developed.

For example, Lee et al. combined their initial genome-wide survey of TF binding in *S. cerevisiae* with a compendium of expression profiles to identify sets of genes with highly correlated expression patterns and coordinated TF binding [123]. Tavazoie et al. developed a method in which genes are first clustered by coexpression, then a motif discovery algorithm is used to predict TF binding sites common to each gene cluster [189]. Bar-Joseph et al. developed the Gene Regulatory Modules (GRAM) algorithm to identify modules of genes with similar expression profiles and a common set of TFs bound to their promoters [5]. Bussemaker et al. modeled each gene expression level as an additive function of different TF binding site motif occurrences in the gene promoters [19], and Gao et al. later extended this method to use ChIP-chip data in place of sequence motifs [61]. These methods all have the same primary goal of identifying regulons—modules of functionally related and co-regulated genes—and the specific TFs that coordinate their expression.

Chen et al. developed a probabilistic framework—Clustering of Genes into Regulons using Integrated Modeling (COGRIM)—for studying regulons by integrating experimental TF binding data, predicted binding sites from PWMs, and expression data. In brief, COGRIM models the observed expression value of each target gene as a linear function of the expression of TFs regulating each gene. Individual TF-Gene edges in the network are given prior probabilities based on ChIP-chip TF binding data and/or PWM analysis of the gene promoters. The network model is then fit to an expression compendium, and used to infer posterior probabilities of TF-Gene edges given the expression and binding site data. COGRIM also has the option to include synergistic effects from pairs of TFs that functionally interact [22].

The methods reviewed above are focused on network interactions between TFs
and target genes, such that expression of each gene transcript is viewed as a function of TFs only. However, TF function is often dependent on co-factors, chromatin structure, and changes to the TF protein catalyzed by modifying enzymes. There is a broad need for methods that expand the scope of transcriptional regulatory networks to include these additional components [12, 14, 112]. In this work, I focus on the post-translational regulation of TFs, although many of the methods applicable to this problem can also be adapted to study regulation via co-factors, chromatin modification, and other sources of network dynamics. In the next section, I introduce the general concept of post-translational regulation and discuss novel developments in relevant experimental methods. I then review related methods for the more general problems of inferring condition-specific connectivity in TF networks and TF modulators, and discuss the potential for specifically applying these methods to study post-translational TF regulation.

1.3 Signaling Pathways and Post-Translational Modifications

Living cells constantly sense internal and external chemical states. The means by which a cell coordinates a response to a particular stimuli typically involves a signaling pathway—a series of molecular interactions that amplify a signal, and ultimately coordinate the appropriate response, including changes in gene transcription [1]. Covalent chemical changes to the amino acids in proteins—called Post-Translational Modifications (PTMs) [211]—are critical for the regulation of protein activities in signaling pathways [101]. A typical example of a signaling pathway includes a sequence of PTM reactions, with each modified form of a protein catalyzing the
modification of the next protein in the sequence, and ultimately activating a set of appropriate response proteins. Hundreds of distinct types of PTMs have been reported [36, 62], including phosphorylation, acetylation, methylation, and glycosylation. Many PTMs involved in signaling are reversible, and the proportion of modified and unmodified forms is determined by opposing enzymes that catalyze the forward and reverse PTM reactions, allowing for fine-scale regulation of signal strength [1, 101].

Emerging “proteomic” technologies are now making it possible to study PTMs on a near-global scale, resulting in a growing compendium of high-throughput data related to protein modifications. For example, protein microarrays [56] have been used to identify novel substrates for protein kinases [157] and one acetyltransferase [128], and can be applied to a much broader range of protein-modifying enzymes. Turk et al. have developed a peptide library screening method for determining the specificity of protein kinases [194] that is now broadening our understanding of phosphorylation substrate specificity [140]. Recent improvements to the sensitivity, accuracy, and speed of mass spectrometers have made it possible to assay a much wider range of PTM types [100, 210, 223]. These methods can be used to both monitor known PTMs and discover novel modification sites.

Advances in the study of protein modifying enzymes and PTMs are driving improvements to computational methods for PTM prediction. High-throughput data on the substrate specificity of a particular modifying enzyme now make it possible to build powerful models that go beyond basic motifs [13, 140, 217]. Furthermore, knowledge of which residues are modified in vivo makes it possible to apply predictive models to a smaller set of biological sequences, rather than blindly searching whole proteomes for possible PTM sites [129]. These collective breakthroughs in PTM research represent a major opportunity to expand regulatory network models.
beyond the scope of simple TF-Gene interactions.

### 1.4 PTM-Dependent Transcriptional Regulation

Many signaling pathways relay information to the nucleus and induce changes in the transcriptional state of the cell, often through the modification of TF proteins. PTMs can alter many properties of a TF, including the rate of nuclear trafficking, the rate of DNA binding at specific sequences, the interactions with co-factors, and the stability of the TF protein [11, 17, 110, 191]. Numerous types of chemical modifications of TF proteins have been documented, including phosphorylation [86], acetylation [59, 183], and methylation [120] (see Table 1.1). On one extreme, such regulation can be simple and binary—i.e., PTMs that serve as “on/off” switches for TF activity. More often, however, this regulation is highly complex [216], with multiple signal inputs integrated into tightly regulated transcript levels, and each TF-PTM potentially affecting each target gene in a manner dependent on the larger promoter context [8]. In this way, TFs can act as molecular switchboards that map multiple incoming signals to their appropriate multigenic expression responses [17, 110].

A canonical example of a regulatory TF-PTM is the phosphorylation of CREB at Ser133 (Figure 1.1). This TF-PTM has long been characterized as a key event in cyclic-AMP (cAMP) signaling. Protein kinase A (PKA) is activated by increased levels of cAMP in the cell, and phosphorylates CREB at the Ser133 residue [11]. The Ser133 phosphorylation allows CREB to interact with its co-activator, CBP, thereby recruiting the core transcriptional machinery and inducing the expression of target genes. In some contexts, Ser133 phosphorylation acts a simple on/off switch,

---

4RNAPII and histone proteins are also regulated via PTMs, although these tend to be part of general transcriptional mechanisms, rather than the coordination of specific gene modules [110].
Table 1.1: PTM types relevant to TF regulation, with known examples

<table>
<thead>
<tr>
<th>Modification</th>
<th>Amino Acid(s)</th>
<th>Example TF(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Phosphorylation</td>
<td>Ser, Thr, Tyr</td>
<td>CREB [11, 110, 138], Sp1, PU.1, SRF, c-Myc, ER, Pax-6, p53, NF-κB [215], STATS [39]</td>
</tr>
<tr>
<td>Acetylation</td>
<td>Lys</td>
<td>CREB [110], p53 [97], STAT1 [117], c-Myb, GATA-1, MyoD, E2F, EKLF, dTCF [183]</td>
</tr>
<tr>
<td>Methylation</td>
<td>Lys, Arg</td>
<td>p53 [91, 98], STAT1 [207]</td>
</tr>
<tr>
<td>Glycosylation</td>
<td>Ser, Thr</td>
<td>CREB [110], PDX-1, AP-1, Sp1 [219], Pax-6, c-Myc, p53 [203], NF-κB, NFAT [70], Elf-1 [191]</td>
</tr>
<tr>
<td>Ubiquitylation</td>
<td>Lys</td>
<td>Spt23, Met4, Gcn4, LIM, p53, HIF1α, SMAD2 [143], STAT1 [180]</td>
</tr>
<tr>
<td>SUMOylation</td>
<td>Lys</td>
<td>Lef1, Sp3, Myb, p53, AP-1 [143], STAT1 [163], TCF-4, Ikaros, AR, Elk-1, Sp3, SMAD4, GR, Gal4 [67]</td>
</tr>
<tr>
<td>Neddylation</td>
<td>Lys</td>
<td>p53, p73 [159]</td>
</tr>
<tr>
<td>Hydroxylation</td>
<td>Pro, Asp</td>
<td>HIFα [85]</td>
</tr>
</tbody>
</table>

but further experimentation has revealed that other kinases also activate alternate transcriptional programs through this same TF-PTM [138]. Other PTMs that alter CREB activity have also been discovered, including additional phosphorylations, acetylations, and one O-linked N-acetyl glycosylation (O-GlcNAc) that antagonizes the primary activating phosphorylation [110]. CREB exemplifies the potential complexity of TF-PTM regulatory programs beyond simple “on/off” switches.

Few TFs have been studied as extensively as CREB, and numerous TF-PTMs remain to be discovered. To investigate TF-PTM-mediated gene regulation at the level of systems biology, we first need to identify the key regulatory connectivity between modifying enzymes, TFs, and target genes, as well as the specific TF-PTMs involved in transducing these signals. Inclusion of modifying enzymes and higher-order dependency increases the complexity of the network inference problem by several orders of magnitude. However, a variety of high-throughput data sources can be exploited via computational methods to predict the most likely regulatory
Figure 1.1: CREB is a well-studied TF that exemplifies the complexity of TF-PTM regulatory circuits. Canonical CREB regulation begins with phosphorylation of Ser133 by PKA, which facilitates interaction with the co-factor CBP to recruit RNAPII and promote transcription of target genes such as SST. Other kinases can also regulate CREB through Ser133 and other phosphorylation sites. CBP can also regulate CREB through multiple acetylations, and glycosylation can disrupt the activating effect of the Ser133 phosphorylation. Reprinted from Everett L, Vo A, and Hannenhalli S “PTM-Switchboard—a database of post-translational modifications of transcription factors, the mediating enzymes and target genes” Nucleic Acids Research 2009 37:D66–D71 under the Creative Commons Attribution Non-Commercial License.
Genome-Wide Inference of TF-Modifiers

The network inference methods discussed in Section 1.2 only consider regulatory interactions between TFs and target genes, and are therefore limited in terms of the biological phenomena they can explain. Although TF protein concentration is often the primary determinant of target gene expression, TF activity at each target gene promoter is dependent on a number of other factors, including chromatin accessibility and cellular signals [103]. Harbison et al. tested the genome-wide binding of various TFs in *S. cerevisiae* under different experimental conditions and found that many TFs bind different sites under different conditions. Therefore, TF activity at each target gene promoter can, in fact, be specific to a particular subset of cellular conditions, and the TF-Gene network can be dynamically re-wired by other classes of regulatory factors. For this work, I define a *modulator* as any individual protein that influences TF activity post-transcriptionally, including influence on DNA binding and interaction with the core transcriptional machinery. Many types of molecules can act as TF modulators, including chromatin modifying enzymes, cofactors, and TF modifying enzymes. A *regulatory triplet* constitutes a TF, a target gene, and any modulator gene that influences the regulatory interaction between the TF and target gene. A special type of regulatory triplet, of particular interest in this work, includes as the modulator gene a modifying enzyme that directly catalyzes a PTM on the TF protein. I refer to this type of regulatory triplet as a *Modifier-TF-Gene (MFG) triplet* (Figure 1.2).

MFG triplets differ from other regulatory triplets primarily in the physical interaction between the modifying enzyme and the TF, which yields a covalent chemical change in the TF protein. This has several important implications for specifically studying MFG triplets, including: (i) the possibility that the modifying enzyme
Figure 1.2: Example MFG triplet: a factor (F) is unable to regulate the target gene (G) in its initial state. Modification catalyzed by the enzyme (M) transitions F to a new state (marked by *), at which point it is able to regulate G. Reprinted from Everett L, Vo A, and Hannenhalli S “PTM-Switchboard—a database of post-translational modifications of transcription factors, the mediating enzymes and target genes” Nucleic Acids Research 2009 37:D66–D71 under the Creative Commons Attribution Non-Commercial License.

regulates transcription distally, i.e., prior to the TF-DNA interaction and/or outside the nucleus; (ii) the implication that transcript expression data alone may be insufficient for studying MFG triplets, due to post-transcriptional regulation of both the TF and the modifying enzyme; and (iii) a unique set of additional data involving the detection and modeling of enzyme-substrate interactions and PTMs can be applied to the study of MFG triplets (see Section 1.3). In the remainder of this section I review existing methods applicable to the computational prediction of MFG triplets. Most of these methods are designed for the more general problems of inferring TF modulators or condition-specific TF activity, and have inherent limitations motivating the development of new methods as presented in subsequent chapters.

A number of computational methods have been developed to estimate condition-
specific TF activity based on genome-wide expression data. For example, Vadi-gepalli, Gonye et al. developed the Promoter Analysis and Interaction Network Toolset (PAINT) to systematically scan the promoters of genes differentially expressed in specific experimental conditions, thereby predicting candidate TFs coordinating the condition-specific response [71, 197]. McCord et al. ranked genes by TF binding significance and differential expression in specific yeast microarray experiments, then compared these rankings to predict associations between TF activity and experimental conditions [139]. In a similar approach, Boorsma et al. used TF binding data to first establish regulons of target genes for each TF [14], then tested these regulons for differential expression in specific TF and modifying enzyme knock-out experiments [15].

The condition-specific methods above can be used to infer TF modulators in cases where the expression conditions correspond to specific molecular perturbations. To infer the dependence of TF activity on histone modification enzymes, Steinfeld et al. [182] analyzed the expression of TF regulons in S. cerevisiae samples where specific histone modification enzymes were knocked out. Cheng et al. applied a similar method in yeast strains with knockouts of particular kinases, thereby predicting likely MFG triplets mediating changes in life span [25].

An alternate set of methods search for broader patterns of TF and target gene expression across multiple conditions, by broadly dividing the conditions based on certain biological information, and calculating the differential association of gene pairs across subsets of the available samples. Hu et al. [89] have proposed a non-parametric test to detect differentially correlated gene-pairs in two sets of expression samples from different disease classes. In a different study, Hudson et al. [93] analyzed two expression data sets in cattle, one obtained from a less-muscular wild-type, and the other from cattle with a mutation in the TF myostatin. They found that
the coexpression of myostatin with another gene, MYL2, was significantly different between the mutant and the wild-type sets of expression. This differential coexpression led them to detect a change in myostatin activity even though the expression of myostatin gene itself was not significantly different between the mutant and the wild type. Kim et al. developed an algorithm to predict sets of genes which appear to be co-regulated in a subset of conditions within an expression compendium, and uses addition TF binding site data to infer the TFs responsible for these condition-specific regulatory programs [111].

This class of methods can be used to infer TF modulators by partitioning the expression data based on the expression profile of a potential modulator gene, followed by prediction of specific regulatory pairs or regulons that appear to be dependent on the modulator. Zhang et al. have proposed a method in which each gene with a bimodal expression pattern across a compendium is considered as a potential modulator (regardless of function). This bimodality is then used to partition the samples in the expression compendium, and pairs of additional genes can then be tested for a significant difference in correlation between the two partitions to infer regulatory triplets. The regulatory triplets predicted by this method had only weak global correlations, highlighting the difference between pair-wise and higher-order expression models [220]. Wang et al. proposed a similar algorithm, Modulator Inference by Network Dynamics (MINDY), in which the expression compendium is partitioned into equal sizes according to the highest and lowest expression values of a selected modulator [204]. TF-Gene pairs are then tested for a significant difference in mutual information [134] between the partitions. This method has been applied to infer the kinases and other signaling molecules that directly or indirectly modulate activity of the TF c-MYC in B cells.

Segal et al. [175] proposed a related approach in which multiple partitions are
learned according to a decision tree combining TFs and signaling molecules, and each partition is fit to a normal expression model for a particular module of genes. This method was applied in yeast to infer modules of genes regulated by a combination of TFs and upstream signaling genes. Another approach, termed *Liquid Association*, explicitly tries to detect regulatory triplets \((X, Y, Z)\) where the change in correlation between \(X\) and \(Y\) varies continuously with the change in \(Z\), bypassing the need for any partition structure [126].

**Limitations of Existing Methods**

While many of the methods described above have been successful in elucidating conditional or modulated TF activity in general, there remain a number of limitations to overcome, especially with respect to the inference of TF modifying enzymes and MFG triplets. Many of the methods for inferring TF activity in a single condition rely on accurate TF binding data [15, 25, 182]. As shown by Harbison and colleagues, TF binding patterns are also dependent on cellular conditions [81]. Therefore TF binding data in one condition may not be fully informative when analyzing expression from another condition. Furthermore, these condition-specific methods can only infer modulators of TF activity when applied to data in which specific genes have been perturbed.

Methods which must exhaustively analyze many triplet combinations are generally inefficient when applied at a genome-wide scale. Zhang *et al.* only applied their method to 1,000 human genes (less than 5% of known genes), and still needed to analyze approximately 400 million potential triplets. It was impossible to compute p-values with enough precision to overcome the multiple testing problem, even in this limited case, and therefore Zhang *et al.* randomly split the expression compendium
into equal sized training and test sets. The training set was used to rank the significance of each possible triplet, and the top 10,000 triplets (less than 0.0001% of triplets analyzed) were re-analyzed in the test set. Thus, the methodology of Zhang et al. raises substantial concerns about sensitivity and scalability. Attempting to apply this methodology to a larger number of genes, or a larger percentage of the triplets analyzed, is likely to exacerbate the multiple testing problem. Furthermore, the need to split the data into training and test sets means that only half the available data can actually be used to predict regulatory triplets [220]. MINDY has similar limitations, although in this case Wang et al. chose to limit their analysis to a relatively small number of modulators with known roles in cell signaling, and only TFs of interest, rather than exhaustively searching all possible triplets [204].

In general, methods that attempt to infer network structure from expression data alone cannot truly differentiate direct and indirect interactions, and often cannot distinguish the direction of regulation [133, 134]. These problems are only exacerbated in the inference of regulatory triplets. Even when TF binding data are introduced to support the directness and directionality of the TF-Gene interaction, the effect of the inferred modulator can often be indirect [25]. The modulator genes are also transcriptionally regulated, and often have expression profiles correlated to other genes. Furthermore, expression levels for TFs and modulators, especially modifying enzymes, are not a direct measure of activity. Methods for predicting regulatory triplets from an expression compendium struggle to distinguish between several candidate modulators that are coexpressed, and often cannot analyze regulatory triplets where the modulator is coexpressed with either the TF or target gene [126, 204, 220]. Feedback loops and parallel signaling pathways are likely to introduce such artifacts, and therefore present a challenge for many of the existing methods. Ultimately, these problems limit the sensitivity of current network infer-
ence methods, which translates into fewer hypotheses for experimental validation, and an incomplete picture of network behavior. Thus, there is a critical need in the network inference field for methods that can integrate and fully utilize a wide range of available data.

1.5 Overview of Thesis Work

In this thesis, I have developed new methodologies specifically designed to address the challenges discussed above. In Chapter 2, I first establish a database for cataloging known examples, in order to provide a “gold set” for method validation. In Chapter 3, I introduce the Modification-dependent Network-based Transcriptional Estimator (MoNsTER) algorithm for modeling the synergistic effects of TF-Modifier pairs on putative regulons of target genes. MoNsTER also incorporates additional physical evidence for direct TF-Gene and TF-Modifier interactions to overcome the limitations of analyzing expression data alone. In Chapter 4, I apply MoNsTER to study the role of the TF STAT1 in human B cells. In Chapter 5, I propose alterations to common proteomic methods that can improve the reliable detection of PTMs by mass spectrometry. Finally, in Chapter 6, I suggest novel ways in which these and related tools can be improved and combined to continually expand our knowledge of TF-PTM connectivity in regulatory networks. The scope of this work is primarily limited to the regulation of target genes by TF proteins binding to proximal promoter regions, and the PTMs occurring on these TF proteins. This limitation is motivated by the currently available data, and the complexity that results from considering multiple, interacting forms of gene regulation. However, many of the methods presented here will be readily adaptable to other mechanisms of regulation, when data becomes available at the appropriate scale.
Chapter 2

PTM-Switchboard: a Database of MFG Triplets

This chapter adapted from Everett L, Vo A, and Hannenhalli S “PTM-Switchboard—a database of post-translational modifications of transcription factors, the mediating enzymes and target genes” Nucleic Acids Research 2009 37:D66–D71 under the Creative Commons Attribution Non-Commercial License (http://creativecommons.org/licenses/by-nc/2.0/uk/), available online at: http://nar.oxfordjournals.org

2.1 Introduction

Many instances of TF-PTMs and their effect on gene regulation have been experimentally determined (see Table 1.1, e.g.), a systematic meta-analysis of these examples has not been undertaken. Furthermore, while the current knowledge of TF-PTMs is limited to a subset of TFs and modifying enzymes, it is nevertheless highly valuable for the development and validation of computational models. Thus, there is a need for a structured and machine-readable collection of experimentally
derived regulatory triplets. The **PTM-Switchboard** database addresses this need by cataloging known examples of MFG triplets (as defined in Chapter 1) in the model system *S. cerevisiae*. In this critical aspect, **PTM-Switchboard** differs from previous molecular pathway databases [29, 74, 106], which support only pair-wise relationship between genes. The database currently includes a large sample of experimentally characterized instances curated from the literature. In addition to providing a framework for searching and analyzing existing knowledge, the collection as a whole serves to benchmark computational methods, including those presented in subsequent chapters of this work. Methods for predicting MFG triplets are in their infancy and lack a “gold set” by which to gauge their performance. The database currently contains a sufficient set to evaluate computational approaches and seed further cataloging efforts.

**PTM-Switchboard** is also intended to seed a larger community effort to build a more comprehensive database of MFG triplets, as they are extremely laborious to search and curate from the literature. Text-mining approaches [170, 218] are currently limited to identifying pair-wise interactions, and MFG triplets are rarely studied together as part of a single paper. More commonly, the overall effects of a PTM on a TF’s cellular localization, degradation or DNA-binding activity are studied in one reference, while the gene targets of the TF are studied independently of any PTMs in another reference. In some cases, MFG triplets can be inferred from these references together, but only with careful consideration of the molecular mechanisms involved—a task clearly beyond current text-mining methods. **PTM-Switchboard** links the two fields of cell signaling and transcriptional regulation, and provides substantial opportunity to leverage the existing knowledge bases in these fields.
2.2 Database Overview

The primary unit of data used in PTM-Switchboard is a three-way interaction—the MFG triplet—as described in Chapter 1 and illustrated in Figure 1.2. The database currently contains 519 fully characterized and experimentally validated MFG triplets. These triplets cover a total of 14 modifying enzymes and 15 TFs (Table 2.2), targeting 212 genes in 17 cellular contexts, and are derived from 69 different literature and knowledge base references. Each MFG triplet is stored under the data schema\(^1\) shown in Table 2.1. As noted in Chapter 1, the MFG triplet representation allows the effect of a modifying enzyme to be defined for each of a TF-Genes interaction individually, rather than uniformly across all TF targets. Therefore, multiple triplets contained in the database may share one or two members. For example, Hog1 regulates the overall transcriptional activity of Sko1 at a set of target gene promoters, and a separate triplet is included for each target.

The schema shown in Table 2.1 includes descriptive biological information about each MFG triplet, in addition to the identifiers for each gene member. The modifier can either have a positive or negative effect on the activity of the TF, and likewise the TF can be an activator or repressor of each target gene. In some especially complex cases, such as Sko1, the TF can act as both an activator and a repressor, depending on the activity of the modifying enzyme—in this case, kinase Hog1 [161]. To describe the behavior in such cases, the overall activity of the triplet is described by the influence of the TF on the target gene (positive, negative or neutral) in both the cases when the modifying enzyme is active and when the modifying enzyme is inactive. For example, in Figure 1.2 the relationship between \(F\) and \(G\) is neutral when \(M\) is inactive, and positive when \(M\) is active. Note that this is still a

\(^1\)PTM-Switchboard is implemented using the MySQL database engine: http://www.mysql.com
<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>mod_name</td>
<td>SGD official gene name for the modifying enzyme in the triplet</td>
</tr>
<tr>
<td>mod_orfid</td>
<td>SGD systematic ORF ID for the modifying enzyme</td>
</tr>
<tr>
<td>mod_class</td>
<td>Class/type of modifying enzyme, e.g., “Kinase”</td>
</tr>
<tr>
<td>tf_name</td>
<td>SGD official gene name for the TF in the triplet</td>
</tr>
<tr>
<td>tf_orfid</td>
<td>SGD systematic ORF ID for the TF</td>
</tr>
<tr>
<td>gene_name</td>
<td>SGD official gene name for the target gene</td>
</tr>
<tr>
<td>gene_orfid</td>
<td>SGD systematic ORF ID for the target gene</td>
</tr>
<tr>
<td>ptm_ids</td>
<td>List of individual PTM sites (Table 2.3) that have been mapped for this particular TF-Modifier interaction, each with indicator of whether the modifier adds or removes the PTM.</td>
</tr>
<tr>
<td>context_ids</td>
<td>List of cellular contexts in which triplet is known to function</td>
</tr>
<tr>
<td>ref_ids</td>
<td>List of literature and external database references containing evidence for this triplet</td>
</tr>
<tr>
<td>mod_low</td>
<td>Describes the interaction between the TF and target gene when the modifier activity is low. A ‘+’ value indicates the TF promotes target gene expression, ‘-’ indicates the TF represses target gene expression, and ‘0’ indicates a disruption of the regulatory interaction between TF and target gene.</td>
</tr>
<tr>
<td>mod_high</td>
<td>Describes the interaction between the TF and target gene when the modifier activity is high. Possible values are the same as for mod_low.</td>
</tr>
<tr>
<td>evidence</td>
<td>Evidence codes (Table 2.4) indicating the types of evidence found in support of the triplet from all references</td>
</tr>
</tbody>
</table>

Table 2.1: Schema for storing MFG triplets in PTM-Switchboard database. Each triplet record is stored with all fields listed here.

simplification of complex dynamics, and is intended to serve as a primer for more quantitative investigations.

PTM-Switchboard specifically focuses on MFG triplets involving a direct catalysis of a TF-PTM by a modifying enzyme, and does not include other types of regulatory triplets or cases that lack evidence for direct catalysis. Furthermore, the PTM must be relevant to TF activity at the promoter of the included target gene. This basic unit is an essential building block for more advanced studies of kinetics and fine-scale regulation at both the transcriptional and post-transcriptional level.
Table 2.2: Unique Modifiers and TFs included in PTM-Switchboard, listed by SGD gene symbols.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ptm_type</td>
<td>The type of modification, e.g., phosphorylation</td>
</tr>
<tr>
<td>prot_name</td>
<td>The SGD official gene name for the protein that is modified</td>
</tr>
<tr>
<td>prot_orfid</td>
<td>The SGD systematic ORF ID for the protein that is modified</td>
</tr>
<tr>
<td>prot_refseq</td>
<td>The RefSeq accession of the translated protein sequence</td>
</tr>
<tr>
<td>ptm_pos</td>
<td>The position of the modified amino acid (in the official RefSeq sequence)</td>
</tr>
<tr>
<td>ptm_res</td>
<td>The single character abbreviation of the amino acid that is modified, e.g.,</td>
</tr>
<tr>
<td></td>
<td>‘Y’ for tyrosine</td>
</tr>
<tr>
<td>context_ids</td>
<td>List of contexts in which the PTM has been observed</td>
</tr>
<tr>
<td>ref_ids</td>
<td>List of literature and external database references containing evidence for</td>
</tr>
<tr>
<td></td>
<td>this PTM.</td>
</tr>
</tbody>
</table>

Table 2.3: Schema for storing PTMs in PTM-Switchboard database. Each record with this schema corresponds to a unique modification site on a particular yeast protein.

Each unique PTM site (when mapped) is stored separately, according to the schema in Table 2.3. This table records each PTM at a specific residue of a TF protein, but independent of the modifying enzyme responsible. The same PTM can be involved in multiple MFG triplets, and likewise an MFG triplet can involve multiple PTMs. MFG triplets can involve both addition and removal of a chemical group. Even in the well-studied cases, the exact positions of the PTMs often have not been mapped. However, PTM-Switchboard provides a framework to store these PTMs as they are identified or hypothesized.
2.3 Supporting Evidence

PTM-Switchboard is also an exploratory tool for molecular biologists, and provides a considerable amount of supporting data for the curated instances of the MFG triplets, including links to external annotation resources [74, 156]. All genes are recorded using both their official gene symbol and systematic ORF ID according to the Saccharomyces Genome Database (SGD) [58], thus directly linking each gene to its SGD annotation page. All information contained in PTM-Switchboard is derived either from literature or other knowledge bases, and the sources used to derive this information are continually tracked and annotated, as discussed in the next section.

The MFG triplets currently contained in PTM-Switchboard vary in their mechanistic complexity and are supported by heterogeneous experimental evidence. To help the user assess each triplet, the database includes an extensive annotation of the sources for each MFG triplet. The most convincing cases are those in which all three genes are studied together—i.e., the effect of perturbing the modifying enzyme is studied on both the TF and the target gene—using a combination of genetic and biochemical techniques.

While MFG triplets identified through targeted biochemical experiments garner the highest confidence in their accuracy, there is also a great deal of knowledge to be gained from high-throughput and genetic experiments, even if these experiments occasionally provide invalid MFG triplets. Genetic experiments alone can detect MFG triplets, but cannot distinguish them from other forms of regulation and network connectivity. For example, the modifying enzyme may be further upstream from the TF in the signaling pathway, or likewise the known TF may target another TF, which then targets the gene of interest. Many kinases are also known to operate as co-factors, i.e., by binding TFs at promoters to help recruit, activate,
<table>
<thead>
<tr>
<th>Code</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>C</td>
<td>Supported by biochemical experiment</td>
</tr>
<tr>
<td>G</td>
<td>Supported by genetic experiment</td>
</tr>
<tr>
<td>V</td>
<td>Experimentally demonstrated <em>in vivo</em></td>
</tr>
<tr>
<td>T</td>
<td>Experimentally demonstrated <em>in vitro</em></td>
</tr>
<tr>
<td>L</td>
<td>Identified in low throughput experiment</td>
</tr>
<tr>
<td>H</td>
<td>Identified in high throughput experiment</td>
</tr>
<tr>
<td>I</td>
<td>Inferred from multiple sources</td>
</tr>
<tr>
<td>S</td>
<td>Demonstrated in a single experiment</td>
</tr>
<tr>
<td>M</td>
<td>Identified from literature by manual curation</td>
</tr>
<tr>
<td>X</td>
<td>Identified from literature by text-mining</td>
</tr>
<tr>
<td>P</td>
<td>Predicted by computational tool</td>
</tr>
<tr>
<td>O</td>
<td>Identified by homology to a known triplet in another organism</td>
</tr>
</tbody>
</table>

Table 2.4: **PTM-Switchboard** evidence codes used to characterize the experimental evidence for each PTM and MFG triplet.

or block the core transcriptional machinery \([7, 152, 155]\). Such mechanisms do not qualify for inclusion in **PTM-Switchboard**, however any triplet is useful to include in the database if the best model explaining the genetic evidence involves a TF-PTM. Therefore, **PTM-Switchboard** has been designed to integrate current knowledge from all available sources.

To address the potentially subjective nature of weighing different evidence types, we have developed a system of tagging each triplet with a set of codes that briefly and concisely describe the evidence contained in the literature. This allows the user to quickly assess their confidence in individual triplets based on their own criteria without having to examine each literature reference. I also included tags to catalog MFG triplets predicted by text-mining and computational modeling approaches, which may be useful for future content (see Chapter 6). Appropriate evidence codes can be assigned to separate the gold set from computational predictions. A complete set of the evidence codes currently used is shown in Table 2.4.
Of particular interest is the ‘I’ evidence code, which indicates that the triplet was inferred from a combination of literature references. For instance, one paper may demonstrate that a modifying enzyme regulates a TFs overall activity under a specific condition, e.g., by changing its nuclear import/export, while a second paper may demonstrate that the TF is known to regulate a target gene under a similar condition. Triplets marked with the ‘I’ code indicate that the experimental conditions in the referenced papers were comparable enough to justify combining their results. The ability to integrate multiple references is a major advantage of expert curation.

In addition to the gold set of 519 fully characterized and experimentally validated MFG triplets, PTM-Switchboard also tracks incomplete MFG triplets where computational predictions may be of the most immediate value to molecular biologists. These typically occur in the literature when perturbation of a modifying enzyme leads to a significant change in another genes transcript level, but the TF transducing the signal is unknown. Likewise, a PTM may be known to influence the activity of a TF, but the enzyme catalyzing the addition or removal of this PTM may be unknown. These entries are marked with the evidence code ‘U’ to mark them as having exactly one unknown member and they are excluded from the “gold set” for validating prediction algorithms.

2.4 Guidelines for Literature Curation

The current content of PTM-Switchboard was obtained through careful curation of primary literature. Text-mining approaches [170, 218] are currently limited to identifying pair-wise interactions from individual articles. MFG triplets are rarely studied together as part of a single paper, and therefore require the integration
of knowledge from multiple literature sources. Therefore, this process currently requires a curator with knowledge of basic molecular biology to understand the various subtleties involved in drawing inferences from multiple experiments. As such, there is no strict algorithm for extracting an MFG triplet from literature sources, but the following steps provide a general guideline for the annotation process:

1. Given one or more related papers, e.g., starting from a review of a particular TF or signaling pathway, identify the genes explicitly tested and discussed, as well as the connectivity between these genes to infer appropriate triplets.

2. Identify the relevant contexts, i.e., cellular or experimental perturbations, under which each triplet was studied. This is specific only to the type of perturbation, such as “osmotic shock” rather than the exact details of the reagents and conditions used. If a similar context is already present in the database, the vocabulary is kept consistent. If multiple papers are being used to infer a single triplet, the contexts are checked for consistency across all papers.

3. Assess the types of evidence present in the papers and annotate with the relevant evidence codes, as described in Section 2.3.

4. Identify other relevant papers in PubMed\(^2\) and SGD [58]. If the initial paper is a review, references to the original experimental papers are followed. If biochemical, in vivo, or low-throughput evidence is lacking for a triplet, an attempt is made to find other papers that contain such evidence. If no information has been found on the positions of relevant PTMs, an attempt is made to find papers mapping these PTMs. For PTMs that broadly regulate TF activity, e.g., via nuclear trafficking or degradation, a search is also made.

\(^2\)Available online at: http://www.ncbi.nlm.nih.gov/pubmed
for papers that identify other targets of the same TF in the same contexts, in order to identify partially overlapping triplets. Steps 1–3 are repeated on all additional papers.

5. Resolve gene identity and PTM ambiguities. Gene names are translated to their official SGD symbols before being entered in the database. The positions of all PTMs are resolved against the official reference sequence in RefSeq [156] using sequence match.

2.5 Case Study: The FUS3/STE12/FUS1 Triplet

Here, I present one example of PTM-mediated transcriptional regulation included in PTM-Switchboard that highlights the mechanistic complexity of this process and thus the value of expert curation. Additionally, this case study illustrates the guidelines for inclusion of a MFG triplet in the database and the benefit of using evidence codes. Activation of the mitogen-activated protein kinase (MAPK) Fus3 is known to up-regulate the transcription of many genes, including FUS1, via activation of the TF Ste12 at pheromone response elements in gene promoters [76]. Yeast mating factors trigger this response via a well-studied signal transduction pathway culminating in this MFG triplet [6, 24, 45].

On the surface this appears to be a straightforward MFG triplet, with the simplest model suggesting that Fus3 activates Ste12 via phosphorylation. However, a closer inspection of the literature reveals alternative regulatory mechanisms, including inhibitory phosphorylation of the Ste12 repressors Dig1 and Dig2 [24] and direct repression of Ste12 protein by inactive Fus3, which is lifted upon Fus3 activation [7, 38]. Most experiments focused on this triplet are genetic in nature, and cannot distinguish between these mechanisms. A small fraction of the published
research articles contain biochemical evidence demonstrating that active Fus3 also phosphorylates Ste12 [47] in a way that promotes target gene expression [94, 179]. Therefore, this MFG triplet was included in the database, but without such evidence, the triplet would not have been included. Furthermore, triplets involving Dig1 and Dig2 as modulators of Ste12 do not meet inclusion guidelines for the database regardless of evidence, because these triplets do not involve direct catalysis of Ste12 modification.

In a similar case, there is genetic evidence for activation of Ste12 by another MAPK, Kss1 and the most widely accepted model for this evidence includes phosphorylation of Ste12 by Kss1 [7]. In this case, a set of KSS1/STE12 triplets is included in PTM-Switchboard, but assigned a different set of evidence codes to reflect the lack of biochemical experiments. The FUS3/STE12/FUS1 triplet highlights the need to separate biochemical and genetic evidence to study PTM-dependent regulation because genetic evidence can often mask more complex mechanisms. The basic knowledge that a modifying enzyme activates or represses a TF cannot be assumed to involve a PTM in all cases.

2.6 Discussion

In this chapter, I described a new kind of database—PTM-Switchboard—to catalog TF-PTM mediated regulation of gene transcription. As illustrated by the case of the FUS3/STE12/FUS1 triplet, curating known MFG triplets currently requires an expert examination of literature. I have curated over 500 complete MFG triplets, thus establishing the groundwork for continued efforts. The database provides a focused and structured platform to leverage other regulatory network and signaling pathway resources for this multi-faceted task.
The first version of PTM-Switchboard is meant as the starting point towards a comprehensive database, but can also serve several more immediate purposes. For the molecular biologist studying a particular gene or pathway, PTM-Switchboard is available as a repository of structured MFG triplet data that is otherwise tedious to extract from the literature. Indeed, specific protein and transcription factor databases such as the Universal Protein Resource (UniProt) [3], TRANSFAC [209], and dbPTM [122] include information relevant to MFG triplets. However, integration of these resources to extract consistent information needed for MFG triplet ascertainment is not straightforward. For the computational biologist, the entirety of the database—or a subset filtered by an appropriate evidence code—can serve as an ideal “gold set” for training and testing computational methods, such as those reviewed in Chapter 1. In fact, PTM-Switchboard has been critical in validating the computational method presented in Chapter 3, and has also been used to validate the Mimosa algorithm [80]. Moreover, the computational predictions can be conveniently compiled in PTM-Switchboard and made accessible to molecular biologists for experimental validation.

The collection is also immediately useful to researchers interested in the approximately two-dozen modifying enzymes and TFs included in the database (Table 2.2), or as a resource for studying the regulation of over 200 target genes. PTM-Switchboard provides extensive connectivity to other data repositories, making it an ideal portal for researchers studying transcriptional regulation or cell signaling in S. cerevisiae. The knowledge catalogued in PTM-Switchboard will be immediately helpful in designing further validation experiments and to place these experimental observations in a broader context of cell signaling. The database can also serve as a platform to catalog further literature curation, text-mining results, and predictions from computational models (see Chapter 6).
Investigation of transcriptional networks based entirely on genomic and transcriptomic information, as is the current practice, is limited. This collection is designed to encourage more integrative computational approaches based on post-transcriptional data sources, such as the method described in Chapter 3. By cataloging known instances of TF-PTM mediated regulation of gene transcription, PTM-Switchboard bridges the current resources in the fields of cell signaling and transcriptional regulation to facilitate a broader understanding of regulatory networks.
Chapter 3

MoNsTER: An Integrative Model of Transcriptional Regulation via TF-Modifying Enzymes

3.1 Introduction

As noted in Chapter 1, the inference of TF modulators specifically acting through direct catalysis of TF-PTMs can be aided by additional types of high-throughput data. I therefore propose a principled computational model of gene transcription that explicitly incorporates interactions between modifying enzymes and TFs, allowing for the integration of heterogeneous genomic and proteomic data. This model provides the framework for a new network inference method, called Modification-dependent Network-based Transcriptional Estimator (MoNsTER), which combines an expression compendium (in this work, defined as steady-state mRNA levels measured by microarray in multiple experiments) with other data sources indicative of physical protein-DNA and protein-protein interactions to simultaneously infer the target
genes and upstream modifiers of each TF.

In this chapter, I formally define the probabilistic model of gene expression underlying this method. The necessary complexity of this model motivates the development of a novel heuristic algorithm for fitting the model parameters to the available data in a robust and efficient manner. I then use simulated data to demonstrate that the model and parameter estimation procedure are robust against noise from a variety of sources. Next, I use a well-studied stress-response regulatory network in the model system *S. cerevisiae* to demonstrate the accuracy of MoNsTEr on experimental data.

### 3.2 The Integrative Network Model

The computational problem addressed in this chapter is the inference of a regulatory network model that incorporates: (i) interactions between TFs and gene regulatory regions, and (ii) interactions between TFs and their modifying enzymes. In this section, I introduce and explain the underlying mathematics of the model (represented graphically in Figure 3.1), and introduce a novel heuristic approach to efficiently estimate the model parameters. Individual variables are denoted in italics and the corresponding vectors and matrices of variables are denoted in bold. See Tables 3.1 and 3.2 for a guide to the notation used in this chapter.

**Primary Model Equations**

The model used here is an extension of the variable-selection linear regression model first described by Chen *et al.* [22], and is described concisely by the following equations and notation. Given potential target genes indexed by *i* from 1 to *N*, TFs of interest indexed by *j* from 1 to *J*, and modifiers of interest indexed by *k* from 1
Figure 3.1: Conceptual diagram of network model with relationships to model equations. Input data is shown in green and model parameters are shown in blue. Expression matrices g, f, and h correspond to samples t for genes i, TFs j, and enzymes k respectively. Prior matrices b, m, a, and s are derived from TF binding data, TF motif data, protein-protein interaction data, and protein motif data, respectively. Model parameters include TF-Gene edges C and TF-Modifier edges D, TF activities β, TF-Modifier synergy effects γ, TF-specific prior weights w, and modifier-specific prior weights u.

to K, with expression measured under conditions indexed by t from 1 to T: let g denote the $N \times T$ expression matrix for target genes with values $g_{it}$; let f denote the $J \times T$ expression matrix for TFs with values $f_{jt}$; and let h denote the $K \times T$ expression matrix for modifiers with values $h_{kt}$. Thus, the first step of the modeling procedure is the selection of an appropriate set of candidate TFs, modifiers, and target genes for the network of interest. In particular, the target genes $i$ should not overlap with the TFs $j$ or modifiers $k$, and therefore $i$ can include all other genes.
for which there is sufficient data after selecting \( j \) and \( k \), or a subset thereof. The model defines each target gene expression value \( g_{it} \) as a function of four additive components: (i) basal expression, (ii) direct influence from regulating TFs, (iii) synergistic influence from specific TF-Modifier pairs, and (iv) an error component \( \varepsilon_{it} \) encompassing technical and biological noise. These components are formally defined in the following equation:

\[
g_{it} = \alpha_i + \sum_{j=1}^{J} \beta_j C_{ij} f_{jt} + \sum_{j=1}^{J} \sum_{k=1}^{K} \gamma_{jk} C_{ij} D_{jk} \Phi(f_{jt}, h_{kt}) + \varepsilon_{it} \tag{3.1}
\]

Eq 3.1 is applied to all genes \( i \) from 1 to \( N \) and all samples \( t \) from 1 to \( T \). The term \( \alpha_i \) represents a baseline expression value for gene \( i \) independent of condition. The term \( \beta_j \) is a scaling factor for TF \( j \) to describe its linear influence on all target genes in all conditions. The terms \( C_{ij} \), denoted collectively by the \( N \times J \) matrix \( C \), are binary variables indicating whether each TF \( j \) regulates the expression of each gene \( i \). By definition, \( C_{ij} = 1 \) implies that TF \( j \) is a predicted regulator of gene \( i \) (\( C_{ij} = 0 \) otherwise). Thus, in this model, TF \( j \) only directly affects the subset of genes \( i \) where \( C_{ij} = 1 \). Modifiers differ from TFs in the model in that they do not influence target gene expression directly, but rather indirectly by modulating TF activity. Each TF-Modifier pair \((j, k)\) has an influence parameter \( \gamma_{jk} \) and an edge indicator variable \( D_{jk} \), analogous to the \( \beta_j \) and \( C_{ij} \) variables respectively, i.e., \( D_{jk} = 1 \) if modifier \( k \) has a synergistic effect on the target genes of TF \( j \). The nature of these synergistic effects are described in general as a function \( \Phi \) of TF and modifier expression values, discussed in more detail below. The term \( \varepsilon_{it} \) corresponds to the residual error of the model fit for each \( g_{it} \). This term captures technical and biological noise as well as non-model behavior and is sampled from the normal
<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$N$</td>
<td>Number of target genes in input data</td>
</tr>
<tr>
<td>$J$</td>
<td>Number of TFs in input data</td>
</tr>
<tr>
<td>$K$</td>
<td>Number of modifiers in input data</td>
</tr>
<tr>
<td>$T$</td>
<td>Number of expression samples in input data</td>
</tr>
<tr>
<td>$i$</td>
<td>Index over target genes in input data, range $(1, N)$</td>
</tr>
<tr>
<td>$j$</td>
<td>Index over TFs in input data, range $(1, J)$</td>
</tr>
<tr>
<td>$k$</td>
<td>Index over modifiers in input data, range $(1, K)$</td>
</tr>
<tr>
<td>$t$</td>
<td>Index over samples in input data, range $(1, T)$</td>
</tr>
<tr>
<td>$g$</td>
<td>Matrix of target gene expression values for all genes $i$ and all samples $t$, size $N \times T$</td>
</tr>
<tr>
<td>$f$</td>
<td>Matrix of TF expression values for all TFs $j$ and all samples $t$, size $J \times T$</td>
</tr>
<tr>
<td>$h$</td>
<td>Matrix of modifier expression values for all modifiers $k$ and samples $t$, size $K \times T$</td>
</tr>
<tr>
<td>$b$</td>
<td>TF-Gene prior matrix based on experimental binding data for all genes $i$ and TFs $j$, size $N \times J$</td>
</tr>
<tr>
<td>$m$</td>
<td>TF-Gene prior matrix based on promoter motif analysis for all genes $i$ and TFs $j$, size $N \times J$</td>
</tr>
<tr>
<td>$a$</td>
<td>TF-Modifier prior matrix based on protein-protein interaction data for all modifiers $k$ and TFs $j$, size $J \times K$</td>
</tr>
<tr>
<td>$s$</td>
<td>TF-Modifier prior matrix based on substrate profile analysis for all modifiers $k$ and TFs $j$, size $J \times K$</td>
</tr>
<tr>
<td>$\Xi$</td>
<td>The set of all input data matrices, $(g, f, h, b, m, a, s)$</td>
</tr>
</tbody>
</table>

Table 3.1: Summary of notation representing components of input data for MoNsTEr network model distribution with model-wide variance parameter $\sigma^2$:

$$\varepsilon_{it} \sim N(0, \sigma^2) \quad \forall i, t$$  \hspace{1cm} (3.2)

The interaction function $\Phi$ is, generally speaking, intended to be a non-linear representation for synergistic TF-Modifier interaction effects on target gene expression. The function can be computed prior to model-fitting, and therefore different functions do not alter the complexity of the model or the computations discussed in subsequent sections. In this work, I used a sign-corrected product in order to capture a synergistic effect that is positive only when both input expression values
are positive, and negative in all other cases:

\[
\Phi(\mathbf{f}_{jt}, \mathbf{h}_{kt}) = \text{sign}(\min(\mathbf{f}_{jt}, \mathbf{h}_{kt}))|\mathbf{f}_{jt} \ast \mathbf{h}_{kt}| = \begin{cases} 
\mathbf{f}_{jt} \ast \mathbf{h}_{kt} & \text{iff } \mathbf{f}_{jt}, \mathbf{h}_{kt} > 0 \\
-|\mathbf{f}_{jt} \ast \mathbf{h}_{kt}| & \text{otherwise}
\end{cases} \tag{3.3}
\]

This function is intended for normalized expression profiles with mean 0 (as enforced by expression normalization, see Section 3.3), and has the preferable behavior that it returns positive values only when \(\mathbf{f}_{jt}\) and \(\mathbf{h}_{kt}\) are both positive (corresponding to high expression).

The model also allows the consideration of other types of high-throughput data, in addition to expression by defining prior probabilities for the edge variables \(\mathbf{C}\) and \(\mathbf{D}\). Let \(\mathbf{b}\) and \(\mathbf{m}\) denote two \(N \times J\) matrices of prior probabilities for TF-Gene interactions based on different types of biological data regarding protein-DNA interactions. In the applications presented here, \(b_{ij}\) is the prior probability that TF \(j\) binds the proximal promoter of gene \(i\) based on a ChIP-seq binding experiment. Similarly, \(m_{ij}\) is a prior probability derived from scanning the promoter of gene \(i\) with a PWM describing the preferred binding sequence for TF \(j\). Priors for TF-Modifier interactions are derived from additional \(J \times K\) matrices \(\mathbf{a}\) (derived from protein-protein interaction data) and \(\mathbf{s}\) (derived from PWMs describing modifier-specific substrate recognition motifs). Thus, the variable matrices \(\mathbf{C}\) and \(\mathbf{D}\) in the model are given prior probabilities based on a weighted mixture of all available biological priors, as follows:

\[
P(C_{ij} = 1) = b_{ij}^{w_{ij}} m_{ij}^{(1-w_{ij})} \tag{3.4}
\]

\[
P(D_{jk} = 1) = a_{jk}^{u_{jk}} s_{jk}^{(1-u_{jk})} \tag{3.5}
\]

The weight variables in the above equations, denoted collectively as \(\mathbf{w}\) and \(\mathbf{u}\),
each range from 0 to 1 and are also estimated as part of the model-fitting procedure. Thus, no assumptions need to be made about the relative quality of each prior source. Note that there are separate weight variables $w_j$ and $u_k$ for each TF $j$ and modifier $k$ respectively. This was chosen because the quality of a given PWM or ChIP-chip result set will typically vary more widely by TF, rather than by target gene. Likewise, the PWM or interaction data for a modifier is also assumed to vary more widely by modifier, rather than by substrate or interaction partner. Note that when only a single type of prior is available for either $C$ or $D$, then Eq 3.4 or 3.5, respectively, simplifies to an unweighted use of the available prior. In other words, if no PWM data is available for a particular TF $j$, then $w_j = 1$, and $P(C_{ij} = 1) = b_{ij}$ for all genes $i$.

Also note that given any full instantiation of values for $C$ and $D$, Eq 3.1 becomes a linear regression and can be solved using a standard closed-form solution. Although this problem can be solved as a linear regression, the relationship between target gene, TF, and modifier expression values is non-linear because of the synergy function $\Phi$ (Eq 3.3).

**Model Estimation Method**

The goal of this computational method is to estimate the posterior probabilities for all model parameters, given the expression data and the priors. Once estimated, the posterior probabilities of $C$ and $D$ can be used to probabilistically infer a network with connectivity between target genes, TFs, and modifying enzymes. In addition, the best-fit values of other parameters such as $\beta$ and $\gamma$ can be used to infer the strength and directionality of these interactions. Let $\Theta$ denote the set of model parameters $\{\alpha, \beta, \gamma, \sigma, w, u\}$, excluding $C$ and $D$. Let $\Xi$ denote the complete set
Table 3.2: Summary of notation representing parameters in the network model.

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>(\alpha)</td>
<td>Vector of basal expression values for target genes, length (N)</td>
</tr>
<tr>
<td>(\beta)</td>
<td>Vector of TF influence parameters, describing linear effect of TF on all target genes, length (J)</td>
</tr>
<tr>
<td>(\gamma)</td>
<td>Matrix of TF-Modifier synergy parameters, describing the sign and magnitude of TF-Modifier synergy terms, size (J \times K)</td>
</tr>
<tr>
<td>(C)</td>
<td>Matrix of indicator variables describing TF-Gene network connectivity, size (N \times J)</td>
</tr>
<tr>
<td>(D)</td>
<td>Matrix of indicator variables describing TF-Modifier network connectivity, size (J \times K)</td>
</tr>
<tr>
<td>(\Phi)</td>
<td>General synergy function mapping expression values ((f_{jt}, h_{kt})) to synergistic effect of interaction between TF (j) and modifier (k)</td>
</tr>
<tr>
<td>(\varepsilon_{it})</td>
<td>Residual error for observed expression of gene (i) in sample (t)</td>
</tr>
<tr>
<td>(\sigma^2)</td>
<td>Model-wide variance of residual errors</td>
</tr>
<tr>
<td>(w)</td>
<td>Vector of weight variables to apply to (b) and (m) for each TF, length (J), values in range ((0, 1))</td>
</tr>
<tr>
<td>(u)</td>
<td>Vector of weight variables to apply to (a) and (s) for each modifier, length (K), values in range ((0, 1))</td>
</tr>
<tr>
<td>(\tau_\alpha)</td>
<td>Hyperparameter for the standard deviation of normal prior for all members of (\alpha)</td>
</tr>
<tr>
<td>(\tau_\beta)</td>
<td>Hyperparameter for the standard deviation of normal prior for all members of (\beta)</td>
</tr>
<tr>
<td>(\tau_\gamma)</td>
<td>Hyperparameter for the standard deviation of normal prior for all members of (\gamma)</td>
</tr>
<tr>
<td>(\Theta)</td>
<td>The set of all non-edge model parameters, ((\alpha, \beta, \gamma, \sigma^2, w, u))</td>
</tr>
</tbody>
</table>

of biological data sources \({g, f, h, b, m, a, s}\). Thus, the model can be completely described by the set \(\{\Xi, \Theta, C, D\}\). The relative posterior probability of a model configuration \(\{\Theta, C, D\}\), given a set of biological data \(\Xi\) is:

\[
P(\Theta, C, D|\Xi) \propto P(g|f, h, C, D, \Theta) \ast P(C|m, b, w) \ast P(D|a, s, u) \ast P(\Theta) \quad (3.6)
\]

In other words, the posterior probability of a model given input data is proportional to the product of the gene expression likelihood, the edge priors, and the remaining parameter priors. The priors on all model parameters in \(\Theta\) are the same as in ref. [22]. Specifically, \(\alpha\), \(\beta\), and \(\gamma\) are assumed to have normal priors with
standard deviations $\tau_\alpha$, $\tau_\beta$, and $\tau_\gamma$ respectively. The $\tau$ hyperparameters are set to a large value, e.g., 10,000, to make the parameter priors uninformative. The model-wide variance, $\sigma^2$, is assumed to have a prior defined by the $\chi^2$ distribution with $\nu = 2$. The weight variables $w$ and $u$ are given uniform priors in the range $(0,1)$. See Appendix A.1 for full derivations of these posterior distributions.

Solving for the full posterior distribution (Eq 3.6) analytically is an intractable problem. However, it is possible to compute an “individual posterior” on each model parameter given the biological data and fixed values for all other model parameters. These equations form the basis for iterative approaches to estimating the model parameters, and are derived in Appendix A.1.

The full posterior (Eq 3.6) can therefore be estimated in a standard iterative framework, such as Gibbs Sampling [65] or hill-climbing. However, the posterior distributions relevant to this model are typically highly multi-modal. This results in many regions of the parameter space that are “locally good”, i.e. they are considerably more likely than similar model configurations. In other words, there are often multiple solutions for fitting the model to a given set of input data, which are substantially different and roughly equal in their “goodness of fit”. This presents a problem for normal statistical learning techniques, as maximization-based techniques are only guaranteed to find local optima, and sampling techniques require an impractical number of iterations to fully explore such a parameter space.

I present here a heuristic framework that combines several statistical learning approaches to produce a good fit of the network model to the input data, with partial or local estimates for key variables of interest. In short, a local estimation algorithm is run in 3 phases, described in greater detail in Appendix A.2. In the first phase, a fixed starting network is selected and other model parameters are fit to this network using a closed form solution. In the second phase, all model parameters,
including network edges, are iteratively maximized until sufficient convergence at a local optimum. In the third phase, model parameters of interest are iteratively resampled a fixed number of times to provide a more robust estimate around the local optimum. Marginalized posterior distributions for each parameter of interest are derived in Appendix A.3.

In order to robustly estimate the posterior probabilities, the local estimation algorithm should be run multiple times. While the results of individual runs may be of interest, there is a need to summarize across multiple results in order to score each network edge in terms of how likely it is overall given the results of multiple runs. In this work, I summarize multiple runs by averaging together the expected or fixed value of each parameter in $\Theta$ and averaging together the estimated posterior probabilities for each edge variable in $\{C, D\}$ (see Appendix A.3).

The averaging method is only valid under the assumption that all individual result sets are similar, i.e., that each run is an estimate around the same mode of the solution space. To assess the validity of this assumption, I performed hierarchical clustering [82] of the individual network models returned by each run of the algorithm. I used the vector of posterior probabilities assigned to all edge variables $(C, D)$ to represent each model and then compute the Euclidean distance between all pairs of models. From this distance matrix, I performed complete hierarchical clustering as implemented in R [66] and visually inspect the resulting tree. If the overall tree structure does not show any clearly delineated clusters, then the computed network models most likely represent estimates of the same solution mode and can be averaged together for a more robust estimate. The clustering results in the yeast application exemplify this criteria (see Section 3.5).

Conversely, if the tree shows several well-separated clusters, then the computed network models were most likely sampled from multiple divergent solution modes.
The model is generally over-parameterized relative to the available data, and it is therefore possible, but not necessarily incorrect, for subsequent runs to return parameters that encompass substantially different networks. It is possible that the underlying or “true” network that produced the input data cannot be completely captured by this model, e.g., due to a mixture of cell types or genetic backgrounds with different underlying networks included in the input data\(^1\). In this case, each mode may capture a different view of the underlying network. At this time I do not have a solution for summarizing over multiple modes, and prefer instead to analyze the biological significance of each cluster individually. Whenever the dendrogram indicated multiple modes, I selected an appropriate height cut-off for clustering the models, and then averaged together and performed biological inference separately within each cluster. The clustering results from the human B cell application in Chapter 4 appear to be multi-modal based on this criteria (see Figure 4.2).

### 3.3 Data Sources and Pre-Processing

I initially applied MoNsTER to simulated data and a well-studied network in *S. cerevisiae* in order to demonstrate the accuracy of the model and estimation method. In this section, I describe all procedures used to prepare input data for these applications.

In general, when defining the expression matrices, the genes used as TFs and modifiers for a particular model application are always completely excluded from the set of target genes. This is necessary because regulator expression data is used as a proxy for regulatory activity, and attempting to simultaneously model their tran-

---

\(^1\)The phenomenon of multi-modality can also occur due to violations of model assumptions, i.e. a TF acting as activator for one set of genes and repressor for another. Thus, considering multiple modes in the parameter space allows us to overcome these model limitations.
scriptional regulation would more likely result in false positive connections. However, this limitation could be overcome if both protein-level and transcript-level concentration measurements were available for TFs and modifiers in all conditions.

In order to derive PWM-based priors of any kind, the PWM is scanned along a DNA or protein sequence to compute a score at each possible PWM-sequence alignment. Each score can be converted to a p-value by comparing it to a distribution of scores computed for some set of background sequences, e.g., random genomic or protein sequences. This procedure is described in more detail in ref. [124]. A conversion from p-values to prior probabilities was first introduced in ref. [22], and is reviewed or adapted here where applicable. Also, as noted in ref. [22], all priors (PWM-based or otherwise), must be “trimmed” so that they do not include extreme values of 0 or 1, as these values prevent further estimation of posteriors by the model. I limited the range of all priors to (0.05, 0.95) for all applications presented in this work.

Simulated Data

Simulated data was primarily generated to test the accuracy of the model-fitting method described in Section 3.2, and is not intended to demonstrate the biological relevance of the underlying model or the simulated network properties. All input data matrices in \( \Xi \) were initially simulated for a network containing \( N = 200 \) target genes, \( J = 10 \) TFs, and \( K = 100 \) modifiers measured under \( T = 100 \) conditions. This is a network of sufficient size to reliably assess the accuracy of estimated model parameters without requiring substantial computing time. The expression values for all TFs and modifiers in all conditions (\( f \) and \( h \)) were randomly sampled from the normal distribution \( N(0, 1) \). The edge indicator variables \( C_{ij} \) and \( D_{jk} \) were selected
at random with the following constraints:

- Each TF regulates 50–100 genes: \( 50 \leq \left\{ \sum_{i=1}^{N} C_{ij} \right\} \leq 100 \ \forall j \)
- Each TF is targeted by 1–20 modifiers: \( 1 \leq \left\{ \sum_{k=1}^{K} D_{jk} \right\} \leq 20 \ \forall j \)
- Each modifier targets 1–3 TFs: \( 1 \leq \left\{ \sum_{j=1}^{J} D_{jk} \right\} \leq 3 \ \forall k \)

Each parameter \( \beta_j \) was randomly sampled from the normal distribution \( N(0, 0.3) \), but was resampled whenever \( |\beta_j| < 0.05 \). Each parameter \( \gamma_{jk} \) for which the corresponding \( D_{jk} = 1 \) was randomly sampled following the same procedure used for the \( \beta_j \) variables. The prior matrices \( \{b, m, a, s\} \) were all randomly sampled using the standard Beta distribution to generate priors that are correlated to the intended network, but also contain a substantial amount of noise. Each prior value was sampled as follows:

\[
\begin{align*}
 b_{ij}, m_{ij} &\sim B(z, 1) \quad \forall (i, j) : C_{ij} = 1 \\
 b_{ij}, m_{ij} &\sim B(1, z) \quad \forall (i, j) : C_{ij} = 0 \\
 a_{jk}, s_{jk} &\sim B(z, 1) \quad \forall (j, k) : D_{jk} = 1 \\
 a_{jk}, s_{jk} &\sim B(1, z) \quad \forall (j, k) : D_{jk} = 0
\end{align*}
\]

Note that in the equations above, ‘B’ denotes a standard probability distribution (the “Beta” distribution) and should not be confused with the prior matrix \( b \) or the model parameters \( \beta \). The variable \( z \) is randomly sampled from the uniform distribution \( U(1.2, 1.5) \) independently for each column of each matrix, in order to vary the relative quality of priors for each TF and each modifier.

The expression values for all target genes in all conditions (matrix \( g \)) were computed according to Eq 3.1 with all residual error terms \( \varepsilon_{it} \) initially set to 0. Addi-
tional versions of $g$ were calculated with residual errors randomly sampled according to Eq 3.2 using increasing values of $\sigma^2 = 0.05, 0.1, \ldots, 1$, denoted $g_{\sigma^2=0.05}$, etc.

For subsequent simulations, the input data was expanded to include other types of noise, in the form of either additional, uninformative target genes, or additional, uninformative conditions. In both cases, the matrix $g_{\sigma^2=1}$ was used as a starting point, so these subsequent simulations also include a considerable amount of noise in the informative target genes and conditions.

To add additional, unregulated target genes, the model was expanded to include $N = N_0 + N'$ target genes, where $N_0 = 200$ for the original, regulated target genes, and $N' = 20, 40, \ldots, 200$ for the additional, unregulated genes. All unregulated expression values were randomly sampled from $N(0, 1)$. In other words, each additional gene profile was completely random and unregulated by the TFs and modifiers in the network.

To add additional, uninformative conditions, the simulated data was expanded to include $T = T_0 + T'$ conditions, with $T_0 = 100$ for the original, informative conditions, and $T' = 10, 20, \ldots, 100$ for the additional, uninformative conditions. All expression values in the uninformative conditions were sampled from the distribution $N(0, 1)$. In other words, each additional condition contained completely random expression values, with no information related to network structure.

**Yeast Network Input Data**

I also applied MoNsTEr to a stress-response network in *S. cerevisiae* based around a large number of MFG triplets catalogued in PTM-Switchboard (Chapter 2). The network model encompasses $N = 80$ target genes, $J = 2$ TFs (MSN2 and MSN4), and $K = 81$ modifiers (kinases). The data sources used for this application are
Table 3.3: Data sources used as input for MoNsTEr application to yeast stress response network, organized by model variable. All relevant literature citations are shown in the “Source” column.

Yeast Expression Compendium

I derived the expression matrices g, f, and h from an expression compendium, previously compiled and normalized by Chen et al. [22]. This compendium includes T = 314 samples compiled from 18 studies covering a diverse set of conditions such as cellular stresses and cell cycle phases. The matrix f contains expression profiles for the TFs MSN2 (YMR037C) and MSN4 (YKL062W). Matrix h contains the available expression profiles for 81 kinases. Table 3.4 lists the known target genes, decoy target genes, and kinases included in the network model. The matrix g contains: (i) expression profiles for 40 known targets of MSN2/4, compiled in PTM-Switchboard (Chapter 2), and (ii) 40 additional decoy target genes selected at random from the remaining expression data, ensuring that no decoy gene had any Gene Ontology (GO) [4] annotation suggesting involvement in stress response.

All rows (expression profiles) are further normalized to have mean 0 and variance 1, by applying the z-score transformation, for example:

$$g_{it} = \frac{g_{it}^{(raw)} - \mu_i^{(raw)}}{\sigma_i^{(raw)}} \quad \forall i, t$$

(3.7)

Similar transformations are also applied to expression profiles in matrices f and
**Known MSN2/4 Target Genes**

<table>
<thead>
<tr>
<th>Gene Symbol</th>
</tr>
</thead>
<tbody>
<tr>
<td>APJ1</td>
</tr>
<tr>
<td>ECM4</td>
</tr>
<tr>
<td>GRX2</td>
</tr>
<tr>
<td>HSP42</td>
</tr>
<tr>
<td>NTH1</td>
</tr>
<tr>
<td>RAS2</td>
</tr>
<tr>
<td>TPS3</td>
</tr>
<tr>
<td>YGL036W</td>
</tr>
<tr>
<td>CPR1</td>
</tr>
<tr>
<td>GAC1</td>
</tr>
<tr>
<td>GSY2</td>
</tr>
<tr>
<td>HSP78</td>
</tr>
<tr>
<td>PIL1</td>
</tr>
<tr>
<td>SPS100</td>
</tr>
<tr>
<td>TRX2</td>
</tr>
<tr>
<td>YML131W</td>
</tr>
<tr>
<td>CTT1</td>
</tr>
<tr>
<td>GLK1</td>
</tr>
<tr>
<td>HSP104</td>
</tr>
<tr>
<td>HXK1</td>
</tr>
<tr>
<td>PNC1</td>
</tr>
<tr>
<td>SSA4</td>
</tr>
<tr>
<td>TSA2</td>
</tr>
<tr>
<td>YMR090W</td>
</tr>
<tr>
<td>CYC7</td>
</tr>
<tr>
<td>GPH1</td>
</tr>
<tr>
<td>HSP12</td>
</tr>
<tr>
<td>LAP4</td>
</tr>
<tr>
<td>PRX1</td>
</tr>
<tr>
<td>TPS1</td>
</tr>
<tr>
<td>TSL1</td>
</tr>
<tr>
<td>YMR315W</td>
</tr>
<tr>
<td>DDR48</td>
</tr>
<tr>
<td>GRE2</td>
</tr>
<tr>
<td>HSP26</td>
</tr>
<tr>
<td>MDJ1</td>
</tr>
<tr>
<td>PTP2</td>
</tr>
<tr>
<td>TPS2</td>
</tr>
<tr>
<td>YDL124W</td>
</tr>
<tr>
<td>YNL134C</td>
</tr>
</tbody>
</table>

**Decoy Target Genes**

<table>
<thead>
<tr>
<th>Gene Symbol</th>
</tr>
</thead>
<tbody>
<tr>
<td>AAT2</td>
</tr>
<tr>
<td>CUE1</td>
</tr>
<tr>
<td>ILV1</td>
</tr>
<tr>
<td>LYS9</td>
</tr>
<tr>
<td>PIB1</td>
</tr>
<tr>
<td>RIX7</td>
</tr>
<tr>
<td>RPS6B</td>
</tr>
<tr>
<td>VPS41</td>
</tr>
<tr>
<td>ASK1</td>
</tr>
<tr>
<td>EDC2</td>
</tr>
<tr>
<td>IML2</td>
</tr>
<tr>
<td>MEF1</td>
</tr>
<tr>
<td>PTM1</td>
</tr>
<tr>
<td>RPN1</td>
</tr>
<tr>
<td>RPT3</td>
</tr>
<tr>
<td>VPS62</td>
</tr>
<tr>
<td>CAR1</td>
</tr>
<tr>
<td>EMI5</td>
</tr>
<tr>
<td>KAR4</td>
</tr>
<tr>
<td>MPP10</td>
</tr>
<tr>
<td>PUS7</td>
</tr>
<tr>
<td>RPS14B</td>
</tr>
<tr>
<td>SMY2</td>
</tr>
<tr>
<td>VRG4</td>
</tr>
<tr>
<td>CCT4</td>
</tr>
<tr>
<td>FMP52</td>
</tr>
<tr>
<td>LAG1</td>
</tr>
<tr>
<td>NAT1</td>
</tr>
<tr>
<td>RCR2</td>
</tr>
<tr>
<td>RPS29B</td>
</tr>
<tr>
<td>THI20</td>
</tr>
<tr>
<td>YLF2</td>
</tr>
<tr>
<td>CTF3</td>
</tr>
<tr>
<td>HXT5</td>
</tr>
<tr>
<td>LEA1</td>
</tr>
<tr>
<td>NIP100</td>
</tr>
<tr>
<td>RIM4</td>
</tr>
<tr>
<td>RPS4A</td>
</tr>
<tr>
<td>TRM10</td>
</tr>
<tr>
<td>YLH47</td>
</tr>
</tbody>
</table>

**Kinase Genes**

<table>
<thead>
<tr>
<th>Gene Symbol</th>
</tr>
</thead>
<tbody>
<tr>
<td>AKL1</td>
</tr>
<tr>
<td>CTK1</td>
</tr>
<tr>
<td>HSL1</td>
</tr>
<tr>
<td>KIN82</td>
</tr>
<tr>
<td>PKH2</td>
</tr>
<tr>
<td>RIM15</td>
</tr>
<tr>
<td>STE11</td>
</tr>
<tr>
<td>YCK2</td>
</tr>
<tr>
<td>ARK1</td>
</tr>
<tr>
<td>DBF2</td>
</tr>
<tr>
<td>IKS1</td>
</tr>
<tr>
<td>KNS1</td>
</tr>
<tr>
<td>PKH3</td>
</tr>
<tr>
<td>SAK1</td>
</tr>
<tr>
<td>STE20</td>
</tr>
<tr>
<td>YCK3</td>
</tr>
<tr>
<td>ATG1</td>
</tr>
<tr>
<td>DUN1</td>
</tr>
<tr>
<td>IME2</td>
</tr>
<tr>
<td>KSP1</td>
</tr>
<tr>
<td>PKP2</td>
</tr>
<tr>
<td>SAT4</td>
</tr>
<tr>
<td>SWE1</td>
</tr>
<tr>
<td>YGK3</td>
</tr>
<tr>
<td>BCK1</td>
</tr>
<tr>
<td>ELM1</td>
</tr>
<tr>
<td>IPL1</td>
</tr>
<tr>
<td>KSS1</td>
</tr>
<tr>
<td>PRK1</td>
</tr>
<tr>
<td>SKM1</td>
</tr>
<tr>
<td>TDA1</td>
</tr>
<tr>
<td>YPL150W</td>
</tr>
<tr>
<td>CDC15</td>
</tr>
<tr>
<td>FMP48</td>
</tr>
<tr>
<td>IRE1</td>
</tr>
<tr>
<td>MCK1</td>
</tr>
<tr>
<td>PRR1</td>
</tr>
<tr>
<td>SKS1</td>
</tr>
<tr>
<td>TOS3</td>
</tr>
<tr>
<td>CDC28</td>
</tr>
<tr>
<td>FRK1</td>
</tr>
<tr>
<td>KCC4</td>
</tr>
<tr>
<td>MEK1</td>
</tr>
<tr>
<td>PSK1</td>
</tr>
<tr>
<td>SKY1</td>
</tr>
<tr>
<td>TPK1</td>
</tr>
<tr>
<td>CDC5</td>
</tr>
<tr>
<td>FUS3</td>
</tr>
<tr>
<td>KIN1</td>
</tr>
<tr>
<td>MKK1</td>
</tr>
<tr>
<td>PTK2</td>
</tr>
<tr>
<td>SLT2</td>
</tr>
<tr>
<td>TPK2</td>
</tr>
<tr>
<td>CKA1</td>
</tr>
<tr>
<td>GIN4</td>
</tr>
<tr>
<td>KIN2</td>
</tr>
<tr>
<td>MPS1</td>
</tr>
<tr>
<td>RAD53</td>
</tr>
<tr>
<td>SNF1</td>
</tr>
<tr>
<td>TPK3</td>
</tr>
<tr>
<td>CLA4</td>
</tr>
<tr>
<td>HAL5</td>
</tr>
<tr>
<td>KIN28</td>
</tr>
<tr>
<td>NNK1</td>
</tr>
<tr>
<td>RCK1</td>
</tr>
<tr>
<td>SPS1</td>
</tr>
<tr>
<td>VHS1</td>
</tr>
<tr>
<td>CMK1</td>
</tr>
<tr>
<td>HRK1</td>
</tr>
<tr>
<td>KIN3</td>
</tr>
<tr>
<td>PBS2</td>
</tr>
<tr>
<td>RCK2</td>
</tr>
<tr>
<td>SSK22</td>
</tr>
<tr>
<td>YAK1</td>
</tr>
<tr>
<td>CMK2</td>
</tr>
<tr>
<td>HRR25</td>
</tr>
<tr>
<td>KIN4</td>
</tr>
<tr>
<td>PHO85</td>
</tr>
<tr>
<td>RIM11</td>
</tr>
<tr>
<td>SSN3</td>
</tr>
<tr>
<td>YCK1</td>
</tr>
</tbody>
</table>

Table 3.4: All target genes and kinases used as input for yeast **MoNsTEr** application, listed by SGD gene symbol.

**h.** The z-score normalization strengthens the model assumption that residual error terms follow a normal distribution with 0 mean and uniform model-wide error variance (Eq 3.2).

**MSN2/4-Target Gene Priors from PWMs**

In the yeast application, I derived a single matrix \( m \) for TF-Gene priors, using previously published PWMs for MSN2 and MSN4 [81]. For each TF-Gene pair \((i, j)\),
I scanned both strands of the 700bp upstream region for each gene \( i \), and recorded the best (maximum) PWM score, denoted as \( S_{ij} \). I then scanned the same PWMs against random genomic sequences to generate a background score distribution for each PWM, thereby allowing conversion of each score \( S_{ij} \) to a p-value \( P_{ij} \), using the method in ref. [124]. I then converted each p-value to a prior probability following the same equation used in ref. [22]:

\[
m_{ij} = (1 - P_{ij})^{2(\ell - W_j + 1)}
\]

In this case, \( \ell = 700 \) is the fixed promoter length, and \( W_j \) is the width (in bases) of the PWM corresponding to TF \( j \).

**Kinase-Substrate Priors from PWMs**

I generated a single matrix \( s \) of priors for TF-Kinase connectivity by scanning each TF protein sequence with a PWM describing the predicted substrate specificity of each kinase. I first acquired the protein sequence of each known kinase present in the expression compendium from SGD\(^2\) [58]. I then submitted each sequence to the Predikin v2.0 web server\(^3\) [171].

Briefly, Predikin takes a kinase protein sequence as input, and constructs a predicted substrate profile, as a PWM, from the known substrates of related kinases with high accuracy (see ref. [171] for further details). Thus, for each kinase protein sequence, Predikin returns a predicted PWM describing the most likely protein sequences to be targeted by that kinase. The primary advantage of Predikin over similar tools is its ability to generate a PWM for a wider range of kinases, although the Predikin server failed to predict a PWM for a few kinases, and these were

---

\(^2\)Accessed at: [http://www.yeastgenome.org](http://www.yeastgenome.org) on 7/28/08

\(^3\)Accessed at: [http://predikin.biosci.uq.edu.au](http://predikin.biosci.uq.edu.au) on 8/6/08
discarded from the input set (not included in $K = 81$).

Next, I obtained the protein sequences for the TFs MSN2 and MSN4, also downloaded from SGD$^2$. I scanned the PWM for each kinase $k$ against the protein sequence for each TF $j$, and recorded the maximum PWM score, denoted as $S_{jk}$. I also generated and scanned randomized protein sequences to create an appropriate background score distribution for each kinase $k$. For each $k$, I used the background distribution to transform all corresponding scores $S_{jk}$ to p-values, denoted $P_{jk}$. These p-values were then converted to prior probabilities, adapting Eq 3.8 above:

$$s_{jk} = (1 - P_{jk})^{\ell_j - W_k + 1}$$

In this case, $\ell_j$ is the length of the protein sequence for TF $j$, and $W_k = 7$ for all kinase PWMs generated by Predikin. Ultimately, these priors represent a probabilistic estimate of the biochemical constraints on the TF-Modifier network edges.

### 3.4 Analysis of Simulated Data

MoNsTEr was first applied to simulated input data to estimate the accuracy of the model-fitting method described in Section 3.2. Initially, data was simulated from a network of $K = 100$ modifiers, $J = 10$ TFs, and $N = 200$ target genes over $T = 100$ conditions, with noisy priors but no noise in the expression data ($\sigma^2 = 0$) as described in Section 3.3. The model-fitting algorithm was run for 10 replicates, and each run was initialized by randomly selecting each network edge according to the geometric mean of the corresponding priors. This initialization procedure is fairly stochastic, representing wide coverage of the possible solution space, and is suitable for smaller networks. The results from all 10 runs of the model-fitting
algorithm were then averaged together to compute posterior probabilities for the edge parameters $C$ and $D$.

To visualize the performance of the model-fitting method, I plotted a Receiver Operating Characteristic (ROC) curve for the posterior probabilities $C$ and $D$ compared to the known network structure used to simulate the data (Figure 3.2A,B). For comparison ROC curves are also shown for each individual type of prior probability, and for TF-Gene expression correlations. The performance observed on the ROC curves can be quantified by computing the Area Under the Curve (AUC) metric. This metric indicates that MoNsTER perfectly recovered the TF-Gene edges (AUC = 1, solid blue line in Figure 3.2A). I evaluated the relative advantage of the full model, as compared with using either the priors $b$ or $m$ alone, or expression correlations between TFs and target genes, to predict the TF-Gene edges. These alternatives all performed with lower accuracy than the full model, resulting in AUC values < 0.75 (dotted and dashed blue lines in Figure 3.2). TF-Modifier edges were substantially harder to infer, owing to their indirect effect in the expression model (AUC = 0.78, solid green line in Figure 3.2B). However, MoNsTER predicted these edges with better accuracy than either simulated prior type alone (AUCs < 0.7, Figure 3.2B). These results demonstrate that the observed performance of MoNsTER cannot be reproduced by simpler analyses of the individual data sources.

Simulated Sources of Noise

To further assess the effect of various types of noise on the accuracy of MoNsTER, three possible sources of noise were simulated: (i) individual expression errors $\varepsilon_{it}$, (ii) the inclusion of additional genes unconnected to the network, and (iii) the inclusion of additional, uninformative expression samples (see Section 3.3). Also note
Figure 3.2: Accuracy of MoNsTEr using simulated data, measured separately for posterior probabilities of TF-Gene connectivity (solid blue lines) and TF-Modifier connectivity (solid green lines), by the area under the ROC curve (AUC) for each network model. The accuracy of network priors (dotted blue and green lines) and TF-Gene expression correlation (dashed blue lines) are shown for comparison. A. ROC curves for TF-Gene edges with no expression noise. B. ROC curves for TF-Modifier edges with no expression noise. C. Accuracy (AUC) for increasing model-wide variance of gene expression residual errors. D. Accuracy (AUC) for increasing percentage of uninformative genes as compared to the number of informative genes. E. Accuracy (AUC) for increasing percentage of uninformative expression samples as compared to the number of informative samples.

that the priors are consistently noisy throughout this analysis. For each set of noisy input data, the model-fitting algorithm was run 30 times, using the same stochastic initialization procedure as the previous analysis, and the model estimates were averaged within groups of 10 runs, in order to also assess the variability of algorithm performance. For each group of 10 runs, the average posterior probabilities for C and D were used to recompute the AUC as in the previous analysis above. This resulted in 3 AUC values for each analysis (1 for each group of 10 runs), which were used to plot median performance as a function of different noise sources. “I-
bars” are also drawn, spanning the min and max AUC values at each noise level (Figure 3.2C-E).

Noise within the gene expression values was simulated by gradually increasing the model-wide variance $\sigma^2$ from 0 to 1 (matrices $g_{\sigma^2=0}$ thru $g_{\sigma^2=1}$ from Section 3.3). Figure 3.2C shows that algorithm performance is unaffected by noise in the individual expression values, up to a degree of noise equivalent to the expression variance for each regulator (TF or modifier). Note that this limit is essentially guaranteed in other applications by the z-score transformation applied to real expression data (Eq 3.7).

For subsequent analyses, the value of $\sigma^2 = 1$ was kept constant, and further noise was added in the form of either additional gene profiles, or additional expression conditions, neither of which contain any information relevant to the original network. The motivation for this analysis is that real expression compendia will contain target genes that are not targeted by any of the TFs of interest, and may also contain expression samples in which the underlying network is disrupted due to non-model elements, e.g., samples from a different cell-type. As described in Section 3.3, I added either $N'$ uninformative genes, or $T'$ uninformative conditions. Figure 3.2D shows the results for increasing numbers of unregulated genes, represented as a percentage of the number of regulated genes ($N_0 = 200$). Again, performance is essentially constant. Similarly, Figure 3.2E shows the results for increasing numbers of uninformative samples, plotted as a percentage of the number of informative samples ($T_0 = 100$). In this case, the inference of TF-Modifier connections were negatively affected by this type of noise, but still out-performed individual prior sources. Based on these detailed simulation studies, I conclude that the model-fitting method can reconstruct an underlying network model despite substantial noise, and with greater accuracy than the individual input data sources.
3.5 Analysis of Yeast MSN2/4 Network

To assess the utility of this model for biological inference from experimental data, I applied MoNsTER to a well-studied stress response network in *S. cerevisiae*. I chose a network based around the TFs MSN2 and MSN4 in order to validate against existing knowledge of target genes [141] and upstream signaling components [137, 178] for these two TFs, previously cataloged in PTM-Switchboard (Chapter 2). MSN2 and MSN4 function redundantly to regulate a core transcriptional response to most cellular stresses [21, 64]. Both TFs are regulated primarily at the post-translational level, with various kinases controlling their nuclear localization [137, 178]. The yeast input data set was prepared from the sources listed in Table 3.3, and includes data for TFs MSN2/4, all yeast kinases for which sufficient data was available, 40 known MSN2/4 target genes, and 40 decoy target genes (see Section 3.3).

The MoNsTER model-fitting algorithm was run 100 times to fit all model parameters to the network of $N = 80$ target genes, $J = 2$ TFs, and $K = 81$ modifiers. In each run, the algorithm was seeded using the same stochastic method as for the simulated data analysis (see Section 3.4). A substantially larger number of runs were used here as compared to the simulated data because fewer separate sets of input data were analyzed, and therefore more run time could be dedicated to each set of input data. There was also a clear increase in accuracy using a larger number of runs in this case (data not shown). The 100 individual network model estimates were hierarchically clustered as described in Section 3.2 (Figure 3.3A). In the observed dendrogram, the clusters are not well-separated (most of the tree height is within these clusters). This meets the criteria for a single solution mode, although there is clearly some variability between individual model estimates. Therefore, the parameter estimates for all 100 replicate models were averaged together to produce
Figure 3.3: Hierarchical clustering of individual yeast models from 100 runs of the model-fitting algorithm for: (A) the full model and (B) the TF-only model. In both model sets, there are no well-separated clusters to indicate distinct solution modes.

A single estimate of model parameter values and posterior probabilities for C and D. This entire model-fitting procedure was also repeated for a network model containing only TFs and target genes, but no modifiers (see below), with hierarchical clustering results shown in Figure 3.3B. Although the scale of this dendrogram is altered due to the smaller number of edges in the TF-only model, there is a similar lack of separation between clusters, and again all 100 model estimates were averaged together.

Accurate Prediction of MSN2/4 Modifiers and Target Genes

I first assessed the accuracy with which MoNsTEr predicted known target genes for MSN2/4. A ROC curve measuring the overall separation of true and decoy target genes was constructed using the posterior edge probabilities estimated for C. For comparison, additional ROC curves were constructed using the PWM-based prior
Figure 3.4: ROC curves of MSN2/4 target gene selection in yeast network models. Curves are drawn by computing the sensitivity and specificity from known and decoy target genes at all possible thresholds for: (i) the magnitude of correlation between TF and target gene expression profiles (dashed blue lines, AUC = 0.63); (ii) the prior probabilities $m$ derived from PWMs (dotted blue lines, AUC = 0.73); (iii) the posterior probabilities from MoNsTER with no kinases (solid blue lines, AUC = 0.82); (iv) the posterior probabilities from MoNsTER with TF-kinase synergy terms (solid green lines, AUC = 0.87). The difference between the AUC values for the MoNsTER models with and without kinases is significant with p-value = 0.0018 based on bootstrap tests.

probabilities $m$ and the TF-Gene expression correlations as possible selection criteria (Figure 3.4). The posterior edge probabilities estimated by the model had the highest AUC (0.87), indicating the best separation of true and false TF-Gene edges. In particular, there was a substantial gain in sensitivity at low false positive rates appropriate for subsequent experimental design or network analysis. For example, at a fixed false positive rate of 10%, MoNsTER selected TF-Gene edges with a true positive rate of 77.5%, while the PWM priors and expression correlation alone had true positive rates of only 37.5% and 18.75% respectively.

I next assessed the accuracy with which MoNsTER identified known modifiers of TF activity. MSN2/4 are regulated in response to a wide range of cellular conditions
Table 3.5: Kinases known to regulate MSN2/4 with corresponding results from MoNaTEr applied to yeast input data. MoNaTEr results are reported for the TF with the highest posterior of connectivity to each kinase. Results include the posterior probability \( P(D_{jk} = 1) \), the associated influence parameter \( \gamma_{jk} \), and the rank of the posterior among all kinase connections to the TF. For comparison, the prior probability and associated rank are shown in the right-most columns.

<table>
<thead>
<tr>
<th>Known Kinases</th>
<th>MoNaTEr Results</th>
<th>Prior</th>
</tr>
</thead>
<tbody>
<tr>
<td>Symbol</td>
<td>Ref.</td>
<td>Role</td>
</tr>
<tr>
<td>TPK1</td>
<td>[63, 72, 178]</td>
<td>-</td>
</tr>
<tr>
<td>TPK2</td>
<td>[63, 72, 178]</td>
<td>-</td>
</tr>
<tr>
<td>TPK3</td>
<td>[63, 72, 178]</td>
<td>-</td>
</tr>
<tr>
<td>SNF1</td>
<td>[137]</td>
<td>-</td>
</tr>
<tr>
<td>RIM11</td>
<td>[84]</td>
<td>+</td>
</tr>
<tr>
<td>SSN3</td>
<td>[26]</td>
<td>-</td>
</tr>
<tr>
<td>YAK1</td>
<td>[121]</td>
<td>+</td>
</tr>
</tbody>
</table>

[64, 137], and many of the kinases in the input data are likely to have some indirect effect on MSN2/4 activity. However, there is a small set of kinases in the input data that have experimental evidence for direct regulation of MSN2/4 proteins, specifically PKA [63, 72, 178], SNF1 [137], RIM11/GSK3 [84], SSN3 [26], and YAK1 [121] (Table 3.5). MoNaTEr estimated significantly higher posterior probabilities \( D \) and influence parameters \( \gamma \) for these known kinases, as compared to all other input kinases (Mann-Whitney test p-value = 0.007 for \( D \), p-value = 0.047 for |\( \gamma \)|). By comparison, prior probabilities \( s_{jk} \) alone do not significantly favor the known kinases modifying MSN2/4 activity. Furthermore, the ranks of known kinases in the MoNaTEr network are significantly higher than those based on prior probabilities \( s_{jk} \) (Table 3.5, “Rank” columns, Mann-Whitney test p-value = 0.047).

To specifically assess the contribution of TF-Modifier synergy terms (Eq 3.3) to target gene expression prediction, I repeated the model fitting procedure on a network containing only the \( N = 80 \) target genes and \( J = 2 \) TFs, but no kinases (\( K = 0 \)). I found that the TF-only model had a lower AUC than the full model.
(see Figure 3.4), and a lower sensitivity (37%, interpolated from the ROC curve) at a fixed 10% false positive rate.

Significance Analysis of ROC Curves

In order to assess the significance of the ROC curves in Figure 3.4, I performed additional statistical analyses. *Permutation tests* [54] were performed for each individual ROC curve, to determine the significance of the computed AUC values. In this test, the edge discriminants, e.g., the posterior estimates $C$, are randomly permuted relative to the known true/false labels on these same edges. The AUC value is recomputed for each permutation, and the p-value is the proportion of permutations in which the randomized AUC value is higher than the originally observed AUC value. I performed 10,000 permutations for the AUC values corresponding to the following discriminants: full model posteriors $C^{(+)}$, TF-only model posteriors $C^{(−)}$, PWM-based priors $m$, and the magnitude of expression correlation between $g$ and $f$. The resulting p-values are summarized in Table 3.6 (3rd column). All AUC values were significant according to this test, indicating that each discriminant performs significantly better than random selection of edges. In other words, none of the AUC values in Table 3.6 are expected to occur by chance.

<table>
<thead>
<tr>
<th>Discriminant</th>
<th>AUC</th>
<th>Permutation</th>
<th>P-value</th>
<th>95% Conf. Int.</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C^{(+)}$</td>
<td>0.87</td>
<td>$&lt; 10^{-5}$</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>$C^{(−)}$</td>
<td>0.82</td>
<td>$&lt; 10^{-5}$</td>
<td>0.0018</td>
<td>(0.015, 0.095)</td>
</tr>
<tr>
<td>$m$</td>
<td>0.73</td>
<td>$&lt; 10^{-5}$</td>
<td>0.00458</td>
<td>(0.035, 0.25)</td>
</tr>
<tr>
<td>$</td>
<td>\text{cor}(g,f)</td>
<td>$</td>
<td>0.63</td>
<td>0.00229</td>
</tr>
</tbody>
</table>

Table 3.6: Significance of yeast application AUC values.
full model posteriors $C^{(+)}$ as compared to other observed AUC values, I performed a bootstrap analysis [54], as follows. For each of 10,000 bootstrap samples, I randomly sampled 160 TF-Gene edges from the actual network with replacement and with all assigned posteriors, priors, etc. Thus, for each individual bootstrap sample, some edges may be ignored, and other edges may count multiple times, although the overall sample size remains constant. If the observed improvement in AUC is based on a small percentage of the overall network, then this apparent improvement should disappear in many of the bootstrap samples. Thus, the p-value based on bootstrap analysis is the proportion of bootstrap samples in which the AUC for the full model, $C^{(+)}$, is no longer greater than another comparative AUC value (e.g., $C^{(-)}$, the posteriors from the TF-only model). These p-values are reported for each of the AUC values as compared to the full model AUC (Table 3.6, 4th column). For each bootstrap sample, I also computed the exact difference in AUC values between $C^{(+)}$ and each other discriminant, and summarized these values as a 95% confidence interval (Table 3.6, 5th column). All bootstrap p-values were significant, and all differences in AUC had positive confidence intervals, indicating that the apparent improvement in accuracy observed in the full model compared to all other tested discriminants is unlikely to occur by chance.

Thus, in predicting the TF-Gene network, MoNsTEr provides a significant improvement over each individual data source used as input (bootstrap p-value = 0.00458). Furthermore, inclusion of kinases in the model resulted in a significant improvement to the overall accuracy of MSN2/4 target selection (Figure 3.4, solid lines, bootstrap p-value = 0.0018). Overall, the inclusion of modifiers in MoNsTEr improves the model of MSN2/4 regulation of target genes, and enables the prediction of upstream components for the transcriptional program.
3.6 Discussion

In this chapter, I have presented a novel model-based method to infer regulatory networks, MoNsTEr, that simultaneously predicts TF-Gene and TF-Modifier interactions by integrating heterogeneous data types in a probabilistic framework. Simulation studies show that MoNsTEr is robust against a variety of noise sources. I have further demonstrated the validity and usefulness of this method through an application to the stress response network in *S. cerevisiae* mediated by MSN2/4.

**Development of the Heuristic Model-Fitting Framework**

A fundamental challenge in computational biology is the construction of models that are both biologically comprehensive and computationally tractable. The TF-Modifier interactions capture a critical and often ignored aspect of transcriptional regulatory networks, but also render the model highly parameterized. As a result, the exploration of possible models presents a computational and statistical challenge.

Chen *et al.* used a similar—but simpler—model [22] and were able to estimate the joint posterior distribution over all parameters using standard Gibbs Sampling [65]. The extended model presented in this chapter is substantially more complex due to the additional network layer of TF-Modifier connectivity, and tends to result in multi-modal distributions not easily traversed by the Markov Chain approach used in Gibbs Sampling. As evidence of this, I initially attempted to estimate the posterior of the MoNsTEr model using Gibbs Sampling. Reliable estimation using Gibbs Sampling requires the sampling algorithm to reach convergence, which is typically assessed by running several parallel runs (called *chains*) of the sampling algorithm, and measuring a convergence statistic, $\hat{R}$, based on observed parameter values within and between chains [18]. It is not valid to begin drawing samples from
the Markov Chains until they have converged, indicated by an $\hat{R}$ value close to 1 for all parameters. I was unable to reach a satisfactory value of $\hat{R}$ for many parameters after a substantial number of iterations—in some cases greater than 50,000—even for a small network (data not shown). On the contrary, attempting to fit a model with only TF regulators (no modifiers) via Gibbs Sampling converged in less than 5,000 iterations. Thus, I concluded that the convergence problems were specific to the extended model.

As a practical alternative, I fit the model parameters using a heuristic that identifies a sufficiently good solution for biological inference in the cases studied here. In general terms, MoNsTEr leverages the prior knowledge of network structure, and combines the merits of both maximization and sampling approaches. The accuracy of MoNsTEr is improved by combining multiple independent runs of the model-fitting procedure, given that each run estimates a similar network model, as demonstrated by the results in this chapter.

**Interpreting the TF-Modifier Influence Parameter**

In addition to assigning a posterior probability for each TF-Modifier interaction, I also estimate an influence parameter $\gamma_{jk}$. The magnitude of this parameter can be used as an additional filter for the inference of TF-Modifier interactions that affect gene transcription. In the yeast application, the magnitudes of $\gamma_{jk}$ parameter for the known MSN2/4-regulating kinases were all greater than 0.05, and were significantly larger compared to the other kinases (Table 3.5, p-value = 0.047).

The sign of each $\gamma_{jk}$ parameter potentially provides information on whether the modifier up- or down-regulates the activity of the target TF. For example, in the yeast application, the $\gamma_{jk}$ values associated with several known modifiers are con-
sistent with their known inhibitory or activating roles (Table 3.5). On the other hand, the model is fit to steady state expression values, and therefore cannot capture any feedback loops that may alter the observed relationship between modifier and TF. This suggests that while MoNsTEr accurately infers TF-Modifier edges, the dynamics of these regulatory relationships need to be further explored under specific experimental conditions.

Additionally, the function $\Phi$ (Eq 3.3) used to represent these interactions is based on the biological intuition that synergistic effects only occur when both members are sufficiently expressed, and has been successful in the applications tested. However, this function does not have a symmetric intuition for negative synergistic effects, i.e., when $\gamma_{jk} < 0$, this intuition may be lost. Ideally, a non-linear function that is also monotonic in both $f_{jt}$ and $h_{kt}$, and which fits the biological intuitions for both positive and negative values of $\gamma_{jk}$ could be developed. I leave the identification of such a function for future work, but also note that it is trivial to implement new interaction functions into the overall methodological framework.

**Additional Data Available for Yeast**

The yeast input data set used in this chapter lacks a TF-Gene prior matrix $b$ based on ChIP experiments. While there is publicly available ChIP-chip data for the factors of interest, these experiments were performed in the absence of any environmental stress [123]. The TFs MSN2 and MSN4 are primarily controlled at the level of nuclear transport, and in the absence of stress, these factors are not present at high concentrations in the nucleus. Unsurprisingly, few binding sites were identified for these particular factors in normal growth conditions. I attempted to derive a prior probability matrix $b$ from this data set, but found that it resulted in
an overly sparse network with lower performance (data not shown).

There is also publicly available data for yeast kinase substrates identified \textit{in vitro} through the use of protein microarrays [157]. I attempted to use this data set to derive the prior matrix $a$, but ultimately rejected it for several reasons. For one, quantitative data was not made public, and so I could only assign a general “high” prior probability (e.g., 0.9) to their predicted kinase-substrate connections, and a general “low” prior probability (e.g., 0.1) to all other interactions. This data set was also especially sparse for the TFs of interest, and in particular none of the 81 kinases were predicted to phosphorylate MSN2. I also attempted to derive priors from STRING [99], but once again, this resulted in an overly sparse network with lower performance (data not shown).

Taken together, these unsuccessful attempts to include additional data sources suggest that the use of multiple network priors is only beneficial when the priors are of a sufficient density. From a theoretic perspective, I expect the method described in this chapter to be more effective for filtering potential network structures suggested by noisy prior knowledge, rather than for discovering completely novel interactions with no prior evidence. This expectation is further supported by results in the next chapter, where I apply MoNsTEr to a network mediated by the TF STAT1.
Chapter 4

Analysis of the STAT1-Mediated Network in Human B Cells

4.1 Introduction

I applied MoNsTER (Chapter 3) to decipher the complex regulatory network mediated by the TF STAT1 in human B cells. B cells play a critical role in the adaptive immune response, and dysregulation of B cell networks can lead to a number of human diseases including autoimmune disorders [201], leukemias [127], and lymphomas [176]. STAT1 is a pleiotropic and critical mediator of cellular responses to a broad range of cytokines and growth factors, as well as B cell development and function. STAT1 is subject to complex post-translational regulation, and improper activity is implicated in human immune disorders and cancers [16, 39, 154, 177]. For example, STAT1 is the primary regulator of IFN-γ signaling, which is essential for anti-bacterial immune responses. IFN-γ signaling has both STAT1-dependent and STAT1-independent components, but the role of each component and the amount of cross-talk between downstream pathways remain unknown [73]. To gain a better
mechanistic understanding of STAT1’s pleiotropic function, additional knowledge is needed regarding the modifying enzymes and other modulators targeting STAT1 and their influence on the expression of specific STAT1 target genes.

The STAT1-mediated regulatory network is a particularly attractive application for integrative methods, due to the availability of a variety of relevant high throughput data (summarized in Table 4.1). Basso et al. previously generated a compendium of 336 expression microarray samples from 62 different human B cell sources, including cord blood, tumor samples, and 39 different cell lines [10]. STAT1 binding has been characterized by multiple groups, resulting in several high-quality PWMs [209] and genome-wide ChIP-seq data [165]. Thus, there is sufficient data to study the transcriptional regulatory network mediated by STAT1 in human B cells using the integrative model-based method developed in Chapter 3. In this chapter, I demonstrate how the network model can elucidate coordinated cellular responses relevant to human physiology and disease. MoNsTEr predicts a module of STAT1 target genes and modifying enzymes active in B cells that is well-supported by the STAT1 literature, but also includes novel hypotheses about the role of STAT1 in specific signaling pathways.

4.2 Materials and Methods

Human B Cell Expression Compendium

I downloaded the expression compendium of $T = 336$ Affymetrix HGU95A microarray samples of human B cell from ref. [10] in RAW format from GEO\(^2\) [9]. I processed the raw data with RMA [96] in BioConductor [66], which outputs normalized expres-

<table>
<thead>
<tr>
<th>Variable(s)</th>
<th>Summary</th>
<th>Data Type</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>g, f, h</td>
<td>Human B cells, B cell-derived cancers, and cell lines</td>
<td>Affymetrix HGU95A gene expression</td>
<td>GEO: GSE2350, ref. [10]</td>
</tr>
<tr>
<td>b</td>
<td>STAT1 ChIP-seq in IFN-γ treated HeLa S3 cell line</td>
<td>Illumina DNA sequence reads</td>
<td>GEO: GSE12782, ref. [165]</td>
</tr>
<tr>
<td>m</td>
<td>STAT1 binding sites in response to IFN-α/β (ISRE)</td>
<td>PWM</td>
<td>TRANSFAC: M00258, ref. [209]</td>
</tr>
<tr>
<td>m</td>
<td>STAT binding sites in response to IFN-γ (GAS)</td>
<td>PWM</td>
<td>ref. [162]</td>
</tr>
<tr>
<td>a</td>
<td>STRING Database, “experimental” and “pathway” channels¹</td>
<td>Protein-protein interactions</td>
<td>ref. [99]</td>
</tr>
</tbody>
</table>

Table 4.1: Data sources used as input for MoNsTEr application in human B cells, organized by model variable. All relevant literature citations and database accessions are shown in the “Source” column.

sion values on the log₂-scale. I then filtered out all probe sets with low expression in all conditions (max expression < log₂(100)), or insufficient perturbation across the samples (variance < 0.03). As for the *S. cerevisiae* data used in Chapter 3, each remaining gene expression profile was further normalized to have mean of 0 and standard deviation of 1 using Eq 3.7.

The expression compendium was then split into matrices g, f, and h, as follows. The HGU95A microarray contained 4 different probe sets corresponding to STAT1, which were highly correlated ($r^2 > 0.9$ for all pairs). I calculated the mean of these expression values within each sample to create a single representative expression profile for STAT1 (matrix f, with $J = 1$). I also identified groups of probe sets that corresponded to the same kinase or phosphatase gene and combined redundant expression profiles. For each *Entrez* gene ID [132] corresponding to multiple probe sets in a list of kinases and phosphatases on the HGU95A microarray, I applied the clustering procedure shown in Box 4.1. Note that I still allow for a modifier to
1. Identify all corresponding probe sets (expression profiles)

2. Identify the pair of expression profiles with the largest positive correlation, \( r^2 \)

3. If \( r^2 > 0.5 \), average the corresponding pair of expression values within each sample, and replace both profiles with the average expression profile

4. Repeat steps 2 and 3 until the criteria in step 3 is not met, or until a single expression profile remains.

Box 4.1: Procedure for clustering probe sets annotated for the same kinase or phosphatase gene.

be represented by multiple expression profiles in the model input data if the probe sets are not strongly correlated across the expression compendium. This process resulted in \( K = 510 \) expression profiles for kinases and phosphatases (matrix \( h \)), corresponding to 323 unique Entrez gene IDs. The remaining \( N = 8,973 \) probe sets that passed the initial filtering, but did not correspond to STAT1, kinase, or phosphatase genes, were treated as target genes (matrix \( g \)). No clustering was applied to target gene probe sets, regardless of correlation. Thus, the probe sets correspond to only 7,026 unique Entrez gene IDs. Combined expression profiles for STAT1 and relevant modifiers were z-score normalized by the same method as for individual probe sets (Eq 3.7).

**STAT1-Target Gene Priors from ChIP-seq**

To derive the TF-Gene prior matrix \( b \), I obtained ChIP-Seq data for STAT1 from GEO\(^3\) [9], previously published by Rozowsky *et al.* [165]. This data set includes mapped reads for STAT1 ChIP-seq experiments and input DNA controls from IFN-\(\gamma\)-treated HeLa S3 cells. I first filtered the data by removing all ambiguous or unmapped reads, then applied the Global Identifier of Target Regions (GLITR)\(^3\)

algorithm [195] for peak-calling. Briefly, I used the filtered input DNA controls from ref. [165] as “Pseudo” reads, and all input human DNA reads from ref. [195] as “Background”. I removed duplicate start coordinates, then randomly removed a small percentage (< 0.1%) from the remaining ChIP-seq start coordinates, such that both the ChIP-seq and Pseudo input sets had the same number of unique start coordinates. All other GLITR parameters were left at their default settings. The details of mapping GLITR peak scores to prior probabilities are provided in Appendix B. Each peak with probability > 0 after mapping was assigned to proximal target genes based on a presumed promoter region of 1kb upstream from each annotated gene region.

STAT1-Target Gene Priors from PWMs

STAT1 binds to two distinct motifs, depending on its dimerization partner and the upstream signal triggering its activity [154]. The IFN-Stimulated Response Element (ISRE) is typically bound by STAT1 in response to Type I IFNs (i.e. IFN-α/β), while the IFN-γ-Activated Site (GAS) is typically bound by STAT1 in response to the Type II IFN (IFN-γ). I used a PWM from TRANSFAC v10.2 [209] to represent the ISRE motif (accession #M00258), and derived a PWM for GAS sites using 19 exemplary sequences compiled by Robertson et al. [162] (Figure 4.1).

I then scanned the same presumed promoter regions used for the ChIP-seq analysis (1kb upstream from annotated gene regions) with both PWMs. As in Chapter 3, each PWM match score was converted to a p-value following the strategy in ref. [124]. For each promoter i, I used the most significant p-value for either PWM, denoted here as $P_{ij}$. I then used an adjusted version of Eq 3.8 that accounts for the use of
Figure 4.1: PWMs used to derive STAT1-Gene interaction priors $m_{ij}$. The upper PWM corresponds to the IFN-Stimulated Response Element (ISRE) and the lower PWM corresponds to the IFN-$\gamma$-Activated Site (GAS). Both PWMs are represented using WebLogo [37].

multiple PWMs to convert p-values to prior probabilities:

$$m_{ij} = (1 - P_{ij})^{2[\ell-(W_I+W_G-2)]}$$  \hspace{1cm} \text{(4.1)}$$

Here, $\ell = 1000$ is the length of the promoter sequence, $W_I = 15$ is the width of the ISRE PWM, and $W_G = 11$ is the width of the GAS PWM. The exponent computed in Eq 4.1 is the number of tests done against both PWMs for a single promoter (both strands of sequence are scanned).

**Modifier-STAT1 Priors from STRING**

In this application, I considered only modifiers related to phosphorylation (kinases and phosphatases), because two phosphorylation sites (Tyr-701 and Ser-727) are known to be the primary regulators of STAT1 activity [39]. I derived a single TF-
Modifier prior matrix $a$ using selected channels of interest from the STRING database v8.2\textsuperscript{4} [99]. I then mapped STAT1 and all modifiers in the expression matrix $h$ to their corresponding STRING identifier via Entrez gene ID [132], and extracted all protein links between STAT1 and the input modifiers. I used only the channel scores titled “experimental” (denoted here as $S_e$) and “database” (denoted here as $S_d$), in order to focus on physical interactions, rather than the more generally defined functional associations. I recomputed the combined score $S$ for each link using the Bayesian integration equation from ref. [202]:

$$S = 1 - ((1 - S_e) (1 - S_d))$$  \hspace{1cm} (4.2)

Thus, each prior probability $a_{jk}$ corresponds to the recomputed score $S$ for the link between $j = \text{STAT1}$ and modifier $k$. All missing links were presumed to have $a_{jk} = 0$ (therefore transformed to 0.05 by the restricted prior range described in Section 3.3).

**Application of MoNsTER to Human Data**

Model-fitting procedures similar to those in Chapter 3 were applied to input matrices \{g, f, h, b, m, a\}. The network modeled here is approximately 30-fold larger than the yeast network in Chapter 3, and therefore each run of the model-fitting algorithm requires substantially more computing time. Rather than seed the algorithm with a highly stochastic network, as I did for the yeast application, I opted to use a smaller number of runs seeded with initial models closer to the network predicted by the priors. I began by seeding the model-fitting method with a fixed network containing only those edges with high prior probabilities. I ran the method 3 times with a fixed start, in order to capture the small variation arising in the estimation phase. I

\textsuperscript{4} Accessed at: http://string-db.org on 10/23/09
• Non-Stochastic (NS): All network edges initialized by: \( C_{ij} = 1 \) iff \( \sqrt{b_{ij}m_{ij}} \geq 0.75 \) and \( D_{jk} = 1 \) iff \( a_{jk} \geq 0.75 \), all other parameters fit by OLS (3 runs)

• Perturbed Network (PN): All network edges initialized as in NS, then 1% of edges switched at random, all other parameters fit by OLS (10 runs)

• Perturbed Parameters (PP): All network edges initialized as in NS, all other parameters fit by OLS, then added random noise sampled from \( N(0, 0.05) \) (10 runs)

Box 4.2: Initialization procedures for multiple runs of MoNaSTEr on human B cell data

then added a moderate amount of perturbation to the starting model, in order to explore a larger portion of the solution space. I perturbed the network structure by adding/removing edges at random. Alternatively, I perturbed the starting values of numerical parameters by adding noise sampled from a normal distribution. Thus, I ran the model-estimation heuristic multiple times, summarized in Box 4.2. Further perturbations to the initialization phase did not identify any additional solution modes (data not shown).

For all models, I used the thresholds \( P(C_{ij} = 1) \geq 0.9 \) to predict high-confidence STAT1 target genes, and \( P(D_{jk} = 1) \geq 0.9, |\gamma_{jk}| \geq 0.05 \) to predict high-confidence STAT1 modifying enzymes.

Human B Cell Expression Permutation

As a control for the MoNaSTEr network model, the expression values within each profile (row) in \( g, f, \) and \( h \) were each randomly permuted with respect to conditions \( t \), in order to disrupt any biological regulatory signal in the data. I then repeated the model-fitting method 10 times, using the same prior probabilities in \( m, b, \) and \( a \), and averaged together the model results. I repeated this entire permutation and model-fitting procedure 10 times, and compared the distributions of model
parameters across all permuted models to those in the primary STAT1 model.

Annotation Enrichment Analysis

To summarize the biological relevance of each network model, enrichment analysis for predicted STAT1 target genes and modifiers was performed on each possible annotation using the BioConductor packages GOstats, GO.db, KEGG.db, and org.Hs.eg.db [66]. I excluded all GO terms one or two levels below the ontology root, as these tend to be the broadest and least informative. The significance of annotation enrichment for putative target genes was performed using the hypergeometric test with the full set of $N$ candidate target genes from input matrix $g$ as the background set. As a comparison to individual data sources used for the MoNsTEr application, I selected the same number of probe sets from lists of target genes ranked by each of the input data sets alone, and repeated the enrichment analysis for each of these lists. All reported p-values are corrected for multiple testing by the Bonferroni method, i.e., multiplied by the number of tests performed for that particular annotation class [54]. P-values that were greater than 1 after Bonferroni correction are reported as ‘–’.

Due to the small sample size, and the general sparsity of pathway annotations, it was not possible to reliably assess the significance of annotation enrichments for STAT1 modifiers predicted by MoNsTEr against those predicted by STRING. Instead, I selected the annotations with the strongest overall enrichment based on an odds ratio. For each annotation “X” associated with at least one MoNsTEr-predicted modifier, I computed the odds ratio of enrichment against the STRING-only list of
modifiers, as follows:

$$\text{Odds Ratio (X)} = \frac{\% \text{ of MoNsTEr-predicted modifiers w/ annotation X}}{\% \text{ of STRING-only modifiers w/ annotation X}}$$  \hspace{1cm} (4.3)

For GO Biological Process (GOBP) annotations, I selected annotations associated with at least 5 of the 21 MoNsTEr-predicted modifiers, and odds ratio > 2. For KEGG Pathway annotations, which are sparser than GOBP annotations, I selected terms with at least 3 of 21 MoNsTEr-predicted modifiers, and once again an odds ratio > 2.

### 4.3 MoNsTEr Predicts Three Distinct Network Models

I applied MoNsTEr to input data for 7,026 unique target genes, 1 TF (STAT1), and 323 unique modifier genes encompassing most known kinases and phosphatases (see Section 4.2). Consistent with a pleiotropic role for STAT1, and the fact that the expression data covers a wide range of network perturbations, MoNsTEr predicted three distinct network models given the available data (Table 4.1). I ran MoNsTEr using several different initialization configurations to explore the overall model solution space and compute robust parameter estimates with reasonable efficiency (see Section 4.2 for details). Model parameters from individual runs were hierarchically clustered (see Section 3.2) and the resulting dendrogram is shown in Figure 4.2. Note that there are much longer branches at the top of the hierarchy, indicating well-separated clusters. Runs are labeled by the type of seeding method described in Box 4.2, and the three major clusters are numbered I–III and outlined in green, purple, and blue, respectively. I averaged model estimates together within each of these clusters to produce three distinct network models.
As an overall comparison of the three network models, I first looked at basic model properties, summarized in Table 4.2. The term $\beta_1$ is the influence parameter for STAT1, and describes the influence of STAT1, in general (independent of modifiers), and whether it is an activator ($\beta_1 > 0$) or repressor ($\beta_1 < 0$). The term $w_1$ is the prior weight parameter for STAT1, and describes whether the inferred network is more dependent on the ChIP-based prior matrix $b$, or PWM-based prior matrix $m$. The number of targets is given in terms of probe sets, and is based on the
Table 4.2: Properties of STAT1 network models averaged after hierarchical clustering. The number of predicted targets is based on a threshold of $P(C_{ij} = 1) \geq 0.9$, and is shown as both the number of individual probe sets (4th column), and the number of unique genes after mapping all probe sets (5th column). For comparison, the total number of probe sets ($N$) and unique genes mapping to these probe sets are shown in the bottom row.

<table>
<thead>
<tr>
<th>Model</th>
<th>$\beta_1$ (s.d.)</th>
<th>$w_1$ (s.d.)</th>
<th>target probe sets</th>
<th>target genes</th>
</tr>
</thead>
<tbody>
<tr>
<td>I</td>
<td>0.24 (0.003)</td>
<td>0.51 (0.003)</td>
<td>1,803</td>
<td>1,559</td>
</tr>
<tr>
<td>II</td>
<td>-0.29 (0.07)</td>
<td>0.62 (0.008)</td>
<td>2,031</td>
<td>1,735</td>
</tr>
<tr>
<td>III</td>
<td>-0.004 (0.0007)</td>
<td>0.26 (0.0006)</td>
<td>3,190</td>
<td>2,679</td>
</tr>
<tr>
<td>(Input)</td>
<td>-</td>
<td>-</td>
<td>8,973</td>
<td>6,975</td>
</tr>
</tbody>
</table>

threshold $P(C_{ij} = 1) \geq 0.9$. There is little overlap in the target genes predicted by each model. Network models I and III share only 5 predicted targets, and networks II and III share 95 predicted targets (by probe set). Network models I and II do not share any predicted targets. This supports the notion that all three clusters of model estimates are well-separated in the possible solution space.

The network model derived from cluster I produces the most reliable model for several reasons. Most notably, the predicted targets in this model are significantly enriched for known direct targets of STAT1, previously compiled by Roberston et al. [162]. 23 of these known targets mapped to 39 probe sets in matrix $g$ and of these, 25 (64%) were predicted by MoNsTEr (hypergeometric test p-value = 2.6x10$^{-9}$). The 23 known direct target genes are listed in Table 4.3 with the corresponding HGU95A probe set IDs present in the human B cell expression compendium.

Network I is also the only model with $\beta_1 > 0$, which is in agreement with the established role of STAT1 as an activator of transcription [39, 177]. This model has $w_1$ closest to 0.5, which indicates balanced use of the ChIP and PWM-based TF-Gene priors. By comparison, model II is biased somewhat towards the ChIP-based priors, which are limited to a singular cellular condition. Model III is heavily biased.
Table 4.3: Gene symbols for verified direct STAT1 targets compiled by Robertson et al. [162], with matching probe sets in the human B cell expression compendium. √ indicates STAT1 target is correctly predicted by MoNSTer primary model.

Towards the PWM-based priors, which are likely to have lower specificity in general compared to the ChIP-based priors. Model III also predicts that STAT1 has very weak influence on the predicted target genes ($\beta_1 \approx 0$).

To further compare these network models, I plotted histograms of the TF-Gene priors, expression correlations to STAT1, and expression entropy [173] for the target genes predicted by each network (Figure 4.3). These distributions further support the selection of the model I as the most interesting and reliable solution mode. Despite the fact that model II (purple bars) is more heavily weighted towards the ChIP-seq prior matrix $b$, and is generally more enriched for target genes with higher
Figure 4.3: Distributions of STAT1 network model properties. Distributions are shown for: (A) ChIP-seq-based priors, (B) PWM-based priors, (C) STAT1-Target Gene expression correlation, (D) Entropy of target genes. White bars represent background distribution for all target genes used as input. Green, purple, and blue bars represent STAT1 target genes predicted by network solution modes I–III, respectively.

$b_{ij}$ priors, model I (green bars) is actually the most enriched for the group of target genes with the highest $b_{ij}$ scores (Figure 4.3A). This is also true for the PWM-based priors $m_{ij}$ (Figure 4.3B). In other words, model I is the most enriched for target genes
with the strongest prior scores based on both ChIP-seq and PWM. Model I is also enriched for genes with the strongest correlation to STAT1 expression (Figure 4.3C).

Entropy [173] measures the overall broadness or lack of condition-specificity for each target gene. Lower entropy indicates that a gene is specifically expressed in a smaller set of expression samples in the compendium. Gene expression profiles with low entropy are the most likely to represent active regulatory patterns rather than random noise. Model III (blue bars), which also contains the largest number of predicted targets, and the weakest STAT1 regulatory influence (Table 4.2), is heavily biased towards target genes which appear to be the most random in their expression patterns (Figure 4.3D). Model I is highly enriched for genes with lower entropy, and therefore represents the strongest regulatory signal in the input data.

Based on the model comparisons described above, I chose model I as the “primary” model, i.e., the model most likely to capture the direct regulatory effects of STAT1. This model is discussed in detail in the next section. In Section 4.5, I discuss the possible biological implications of alternate network models II and III.

4.4 Analysis of the Primary STAT1-Mediated Network Model

The primary STAT1 network model predicts that STAT1 regulates the transcription of 1,559 unique target genes, represented by 1,803 probe sets in the input data. As shown in Figure 4.4A, the predicted target genes generally have a greater expression correlation with STAT1 than the remaining input genes. The same is true when comparing the priors $b_{ij}$ and $m_{ij}$ for the predicted target genes against all remaining input genes (Figure 4.4B,C). However, none of these properties, by themselves, ac-
curately discriminate the same set of target genes predicted by MoNsTEr. Therefore, the model-based prediction of network edges integrates the information embedded in the priors and the expression data. Interestingly, the model correctly predicts STAT1 as an activator, rather than repressor, of these target genes [39, 177].

As noted in Section 4.3, the primary network model is significantly enriched for known direct targets of STAT1. Overall, the list of putative target genes is also enriched for GO [4] and KEGG pathway [106] annotations relevant to both STAT1 and B cell functions (Table 4.4). With one exception, these annotations are not significantly enriched among the top target genes predicted by any of the individual data sources (Table 4.4, right-hand columns). A few additional, but biologically unrelated annotations are specifically enriched among the top target genes predicted by ChIP-seq, but otherwise there is little functional enrichment detectable in the target genes predicted by individual data sources.

This network model also predicts 20 kinases and 1 phosphatase (represented by 23 probe sets, Figure 4.5) that influence STAT1 activity, including all 4 members
<table>
<thead>
<tr>
<th>GO Biological Process</th>
<th>MoNsTER</th>
<th>Expr</th>
<th>ChIP</th>
<th>PWM</th>
</tr>
</thead>
<tbody>
<tr>
<td>signal transduction</td>
<td>6E-08</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>regulation of I-kappaB kinase/NF-kappaB cascade</td>
<td>7E-06</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>regulation of signal transduction</td>
<td>5E-05</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>I-kappaB kinase/NF-kappaB cascade</td>
<td>2E-04</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>inflammatory response</td>
<td>4E-04</td>
<td>–</td>
<td>–</td>
<td>0.049</td>
</tr>
<tr>
<td>positive regulation of I-kappaB kinase/NF-kappaB cascade</td>
<td>4E-04</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>intracellular signaling cascade</td>
<td>7E-04</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>response to wounding</td>
<td>8E-04</td>
<td>–</td>
<td>–</td>
<td>0.346</td>
</tr>
<tr>
<td>cell death</td>
<td>0.002</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>response to virus</td>
<td>0.003</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>protein kinase cascade</td>
<td>0.004</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>programmed cell death</td>
<td>0.005</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>apoptosis</td>
<td>0.007</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>negative regulation of biological process</td>
<td>0.01</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>biopolymer metabolic process</td>
<td>–</td>
<td>–</td>
<td>3E-07</td>
<td>–</td>
</tr>
<tr>
<td>nucleobase, nucleoside, nucleotide and nucleic acid metabolic process</td>
<td>–</td>
<td>–</td>
<td>2E-06</td>
<td>–</td>
</tr>
<tr>
<td>RNA metabolic process</td>
<td>–</td>
<td>–</td>
<td>0.006</td>
<td>–</td>
</tr>
<tr>
<td>regulation of cellular metabolic process</td>
<td>–</td>
<td>–</td>
<td>0.007</td>
<td>–</td>
</tr>
<tr>
<td>regulation of nucleobase, nucleoside, nucleotide and nucleic acid metabolic process</td>
<td>–</td>
<td>–</td>
<td>0.016</td>
<td>–</td>
</tr>
<tr>
<td>response to DNA damage stimulus</td>
<td>–</td>
<td>–</td>
<td>0.037</td>
<td>–</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>KEGG Pathway</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Leukocyte transendothelial migration</td>
<td>3E-07</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>Focal adhesion</td>
<td>1E-05</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>Cell adhesion molecules (CAMs)</td>
<td>0.009</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>T cell receptor signaling pathway</td>
<td>0.01</td>
<td>0.835</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>Cytokine-cytokine receptor interaction</td>
<td>0.012</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>ECM-receptor interaction</td>
<td>0.02</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
</tbody>
</table>

Table 4.4: Significantly enriched annotations (1st column) for gene targets in primary STAT1 network, using all gene targets in the input data as the background, with Bonferroni-corrected p-values (2nd column). As a control, corrected p-values are also shown for the most correlated genes (3rd column), and genes with the highest ChIP-seq and PWM-based priors (4th and 5th columns). ‘–’ indicates a p-value > 1 after Bonferroni correction.
Figure 4.5: STAT1 modifiers predicted by MoNsTEr primary network model. All predicted modifiers have $P(D_{jk} = 1) > 0.9$ and $|\gamma_{jk}| > 0.05$, and are ordered by decreasing $|\gamma_{jk}|$. DUSP3 and LCK are modeled by multiple, uncorrelated probe sets, and therefore have multiple $\gamma_{jk}$ parameters.

of the Janus Kinase family, which are well-characterized activators of the STAT family [177]. Based on the $\gamma_{jk}$ values, the most influential modifier in this STAT1 network model is JAK1, which has experimentally characterized roles in both Type I and II interferon signal transduction through STAT1 [16]. Additional literature supporting each predicted modifier is summarized in Table 4.5. Overall, 18 (86%) of the predicted modifiers have existing evidence for specific regulation of STAT1 transcriptional activity.

All predicted modifiers have high STAT1 interaction priors $a_{jk}$ derived from STRING [99], but not all modifiers with high priors were predicted by MoNsTEr (see Discussion). Therefore the method specifically predicts modifiers likely to affect STAT1 transcriptional activity in B cells and related cancers, given some prior knowledge of general protein-protein interaction with STAT1. To further assess the functional implications of the modifier list predicted by MoNsTEr as compared to
Table 4.5: STAT1-Modifiers predicted by primary network model, listed by gene symbol, with $\gamma_{jk}$ value and summary of known role in STAT1 regulation. $\gamma_{jk}$ values marked * indicate a probe set annotated for the same modifier, which corresponds to an incomplete transcript.
<table>
<thead>
<tr>
<th>GOBP Term</th>
<th>MoNSTEr Count</th>
<th>Odds Ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td>intracellular signaling cascade</td>
<td>15</td>
<td>5.69</td>
</tr>
<tr>
<td>protein kinase cascade</td>
<td>11</td>
<td>3.45</td>
</tr>
<tr>
<td>hemopoiesis</td>
<td>5</td>
<td>4.82</td>
</tr>
<tr>
<td>apoptosis</td>
<td>5</td>
<td>4.82</td>
</tr>
<tr>
<td>cell death</td>
<td>5</td>
<td>4.82</td>
</tr>
<tr>
<td>programmed cell death</td>
<td>5</td>
<td>4.82</td>
</tr>
<tr>
<td>hemopoietic or lymphoid organ development</td>
<td>5</td>
<td>4.82</td>
</tr>
<tr>
<td>protein amino acid phosphorylation</td>
<td>19</td>
<td>2.84</td>
</tr>
<tr>
<td>phosphorylation</td>
<td>19</td>
<td>2.84</td>
</tr>
<tr>
<td>organ development</td>
<td>10</td>
<td>2.14</td>
</tr>
<tr>
<td>system development</td>
<td>10</td>
<td>2.14</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>KEGG PATHWAY Term</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>T cell receptor signaling pathway</td>
<td>3</td>
<td>∞</td>
</tr>
<tr>
<td>Tight junction</td>
<td>3</td>
<td>∞</td>
</tr>
<tr>
<td>Adipocytokine signaling pathway</td>
<td>5</td>
<td>4.82</td>
</tr>
<tr>
<td>Jak-STAT signaling pathway</td>
<td>5</td>
<td>2.29</td>
</tr>
</tbody>
</table>

Table 4.6: Pathway annotations with strong odds ratios when comparing MoNSTEr-predicted modifiers to other modifiers predicted by STRING.

other interacting partners predicted by STRING, I compared the list of 21 MoNSTEr-predicted modifiers to the remaining 17 modifiers (kinases and phosphatases only) predicted only by STRING. These lists are too short for a robust statistical analysis, but I observed obvious trends in the functional enrichment of the MoNSTEr-predicted list as compared to the STRING-only list, based on the odds ratios (Eq 4.3). It is striking that many of the annotations in Table 4.6 recapitulate known pathways for STAT1, such as apoptosis, and related terms also enriched among the predicted target genes (Table 4.4). The list also includes annotations relevant to B cells and related cancers, such as hemopoiesis and hemopoietic or lymphoid organ development.

As an additional control, I repeatedly permuted the expression values within
each gene profile and recomputed the MoNsTEr network model (see Section 4.2). In all cases, the permuted model predicted fewer target genes and upstream modifiers for STAT1, with weaker $\beta$ and $\gamma$ influence parameters (Figure 4.6). This control empirically demonstrates that the predicted network is the result of coherent signal in both the expression and network prior data.

Additionally, I analyzed the known functional associations between STAT1 target genes and modifiers predicted by this network model. The predicted modifiers of STAT1 should be functionally linked to the predicted targets of STAT1 if the network model truly captures biological functions mediated through STAT1. To test this, I first extracted from the STRING database [99] functional association scores for all pairs $(i, k)$ in the human B cell input data. I recomputed the association scores by excluding the “expression correlation” channel to avoid any commonality with the input data that might bias the results. Note that these association scores were not used to compute the STAT1-Modifier prior matrix $a$, and therefore are independent of the input data used by MoNsTEr.

I separated the association scores into those within the primary network model, i.e., for which both $i$ was a predicted target of STAT1 and $k$ was a predicted modifier of STAT1, and those outside the network (all remaining associations). I first analyzed the density of associations in each category, based on the number of associations contained in STRING vs. the number of possible pairs $(i, k)$. I found that the density of defined associations within the network was approximately 4.5-fold greater within the network (0.0219 vs. 0.0049), and this difference is highly significant according to Fisher’s exact test (p-value < 2.2E-16). I also compared the distribution of the defined STRING scores, independent of density, and found that in general the association scores defined for $(i, k)$ pairs within this network were higher than those outside the network (Figure 4.7, Mann-Whitney Test p-value < 2.2E-16).
Figure 4.6: Primary STAT1 network model vs. permuted expression controls. Distributions of model parameters for primary STAT1 network (green), and for the networks obtained from randomly permuted expression data (blue) for: (A) STAT1-Gene posterior probabilities, (B) STAT1-Modifier posterior probabilities, (C) STAT1 $\beta_j$ parameters, and (D) STAT1-Modifier $\gamma_{jk}$ magnitudes.
Moreover, the genes in the module are enriched for STAT1 in human B cells. This module is significantly enriched for known direct processes relevant to STAT1, such as biological insights from the primary model.

Therefore, I conclude that this network predicts STAT1 target genes and modifiers which are functionally coherent, i.e., involved in common biological processes.

4.5 Discussion

Biological Insights from the Primary Model

The primary MoNsTER model predicted a large module of target genes regulated by STAT1 in human B cells. This module is significantly enriched for known direct targets of STAT1. Moreover, the genes in the module are enriched for GO biological processes relevant to STAT1, such as apoptosis [92] and I-kappaB kinase/NF-kappaB
cascade [206]. Other enriched GO terms suggest general STAT1 and B cell functions, e.g., response to virus and inflammatory response (Table 4.4). Importantly, nearly all of these annotations are not enriched among the top target genes as predicted by ChIP-seq, PWM, or expression correlation analysis alone. Overall the functional enrichment analysis of predicted target genes support the relevance of this network model to the TF and cell type analyzed.

The primary STAT1 network model inferred by MoNsTER also provides a compelling list of potential STAT1 modifiers, including known, suspected, and novel regulators of STAT1 transcriptional activity. This network model correctly identifies JAK1 (\(\gamma_{jk} = 0.29\)), TYK2 (\(\gamma_{jk} = 0.08\)), and JAK2 (\(\gamma_{jk} = 0.05\)) as positive regulators of STAT1 activity. Notably, JAK1 has the strongest effect on STAT1 (\(\gamma_{jk} = 0.29\)) in this model and JAK1 is known to be a highly influential regulator of both IFN-\(\alpha/\beta\) and IFN-\(\gamma\)-dependent STAT1 activity [16]. Many of the other \(\gamma_{jk}\) parameter values for predicted STAT1 modifiers are supported by literature evidence (Table 4.5). Many of the STAT1 modifiers predicted by MoNsTER are annotated for pathways related to the immune system, while STAT1 interaction partners predicted only by STRING are not (Table 4.6). However, there are relatively few modifiers predicted by either method, as is expected, and many of these modifiers are poorly annotated, limiting the power of any statistical test for this observation. Overall, 86% of the predicted modifiers are supported by existing literature (Table 4.5), with the exception of several novel predictions discussed below. Furthermore, a role in B cell function or STAT1 regulation specific to this cell type is a novel prediction in most cases.

MoNsTER predicts a novel association between the dual-specificity phosphatase DUSP3/VHR and STAT1 transcriptional activity. This phosphatase has been shown to dephosphorylate STAT5 in interferon signaling [88] and inactivate Erk2 and Jnk
downstream of the T cell antigen receptor [2]. VH1, the *vaccinia* virus homolog of DUSP3, is known to block STAT1 activation during infection [145]. However, this analysis is the first to suggest a link between endogenous DUSP3 and STAT1 transcriptional control.

MoNsTER also predicts Receptor-Interacting Protein Kinase 1 (RIPK1) as a direct modifier of STAT1 transcriptional activity. RIPK1 is a kinase known to interact with TNF receptor 1 (TNFR1) resulting in a switch between the pro-apoptotic and anti-apoptotic responses to TNF-α. Experiments in 293T cells have shown that STAT1 can competitively bind TNFR1, displacing the interaction with RIPK1, and disrupting downstream signaling through the NF-κB pathway independently of STAT1 nuclear import or DNA binding [206]. Cancer progression models in mice have shown that TNF-α can suppress tumor growth in a STAT1-dependent manner [213]. The model presented here is the first to suggest that RIPK1 may antagonize signaling through STAT1 in a way that affects STAT1 target gene transcription. In combination with the predicted STAT1 target genes related to apoptosis and NF-κB signaling (Table 4.4), the network suggests a critical role for STAT1 in balancing the pro-apoptotic and pro-survival responses to TNF-α in tumor progression.

**Influence of Input Data Sources on Model Parameters**

The primary STAT1 network model inferred by MoNsTER provides a compelling list of potential STAT1 modifiers, including known, suspected, and novel regulators of STAT1 transcriptional activity. I used two thresholds for the prediction of high-confidence STAT1 modifiers, specifically: TF-Modifier edge posterior \( P(D_{jk} = 1) \geq 0.9 \) and the influence parameter for this edge has magnitude \( |\gamma_{jk}| \geq 0.05 \). This criteria selects for TF-Modifier interactions with both an overall strong probability and
a substantial effect on the expression of downstream target genes. This is important because the overall posterior probabilities $D_{jk}$ are influenced by the corresponding prior probabilities, as well as by the expression data. It is possible, in the model-fitting procedure, for a TF-Modifier interaction with a strong prior probability to be included in the model, even while its impact in Eq 3.1 is minimized by setting $\gamma_{jk} \approx 0$, especially when only one type of TF-Modifier prior matrix is available. I chose the particular threshold $|\gamma_{jk}| \geq 0.05$ based on the earlier *S. cerevisiae* analysis (Section 3.5) in which all known modifiers met this threshold, and also because it was outside the range of $\gamma_{jk}$ values estimated using the permuted B cell expression data as a control.

To further explore this issue, I generated scatter plots comparing the estimated modifier interaction parameters ($D$ and $\gamma$) to the individual input data sources. First, I noticed that the edge posteriors $D_{jk}$ are significantly biased by the edge priors $a_{jk}$ (Figure 4.8A). This is partly because of the limited sensitivity demonstrated in the simulation experiments, and also because we only have a single source of prior information for TF-Modifier edges, as noted above. Overall, the selection of STAT1 modifiers in the network model appears to be restricted to only those with above average prior probabilities based on the STRING database, even though no such threshold is explicitly stated in our model. However, a strong prior probability does not guarantee inclusion in the final network model, and our prediction criteria also include a threshold on the $\gamma_{jk}$ parameter. I confirmed that $|\gamma_{jk}|$ is not biased by the prior probabilities $a_{jk}$ or by expression correlations between STAT1 and individual modifiers $k$ in our primary model (Figure 4.8C,D).

Furthermore, the probabilities derived from STRING are only meant to predict general protein-protein interactions, while our model specifically predicts interactions that influence TF activity in the cell type analyzed. As previously discussed
in Section 3.6, MoNsTEr can be thought of as a way to filter a list of possible interactions represented as prior knowledge to a more specific set of putative functional modifiers that fit the observed expression values. This idea is supported here by the observation that many of the STAT1 modifiers predicted by MoNsTEr are annotated for pathways related to the immune system, while STAT1 interaction partners predicted only by STRING are not (Table 4.6).

**Alternate STAT1-Mediated Network Models**

As noted in Chapter 3, it is possible for multiple models fit to the same input data to discover distinct and equally correct biological aspects of a network. For example, if a TF acts as an activator at some promoters, but as a repressor at others, there is no way for a single version of the MoNsTEr model to capture this. However, the model-fitting algorithm can be run multiple times to identify two distinct, but equally valid, models—one encompassing the activated targets of the TF, and another encompassing the repressed targets.

In this chapter, the model-fitting algorithm produced three distinct network models. I chose to follow up on one model that was well supported by the existing knowledge of STAT1 targets and pathways, but it is likely that the remaining two models capture some relevant aspects of STAT1 biology, discussed below. However, it is unclear, without further experimental validation, whether these alternate models represent direct STAT1 activities, or broader downstream effects correlated to STAT1. Therefore, I did not investigate the modifiers predicted by these network models, because the TF-Modifier connections are ultimately inferred from their effects on target gene expression (Eq 3.1). In other words, questions remain about the validity of the putative target genes in these models, and the modifier predictions
Figure 4.8: Scatter plots investigating the dependence of STAT1-Modifier edge parameters $D_{jk}$ and $\gamma_{jk}$ on STRING-based priors $a_{jk}$ and STAT1-Modifier expression correlation. Plots shown for (A) $D_{jk}$ vs. $a_{jk}$, (B) $D_{jk}$ vs. expression correlation, (C) $\gamma_{jk}$ vs. $a_{jk}$, and (D) $\gamma_{jk}$ vs. expression correlation. Only the dependence in (A) is significant.
can only be as reliable as the target gene predictions.

The average network model derived from cluster II in Figure 4.2 linked STAT1 to the negative regulation of 2,031 probe sets corresponding to 1,735 unique genes. This set of genes is significantly enriched for annotations related to metabolic and biosynthetic processes, as shown in Table 4.7, and once again these enrichments are not found in top genes based on each individual input. However, this set of genes contains none of the direct STAT1 targets compiled by Robertson et al. [162]. This model also suggests a repressive role for STAT1 ($\beta_1 = -0.29$), although STAT1 is primarily characterized as an activator in the literature [39, 154]. Therefore I argue that this network model is probably representative of downstream effects. However, it does suggest that in general STAT1 activity is negatively correlated with cell growth and proliferation in the analyzed expression data. Negative regulation of these functions is a known downstream effect of STAT1 activity in many cell and tumor types [160]. As with all of the network models, I cannot completely rule out indirect or parallel effects, both in terms of the target genes and STAT1 modifiers.

I hypothesize that the network model derived from cluster III is most likely artifactual or representative of broad cellular trends, rather than STAT1-specific regulation. The value of the $\beta_1$ parameter in this model suggests that STAT1 has little direct influence on the predicted target genes, although this could simply mean that STAT1 is completely dependent on upstream modifiers for the regulation of target genes in this network model. The value of $\beta_1$ is also within the range of values observed when the input expression data is randomly permuted, making it difficult to assess the significance of this model. Furthermore, the list of predicted target genes is heavily biased towards high-entropy genes which are unlikely to carry significant biological regulatory signal in the expression set (Figure 4.2D). This target gene list is also enriched primarily for functional annotations related
to the nervous system rather than the immune system, although these functional annotations are still highly significant and would not be predicted using the PWM scores alone (Table 4.8).

I offer several alternate hypotheses for this model. One possibility is that the network reflects indirect regulatory connections between the modifiers and target genes, but mediated by TFs other than STAT1. This may occur in the procedure because other mediating TFs were left out due to lack of sufficient data. Another possibility is that this model does represent a STAT1-mediated network, albeit one that is not primarily active in B cells. This model is biased towards the PWM-based priors, and therefore is more likely to identify STAT1 binding sites across all cell types and signaling pathways. A role for STAT1 in regulating gene expression in neurons has been experimentally demonstrated [27, 28]. In this case, the network model parameters may be indicative that STAT1 is being decoupled from the regulation of these target genes by the upstream modifiers in the samples covered by this particular expression compendium. Even if this is the case, it is likely that additional false positive target genes were picked up due to the lack of a stringent expression model. Ultimately, I cannot differentiate among these possibilities using existing data.
Table 4.7: Significantly enriched annotations (1st column) for gene targets in STAT1 network model II, using all gene targets in the input data as the background, with Bonferroni-corrected p-values (2nd column). As a control, corrected p-values are also shown for the most correlated genes (3rd column), and genes with the highest ChIP-seq and PWM-based priors (4th and 5th columns). ‘–’ indicates a p-value > 1 after Bonferroni correction.

<table>
<thead>
<tr>
<th>GO Biological Process</th>
<th>MoNsTEr</th>
<th>Expr</th>
<th>ChIP</th>
<th>PWM</th>
</tr>
</thead>
<tbody>
<tr>
<td>nucleobase, nucleoside, nucleotide and nucleic acid metabolic process</td>
<td>5E−38</td>
<td>–</td>
<td>7E−08</td>
<td>–</td>
</tr>
<tr>
<td>biopolymer metabolic process</td>
<td>2E−27</td>
<td>–</td>
<td>4E−09</td>
<td>–</td>
</tr>
<tr>
<td>DNA metabolic process</td>
<td>4E−24</td>
<td>–</td>
<td>0.045</td>
<td>–</td>
</tr>
<tr>
<td>ribonucleoprotein complex biogenesis and assembly</td>
<td>4E−22</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>mRNA metabolic process</td>
<td>2E−21</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>mRNA processing</td>
<td>9E−21</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>RNA splicing</td>
<td>1E−18</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>DNA replication</td>
<td>3E−18</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>response to DNA damage stimulus</td>
<td>9E−17</td>
<td>–</td>
<td>0.003</td>
<td>–</td>
</tr>
<tr>
<td>DNA repair</td>
<td>4E−16</td>
<td>–</td>
<td>0.15</td>
<td>–</td>
</tr>
<tr>
<td>ribosome biogenesis and assembly</td>
<td>4E−16</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>cell cycle phase</td>
<td>1E−15</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>RNA metabolic process</td>
<td>1E−14</td>
<td>–</td>
<td>0.001</td>
<td>–</td>
</tr>
<tr>
<td>M phase</td>
<td>5E−14</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>oxidative phosphorylation</td>
<td>5E−14</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>mitotic cell cycle</td>
<td>2E−12</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>rRNA processing</td>
<td>3E−11</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>rRNA metabolic process</td>
<td>3E−11</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>electron transport</td>
<td>3E−11</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>M phase of mitotic cell cycle</td>
<td>1E−10</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>ATP synthesis coupled electron transport</td>
<td>4E−10</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>organelle ATP synthesis coupled electron transport</td>
<td>4E−10</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>macromolecular complex assembly</td>
<td>6E−10</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>mitosis</td>
<td>7E−10</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>DNA-dependent DNA replication</td>
<td>9E−10</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>cellular component assembly</td>
<td>1E−08</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>tRNA metabolic process</td>
<td>1E−08</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>mitochondrial electron transport, NADH to ubiquinone</td>
<td>2E−08</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>cellular biosynthetic process</td>
<td>5E−08</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>nuclear mRNA splicing, via spliceosome</td>
<td>7E−08</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>RNA splicing, via transesterification reactions</td>
<td>7E−08</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
</tbody>
</table>

Continued on next page
Table 4.7: Enriched annotations for gene targets in STAT1 network model II (continued).

<table>
<thead>
<tr>
<th>GO Biological Process</th>
<th>MoNsTEr</th>
<th>Expr</th>
<th>ChIP</th>
<th>PWM</th>
</tr>
</thead>
<tbody>
<tr>
<td>RNA splicing, via transesterification reactions with bulged adenosine as nucleophile</td>
<td>7E−08</td>
<td>−</td>
<td>−</td>
<td>−</td>
</tr>
<tr>
<td>macromolecule catabolic process</td>
<td>2E−07</td>
<td>−</td>
<td>−</td>
<td>−</td>
</tr>
<tr>
<td>protein-RNA complex assembly</td>
<td>2E−07</td>
<td>−</td>
<td>−</td>
<td>−</td>
</tr>
<tr>
<td>organelle organization and biogenesis</td>
<td>8E−07</td>
<td>−</td>
<td>−</td>
<td>−</td>
</tr>
<tr>
<td>biopolymer catabolic process</td>
<td>3E−06</td>
<td>−</td>
<td>−</td>
<td>−</td>
</tr>
<tr>
<td>cellular macromolecule catabolic process</td>
<td>4E−06</td>
<td>−</td>
<td>−</td>
<td>−</td>
</tr>
<tr>
<td>purine nucleotide biosynthetic process</td>
<td>6E−06</td>
<td>−</td>
<td>−</td>
<td>−</td>
</tr>
<tr>
<td>ribonucleotide biosynthetic process</td>
<td>1E−05</td>
<td>−</td>
<td>−</td>
<td>−</td>
</tr>
<tr>
<td>protein metabolic process</td>
<td>5E−05</td>
<td>−</td>
<td>0.051</td>
<td>−</td>
</tr>
<tr>
<td>ribonucleotide metabolic process</td>
<td>6E−05</td>
<td>−</td>
<td>−</td>
<td>−</td>
</tr>
<tr>
<td>intracellular transport</td>
<td>6E−05</td>
<td>−</td>
<td>−</td>
<td>−</td>
</tr>
<tr>
<td>purine nucleotide metabolic process</td>
<td>8E−05</td>
<td>−</td>
<td>−</td>
<td>−</td>
</tr>
<tr>
<td>purine ribonucleotide biosynthetic process</td>
<td>1E−04</td>
<td>−</td>
<td>−</td>
<td>−</td>
</tr>
<tr>
<td>intracellular protein transport across a membrane</td>
<td>2E−04</td>
<td>−</td>
<td>−</td>
<td>−</td>
</tr>
<tr>
<td>protein folding</td>
<td>3E−04</td>
<td>−</td>
<td>−</td>
<td>−</td>
</tr>
<tr>
<td>purine ribonucleotide metabolic process</td>
<td>3E−04</td>
<td>−</td>
<td>−</td>
<td>−</td>
</tr>
<tr>
<td>cell cycle checkpoint</td>
<td>3E−04</td>
<td>−</td>
<td>−</td>
<td>−</td>
</tr>
<tr>
<td>macromolecule biosynthetic process</td>
<td>4E−04</td>
<td>−</td>
<td>−</td>
<td>−</td>
</tr>
<tr>
<td>spindle organization and biogenesis</td>
<td>5E−04</td>
<td>−</td>
<td>−</td>
<td>−</td>
</tr>
<tr>
<td>cellular catabolic process</td>
<td>5E−04</td>
<td>−</td>
<td>−</td>
<td>−</td>
</tr>
<tr>
<td>cellular protein metabolic process</td>
<td>9E−04</td>
<td>−</td>
<td>0.658</td>
<td>−</td>
</tr>
<tr>
<td>ubiquitin-dependent protein catabolic process</td>
<td>9E−04</td>
<td>−</td>
<td>−</td>
<td>−</td>
</tr>
<tr>
<td>tRNA aminoacylation for protein translation</td>
<td>0.001</td>
<td>−</td>
<td>−</td>
<td>−</td>
</tr>
<tr>
<td>amino acid activation</td>
<td>0.001</td>
<td>−</td>
<td>−</td>
<td>−</td>
</tr>
<tr>
<td>tRNA aminoacylation</td>
<td>0.001</td>
<td>−</td>
<td>−</td>
<td>−</td>
</tr>
<tr>
<td>tRNA processing</td>
<td>0.001</td>
<td>−</td>
<td>−</td>
<td>−</td>
</tr>
<tr>
<td>modification-dependent macromolecule catabolic process</td>
<td>0.001</td>
<td>−</td>
<td>−</td>
<td>−</td>
</tr>
<tr>
<td>microtubule-based process</td>
<td>0.001</td>
<td>−</td>
<td>−</td>
<td>−</td>
</tr>
<tr>
<td>modification-dependent protein catabolic process</td>
<td>0.001</td>
<td>−</td>
<td>−</td>
<td>−</td>
</tr>
<tr>
<td>proteolysis involved in cellular protein catabolic process</td>
<td>0.001</td>
<td>−</td>
<td>−</td>
<td>−</td>
</tr>
<tr>
<td>cellular protein catabolic process</td>
<td>0.002</td>
<td>−</td>
<td>−</td>
<td>−</td>
</tr>
<tr>
<td>cellular macromolecule metabolic process</td>
<td>0.002</td>
<td>−</td>
<td>−</td>
<td>−</td>
</tr>
<tr>
<td>nucleobase, nucleoside and nucleotide metabolic process</td>
<td>0.003</td>
<td>−</td>
<td>−</td>
<td>−</td>
</tr>
</tbody>
</table>

Continued on next page
Table 4.7: Enriched annotations for gene targets in STAT1 network model II (continued).

<table>
<thead>
<tr>
<th>GO Biological Process</th>
<th>MoNsTEr</th>
<th>Expr</th>
<th>ChIP</th>
<th>PWM</th>
</tr>
</thead>
<tbody>
<tr>
<td>RNA transport</td>
<td>0.004</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>nucleic acid transport</td>
<td>0.004</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>nucleobase, nucleoside, nucleotide and nucleic acid transport</td>
<td>0.004</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>DNA recombination</td>
<td>0.006</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>double-strand break repair</td>
<td>0.006</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>RNA localization</td>
<td>0.007</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>protein catabolic process</td>
<td>0.007</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>interphase of mitotic cell cycle</td>
<td>0.01</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>transcription initiation</td>
<td>0.011</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>nucleoside monophosphate biosynthetic process</td>
<td>0.012</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>nucleoside monophosphate metabolic process</td>
<td>0.012</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>interphase</td>
<td>0.012</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>amino acid metabolic process</td>
<td>0.017</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>nucleotide biosynthetic process</td>
<td>0.017</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>sterol biosynthetic process</td>
<td>0.022</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>mRNA transport</td>
<td>0.023</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>translational initiation</td>
<td>0.025</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>recombinational repair</td>
<td>0.026</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>double-strand break repair via homologous recombination</td>
<td>0.026</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>nucleoside phosphate metabolic process</td>
<td>0.032</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>nucleotide metabolic process</td>
<td>0.032</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>DNA replication initiation</td>
<td>0.038</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>ribonucleoside monophosphate biosynthetic process</td>
<td>0.039</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>ribonucleoside monophosphate metabolic process</td>
<td>0.039</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>coenzyme metabolic process</td>
<td>0.044</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>regulation of cellular metabolic process</td>
<td>–</td>
<td>–</td>
<td>0.002</td>
<td>–</td>
</tr>
<tr>
<td>regulation of nucleobase, nucleoside, nucleotide and nucleic acid metabolic process</td>
<td>–</td>
<td>–</td>
<td>0.002</td>
<td>–</td>
</tr>
<tr>
<td>ubiquitin cycle</td>
<td>–</td>
<td>–</td>
<td>0.024</td>
<td>–</td>
</tr>
<tr>
<td>regulation of RNA metabolic process</td>
<td>–</td>
<td>–</td>
<td>0.025</td>
<td>–</td>
</tr>
<tr>
<td>regulation of transcription</td>
<td>–</td>
<td>–</td>
<td>0.03</td>
<td>–</td>
</tr>
<tr>
<td>DNA damage response, signal transduction resulting in induction of apoptosis</td>
<td>–</td>
<td>–</td>
<td>0.038</td>
<td>–</td>
</tr>
<tr>
<td>regulation of transcription, DNA-dependent</td>
<td>–</td>
<td>–</td>
<td>0.044</td>
<td>–</td>
</tr>
</tbody>
</table>

Continued on next page
Table 4.7: Enriched annotations for gene targets in STAT1 network model II (continued).

<table>
<thead>
<tr>
<th>GO Biological Process</th>
<th>MoNsTEr</th>
<th>Expr</th>
<th>ChIP</th>
<th>PWM</th>
</tr>
</thead>
<tbody>
<tr>
<td>negative regulation of nucleobase, nucleoside, nucleotide and nucleic acid metabolic process</td>
<td>–</td>
<td>–</td>
<td>0.049</td>
<td>–</td>
</tr>
<tr>
<td>inflammatory response</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>0.045</td>
</tr>
</tbody>
</table>

**KEGG Pathway**

<table>
<thead>
<tr>
<th>Pathway</th>
<th>p-value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oxidative phosphorylation</td>
<td>2E−19</td>
</tr>
<tr>
<td>Proteasome</td>
<td>7E−11</td>
</tr>
<tr>
<td>Cell cycle</td>
<td>2E−09</td>
</tr>
<tr>
<td>Pyrimidine metabolism</td>
<td>3E−09</td>
</tr>
<tr>
<td>Aminoacyl-tRNA biosynthesis</td>
<td>1E−05</td>
</tr>
<tr>
<td>Purine metabolism</td>
<td>3E−05</td>
</tr>
<tr>
<td>RNA polymerase</td>
<td>3E−04</td>
</tr>
<tr>
<td>DNA polymerase</td>
<td>4E−04</td>
</tr>
<tr>
<td>Basal transcription factors</td>
<td>0.003</td>
</tr>
<tr>
<td>One carbon pool by folate</td>
<td>0.017</td>
</tr>
<tr>
<td>Ubiquitin mediated proteolysis</td>
<td>0.02</td>
</tr>
<tr>
<td>Biosynthesis of steroids</td>
<td>0.023</td>
</tr>
</tbody>
</table>
Table 4.8: Significantly enriched annotations (1st column) for gene targets in STAT1 network model III, using all gene targets in the input data as the background, with Bonferroni-corrected p-values (2nd column). As a control, corrected p-values are also shown for the most correlated genes (3rd column), and genes with the highest ChIP-seq and PWM-based priors (4th and 5th columns). ‘–’ indicates a p-value > 1 after Bonferroni correction.

<table>
<thead>
<tr>
<th>GO Biological Process</th>
<th>MoNsTEr</th>
<th>Expr</th>
<th>ChIP</th>
<th>PWM</th>
</tr>
</thead>
<tbody>
<tr>
<td>neurological system process</td>
<td>9E–26</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>G-protein coupled receptor protein signaling pathway</td>
<td>2E–19</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>synaptic transmission</td>
<td>2E–17</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>cell-cell signaling</td>
<td>5E–17</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>system development</td>
<td>9E–16</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>transmission of nerve impulse</td>
<td>4E–15</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>cell surface receptor linked signal transduction</td>
<td>5E–12</td>
<td>0.005</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>nervous system development</td>
<td>4E–11</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>ion transport</td>
<td>3E–10</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>sensory perception</td>
<td>3E–09</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>organ development</td>
<td>1E–07</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>metal ion transport</td>
<td>1E–07</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>neuron development</td>
<td>8E–07</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>tissue development</td>
<td>3E–06</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>neurogenesis</td>
<td>4E–06</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>axonogenesis</td>
<td>7E–06</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>sensory perception of light stimulus</td>
<td>8E–06</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>visual perception</td>
<td>8E–06</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>generation of neurons</td>
<td>9E–06</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>G-protein signaling, coupled to cyclic nucleotide second messenger</td>
<td>1E–05</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>neurite development</td>
<td>2E–05</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>neuron morphogenesis during differentiation</td>
<td>2E–05</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>neurite morphogenesis</td>
<td>2E–05</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>cellular morphogenesis during differentiation</td>
<td>4E–05</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>cyclic-nucleotide-mediated signaling</td>
<td>6E–05</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>potassium ion transport</td>
<td>7E–05</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>neuron differentiation</td>
<td>1E–04</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>cation transport</td>
<td>2E–04</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>feeding behavior</td>
<td>2E–04</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
</tbody>
</table>

Continued on next page
Table 4.8: Enriched annotations for gene targets in STAT1 network model III (continued).

<table>
<thead>
<tr>
<th>GO Biological Process</th>
<th>MoNsTEr</th>
<th>Expr</th>
<th>ChIP</th>
<th>PWM</th>
</tr>
</thead>
<tbody>
<tr>
<td>muscle system process</td>
<td>5E−04</td>
<td>−</td>
<td>−</td>
<td>−</td>
</tr>
<tr>
<td>muscle contraction</td>
<td>5E−04</td>
<td>−</td>
<td>−</td>
<td>−</td>
</tr>
<tr>
<td>monovalent inorganic cation transport</td>
<td>5E−04</td>
<td>−</td>
<td>−</td>
<td>−</td>
</tr>
<tr>
<td>signal transduction</td>
<td>7E−04</td>
<td>−</td>
<td>−</td>
<td>−</td>
</tr>
<tr>
<td>G-protein signaling, coupled to cAMP nucleotide</td>
<td>0.003</td>
<td>−</td>
<td>−</td>
<td>−</td>
</tr>
<tr>
<td>second messenger</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>second-messenger-mediated signaling</td>
<td>0.003</td>
<td>−</td>
<td>−</td>
<td>−</td>
</tr>
<tr>
<td>cAMP-mediated signaling</td>
<td>0.013</td>
<td>−</td>
<td>−</td>
<td>−</td>
</tr>
<tr>
<td>neuropeptide signaling pathway</td>
<td>0.014</td>
<td>−</td>
<td>−</td>
<td>−</td>
</tr>
<tr>
<td>sodium ion transport</td>
<td>0.016</td>
<td>−</td>
<td>−</td>
<td>−</td>
</tr>
<tr>
<td>skeletal development</td>
<td>0.021</td>
<td>0.860</td>
<td>−</td>
<td>−</td>
</tr>
<tr>
<td>ectoderm development</td>
<td>0.026</td>
<td>−</td>
<td>−</td>
<td>−</td>
</tr>
<tr>
<td>central nervous system development</td>
<td>0.028</td>
<td>−</td>
<td>−</td>
<td>−</td>
</tr>
<tr>
<td>brain development</td>
<td>0.041</td>
<td>−</td>
<td>−</td>
<td>−</td>
</tr>
<tr>
<td>biopolymer metabolic process</td>
<td>−</td>
<td>−</td>
<td>6E−18</td>
<td>−</td>
</tr>
<tr>
<td>nucleobase, nucleoside, nucleotide and nucleic acid metabolic process</td>
<td>−</td>
<td>−</td>
<td>7E−15</td>
<td>−</td>
</tr>
<tr>
<td>RNA metabolic process</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>response to DNA damage stimulus</td>
<td>−</td>
<td>−</td>
<td>9E−06</td>
<td>−</td>
</tr>
<tr>
<td>ubiquitin cycle</td>
<td>−</td>
<td>−</td>
<td>2E−04</td>
<td>−</td>
</tr>
<tr>
<td>regulation of cellular metabolic process</td>
<td>−</td>
<td>−</td>
<td>3E−04</td>
<td>−</td>
</tr>
<tr>
<td>DNA metabolic process</td>
<td>−</td>
<td>−</td>
<td>4E−04</td>
<td>−</td>
</tr>
<tr>
<td>protein metabolic process</td>
<td>−</td>
<td>−</td>
<td>4E−04</td>
<td>−</td>
</tr>
<tr>
<td>RNA biosynthetic process</td>
<td>−</td>
<td>−</td>
<td>0.002</td>
<td>−</td>
</tr>
<tr>
<td>DNA repair</td>
<td>−</td>
<td>−</td>
<td>0.002</td>
<td>−</td>
</tr>
<tr>
<td>transcription, DNA-dependent</td>
<td>−</td>
<td>−</td>
<td>0.002</td>
<td>−</td>
</tr>
<tr>
<td>regulation of nucleobase, nucleoside, nucleotide and nucleic acid metabolic process</td>
<td>−</td>
<td>−</td>
<td>0.003</td>
<td>−</td>
</tr>
<tr>
<td>cellular protein metabolic process</td>
<td>−</td>
<td>−</td>
<td>0.005</td>
<td>−</td>
</tr>
<tr>
<td>regulation of RNA metabolic process</td>
<td>−</td>
<td>−</td>
<td>0.006</td>
<td>−</td>
</tr>
<tr>
<td>regulation of transcription</td>
<td>−</td>
<td>−</td>
<td>0.008</td>
<td>−</td>
</tr>
<tr>
<td>regulation of transcription, DNA-dependent</td>
<td>−</td>
<td>−</td>
<td>0.009</td>
<td>−</td>
</tr>
<tr>
<td>cellular macromolecule metabolic process</td>
<td>−</td>
<td>−</td>
<td>0.009</td>
<td>−</td>
</tr>
<tr>
<td>biopolymer modification</td>
<td>−</td>
<td>−</td>
<td>0.019</td>
<td>−</td>
</tr>
<tr>
<td>RNA splicing</td>
<td>−</td>
<td>−</td>
<td>0.029</td>
<td>−</td>
</tr>
</tbody>
</table>

Continued on next page
Table 4.8: Enriched annotations for gene targets in STAT1 network model III (continued).

<table>
<thead>
<tr>
<th>GO Biological Process</th>
<th>MoNsTEr</th>
<th>Expr</th>
<th>ChIP</th>
<th>PWM</th>
</tr>
</thead>
<tbody>
<tr>
<td>mRNA processing</td>
<td>–</td>
<td>–</td>
<td>0.038</td>
<td>–</td>
</tr>
<tr>
<td>mRNA metabolic process</td>
<td>–</td>
<td>–</td>
<td>0.044</td>
<td>–</td>
</tr>
<tr>
<td>response to wounding</td>
<td>–</td>
<td>–</td>
<td>0.048</td>
<td></td>
</tr>
</tbody>
</table>

**KEGG Pathway**

<table>
<thead>
<tr>
<th>KEGG Pathway</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Neuroactive ligand-receptor interaction</td>
<td>2E−26</td>
<td>0.77</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>Calcium signaling pathway</td>
<td>0.001</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>Cell Communication</td>
<td>0.013</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>C21-Steroid hormone metabolism</td>
<td>0.042</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>p53 signaling pathway</td>
<td>–</td>
<td>–</td>
<td>0.045</td>
<td>–</td>
</tr>
</tbody>
</table>
Chapter 5

Reliable Identification of PTMs Using Mass Spectrometry

This chapter adapted from Everett LJ, Bierl C, and Master SR “Unbiased statistical analysis for multi-stage proteomic search strategies” Journal of Proteome Research 2010 9(2):700–707 with permission from the American Chemical Society

5.1 Introduction

A major paradigm in proteomics is that of bottom-up sequencing by tandem mass spectrometry (tandem MS). In bottom-up proteomics, proteins are digested into shorter peptides with a sequence-specific protease, separated by high-performance liquid chromatography, and ionized into the mass spectrometer by an electrospray interface. Inside the mass spectrometer, the mass-to-charge ratio ($m/z$) is measured for each ionized peptide, and the intensity of each peak in the spectrum represents the relative abundance. In tandem MS, the dominant peptide species in this initial, or precursor, scan (MS1) are each isolated in an ion trap and fragmented, typically by
collision with an inert gas. Fragmentation methods are designed such that peptides tend to fragment at an amide bond along the peptide backbone, and one of the resulting fragments retains a charge. After fragmentation, the resulting product ions are scanned again (MS2), measuring the m/z values for the peptide fragments. Thus, the precursor mass and MS2 spectrum can be observed for thousands of peptides in a single experiment. Each PTM alters the mass of the modified peptide, and therefore alters the intact m/z value and a subset of peaks in the MS2 spectrum generated by the PTM-containing peptide [101]. For a detailed review of typical LC-MS/MS workflows, see ref. [181].

The primary analytical task in computational proteomics is to identify the biological peptide that generated each MS2 spectrum. In theory, it is possible to identify a series of peaks in the MS2 spectrum such that the difference in masses correspond to each residue in the peptide (with any PTM). However, this task is often complicated by the fact that a typical MS2 spectrum contains at least two such ion series, corresponding to charge retention on either the amino- or carboxy-terminus, and neither series is complete in most cases. Noise peaks, fragmentation of other bonds, higher charge states, and imperfect mass accuracy further complicate the task of sequencing an MS2 spectrum de novo [181].

A popular solution to sequencing by tandem MS is to instead compare experimental spectra to a database\(^1\) of known protein sequences, typically translated from open reading frames in the genome. Each matched pair in a database search—consisting of a MS2 spectrum and its corresponding peptide—is designated a spectrum match. Thus, the complete solution space for any database search problem is the set of all peptide sequences in the database, with all possible PTM states.

\(^1\)The known sequences used for proteomic database searching are actually stored in a simple text format, but this file is often referred to as the sequence “database”.
Given the large number of possible PTMs, and the combinatoric complexity of multiple PTMs, it is impractical to search this solution space exhaustively, especially when the problem must be solved for thousands of spectra generated in a single experiment.

A common analysis strategy in proteomic database search algorithms, such as X!Tandem [34] and Mascot [153], is to further limit the search space in several ways. For each individual spectrum, the search space is also limited to peptides within a predefined window around the measured parent ion mass. Furthermore, the peptides derived from the sequence database can be limited to the cleavage patterns expected from the digest enzyme used in sample preparation. At best, only a few potential PTM types can be considered in a standard proteomics data analysis, even though a much wider variety could be potentially be inferred from the available data [147, 150]. In other words, database search algorithms search only the a priori most likely portions of the solution space at the cost of unexpected, but nonetheless biologically interesting, spectrum matches.

**Multi-Stage Search Strategies**

Any search strategy can be considered to have multiple “stages” if one or more additional peptide identification steps are taken based on information from spectrum matches identified by searching the same data set. A typical example of this is the model refinement option in X!Tandem [33], which first identifies a list of high-confidence proteins based on fully enzymatic, unmodified peptides (initial stage) and then searches this list of proteins for a larger range of possible peptides, including peptides with one or more PTMs (refinement stage). The initial stage is also typically limited based on knowledge of the enzyme used for protein digestion,
most commonly trypsin. The refinement stage is therefore typically used to identify nonenzymatic peptides resulting from in vivo cleavage, atypical digestion, or unexpected fragmentation during the ionization process, in addition to PTM identification. A related problem to PTM identification is that of identifying single amino acid substitutions—point mutations—which alter the mass of a residue in the peptide sequence and therefore are mathematically equivalent to PTMs within the context of the spectrum matching problem.

Other examples of multi-stage search include error tolerant search option in Mascot [35], and the iterative search options in VEMS [135] and Spectrum Mill. The multi-stage concept does not include strategies in which multiple searches are performed independently and then combined, e.g., Scaffold [174]. This concept also does not necessarily include strategies in which a single search stage is followed by one or more filtering or statistical analysis steps, e.g., PeptideProphet [108]. In other words, for the method to be considered a multi-stage search strategy as discussed in this work, each stage of the search must result in new matches, and stages must be performed in an iterative manner.

The primary motivation for a multi-stage search strategy is that wide searches—e.g., searches including peptides with nonenzymatic cleavages or multiple PTMs—are often too slow and prone to false positives to be efficiently performed using a single-stage strategy. This is especially true for mammalian proteomes, in which even the limited search space of fully enzymatic and unmodified peptides is already substantially large.

A secondary motivation for performing a multi-stage search is that the goals of narrow searches are often different than the goals of wide searches. In a narrow search, the primary goals are typically protein identification and quantification. A wide search may include additional goals such as protein characterization through
analyzing the biologically relevant PTMs of the protein. The basis of X!Tandem model refinement is that narrow searches are sensitive enough for the goal of protein identification, and thus should be used initially for this task [33]. Subsequent goals, including increasing the confidence of the initial protein identifications and characterizing the PTM states of these proteins, can then be achieved through wider searches that are specific to the initially identified proteins.

**False Discovery Rate in Computational Proteomics**

A single LC-MS/MS experiment can generate tens of thousands of MS2 spectra. Thus, database searching requires the evaluation a large number of hypotheses, and is prone to false positives if not corrected for multiple-testing. Furthermore, it is well-established that the expected null distribution of peptide match scores varies depending on the instrument parameters [30, 108, 114]. This presents a serious challenge when filtering to a high-confidence set of peptide matches.

Proteomic database search results are typically controlled by *False Discovery Rate* (FDR)—the expected proportion of false positive spectrum matches among all accepted matches. However, the lack of a reliable null distribution makes it difficult to use p-value-based approaches for FDR estimation [54]. One common solution to this problem is the empirical estimation of the FDR by searching the same set of experimental spectra against an equivalent database of “decoy” peptides, e.g., the reverse of all sequences in the target database [46, 104, 105].

While impossible to compute analytically, the accuracy of decoy-based methods have been demonstrated empirically using control samples of known protein analytes [109, 114]. Spectra generated from control samples can be searched against a database containing the sequences of the true protein components, as well as a
large number of additional sequences, such as the complete proteome of a distantly related species. In this case, two forms of the FDR can be estimated. The **control FDR** is computed from the known identities of proteins in the control sample, and relies on the assumption that spurious spectrum matches to these proteins are negligibly rare if the sequence database contains a sufficient number of additional sequences. The **decoy FDR** is computed using a reverse database, i.e., the same strategy used for experimental samples of unknown composition. The control and decoy FDR estimates can be compared at varying spectrum match score thresholds to demonstrate the overall accuracy of the decoy FDR estimate.

**FDR Estimation in Multi-Stage Search Strategies**

Despite the fact that multi-stage search strategies have been proposed in the literature and implemented in several common search algorithms [33, 35, 147], little consideration has been given to their compatibility with popular statistical methods such as decoy-based estimates of the FDR [46, 104, 105] and validation using control samples [109, 114]. In this chapter, I identify key assumptions made by these methods that can become problematic in validating and using a multi-stage search strategy. I demonstrate these problems using a set of spectra generated from control samples and analyzed with the most widely used multi-stage search tool: the model refinement option in **X!Tandem** [33]. I present solutions to these problems, and implement one critical solution—the maintenance of forward and reverse protein pairs in the model refinement stage—as a patch for the publicly available **X!Tandem** source code [34]. I argue that, in general, multi-stage search strategies can be used both to speed up protein identification tasks and to expand the scope of proteomics analyses. However, special care must be taken to ensure that downstream statistical
methods are not compromised in the process. The methods presented here provide a template for validating other combinations of multi-stage search strategies and statistical methods.

5.2 Materials and Methods

Preparation of Control Solution

A control solution of known proteins was prepared using the Universal Proteomics Standard (Sigma-Aldrich, UPS-1), an equimolar mixture of 48 known proteins. Proteins were resuspended in 50 µL of extraction reagent prior to reduction and alkylation with iodoacetamide (IAA) per the manufacturer’s instructions (PROT-RA ProteoPrep kit, Sigma-Aldrich). Proteins were digested using a Trypsin Spin Column (TT0010, Sigma-Aldrich), and peptides were eluted with 2 rounds of 150 µL of enzyme reaction buffer. The final volume was diluted to 500 µL and aliquotted into 10 µL quantities, which were further diluted (10×) for an estimated final concentration of 1 fmol/µL.

LC-MS/MS Protocol

A single aliquot of control solution was analyzed in 6 technical replicates. For each replicate, 1 µL (estimated 1 fmol of protein) of sample was loaded onto a reverse phase LC column for 10 min, then separated using a 30-min gradient of 12-34% acetonitrile at 300 nL/min. nESI-MS/MS was performed using an Advion TriVersa Nanomate connected to a Thermo LTQ-Orbitrap. Spectra were obtained in data-dependent mode, with MS1 spectra obtained with high mass accuracy and resolution ($R = 60,000$) in the Orbitrap and MS2 spectra for the top 5 peaks obtained in the
<table>
<thead>
<tr>
<th>Replicate</th>
<th>MS2 spectra</th>
<th>Search Times (sec)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Single</td>
</tr>
<tr>
<td>1</td>
<td>4,519</td>
<td>698</td>
</tr>
<tr>
<td>2</td>
<td>4,344</td>
<td>845</td>
</tr>
<tr>
<td>3</td>
<td>4,414</td>
<td>845</td>
</tr>
<tr>
<td>4</td>
<td>4,039</td>
<td>600</td>
</tr>
<tr>
<td>5</td>
<td>4,245</td>
<td>678</td>
</tr>
<tr>
<td>6</td>
<td>4,432</td>
<td>842</td>
</tr>
<tr>
<td>Total</td>
<td>25,993</td>
<td>4,508</td>
</tr>
</tbody>
</table>

Table 5.1: Replicate Data for Proteomics Analysis. Each replicate was generated from a control sample of 48 known proteins. The number of MS2 spectra extracted from each RAW file is shown in the second column. Run times for semi-tryptic searches are shown in columns 3 (single-stage search), 4 (multi-stage search), and 5 (multi-stage search with corrected X!Tandem code). Searches were run on a laptop with a 2.4 GHz Intel Core 2 Duo using a single thread.

linear ion trap. The 6 replicate runs resulted in a total of 25,993 MS2 spectra (Table 5.1).

**Preparation of Search Databases**

The complete sequences for all proteins in the control mix (true positives) were supplemented with 4,939 *E. coli* proteins that serve as known false positives. In addition, the sequences for porcine trypsin and common contaminant proteins such as keratins were added to the database used for all searches.

A separate version of this database was also generated by manually introducing point mutations *in silico* into the sequences of control proteins. The locations of point mutations were chosen to avoid changing any potential tryptic cleavage sites. Locations were also chosen to occur in peptides that were identifiable in an initial search of spectra using the database of correct sequences. At least one fully tryptic, unmodified peptide per protein was left unchanged in this database to guarantee that each protein could be identified in the initial search stage of X!Tandem.
Spectrum File Conversion and Database Searches

Files in RAW format were converted to DTA format using DeconMSn v1.6 [136], and then DTA files from each RAW file were converted to single mzXML files using the dta2mzxml tool packaged in the Trans-Proteomic Pipeline (TPP) v3.5.1 [107]. mzXML files corresponding to each technical replicate were searched individually against the databases described above using X!Tandem v2008.02.01.1 [34]. All searches were performed using a 50 ppm error tolerance on the parent ion mass and a 0.4 Da error tolerance on the fragment masses. Scoring of spectra was performed using the K-score module [131], unless otherwise noted. A fixed mass shift of 57.022 Da was applied to Cysteine residues to account for alkylation, and a potential mass shift of 15.994915 Da was allowed for Methionine oxidation. In most searches, reversed sequences were generated on-the-fly by X!Tandem and searched simultaneously with the forward sequences. Alternatively, a reversed sequence database was pre-generated and searched separately from the forward sequences, where noted. In multi-stage searches, the refinement threshold was set to an Expect value of 0.1. I developed a modified version of the X!Tandem source code to include the “Re-Rev” option, which can only be used with model refinement and on-the-fly generation of reversed sequences. This option changes how X!Tandem constructs the refinement list—the shortened list of proteins to consider in the refinement stage—by first filtering all reversed sequences from the list, then including the reversed sequence of each protein on the list.

Fully tryptic peptides are defined as those with a cleavage site matching the pattern [RK]|{P}, i.e., cleavage C-terminal to an Arginine or Lysine not followed by a Proline, at both termini (except at the protein termini), and no more than two sites matching this pattern within the peptide sequence (missed cleavages).
“Semi-tryptic” peptides are defined as having one terminus that does not match the cleavage pattern. Peptides containing 3–5 missed cleavage sites are also considered; for convenience of nomenclature, I group these together into the “semi-tryptic” peptide definition. For semi-tryptic multi-stage searches, the initial search stage was used to identify fully tryptic peptides, and the refinement stage was used to identify semi-tryptic peptides. For the comparable single-stage searches, the initial search was used to identify both fully tryptic and semi-tryptic peptides for all proteins in the database; no refinement stage was performed. For the point mutation searches, an initial stage was used to identify fully tryptic peptides in the mutated database described above, and the refinement stage was used to consider all possible point mutations. Semi-tryptic peptides were not considered in the point mutation searches, or vice versa.

**FDR Analysis**

The search results for each technical replicate were pooled together into a single spectrum match set for each search strategy performed. For each unique spectrum match score, the FDR for the subset of all matches with scores less than or equal to this threshold was estimated using two different strategies: (i) using the known identity of control proteins as true positives, called *control FDR* [109, 114], and (ii) using the number of reverse matches divided by the number of forward matches, called *decoy FDR* [46, 104, 105]. These FDR calculations are shown in Eq 5.1 and Eq 5.2, respectively, for an arbitrary scoring method that produces scores \( x \). In different analyses, I applied these FDR calculations to thresholds on the default “hyperscore” [34], the \( K \)-score [131], and the Expect score [57] (in which case, the threshold was \( x \leq X \)). Unless otherwise noted, FDR calculations are based on
Expect score thresholds. Additional FDR values were also calculated for: (i) only those spectrum matches identified in the initial search, and (ii) only those spectrum matches identified in the refinement stage.

\[
\text{Control FDR}(X) = \frac{\text{# of Incorrect Matches s.t. } x \geq X}{\text{# of Total Matches s.t. } x \geq X} \quad (5.1)
\]

\[
\text{Decoy FDR}(X) = \frac{\text{# of Reverse Matches s.t. } x \geq X}{\text{# of Forward Matches s.t. } x \geq X} \quad (5.2)
\]

When computing control FDR for a multi-stage semi-tryptic search, matches from the refinement stage were only considered correct if they could also be obtained from the comparable single-stage search. When computing control FDR for a point mutation search, matches from refinement were only considered correct if the point mutation matched the true control sequence of the protein in the unaltered database. Spectrum matches to contaminant proteins, including the trypsin enzyme used for digestion, were excluded from all FDR calculations.

The control FDR and decoy FDR estimates were paired for each score threshold and plotted to visually assess the accuracy of the decoy FDR. In these plots, a line that roughly follows the diagonal (decoy FDR \(\approx\) control FDR) indicates a reliable estimate of the FDR using decoy matches. A line that skews into the top-left quadrant of the plot (decoy FDR \(\ll\) control FDR) indicates an underestimation of the FDR using decoy matches. A line that skews into the bottom-right quadrant of the plot (decoy FDR \(\gg\) control FDR) indicates an overestimation of the FDR using decoy matches. Plots were smoothed by replacing the FDR value at each threshold \(X\) with the lowest FDR value from the same estimation method at a threshold greater than or equal to \(X\). This is equivalent to computing a \(q\)-value [30, 104, 105].

At stricter thresholds \(X\) corresponding to low FDR values (1–5%), the decoy FDR estimate is computed from a small number of decoy matches, and therefore
is expected to have a larger variance due to sampling effects. Therefore, I also applied a previously published error model [95] to the decoy FDR estimates, with the alteration that I only computed $F(a|n)$ for values of $a$ ranging from 0 to the number of observed forward matches. This allows for a comparison between the control FDR and the 95% confidence interval of the decoy FDR estimate, to account for sampling effects at low FDR.

5.3 Results

Control Sample Validation Bias in Multi-Stage Search Strategies

The reliability and accuracy of statistical methods in proteomics can be demonstrated empirically using spectra generated from control solutions of known protein content [109, 114]. Although simplified control mixtures may not fully capture the complexity seen in some experimental samples, they nonetheless have functioned as a valuable standard for validating spectrum matches. The basic assumption of this approach is that spectrum matches to control proteins (hereafter designated control matches) are correct based on a priori knowledge of the analyzed sample. As long as the search database contains a substantial number of additional, unrelated proteins, the probability of a false positive match to a control protein is negligible. Thus, a highly accurate estimate of the FDR can be computed from the control sequence matches (the control FDR) and compared to more generally applicable estimates of the FDR, such as the decoy FDR computed from reverse sequences.

The simple strategy just described is sufficient for validating statistical methods for single-stage searches, but it is insufficient for multi-stage searches. The key
problem is that secondary search stages, such as X!Tandem model refinement, are often used for characterizing the state of a protein, such as the presence of PTMs. While a simple strategy can take advantage of a priori knowledge of the sample content, there is often no comparable a priori knowledge regarding sample characterization. That is, while a subset of well-known modifications on control proteins may be known, there is insufficient knowledge on the complete set of potential modifications across all control proteins to unambiguously label every modified or nonenzymatic spectrum match as true or false. Thus, a search stage that specifically aims to identify these types of peptides has no equivalent rule on which to base the computation of the control FDR.

Another way to conceptualize this problem is as follows: during the initial search stage (or, equivalently, for a single-stage search), the set of control proteins is only a small portion of the overall search space. Thus, false positive matches to the control proteins occur rarely by chance and the correctness of control matches can be assumed. However, in the model refinement stage of a multi-stage search, the list of proteins to search becomes enriched for control sample proteins, and false positive matches occur with high probability.

To demonstrate this problem, I searched spectra from a set of six control sample replicates with X!Tandem using a database of 48 known control solution proteins and 4,939 E. coli proteins, using the multi-stage model refinement option to identify semi-tryptic peptides in the refinement stage only. These searches identified a total of 2,047 semi-tryptic matches from the control proteins (Figure 5.1). However, when these spectra were searched against the same database using a single-stage search for both fully tryptic and semi-tryptic peptides, only 261 semi-tryptic control matches were identified, the majority of which overlapped matches from the multi-stage search (Figure 5.1B). The majority of refinement-stage control matches could not
be corroborated by a single-stage search and are likely to be false positives based on the violation of the key assumption discussed above. Conversely, the matches from the initial search stage, in which the necessary assumption is valid, are mostly confirmed by a single-stage search (Figure 5.1A).

To test the assumption that semi-tryptic control matches from the refinement stage are invalid if not confirmed by a single-stage search, I analyzed the distributions of delta mass (Figure 5.2A), $K$-score (Figure 5.2B), and Expect score (Figure 5.2C) on the semi-tryptic control matches occurring in both search results (Figure 5.2, solid lines) or only in the refined search (Figure 5.2, dashed lines). For comparison, I also plot these distributions for known false positive matches to *E. coli* proteins (Figure 5.2, dotted line). Delta mass is the difference between the observed parent ion mass, measured at high mass accuracy for this data set, and the theoretical

---

**Figure 5.1**: Overlap of matches between single-stage search and multi-stage search for (A) fully tryptic matches (Initial stage of multi-stage search) and (B) semi-tryptic matches (Refinement stage of multi-stage search). Areas are not drawn to scale. Additionally, a negligible number (9) of control matches were identified in both search strategies but matched different control proteins or peptides (not shown). Reprinted from Everett LJ, Bierl C, and Master SR “Unbiased statistical analysis for multi-stage proteomic search strategies” *Journal of Proteome Research* 2010 9(2):700–707 with permission from the American Chemical Society.
mass of the peptide from the database. For correct matches, a narrow delta mass distribution centered near 0 is expected, while a wider delta mass distribution is indicative of incorrect matches. K-Score [131] is a measure of similarity between the observed MS2 spectrum and the theoretical spectrum. Lower K-Scores indicate poorer matches and in general are more likely to be false positives. Expect score [57] is the number of matches expected to occur by chance for a particular spectrum and is estimated using the K-score distribution from all theoretical spectra within the parent mass window of the observed spectrum. Lower Expect scores indicate more significant matches, and thus, the log transformation also follows this trend. All of these distributions are highly similar between the known false positive matches and the semi-tryptic control matches identified only in the multi-stage search. Therefore, simply using the list of known control proteins to mark matches as correct or incorrect is insufficient to accurately compute the control FDR when using multi-stage search strategies.
To more reliably identify true positive semi-tryptic matches, I next compared each match from the refinement stage to the match for the same spectrum in a comparable single-stage search (Figure 5.3A). If a spectrum matches the same control protein sequence in both the single-stage and multi-stage searches, then the spectrum match from the refinement search is marked as correct. Otherwise, it is considered a false positive when computing the control FDR. The decoy FDR, computed for the refinement search results based on the reverse matches, is then compared to the control FDR to determine its approximate accuracy.

**Decoy-Based FDR Bias in the Refinement Stage**

The decoy-based estimates of FDR [46, 104] rely on the basic assumption that false positive matches occur against the forward and reverse sequences with equal probability [30]. While this has been repeatedly demonstrated empirically for single-stage searches, the list of refinement proteins prepared by X!Tandem does not guarantee an equal number of forward and reverse proteins, nor should such a proportion be expected in practice.

There are typically very few reverse proteins in the refinement list and the probability of a false positive spectrum match to a reverse sequence in general becomes much lower than the probability of a false positive spectrum match to a forward protein. The result is an unacceptable underestimation of the FDR specifically within the set of matches obtained from the refinement stage (Figure 5.4B, dotted lines). This bias is masked in cases where most matches are obtained from the initial search (Figure 5.4A, dotted lines), but can still be problematic when the refinement stage is used for protein characterization (e.g., PTM identification) and subsequent analyses focus on is the spectrum matches returned by the refinement stage. I also analyzed
Figure 5.3: Analysis workflows to analyze FDR estimates on multi-stage search using the refinement stage to identify (A) semi-tryptic peptides and (B) point mutations. Semi-tryptic control matches are marked correct only if they can also be identified in a single-stage search. Control matches with point mutations are only correct if they match the original control sequence. *Marks the key step introduced into the X!Tandem source code. Reprinted from Everett LJ, Bierl C, and Master SR “Unbiased statistical analysis for multi-stage proteomic search strategies” Journal of Proteome Research 2010 9(2):700–707 with permission from the American Chemical Society.

these results for the refinement hits in the FDR range of 0.1% to 10% FDR, as these encompass FDRs used in most applications. The decoy FDR is compared to the control FDR for a number of typical thresholds in Table 5.2. All cases where the control FDR falls outside the 95% confidence interval for the estimated error range of the decoy FDR [95] are shown in bold.
Figure 5.4: Plot of the Control FDR, computed from known set of true and false positives, and the Decoy FDR, computed from reverse sequence matches. FDR values are plotted for subsets of the full result set over a range of possible Expect thresholds. A line close to the diagonal (black line) indicates a reliable Decoy FDR. The results using the semi-tryptic workflow without the Re-Rev patch (dotted blue line) and with the Re-Rev patch (dashed green line) are shown for (A) all spectrum matches and (B) only the semi-tryptic spectrum matches. Reprinted from Everett LJ, Bierl C, and Master SR “Unbiased statistical analysis for multi-stage proteomic search strategies” *Journal of Proteome Research* 2010 9(2):700–707 with permission from the American Chemical Society.

<table>
<thead>
<tr>
<th>Decoy FDR</th>
<th>Control FDR</th>
<th>Rev / Fwd Counts</th>
<th>95% CI</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1%</td>
<td>0%</td>
<td>0 / 101</td>
<td>NA</td>
</tr>
<tr>
<td>1%</td>
<td>0%</td>
<td>1 / 122</td>
<td>0%–4.1%</td>
</tr>
<tr>
<td>2%</td>
<td>1.6%</td>
<td>2 / 126</td>
<td>0%–5.6%</td>
</tr>
<tr>
<td>3%</td>
<td>3%</td>
<td>4 / 134</td>
<td>0%–7.5%</td>
</tr>
<tr>
<td>4%</td>
<td>9.9%</td>
<td>6 / 152</td>
<td>0%–8.6%</td>
</tr>
<tr>
<td>5%</td>
<td>20%</td>
<td>9 / 185</td>
<td>0.5%–9.2%</td>
</tr>
<tr>
<td>10%</td>
<td>90%</td>
<td>177 / 2129</td>
<td>NA</td>
</tr>
</tbody>
</table>

Table 5.2: FDR table for Semi-Tryptic analysis refinement-stage spectrum matches obtained from *X!Tandem*. Column 1 contains the limit placed on decoy FDR, Column 2 contains the control FDR calculated for the same Expect threshold, Column 3 contains the actual count of reverse and forward spectrum matches (needed for error model), and Column 4 contains the range of possible FDR values in the 95% Confidence Interval (CI) for the number of reverse matches observed [95]. CI was not computed for cases with 0 or greater than 25 decoys. Control FDR values that fall outside the 95% CI are shown in bold.
Artificial Point Mutations for Control Sample Validation

Identifying PTMs on specific residues in a complex mixture of (initially) unknown proteins is a common use for multi-stage search strategies [33, 35, 135]. As with semi-tryptic searches, this presents a problem for validation using control solutions because the set of dynamic PTMs on all control proteins is not known completely and would be laborious to validate experimentally. With this problem in mind, I propose another validative framework (Figure 5.3B) in which artificial point mutations are introduced in silico into the control sequence database, and these mutations are subsequently identified using the refinement stage. Each match identified in the refinement stage, that is, any match containing a point mutation, can be automatically considered correct or incorrect solely on its sequence matching the original database (prior to in silico mutation), without any need to run a comparable single-stage search. As noted earlier in this chapter, point mutations are conceptually similar to PTMs: both result in predictable mass shifts in the overall parent ion and a subset of the fragment ions.

To test the validity of this approach, I once again compared the property distributions of matches corresponding to intended point mutations, i.e., those added in silico (Figure 5.5, solid lines), and unintended point mutations, i.e., all others predicted by the search algorithm (Figure 5.5, dashed lines). Here, the distributions for all other false positive matches (E. coli, no point mutations) are provided for comparison (Figure 5.5, dotted lines). The distributions for unintended point mutations are more like the distributions for other false positives and are well-separated from the distributions of intended point mutations. Therefore, introducing known point mutations in silico is a reliable method for separating correct and incorrect matches from the refinement stage.
Figure 5.5: Distributions of spectrum match properties for point mutations matching the original control sequences (solid green lines) vs non-matching point mutation matches (dashed blue line) to demonstrate the reliability of in silico point mutations for automated validation on control samples. Also included is the distribution of other known false positive matches with no point mutations (dotted red line). Distributions shown for (A) Delta Mass, (B) K-score, (C) Expect (log transformed). Reprinted from Everett LJ, Bierl C, and Master SR “Unbiased statistical analysis for multi-stage proteomic search strategies” *Journal of Proteome Research* 2010 9(2):700–707 with permission from the American Chemical Society.

Plots of control vs. decoy FDR at varying Expect thresholds using this alternate analysis strategy are shown in Figure 5.6 (blue dotted lines). Once again, a strong bias is demonstrated in the estimation of decoy FDR for spectrum matches in the refinement stage (panel B). Results for the refinement stage spectrum matches at common FDR thresholds are tabulated in Table 5.3.

**An Improved Strategy for Unbiased FDR Estimates**

To correct the bias demonstrated in multi-stage refinement searches, I modified the X!Tandem source code to adjust the refinement list prior to a refinement search stage. Specifically, the initial refinement list contains all proteins for which at least one peptide was identified with an Expect score less than 0.1; this is the list of proteins to be considered in the refinement stage of the search. The adjustment that I implemented filters all reversed proteins matched by chance from this list,
Figure 5.6: Plot of the Control FDR and the Decoy FDR. The results using the point mutation workflow without the Re-Rev patch (dotted blue line) and with the Re-Rev patch (dashed green line) are shown for (A) all spectrum matches and (B) only the spectrum matches containing point mutations. Reprinted from Everett LJ, Bierl C, and Master SR “Unbiased statistical analysis for multi-stage proteomic search strategies” Journal of Proteome Research 2010 9(2):700–707 with permission from the American Chemical Society.

<table>
<thead>
<tr>
<th>Decoy FDR</th>
<th>Control FDR</th>
<th>Rev / Fwd Counts</th>
<th>95% CI</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1%</td>
<td>9.9%</td>
<td>0 / 160</td>
<td>NA</td>
</tr>
<tr>
<td>1%</td>
<td>13.2%</td>
<td>1 / 167</td>
<td>0%-3%</td>
</tr>
<tr>
<td>2%</td>
<td>36.1%</td>
<td>4 / 227</td>
<td>0%-4.4%</td>
</tr>
<tr>
<td>3%</td>
<td>79.5%</td>
<td>18 / 706</td>
<td>1%-4.1%</td>
</tr>
<tr>
<td>4%</td>
<td>79.5%</td>
<td>18 / 706</td>
<td>1%-4.1%</td>
</tr>
<tr>
<td>5%</td>
<td>79.5%</td>
<td>18 / 706</td>
<td>1%-4.1%</td>
</tr>
<tr>
<td>10%</td>
<td>79.5%</td>
<td>18 / 706</td>
<td>1%-4.1%</td>
</tr>
</tbody>
</table>

Table 5.3: FDR table for point mutation analysis refinement-stage spectrum matches obtained without the Re-Rev patch. Control FDR values shown in bold fall outside the 95% Confidence Interval (CI) computed for the corresponding Decoy FDR value [95].
Table 5.4: FDR table for semi-tryptic analysis refinement-stage spectrum matches obtained with the Re-Rev patch. All Control FDR values fall within the 95% Confidence Interval (CI) [95].

and then adds the corresponding reversed (decoy) protein for each forward (target) protein in the refinement list. The goal of this step is to maintain the approximately equal probability of false positive matches to forward and reverse sequences. To test this correction, I compared results from both semi-tryptic and point mutation refinement searches performed with and without refinement list adjustment. The results (Figures 5.4B and 5.6B, dashed lines) demonstrate that the marked FDR bias was removed in both the semi-tryptic and point mutation searches. Results for the refinement spectrum matches at common FDR thresholds are tabulated in Table 5.4 for semi-tryptic searches and Table 5.5 for point mutation searches. In semi-tryptic searches, this adjustment adds only several seconds to the run time of each search while still representing an overall reduction of 96% in the total search time as compared to the corresponding single-stage search (Table 5.1).

### FDR Bias is Independent of Search Parameters

In order to confirm that the results found in this analysis can be generalized to other FDR estimation methods, I performed similar analyses using slightly different search or estimation parameters. First, using the same search results and rules about defining false positives as in the analyses above, I recalculated the control
<table>
<thead>
<tr>
<th>Decoy FDR</th>
<th>Control FDR</th>
<th>Rev / Fwd Counts</th>
<th>95% CI</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1%</td>
<td>3%</td>
<td>0 / 132</td>
<td>NA</td>
</tr>
<tr>
<td>1%</td>
<td>3%</td>
<td>1 / 132</td>
<td>0%–3.8%</td>
</tr>
<tr>
<td>2%</td>
<td>5.9%</td>
<td>2 / 136</td>
<td>0%–5.1%</td>
</tr>
<tr>
<td>3%</td>
<td>6.6%</td>
<td>4 / 137</td>
<td>0%–7.3%</td>
</tr>
<tr>
<td>4%</td>
<td>8.6%</td>
<td>5 / 140</td>
<td>0%–7.9%</td>
</tr>
<tr>
<td>5%</td>
<td>9.2%</td>
<td>7 / 141</td>
<td>0%–9.9%</td>
</tr>
<tr>
<td>10%</td>
<td>13.5%</td>
<td>14 / 148</td>
<td>2.7%–16.2%</td>
</tr>
</tbody>
</table>

Table 5.5: FDR table for Point Mutation analysis refinement-stage spectrum matches obtained with the Re-Rev patch. Control FDR values shown in bold fall outside the 95% Confidence Interval computed for the corresponding Decoy FDR value [95].

FDR and decoy FDR using thresholds on the raw $K$-score value of each spectrum match (instead of the Expect score, used above). The results for the semi-tryptic validation pipeline are shown in Figure 5.7 and the results for the point mutation validation pipeline are shown in Figure 5.8. In both cases, there is still a clear underestimation of the FDR using decoy hits (blue dotted line in Figures 5.7B and 5.8B), which is corrected when using the Re-Rev patch to the X!Tandem code (green dashed line). Thus, I conclude that the observed bias is not an artifact of using the Expect score as a threshold for estimating FDR values.

I repeated the searches performed in the primary analysis using the default X!Tandem scoring module, or “hyperscore”, in place of the $K$-score module. Figures 5.9 and 5.10 show the score distributions for semi-tryptic and point mutation analyses, respectively. In both cases, the distribution of control protein spectrum matches from the refinement stage that I marked as false positives (based on the criteria established above) is more like the distribution of other false positive matches ($E. coli$ decoys) than the other presumed true positive matches. This confirms that the reliability of both validation methods is not specific to the $K$-score module. Figures 5.11 and 5.12 show the FDR estimates for the two validation methods using
Figure 5.7: FDR Analysis using $K$-score thresholds instead of Expect, with semi-tryptic peptides identified in refinement stage. Reprinted from Everett LJ, Bierl C, and Master SR “Unbiased statistical analysis for multi-stage proteomic search strategies” *Journal of Proteome Research* 2010 9(2):700–707 with permission from the American Chemical Society.

different thresholds on the Expect score. In both cases, there is still a clear underestimation of the FDR using decoy matches (blue dotted lines), which is corrected when using the Re-Rev patch to the X!Tandem code (green dashed lines). Thus, I conclude that the observed bias is not an artifact of using the $K$-score module.

Other reports on FDR analysis have suggested that it is generally more accurate to estimate FDR by comparing spectrum matches identified in separate database searches of the forward and reverse sequence databases [104]. I tested whether the observed bias here was simply a result of searching the forward and reverse databases simultaneously. For the semi-tryptic pipeline I performed three additional searches of the experimental spectra:

1. spectra vs. forward database, multi-stage search

2. spectra vs. forward database, single-stage search
3. spectra vs. reverse database, multi-stage search

To compute the control FDR, spectrum matches to the control proteins in search #1 were only considered correct if they were also identified in search #2. To estimate the FDR from decoy matches, at each threshold the decoy FDR was estimated as the number of spectrum matches from search #3 passing the threshold divided by the number of spectrum matches from search #1 passing the threshold.

For the point mutation pipeline, I performed two additional searches of the experimental spectra:

4. spectra vs. forward database (with point mutations), multi-stage search

5. spectra vs. reverse database (with point mutations), multi-stage search

As in the previous analyses, I considered point mutation spectrum matches from search #4 to be correct only if they matched the sequence in the true, unaltered
sequence database. To estimate the FDR from decoy matches, at each threshold the decoy FDR was estimated as the number of spectrum matches from search #5 passing the threshold divided by the number of spectrum matches from search #4 passing the threshold.

Figures 5.13 and 5.14 show the score distributions for the separated semi-tryptic and point mutation search analyses, respectively. As with the previous analyses, the semi-tryptic spectrum matches to control proteins identified in the refinement stage that do not pass the additional criteria indeed follow score distributions more like those of known false positive matches. Thus, both validation methods are also applicable when searches are performed separately on the forward and reverse sequence databases.

Figures 5.15 and 5.16 show the FDR estimates for both validation methods. In both cases, there is still substantial underestimation of the FDR for refinement matches when using reverse decoys (dotted blue lines, B panels). The X!Tandem Rev-Rev patch cannot be applied in the case of separate forward and reverse searches,
because the choice of reverse proteins used for refinement would need to be informed by the forward proteins selected for refinement in the separate forward search. Thus, there is no corrected FDR estimate shown in these figures. The bias in the decoy FDR is not as severe when using model refinement to identify semi-tryptic peptides, and may not be an issue within the desirable range of FDR cut-offs. However, there is still severe bias observed when using the model refinement for point mutations. Therefore, I conclude that separating the searches of the forward and reverse databases is not a general solution to the problems discussed in this chapter.

5.4 Discussion

High-throughput protein identification remains a fundamental challenge for modern proteomics. Additionally, the recent focus on identifying functional PTMs has added complexity by increasing the search space of peptide matches that must be explored. To address this problem, multi-stage search strategies have been developed to allow a more thorough search for PTMs on a subset of high-confidence proteins. Thus,
validation of search results includes both determining whether the protein was detected in some form (protein identification) as well as assessing whether interesting modifications have been reliably identified (protein characterization).

**Validation of Multi-Stage Search Algorithms**

In this chapter, I have identified two critical problems in the validation and use of multi-stage search algorithms such as the one used in X!Tandem model refinement. These problems are generally applicable to any multi-stage search strategy that limits subsequent search stages to a small set of high confidence proteins, such as the Error Tolerant Search option in Mascot [35]. One such problem is the inability to assign each spectrum match as a true or false positive based solely on its identity as a control sequence, a critical step for automated validation of a proteomic analysis.

Figure 5.11: FDR analysis using the default scoring module instead of K-score, threshold on Expect value (semi-tryptic refinement stage). Reprinted from Everett LJ, Bierl C, and Master SR “Unbiased statistical analysis for multi-stage proteomic search strategies” *Journal of Proteome Research* 2010 9(2):700–707 with permission from the American Chemical Society.
method. I have presented two different solutions to this problem. In one case, I used a comparable single-stage search to automatically validate each control sequence match identified in the refinement stage. Eighty-eight percent of the fully tryptic matches from a multi-stage search were identified in a corresponding single-stage search, while only 12% of the semi-tryptic matches (which correspond to refinement stage hits) had a corresponding single-stage match (Figure 5.1). This result is consistent with a marked increase in the false positive rate within control sequences for refinement results. Some misclassifications may occur with this method, and the 309 fully tryptic matches not identified in a single-stage search (Figure 5.1A) were likely missed as a result of interrogating a larger search space. Despite this limitation, the distribution of match characteristics from true- and false-positive groups identified using single-stage validation (Figure 5.2) provides evidence of the
overall validity of this approach.

As a second solution to the problem of identifying true-positive matches to control spectra, I introduced artificial point mutations \textit{in silico} into the control sequences in the database used by the search algorithm, thereby allowing us to identify true matches as those corresponding to the intended point mutations. This approach has the added benefit of more closely mimicking the search for peptides containing PTMs.

It should be noted that experimental samples may differ from these two test cases in certain important ways. For example, an experimental mixture may contain a much larger proportion of modified, semi-tryptic, or mutated peptides, resulting in a larger set of high-scoring spectrum matches in the refinement stage. Furthermore, mixtures derived from whole-cell lysates (rather than, e.g., pull-down experiments to characterize a protein complex) will be substantially more complex than the control mixtures used here, both in terms of the number of proteins present and the dynamic range. First-stage analysis of these complex mixtures will likely result
in a larger refinement list containing more false positive proteins, and this may dampen the extent of the second-stage bias in comparison with shorter refinement lists having only several decoy proteins by chance. However, since the proposed patch also guarantees a more reliable estimate in the case of complex mixtures and has no substantial drawbacks, I suggest that it be used for these sample types as well. The potential gains of performing a subsequent search stage may be assessed beforehand by, for example, applying a quality metric to unmatched spectra after the initial search [147].

**Decoy FDR Estimation with Multi-Stage Search Methods**

Another problem with statistical control of multi-stage searches is the marked bias in decoy FDR. A comparison of the validated control FDR against the estimated decoy FDR demonstrates that such estimates are highly skewed for the subset of matches identified during the search stages performed on high-confidence proteins (Figures 5.5 and 5.6). I have corrected this problem by guaranteeing equal numbers of corre-
Figure 5.15: FDR Analysis using separate searches of the forward and reverse database, with semi-tryptic peptides identified in the refinement stage. Reprinted from Everett LJ, Bierl C, and Master SR “Unbiased statistical analysis for multi-stage proteomic search strategies” *Journal of Proteome Research* 2010 9(2):700–707 with permission from the American Chemical Society.

responding forward and reverse protein pairs through all search stages, implemented as a minor patch to the *X!Tandem* source code. This correction amounts to maintaining the null model used for statistical validation: specifically, that false positive matches to target and decoy proteins occur with approximately equal probability and score distributions.

Because second-stage bias becomes most apparent at higher FDR thresholds, it is reasonable to ask whether the phenomenon is relevant at realistic FDRs used in common practice (e.g., 1%). There are several reasons why it is difficult to directly demonstrate this statistical bias at more stringent FDR thresholds. Because of the relative simplicity of a 48-protein control mixture, the validation examples analyzed here identify relatively few spectrum matches in the refinement stage, and even fewer at stringent FDR thresholds. Decoy-based FDR estimates have larger errors
when using a small number of reverse spectrum matches [95], and this error may mask the true statistical bias. Despite these difficulties, I have identified second-stage bias that cannot be explained by a previously proposed error model [95] at 5% FDR in the semi-tryptic analysis (Table 5.2) and 1% FDR in the point mutation analysis (Table 5.3). The X!Tandem patch largely corrects this error in the low FDR range (Tables 5.4 and 5.5). Of course, it is also possible that, in general, the extent of statistical bias increases with FDR and in some cases only becomes problematic when using more lenient thresholds. In practical terms, however, I see no drawbacks to using a formally corrected model in all cases, including stringent FDR thresholds for which the statistical bias may be less severe.

In addition to the X!Tandem patch, other methods that maintain the validity of this null model should also allow accurate decoy FDR calculations on multi-stage
search results. However, methods that rely on separate forward and reverse database searches [104] will not meet this criterion, because the refinement list generated by the reverse database search will be very different from the one generated by the forward database search; thus, matches from these two model refinement stages may not yield accurate FDR estimates (Figures 5.15 and 5.16).

There is also no guarantee that overall score distributions of spectrum matches will be similar for results from the initial and refinement search stages. The most reliable way to control the FDR within the refinement stage spectrum matches is to perform the FDR analysis separately on these matches. This may present problems for downstream analysis tools that attempt to model the distributions of true and false matches from all search stages pooled together [31, 108], and can be especially problematic for applications in which the PTM or nonenzymatic cleavage peptides are of primary interest. Additionally, the matches from the refinement stage are technically dependent on the matches from the initial search stage, which violates assumptions made in some protein-level inference models [146].

Downstream analysis tools for selecting reliable spectrum matches and reconstructing sample proteins are indispensable to a reliable proteomics analysis pipeline. However, further consideration of the basic assumptions made in these statistical models is needed to avoid underestimating the FDR when using a multi-stage search strategy. A more general strategy, wherein an initial run of a search algorithm and the downstream analysis are used to determine the set of high-confidence proteins, followed by subsequent runs of these tools to match the remaining unmatched spectra, is likely to be both statistically valid and widely compatible with other search algorithms and downstream analysis tools. In other words, each “stage” of the search strategy could include an independent and rigorous statistical analysis of the spectrum matches. Such a strategy would also more clearly separate the goals of
protein identification and characterization. The validative frameworks developed in this chapter can be used to empirically test the accuracy of such methods.
Chapter 6

Conclusions and Future Directions

6.1 Major Contributions in This Work

The inference of biological networks spanning signal transduction and transcriptional regulation remains a critical field of study in genomics research. In this work I have made several major contributions to this field that will improve the integration of high-throughput data into a comprehensive framework. The PTM-Switchboard database addresses a basic need to catalog current knowledge in a structured, searchable, and machine-readable format. The existing content of this database allows for validation and benchmarking of computational methods, and provides a platform for the dissemination of new predictions and experimental results. As of this writing, PTM-Switchboard has been accessed 1,337 times over a period of 18 months (averaging approximately 75 hits per month).

While a number of methods exist to infer condition-specific or modulated TF activity, as reviewed in Chapter 1, many of the regulatory triplets predicted by these methods involve other classes of TF modulators, such as co-factors and post-transcriptional regulators. The focused study of TF modulators that function via
direct post-translational mechanisms can benefit from additional data sources, including protein-protein interactions and enzyme substrate-specificity models. To this end, MoNsTER is designed to integrate expression data with additional data sources in a principled and probabilistic framework. I have demonstrated the power of this probabilistic model-based methodology for integrating heterogeneous data and inferring complex biological networks encompassing multiple processes. This work therefore represents a significant step towards a comprehensive model of gene transcription. MoNsTER is extremely flexible and can be applied to any of the growing compendia of high-throughput data describing gene expression, protein-DNA interactions, and enzyme-substrate interactions.

In particular, MS-based proteomics continues to provide novel methods for studying PTMS and protein modifying enzymes. MS data sources remain largely untapped for integration with transcriptional and other regulatory networks. For this reason, I have also contributed an improvement to the computational methods for identifying PTMs by tandem MS. Collectively, the advances presented in this work can be used to construct a larger-scale data mining and network inference platform, as discussed in this chapter.

6.2 Mimosa: An Additional Tool for MFG Triplet Prediction

I have also contributed to the development of another novel algorithm, Mimosa, for identifying hidden partition structures in expression data, and predicting the modulators of regulatory interactions [80], described briefly here. In contrast to other methods that begin by selecting a modulator of interest, Mimosa begins with a
putative TF-Gene pairwise interaction and finds an appropriate partition structure, in the expression compendium. In short, a mixture model, described below, is fit to the coexpression data, such that the TF and target gene are correlated in a subset of the samples and uncorrelated in the remaining samples. Modulator genes can then be predicted by testing for differential expression between the inferred partitions.

Following the notation from Chapter 3, for any TF $j$ and putative target gene $i$ in an expression compendium of samples $t$, Mimosa can fit the pair of expression profiles $(i,j)$ to a mixture of correlated and uncorrelated expression models, with

$$p_u(g_{it}, f_{jt}) = \frac{1}{2\pi} \exp \left[ -\frac{1}{2} \left( g_{it}^2 + f_{jt}^2 \right) \right]$$  \hspace{1cm} (6.1)

And the correlated model probability function given by:

$$p_c(g_{it}, f_{jt}) = \frac{\exp \left[ -\frac{1}{2} \left( g_{it}^2 + f_{jt}^2 + 2\psi g_{it} f_{jt} \right) / (1 - \psi^2) \right]}{2\pi \sqrt{1 - \psi^2}}$$  \hspace{1cm} (6.2)

Where $-1 \leq \psi \leq 1$ is a free parameter corresponding to the strength and direction of the correlated distribution. The observed data is treated as a random sampling from the two distributions, $p_u$ and $p_c$, with mixing parameter $\varphi$ defined as the fraction of data points that belong to the uncorrelated group, $p_u$. The total likelihood of a data point $(g_{it}, f_{jt})$ is the mixture of likelihoods under each model:

$$p(g_{it}, f_{jt}) = \varphi \ p_u(g_{it}, f_{jt}) + \left( 1 - \varphi \right) \ p_c(g_{it}, f_{jt})$$  \hspace{1cm} (6.3)

In summary, the mixture model has two free parameters, $(\varphi, \psi)$, that determine the fraction of uncorrelated points in the observed data and the strength of correlation in the remaining points. Parameter estimates are chosen to maximize the log
likelihood function:

\[ L(f, \psi) \equiv \sum_t \ln \left[ p(g_{it}, f_{jt}|f, \psi) \right]. \]  

(6.4)

After fitting the mixture model for a selected gene pair \((i, j)\), *Mimosa* computes the probability that each sample \(t\) belongs to the correlated group:

\[ q_t = \frac{(1 - \varphi) \ p_c(g_{it}, f_{jt}|\psi)}{p(g_{it}, f_{jt}|\varphi, \psi)}. \]  

(6.5)

Modulator genes \(k\) are then selected based on the correlation between their expression profile and vector \(q\). See ref. [80] for additional details and validation of this computational method.

Compared to many of the methods reviewed in Chapter 1, *Mimosa* reduces the number of hypotheses that need to be tested, because it first selects informative TF-Gene pairs before multiplying the possible combinations to predict modulators. The ability to discover conditionally coexpressed genes without an *a priori* partition structure provides a unique opportunity to identify novel regulatory interactions, including those involving TF protein modification. Compared to *MoNsTER*, the *Mimosa* algorithm is far more scalable; the complexity increases only linearly with more putative TF-Gene pairs or candidate modulators. On the other hand, *Mimosa* only analyzes expression data, whereas *MoNsTER* is able to integrate additional data sources related to the physical interactions underlying the network structure\(^1\). *MoNsTER* can integrate data sources in a probabilistic manner that does not impose strict thresholds on any of the input data, and should be more sensitive in principle.

\(^1\)In theory, the tests performed by *Mimosa* can be further limited to those predicted on other data sources, such as ChIP-seq (for putative TF-Gene pairs) and protein-protein interactions (for candidate modulators). However, this is not a particularly principled way to integrate heterogeneous data, and requires the selection of additional arbitrary thresholds for each data source.
Overall, it is likely that Mimosa and MoNsTEr have complimentary strengths. For example, Mimosa has also been applied in a preliminary analysis of STAT1 activity in the human B cell expression data [165] used in Chapter 4. Mimosa found a usable mixture model for 10 of the 25 known target genes compiled in ref. [162], and thereby predicted 34 candidate modulators for the regulation of these genes by STAT1 (see ref. [80]). The modifying enzymes predicted in this list are notably different from those predicted by MoNsTEr. Although Mimosa largely predicted other classes of TF modulators, it did predict several protein modifying enzymes as modulators of STAT1 activity. Specifically, the modulator genes GRK5, DUSP1, and SIK1 were all predicted by Mimosa, and were not predicted by MoNsTEr, despite being included in the input data used in Chapter 4. None of these modifiers have an association with STAT1 listed in the STRING database [99] used for TF-Modifier interaction priors, and thus were not predicted by MoNsTEr. However, GRK5 is known to function upstream of JAK-STAT signaling in general, and therefore represents a biologically relevant result, at least in terms of indirect STAT1 modulators. Ultimately, Mimosa is capable of testing a much broader array of TF modulators, while MoNsTEr is specifically designed to predicted direct modifying enzymes of TF-PTMs when sufficient data is available.

6.3 Approximating Regulator Activities from Transcript Levels

MoNsTEr and Mimosa both rely on an assumption common to most network inference methods based on mRNA measurements—that TF and modifier transcript levels are indicative of their relative activity [12, 22, 112, 134, 204]. The validity of this
assumption is known to vary widely between different TFs [15], and is also likely to vary for each modifier. MoNsTER addresses this problem by improving the estimate of TF activity through the use of TF-Modifier synergy effects, exemplified by the results obtained for yeast TFs MSN2 and MSN4. Specifically, MoNsTER identifies known targets of MSN2/4 with improved accuracy when using the full model with TF-Modifier synergistic effects (see Chapter 3).

The initial results also suggest that, when observed over a sufficient number of conditions, there is substantial biological signal in the expression profiles of modifying enzymes. In the MoNsTER yeast application (Section 3.5), all input kinases had at least a 4-fold change in expression across the observed conditions. The expression level of the known MSN2/4 modifier TPK1 varied 78-fold across the included conditions. Similarly, expression of JAK1 varied 14-fold across the B cell samples used in Chapter 4. The median for all STAT1 modifiers predicted by MoNsTER is a 5-fold range of expression. Although many modifying enzymes are regulated at the post-translational level, there seems to be significant perturbation occurring at the transcript level, which is likely to have some effect on their signaling activity, and which can be used to infer their overall effect on downstream transcription. Thus, methods for inferring modifier activity from transcript level rely primarily on an expression compendium encompassing sufficient perturbation to the transcriptional regulation of the modifier, but can be further aided by the use of additional data sources, as is the case for MoNsTER.

Despite the advances achieved by these methods, the results from Chapter 3 indicate that challenges remain for improving the sensitivity of TF-Modifier network edge prediction. It is possible that modifiers with lower transcript variability may go undetected by MoNsTER and other expression-based methods. This reduced variability in expression can be partly mitigated by having a larger and/or different set
of expression samples. **MoNsTER** can also take advantage of improved accuracy of TF-Modifier priors, e.g., via new breakthroughs in proteomic methods. Therefore, the sensitivity of TF-Modifier prediction is reduced, in part, by the available data, and does not necessarily reflect any inherent limitations of the inference method. In the rest of this chapter I discuss potential improvements to these methods, as well as novel data sources—many of which have only become available in the last several years—that can be used to improve the inference of MFG networks.

### 6.4 A Comprehensive Tool Set for MFG Triplet Prediction

The potential uses for **Mimosa** and **MoNsTER** are broad, ranging from large-scale network inference for elucidating systems-level properties, to focused analysis of a specific TF under post-translational regulation. These complex inference problems are unlikely to be solved by a single tool, as evidenced by the fact that **Mimosa** and **MoNsTER** each identified different known modulators of STAT1 activity from the same expression compendium. Thus, there is a need to integrate different computational pipelines to mine existing compendia and analyze new data, in order to guide and prioritize further experiments. Additionally, there is a broad need to “scale-up” the applications discussed in this work. In this section, I lay out a general framework for integrating these and related tools, and discuss some potential improvements to the computational frameworks of both **Mimosa** and **MoNsTER**.

Scalability remains a major issue for many network inference methods. Despite the advances made by **Mimosa**, the prediction of regulatory triplets still faces substantial combinatorial complexity. Model complexity for **MoNsTER** grows expo-
nentially with the number of genes, TFs, and modifiers considered, and thus far I have only applied this method to specific sub-networks of interest. One potential solution for reducing complexity is to first predict coarse-level network modularity, then limit these tools to networks or triplets within these modules.

In prior work, I developed a method for identifying clusters in multi-partite graphs [52] that is of particular interest here. Briefly, this method allows for the construction of a graph containing different classes of nodes, e.g., genes, regulators, annotations, etc. Edge sets connecting nodes of different classes are then defined separately for each pair of classes, based on appropriate data sources. In one of the applications in ref. [52], I analyzed a graph of TFs, target genes, and tissues, with an edge set linking TFs to target genes based on PWMs, and two edge sets linking TFs and target genes to tissues based on a metric of tissue-specific expression [173]. The algorithm developed in this work identifies clusters in the graph containing nodes from all three classes, such that the density of edges between each class is significantly higher than background. A particularly novel aspect of the work was that the significance of density within a cluster was analyzed separately for edge sets between each pair of classes. This is justified by the different types of data used for these edges, which have inherently different false positive and negative rates.

To apply this methodology to the prediction of MFG modules, a tri-partite graph could be used to represent modifiers, TFs, and target genes. TF-Gene edges would be chosen based on PWM and/or ChIP-seq analyses. Likewise, Modifier-TF edges would be chosen based on a combination of protein-protein interactions and enzyme-substrate specificity models—essentially the same data types used as priors for MoNsTEr. Modifiers could also be linked to target genes based on, for example, functional associations summarized in STRING [99]. The clustering algorithm could then be used to identify significant, and potentially overlapping, modules of mod-
ifiers, TFs, and genes, which would generally be of a reasonable size to model in finer detail with either Mimosa or MoNsTER.

Even with a potential reduction in complexity by focusing on modules, there is still room to improve the statistical and computational framework of MoNsTER. For example, I observed that MoNsTER could not predict STAT1 modifiers without a high prior probability of connectivity, despite the fact that I did not use any explicit threshold on the edge priors (see Chapter 4). This is partly a limitation of the model itself, because the effect of each TF-Modifier interaction in the expression model is parameterized by a separate $\gamma_{jk}$ variable, whereas the effects of TF-Gene connections are modeled more robustly by TF-specific $\beta_j$ variables. However, there is a biological motivation for this model design. While a single TF can often act as both an activator or repressor in a condition- or promoter-specific manner, TFs tend to have similar actions across at least a subset of their target gene repertoire. Thus, while the $\beta_j$ variables may not fully capture the behavior of a TF, they should be sufficient to identify a substantial subset of the TF expression program on each run of the model-fitting method. Conversely, the effect of a modifier interaction is highly substrate-specific. There is no reason to assume, biologically, that a modifier interaction with one TF will have a mechanistically similar effect to that with another TF. Thus, each TF-Modifier interaction has its own unique downstream effect in the model. The result of this model property is that every time a TF-Modifier interaction is removed from the model during iterative model-fitting, the associated $\gamma_{jk}$ variable defaults to an uninformative prior (with expected value equal to 0), and the TF-Modifier interaction is typically only added back into the model by chance, based on its prior. The end result is that the sensitivity for predicting TF-Modifier interactions is especially dependent on the prior probabilities derived from non-expression sources, as observed in the simulation and experimental data analyses. Thus, the current
method is sufficient for filtering candidate modifiers, predicted by noisier or more
general data sources, down to those which appear to modulate the transcriptional
activity of a TF of interest. This is especially useful in the context of data sources
with high false positive rates such as kinase-substrate PWMs, but is less useful when
analyzing sparse data sets (see Section 3.6).

Additionally, replicate runs of the MoNsTEr algorithm can produce distinct net-
work models given the same input data. In the human B cell application, clustering
of network parameters estimated from individual runs revealed three distinct net-
works. It is possible that all three predicted networks are correct, and capture dis-
tinct biological roles of the TF of interest, STAT1 (see Chapter 4). Thus, there is a
need for a more rigorous statistical framework, in which to compare distinct network
models, and summarize across all model-fitting results. This can also potentially
be ameliorated by alternate model-fitting procedures that inherently address the
multi-modality of the solution space.

Given the apparent biological validity of this model, these limitations can poten-
tially be addressed by alternate model-fitting procedures, i.e., without making any
compromise in the biological motivations of the model itself. Thus, further work
is merited to explore alternate methods for model estimation and summarization
of multiple network structure predictions. Novel model-fitting approaches may in-
crease sensitivity and better summarize multi-modal solution spaces. Briefly, one
possible approach would be to alternate between the selection of candidate network
structures (parameters C and D only), and fitting of all other parameters (set Θ).
The structure selection step would require a novel method for identifying several
candidate network structures, and would benefit from a metric for comparing struc-
tures, such as information criteria [87]. In between network selection steps, the
fitting of other parameters could be solved in the OLS closed form, or iteratively
sampled for probabilistic estimates.

Regardless of the model-fitting framework, MonSTEr links modifiers to TFs and TFs to target genes, but does not provide a direct link between modifiers and target genes. In theory, the probability of an individual MFG triplet \((i, j, k)\) is the joint posterior probability \(P(C_{ij} = 1, D_{jk} = 1|\Xi)\). However, in the STAT1 network model, most of the individual network edge posteriors were equal to 1, and thus the joint probabilities must also be (trivially) equal to 1. To more deeply probe the dependencies between specific modifiers and target genes, I propose an additional calculation:

\[
\delta_{ik} = \frac{\sum_{t=1}^{T} (\varepsilon_{it}^{(-k)})^2 - \sum_{t=1}^{T} \varepsilon_{it}^2}{\sum_{t=1}^{T} \varepsilon_{it}^2}
\]  

(6.6)

Where \(\varepsilon_{it}\) is the residual error observed in the estimated model and \(\varepsilon_{it}^{(-k)}\) is the adjusted residual error if modifier \(k\) is removed from the model. If gene \(i\) is particularly dependent on modifier \(k\) in the estimated model, then the residual errors corresponding to this gene should increase when the modifier is removed, resulting in a positive value of \(\delta_{ik}\). Reasonable criteria for predicting MFG triplets would then include a threshold on both the joint posterior stated above, and the value of \(\delta_{ik}\).

### 6.5 Improved Integration with Proteomic Data

While tandem MS remains a promising source of information for the methods discussed above, there is a large amount of data generated from global experiments characterizing whole cell or organelle proteomes. TF-PTMs are unlikely to be identified in global proteome experiments because both PTM-forms and TFs tend to exist below the dynamic range of typical tandem MS experiments. PTMs on non-
TF proteins cannot be directly integrated into the methods developed in this work, but can improve models of enzyme substrate-specificity in general [13, 140, 217], and these models in turn can be applied to TF protein sequences to generate more informative predictions of TF-Modifier interactions.

As presented in Chapter 5, much of the publicly available tandem MS data has not been analyzed with consideration for the full range of in vivo PTMs. Most tandem MS experiments are performed primarily for protein identification and detection of several expected PTM types. However, these experiments often produce MS2 spectra indicative of unexpected and biologically relevant PTMs that are ignored in the downstream data analysis [147, 150]. Thus, there remains a potential for data mining efforts to reanalyze much of this data in order to glean additional predictions of novel PTM sites in general. Multi-stage search strategies, such as the model refinement option in X!Tandem, are a powerful and useful approach to fully utilizing MS data, including the identification of PTMs. However, these strategies require careful consideration before applying the statistical methods currently used for single-stage search strategies. As demonstrated in Chapter 5, it is useful to make the required adjustments, rather than tolerate biased statistics or inefficient search strategies. Additionally, recent developments in proteomics have focused on enrichment, and even quantitation, of specific PTM types. Recent developments in mass spectrometry [100, 101, 119, 210, 223] and other technologies [56, 128, 157, 194] have made it possible to assay a much wider range of PTM types with greater sensitivity, and can further refine computational predictions of enzyme-specific substrates.

Knowledge of in vivo PTM sites can be used to filter the application of predictive models, rather than blindly scanning entire protein sequences [129]. However, this is of little help if TF proteins are themselves going undetected in most proteomic experiments. Another approach is to use the PTMs identified in vivo to improve
models of enzyme-substrate occurrence. This is not a straight-forward task because tandem MS data does not indicate the enzyme(s) responsible for any identified PTM. However, it may be possible to use machine-learning and other pattern discovery methods to identify PTMs in similar sequence motifs/regions with similar physico-chemical properties. These motifs could be linked to specific modifying enzymes using additional experiments, such as quantitatively measuring PTM concentrations after perturbing a specific modifier, or by mining existing protein-protein interaction data for common interacting partners of substrates containing similar motifs.

Another promising direction is the study of enzyme-substrate interactions in vitro. For example, protein microarrays [56] have been used to identify novel substrates for protein kinases [157] and an acetyltransferase [128], and can be applied to a much broader range of protein-modifying enzymes. Turk et al. have developed a peptide library screening method for determining the specificity of protein kinases [194] that is now broadening our understanding of phosphorylation substrate specificity [140].

**MoNsTEr** provides a powerful framework to analyze these and future proteomic datasets in conjunction with expression and protein-DNA interaction data. **MoNsTEr** is extremely flexible and can be applied to any of the growing compendia of high-throughput data describing gene expression, protein-DNA interactions, and enzyme-substrate interactions. These collective breakthroughs in PTM research represent a major opportunity to expand regulatory network models beyond the scope of simple TF-Gene interactions.
6.6 Expanding the TF-PTM Knowledge Base

PTM-Switchboard (Chapter 2) is an initial effort to catalog TF-PTMs, with sufficient content for method validation. However, there is a large amount of literature that remains untapped, and thus there is a need for community curation efforts and improved text-mining methods. Furthermore, PTM-Switchboard should be expanded to other organisms in the future, especially humans. Mammalian systems will require a larger, more controlled ontology for describing the cell types and experimental contexts in which MFG triplets are inferred.

PTM-Switchboard also provides an appropriate database structure in which to catalog computational predictions and make these searchable and accessible to molecular biologists. An ideal cycle would populate the database with computational predictions, followed by validation using the existing literature, and targeted experimentation in unstudied cases. S. cerevisiae is particularly ripe for genome-wide application of the computational methods developed in this work. There is a wide range of in vivo [81] and in vitro [148] binding data available for most TFs in this yeast species, along with pilot efforts in proteome-wide characterization of substrates for specific kinases and other modifying enzymes [128, 140, 157]. With the scalability improvements discussed in Section 6.4, it should be possible to predict MFG triplets in yeast on a genome-wide scale.

Another interesting application for future work is the study of STAT1 PTMs other than phosphorylation. Although Chapter 4 focused primarily on STAT1 phosphorylation, STAT1 activity is known to be controlled via a variety of PTMs [117, 163, 180, 207]. This focus was primarily due to the available data in STRING, which was exceptionally sparse for other classes of modifiers. The methods developed are not inherently restricted to studying phosphorylation, and can benefit from
the general proteomic improvements discussed in Section 6.5 to predict candidate modifiers regulating these PTMs as well.

**In Conclusion**

Ultimately, cell behavior in both healthy and diseased states is the result of complex interactions between networks of genes. Modeling the structure of biological networks allows for the generation of novel and testable hypotheses, regarding both individual gene functions and overall cellular behavior. In this work, I have established that it is feasible to model a network with more complexity than just static TF-Gene interactions, with the benefit of additional biological insight into the interplay between cell signaling and transcriptional regulation. Experimental techniques relevant to network inference are progressing on several fronts, and now more than ever, there is a significant need for computational methods to integrate heterogeneous high-throughput data, and maximize the knowledge extracted from each data source.
Appendix A

MoNsTEr Model-Fitting Procedures

A.1 Derivation of Posterior Distributions

The components of the posterior distribution in Eq 3.6 are further defined as:

\[
P(g|f, h, C, D, \Theta) = \prod_{i=1}^{N} \prod_{t=1}^{T} \left(2\pi\sigma^2\right)^{-1/2} \exp\left[-\frac{1}{2\sigma^2}\frac{e_i^2}{e_{it}}\right]
\]

\[
P(C|m, b, w) = \prod_{i=1}^{N} \prod_{j=1}^{J} \left[b_{ij}^{C_{ij}}(1-b_{ij})^{1-C_{ij}}\right]^{w_j} \left[m_{ij}^{C_{ij}}(1-m_{ij})^{1-C_{ij}}\right]^{1-w_j}
\]

\[
P(D|a, s, u) = \prod_{j=1}^{J} \prod_{k=1}^{K} \left[a_{jk}^{D_{jk}}(1-a_{jk})^{1-D_{jk}}\right]^{u_k} \left[s_{jk}^{D_{jk}}(1-s_{jk})^{1-D_{jk}}\right]^{1-u_k}
\]

\[
P(\Theta) = \prod_{i=1}^{N} (\tau_i^2)^{-1/2} \exp\left[-\frac{1}{2\tau_i^2}\frac{\alpha_i^2}{\alpha_i^2}\right] \prod_{j=1}^{J} (\tau_j^2)^{-1/2} \exp\left[-\frac{1}{2\tau_j^2}\frac{\beta_j^2}{\beta_j^2}\right] \prod_{j=1}^{J} \prod_{k=1}^{K} (\gamma_{jk}^2)^{-1/2} \exp\left[-\frac{1}{2\gamma_{jk}^2}\frac{\sigma^2}{\sigma^2}\right]
\]

Iterative algorithms for estimating model parameters require an individual pos-
terior distribution for each parameter, i.e., a posterior distribution that can be computed given the input data and a current instantiation of all other parameter values. The necessary distributions for the model in Chapter 3 are derived here.

First, let $\Theta_{(-Q)}$ denote the set of all parameters in $\Theta$ except some individual parameter $Q$. Let $\varepsilon_{it}'[Q = q]$ denote the residual error of gene $i$ in sample $t$ when some model parameter $Q$ is changed to value $q$. The posterior distribution for a particular $\alpha_i$ given all expression data and all other model parameters is:

$$ P(\alpha_i = q|\Xi, C, D, \Theta_{(-\alpha_i)}) \propto \exp \left[ -\frac{1}{2\sigma^2} \sum_{t=1}^{T} \varepsilon_{it}'[\alpha_i = q]^2 \right] \ast \exp \left[ -\frac{1}{2\tau_{\alpha}^2} q^2 \right] \quad (A.1) $$

This distribution can be sampled as a normal distribution with mean $\mu_{\alpha_i}$ and variance $\nu_{\alpha}$:

$$ \mu_{\alpha_i} = \frac{\nu_{\alpha}}{\sigma^2} \ast \sum_{t=1}^{T} \varepsilon_{it}'[\alpha_i = 0] \quad (A.2) $$

$$ \nu_{\alpha} = \left( \frac{T}{\sigma^2} + 1/\tau_{\alpha}^2 \right)^{-1} \quad (A.3) $$

Alternatively, the value of $\alpha_i$ can be chosen to maximize the posterior by setting $\alpha_i = \mu_{\alpha_i}$. Also note that for an “unregulated” gene, i.e., a gene $i$ such that $C_{ij} = 0 \ \forall j$, Eq A.2 simplifies to an estimate of the mean expression for gene $i$:

$$ \mu_{\alpha} = \frac{1}{T + \sigma^2/\tau_{\alpha}^2} \sum_{t=1}^{T} g_{it} $$

Similarly, the individual posterior for $\beta_j$ is:

$$ P(\beta_j = q|\Xi, C, D, \Theta_{(-\beta_j)}) \propto \exp \left[ -\frac{1}{2\sigma^2} \sum_{i=1}^{N} \sum_{t=1}^{T} \varepsilon_{it}'[\beta_j = q]^2 \right] \ast \exp \left[ -\frac{1}{2\tau_{\beta}^2} q^2 \right] \quad (A.4) $$
Therefore, $\beta_j$ can be resampled from the normal distribution $N(\mu_{\beta_j}, \nu_{\beta_j})$ with:

\[
\mu_{\beta_j} = \frac{\nu_{\beta_j}}{\sigma^2} \sum_{i=1}^{N} \sum_{t=1}^{T} \varepsilon'_{it}[\beta_j = 0] \ast C_{ij}f_{jt}
\]

\[
\nu_{\beta_j} = \left( \frac{\sum_{i=1}^{N} \sum_{t=1}^{T} (C_{ij}f_{jt})^2}{\sigma^2} + \frac{1}{\tau_{\beta}^2} \right)^{-1}
\]

Once again, the posterior can be maximized by setting $\beta_j = \mu_{\beta_j}$. Also note that $\mu_{\beta_j} = 0$ and $\nu_{\beta_j} = \tau_{\beta}^2$ when $C_{ij} = 0 \ \forall i$. In other words, when TF $j$ does not regulate any genes, its posterior becomes equivalent to its uninformative prior. Similar equations are derived below for resampling or maximizing the individual posterior of $\gamma_{jk}$. In this case, the posterior is equivalent to the the uninformative prior whenever $C_{ij} = 0 \ \forall i$ or when $D_{jk} = 0$.

\[
P(\gamma_{jk} = q | \Xi, \mathbf{C}, \mathbf{D}, \Theta_{\{-\gamma_{jk}\}}) \propto \exp \left[ -\frac{1}{2\sigma^2} \sum_{i=1}^{N} \sum_{t=1}^{T} \varepsilon'_{it}[\gamma_{jk} = q]^2 \right] \ast \exp \left[ -\frac{1}{2\tau_{\gamma}^2} q^2 \right]
\]

\[
\mu_{\gamma_{jk}} = \frac{\nu_{\gamma_{jk}}}{\sigma^2} \sum_{i=1}^{N} \sum_{t=1}^{T} \varepsilon'_{it}[\gamma_{jk} = 0] \ast C_{ij}D_{jk} \Phi(f_{jt},h_{kt})
\]

\[
\nu_{\gamma_{jk}} = \left( \frac{\sum_{i=1}^{N} \sum_{t=1}^{T} (C_{ij}D_{jk} \Phi(f_{jt},h_{kt}))^2}{\sigma^2} + \frac{1}{\tau_{\gamma}^2} \right)^{-1}
\]

The individual posterior for the model-wide residual error variance $\sigma^2$, given all input data and all other parameters is:

\[
P(\sigma^2 | \Xi, \mathbf{C}, \mathbf{D}, \Theta_{\{-\sigma^2\}}) \propto (\sigma^2)^{-\left(\frac{TN}{2}+2\right)} \ast \exp \left[ -\frac{1}{2\sigma^2} \sum_{i=1}^{N} \sum_{t=1}^{T} \varepsilon^2_{it} \right] \ast \exp \left[ -\frac{1}{2\sigma^2} \right]
\]

155
The above equation implies that $\sigma^2$ can be sampled from a scaled-inverse $\chi^2$ distribution with degrees of freedom parameter $TN + 2$ and scale parameter $s^2$ defined in Eq A.11 below. This posterior can be maximized by setting $\sigma^2 = s^2$.

$$s^2 = \frac{1 + \sum_{i=1}^{N} \sum_{t=1}^{T} \varepsilon^2_{it}}{TN + 2} \quad (A.11)$$

The $C$ and $D$ variables are binary, and therefore the individual posteriors can be computed from the relative probability of values 1 and 0 for each variable. For $C_{ij}$, the individual posterior is given by Eq A.12, with $q \in \{0, 1\}$:

$$P(C_{ij} = q|\Xi, C_{(-C_{ij})}, D, \Theta) \propto \exp \left[ \frac{-1}{-2\sigma^2} \sum_{t=1}^{T} \varepsilon'_{it}[C_{ij} = q]^2 \right] * [b_{ij}^q (1 - b_{ij})^{1-q}]^{w_j} * [m_{ij}^q (1 - m_{ij})^{1-q}]^{1-w_j} \quad (A.12)$$

To find the exact posteriors for $C_{ij} = 0$ and $C_{ij} = 1$, first compute the proportional probability values $Z_0^{(C_{ij})}$ and $Z_1^{(C_{ij})}$, respectively:

$$Z_0^{(C_{ij})} = \exp \left[ \frac{-1}{-2\sigma^2} \sum_{t=1}^{T} \varepsilon'_{it}[C_{ij} = 0]^2 \right] * (1 - b_{ij})^{w_j} (1 - m_{ij})^{1-w_j} \quad (A.13)$$

$$Z_1^{(C_{ij})} = \exp \left[ \frac{-1}{-2\sigma^2} \sum_{t=1}^{T} \varepsilon'_{it}[C_{ij} = 1]^2 \right] * b_{ij}^{w_j} m_{ij}^{1-w_j} \quad (A.14)$$

Therefore, $C_{ij}$ can be resampled by setting $C_{ij} = 1$ with probability $\frac{Z_1^{(C_{ij})}}{Z_0^{(C_{ij})} + Z_1^{(C_{ij})}}$ and can be maximized by setting $C_{ij} = 1$ iff $Z_1^{(C_{ij})} > Z_0^{(C_{ij})}$. Using a similar posterior...
equation for $D_{jk}$ yields $\zeta_0^{(D_{jk})}$ and $\zeta_1^{(D_{jk})}$ analogous to $Z_0^{(C_{ij})}$ and $Z_1^{(C_{ij})}$, respectively:

$$
\zeta_0^{(D_{jk})} = \exp \left[ -\frac{1}{2\sigma^2} \sum_{i=1}^{N} \sum_{t=1}^{T} \varepsilon_{it}^{'2} [D_{jk} = 0] \right] \ast (1 - a_{jk})^{u_k} (1 - s_{jk})^{1-u_k} \quad (A.15)
$$

$$
\zeta_1^{(D_{jk})} = \exp \left[ -\frac{1}{2\sigma^2} \sum_{i=1}^{N} \sum_{t=1}^{T} \varepsilon_{it}^{'2} [D_{jk} = 1] \right] \ast a_{jk}^{u_k} s_{jk}^{1-u_k} \quad (A.16)
$$

The individual posteriors of the weight parameters $w_j$ and $u_k$ are independent of the expression data and other parameters, given instantiated values for edge variables $C$ and $D$. The individual posterior distributions are calculated as follows:

$$
P(w_j|C, b, m) \propto \left[ \prod_{i=1}^{N} b_{ij}^C (1-b_{ij})^{1-C_{ij}} \right]^{w_j} \left[ \prod_{i=1}^{N} m_{ij}^C (1-m_{ij})^{1-C_{ij}} \right]^{1-w_j} \frac{\prod_{i=1}^{N} \left[ b_{ij}^w m_{ij}^{1-w_j} + (1-b_{ij})^{w_j} (1-m_{ij})^{1-w_j} \right]}{\prod_{i=1}^{N} \left[ b_{ij}^w m_{ij}^{1-w_j} + (1-b_{ij})^{w_j} (1-m_{ij})^{1-w_j} \right]}
$$

(A.17)

$$
P(u_k|D, a, s) \propto \left[ \prod_{j=1}^{J} a_{jk}^D (1-a_{jk})^{1-D_{jk}} \right]^{u_k} \left[ \prod_{j=1}^{J} s_{jk}^D (1-s_{jk})^{1-D_{jk}} \right]^{1-u_k} \frac{\prod_{j=1}^{J} \left[ a_{jk}^{u_k} s_{jk}^{1-u_k} + (1-a_{jk})^{u_k} (1-s_{jk})^{1-u_k} \right]}{\prod_{j=1}^{J} \left[ a_{jk}^{u_k} s_{jk}^{1-u_k} + (1-a_{jk})^{u_k} (1-s_{jk})^{1-u_k} \right]}
$$

(A.18)

In this work, I used grid sampling to estimate the distributions of $w_j$ and $u_k$. I describe the process here, without loss of generality, for a single parameter $w_j$. First, let $\hat{p}_\lambda = P(w_j = \lambda|C, b, m)$ from Eq A.17 for all $\lambda$ values in \{0.01, 0.02, \ldots, 0.99\}. These values are then normalized to probabilities: $p_\lambda = \hat{p}_\lambda / \sum_\lambda \hat{p}_\lambda$. Therefore, $w_j$ can be resampled from all possible values of $\lambda$, each with probability $p_\lambda$. Alternatively, the posterior of $w_j$ can be maximized by selecting the $w_j = \arg \max_\lambda (p_\lambda)$. 

157
A.2 Local Estimation Algorithm

Phase 1: Model Initialization

To seed the algorithm, an initial network \( \{C, D\} \) is selected, as summarized in Box A.1. The simplest way to select this network is deterministically, by setting a threshold \( q \) on the geometric mean prior of each edge, i.e. \( C_{ij} = 1 \) if and only if \( \sqrt{b_{ij}m_{ij}} \geq q \). This approach is generally appropriate for the first few runs of the estimation algorithm. For increased exploration of the solution space away from the most strongly predicted prior network, this initial deterministic starting network can be perturbed by randomly flipping the state of some fixed proportion of edges. For a more stochastic starting network, each edge parameter can be randomly sampled according to the geometric mean of its priors, i.e. \( P(C_{ij} = 1) = \sqrt{b_{ij}m_{ij}} \).

Note that in all cases, using the geometric mean of priors is equivalent to assuming \( w_j, u_k = 0.5 \forall j, k \), and these parameters are initialized accordingly.

Given a fixed network \( \{C, D\} \), the parameters \( \{\alpha, \beta, \gamma\} \) can be fit using the Ordinary Least Squares (OLS) solution to linear regression. To add a stochastic aspect to the selection of values for these parameters, the best-fit values can be perturbed by adding noise sampled from a normal distribution with \( \mu = 0 \) and small \( \nu \) (i.e. 0.05). The model-wide variance is always initialized by \( \sigma^2 = s^2 \) (from Eq A.11).

Phase 2: Local Maximization

Given initial values for all model parameters, it is possible to compute the individual posterior of any model variable according to the equations in Appendix A.1. A hill-climbing algorithm (Box A.2) is run from the starting point computed in Phase
1, such that each parameter is iteratively reassigned in sequence by maximum likelihood. The hill-climbing algorithm is run until the model parameters converge on an optimal solution. Convergence is determined by tracking the log of the model likelihood (rather than the model posterior, because the parameter priors are constant), given by Eq A.19 below. Convergence is assumed when $\Delta \ln(L) < 0.0001$ (approximately 0).

\[
\ln(L) = \frac{-NT}{2} \ln \left(2\pi\sigma^2\right) - \frac{1}{2\sigma^2} \sum_{i=1}^{N} \sum_{t=1}^{T} \varepsilon_{it}^2
\]

\[
+ \sum_{i=1}^{N} \sum_{j=1}^{J} w_j [C_{ij} \ln(b_{ij}) + (1 - C_{ij}) \ln(1 - b_{ij})]
\]

\[
+ \sum_{i=1}^{N} \sum_{j=1}^{J} (1 - w_j) [C_{ij} \ln(m_{ij}) + (1 - C_{ij}) \ln(1 - m_{ij})]
\]

\[
+ \sum_{j=1}^{J} \sum_{k=1}^{K} u_k [D_{jk} \ln(a_{jk}) + (1 - D_{jk}) \ln(1 - a_{jk})]
\]

\[
+ \sum_{j=1}^{J} \sum_{k=1}^{K} (1 - u_k) [D_{jk} \ln(s_{jk}) + (1 - D_{jk}) \ln(1 - s_{jk})]
\]

(A.19)

Phase 3: Local Network Estimation

The final phase of the algorithm collects local samples of specific variables of interest, notably $C$ and $D$, while leaving other parameters, notably $\alpha$ and $\beta$, fixed at the values identified in Phase 2. This results in a single estimate for each parameter in $\alpha$ and $\beta$, and fixing these variables helps to anchor the subsequent sampling iterations around the local optimum. The $\gamma_{jk}$ parameters must also be resampled in this phase, to allow for informative estimation of each corresponding $D_{jk}$. Specifically, if a parameter $\gamma_{jk} = 0$ at the end of Phase 2, and if this value were to remain fixed, then the posterior for $D_{jk}$ would become equivalent to its prior, and therefore
uninformative. The variables $\sigma^2$, $w$, and $u$ are also resampled, to allow for more robust resampling of $C$ and $D$. These variables are resampled in the same order as they were maximized in Phase 2, using sampling rules derived from the equations in Appendix A.1. This phase of the algorithm (Box A.3) is repeated a fixed number of times.

The output of this phase is a set of locally optimal fixed values for parameters $\alpha$ and $\beta$, denoted from here on as $\hat{\alpha}$ and $\hat{\beta}$, and a fixed number of samples for all other parameters. To avoid auto-correlation between the samples, a lag is introduced, such that samples are only used for model inference every $\ell$ iterations, and all other samples are dropped. From here on, let $X$ denote the number of samples used for model inference, and let $x$ denote a specific sample from 1 to $X$. In other words, phase 3 is run for a total of $\ell \times X$ iterations, but only $X$ samples (every $\ell^{th}$ iteration) are used for model inference. For the applications discussed in this work, I generally found that $X = 100$ and $\ell = 20$ provided an acceptable balance of speed and accuracy.

A.3 Parameter Estimation and Inference

Using a single run of the algorithm outlined above, it is possible to compute a maximum a posteriori (MAP) value for each parameter $\alpha_i$ and $\beta_j$ from $\hat{\alpha}$ and $\hat{\beta}$ fixed at the end of Phase 2. Subsequently, the posterior distribution for all other parameters, given both the input data, and these MAP values can be computed using the samples from Phase 3:

$$\hat{P}(C, D, \gamma, \sigma^2, w, u|\Xi, \hat{\alpha}, \hat{\beta})$$  \hspace{1cm} (A.20)
However, it is generally more interesting to marginalize this joint posterior in order to compute an independent posterior for each parameter. For example, the estimated probability of a particular TF-Gene edge $C_{ij} = \hat{P}(C_{ij} | \Xi, \hat{\alpha}, \hat{\beta})$. The marginalized posteriors are computed from the $X$ samples computed in Phase 3, with $[x]$ denoting the value of a particular parameter in sample $x$, as follows:

\[
\hat{P}(C_{ij} = 1 | \Xi, \hat{\alpha}, \hat{\beta}) = I(\hat{\beta}_j \neq 0) \frac{1}{X} \sum_{x=1}^{X} C_{ij}[x] \tag{A.21}
\]

\[
\hat{P}(D_{jk} = 1 | \Xi, \hat{\alpha}, \hat{\beta}) = \frac{1}{X} \sum_{x=1}^{X} I(\gamma_{jk}[x] \neq 0) D_{jk}[x] \tag{A.22}
\]

\[
\hat{E}(\gamma_{jk} | \Xi, \hat{\alpha}, \hat{\beta}) = \frac{1}{X} \sum_{x=1}^{X} I(D_{jk} = 1) \gamma_{jk}[x] \tag{A.23}
\]

\[
\hat{E}(\sigma^2 | \Xi, \hat{\alpha}, \hat{\beta}) = \frac{1}{X} \sum_{x=1}^{X} \sigma^2[x] \tag{A.24}
\]

\[
\hat{E}(w_j | \Xi, \hat{\alpha}, \hat{\beta}) = \frac{1}{X} \sum_{x=1}^{X} w_j[x] \tag{A.25}
\]

\[
\hat{E}(u_k | \Xi, \hat{\alpha}, \hat{\beta}) = \frac{1}{X} \sum_{x=1}^{X} u_k[x] \tag{A.26}
\]

The function $I$ denotes the identity function, and returns 1 if the specified condition is true, or 0 otherwise. Eq A.21 corresponds to the probability that TF $j$ regulates gene $i$. Eq A.22 corresponds to the probability that modifier $k$ has a synergistic effect on the activity of TF $j$. Eq A.23 corresponds to the expected value of $\gamma_{jk}$ for a particular synergistic effect, and the magnitude of this value can be used as an additional filter to select only those TF-Modifier edges with substantial effects on target gene expression. In this work, I typically use a threshold of $|\gamma_{jk}| \geq 0.05$ to rule out TF-Modifier connections with inconsequential effects on target gene ex-
pression. In general, $\beta_j = 0$ (TF $j$ has no influence on target genes) is considered equivalent to $C_{ij} = 0 \forall i$ (TF $j$ has no target genes), and $\gamma_{jk} = 0$ (TF $j$ and modifier $k$ do not have a synergistic effect) is considered equivalent to $D_{jk} = 0$ (TF $j$ and modifier $k$ do not interact). These equivalencies are built into the inference calculations above. Eq A.25 is used to estimate the relative quality of priors for each TF $j$, with values of $w_j > 0.5$ favoring $b$ and values of $w_j < 0.5$ favoring $m$. Likewise, Eq A.26 estimates the relative quality of priors for each modifier $k$, with values of $u_k > 0.5$ favoring $a$ and values of $u_k < 0.5$ favoring $s$. Thus, given a compendium of reliable experimental data as input, the parameters of this model can be fit to the data in order to make inferences about the underlying biological network.

| 1. Set $w_j, u_k = 0 \forall j, k$ |
| 2. Select $C$ and $D$ values based on priors \{b, m, a, s\} |
  - Deterministically: $C_{ij} = 1$ iff $\sqrt{b_{ij}m_{ij}} \geq q$; $D_{jk} = 1$ iff $\sqrt{a_{jk}s_{jk}} \geq q$, or
  - Stochastically: sampling $C_{ij} = 1$ with probability $\sqrt{b_{ij}m_{ij}}$ and $D_{jk} = 1$ with probability $\sqrt{a_{jk}s_{jk}}$
| 3. Randomly perturb (flip) fixed proportion of edges in $C$ and $D$ (optional) |
| 4. Fit $\{\alpha, \beta, \gamma\}$ to network by OLS |
| 5. Randomly perturb values in $\{\alpha, \beta, \gamma\}$ by adding noise sampled from $N(0, \nu)$ (optional) |
| 6. Set $\sigma^2 = s^2$ (from Eq A.11) |

Box A.1: Parameter initialization procedure for model estimation.
1. Set each $\alpha_i = \mu_{\alpha_i}$ (from Eq A.2)

2. Set each $\beta_j = \mu_{\beta_j}$ (from Eq A.5)

3. Set each $\gamma_{jk} = \mu_{\gamma_{jk}}$ (from Eq A.8)

4. Set $\sigma^2 = s^2$ (from Eq A.11)

5. For each $C_{ij}$:
   - Compute $Z_0^{(C_{ij})}, Z_1^{(C_{ij})}$ (from Eq A.13, A.14)
   - Set $C_{ij} = 1$ iff $Z_1^{(C_{ij})} > Z_0^{(C_{ij})}$, otherwise set $C_{ij} = 0$

6. For each $D_{jk}$:
   - Compute $\zeta_0^{(D_{jk})}, \zeta_1^{(D_{jk})}$ (from Eq A.15, A.16)
   - Set $D_{jk} = 1$ iff $\zeta_1^{(D_{jk})} > \zeta_0^{(D_{jk})}$, otherwise set $D_{jk} = 0$

7. Set each $w_j = \arg\max_{\lambda} (p_\lambda)$ (from Eq A.17).

8. Set each $u_k = \arg\max_{\lambda} (p_\lambda)$ (from Eq A.18).

9. Repeat until $\Delta \ln(L) < 0.0001$ (from Eq A.19)

Box A.2: Local maximization procedure for model estimation.
1. Sample each $\gamma_{jk} \sim N(\mu_{\gamma_{jk}}, \nu_{\gamma_{jk}})$ (from Eq A.9, A.8)

2. Sample $\sigma^2$ from the scaled-inverse $\chi^2$ distribution with $TN + 2$ degrees of freedom and scale parameter $s^2$ (from Eq A.11)

3. Sample each $C_{ij}$ from the Bernoulli distribution with $p = \frac{Z_1^{(C_{ij})}}{Z_1^{(C_{ij})} + Z_0^{(C_{ij})}}$ (from Eq A.13, A.14)

4. Sample each $D_{jk}$ from the Bernoulli distribution with $p = \frac{\zeta_1^{(D_{jk})}}{\zeta_1^{(D_{jk})} + \zeta_0^{(D_{jk})}}$ (from Eq A.15, A.16)

5. Sample each $w_j$ using grid sampling for Eq A.17

6. Sample each $u_k$ using grid sampling for Eq A.18

7. Repeat for $\ell \times X$ iterations

Box A.3: Local sampling procedure for model estimation.
Appendix B

Derivation of TF-Gene Interaction Priors from ChIP-seq Data

Where applicable, I used similar methods to those in ref. [22] in order to derive interaction priors for MoNsTEr. However, Chen et al. did not handle ChIP-seq data, which is substantially different, mathematically speaking, from the ChIP-chip data analyzed in ref. [22]. In this work, I used ChIP-seq data in the form of mapped reads, i.e., short sequence reads that have already been mapped to a unique genomic location (see ref. [165]). I then assembled the individual reads into peak regions and assessed the relative confidence of these regions using GLITR [195]. From this point, I developed a novel method for mapping the output from GLITR to TF-Gene interaction priors for use with MoNsTEr (Chapter 3).

GLITR outputs a score $X \in \{0, 1, \ldots, 100\}$ for each ChIP and Pseudo region, with 0 corresponding to the highest confidence. The proportion of Pseudo regions passing a given threshold $X \leq x$ is used as an estimate of the number of false positives among the ChIP regions passing this same threshold. Thus, the FDR at
any threshold $x$ is given by Eq B.1, originally from ref. [195]:

$$FDR(x) = \frac{\text{Proportion of Pseudo peaks with } X \leq x}{\text{Proportion of ChIP peaks with } X \leq x} \quad (B.1)$$

For use in MoNsTER, each peak score $X$ must be mapped to a prior probability, rather than an FDR. To map scores $X$ to prior probabilities $p_X$, I rely on the same assumption as the FDR calculation, namely that the distribution of scores for the Pseudo peaks is an estimate of the false positives among the ChIP peaks. Thus, I can derive a reasonable estimate of the desired probability:

$$p_x = 1 - \left( \frac{\text{Proportion of Pseudo peaks with } X = x}{\text{Proportion of ChIP peaks with } X = x} \right) \quad (B.2)$$

Note that $p_X$ (Eq B.2) differs from the FDR estimate (Eq B.1) in that it uses only the proportion of regions at a precise score, rather than all regions passing a threshold. In practice, the number of regions are under-sampled for some scores, and therefore the approximation of $p_x$ is not robust for all $x$. To adjust for this problem, I took two basic steps. First, I binned the scores $X$ into bins of size 5, and computed a more robust probability for the entire bin. In other words, $p_0 = \cdots = p_4 = p_{X \in [0,4]}$, and so on. The majority of peaks have $X = 100$, and so $p_{100}$ was computed as a single bin. Second, I assumed that a correct mapping $X \rightarrow p_X$ should be monotonically decreasing. In other words, if score $x_1$ is better (lower) than another score $x_2$, then $p_{x_1} \geq p_{x_2}$ should be true in all such cases. I enforce monotonicity by iterating through $x$ from 1..100, and assign $p_x = \max(p_x, p_{x-1})$. The final mapping from $X \rightarrow p_X$ for the ChIP-seq data used in Chapter 4 is shown in Figure B.1. Note that it is desirable that $p_{100} = 0$, as these are the majority of the false positives in the data, and regions with $p = 0$ can be ignored for subsequent analysis. In this
Figure B.1: Mapping STAT1 ChIP-seq peak scores to probabilities, based on output from GLITR algorithm [195]. $X = 0$ corresponds to the highest confidence peaks.

application, $p_x = 0 \ \forall x > 64$, effectively filtering out most poor-scoring peaks in the data.

The GLITR algorithm does not explicitly compare Pseudo and ChIP peaks that overlap in the genome. Such cases may be indicative of regions where DNA is more accessible, but where there is no actual TF binding site. To account for these regions, I applied the same mapping $X \rightarrow p_X$, derived for ChIP peaks, to the scores for all Pseudo peaks. Let $p_c$ be the probability of a particular ChIP peak, and let $p_u$ be the probability of some overlapping Pseudo peak. Let $p_f$ denote the final probability assigned to the ChIP region after consideration of overlapping Pseudo...
regions, calculated by:

\[ p_f = \max(0, p_c - \max p_u) \]  

(B.3)

In other words, simply subtract the probability of the strongest overlapping Pseudo peak from the probability of the ChIP peak. This essentially enforces a penalty on ChIP peaks with overlapping Pseudo regions. If the Pseudo peak has probability greater than or equal to that of the ChIP peak (i.e., \( \max p_u \geq p_c \)) then the ChIP peak is removed entirely (\( p_f = 0 \)).

The goal of this analysis is to derive prior probabilities for the regulation of target genes \( i \) by TF \( j \), rather than for binding to genomic regions in general. Therefore, I must also map individual peaks with \( p_f > 0 \) to proximal genes in order to derive final values for \( b \). The most reasonable and straightforward way to perform this mapping is to simply define the presumed promoter regions as a fixed amount of sequence upstream of each gene start site, then use the maximum \( p_f \) for all peaks overlapping the presumed promoter. In this analysis, I defined the presumed promoter as the 1kb region upstream of each gene start site as defined in RefSeq [156]. Thus, for the analysis in Chapter 4, \( j = \text{STAT1} \), and \( b_{ij} = \max p_f \) using all peaks overlapping the designated 1kb upstream region for each gene \( i \).
List of Abbreviated Terms

**ARACNE** Algorithm for the Reconstruction of Gene Regulatory Networks

**AUC** Area Under the Curve

**BP** Biological Process

**cAMP** Cyclic Adenosine Monophosphate

**CBP** CREB Binding Protein

**ChIP** Chromatin Immunoprecipitation

**CI** Confidence Interval

**CID** Collision-Induced Dissociation

**CLR** Context Likelihood of Relatedness

**COGRIM** Clustering of Genes into Regulons using Integrated Modeling

**CRE** cAMP Response Element

**CREB** cAMP Response Element-Binding

**Da** Dalton ($m/z$ unit)

**DNA** Deoxyribonucleic Acid

**FDR** False Discovery Rate

**FOCI** First-Order Conditional Independence

**GAS** Interferon-$\gamma$-Activated Site
GEO  Gene Expression Omnibus

GLITR  Global Identifier of Target Regions

GO  Gene Ontology

GRAM  Gene Regulatory Modules

IAA  Iodoacetamide

IFN  Interferon

ISRE  Interferon-Stimulated Response Element

JAK  Janus Kinase

KEGG  Kyoto Encyclopedia of Genes and Genomes

LC  Liquid Chromatography

MAPK  Mitogen-Activated Protein Kinase

MFG  Modifier-TF-Gene

MINDY  Modulator Inference by Network Dynamics

MoNsTEr  Modification-dependent Network-based Transcriptional Estimator

mRNA  Messenger Ribonucleic Acid

MS  Mass Spectrometry

ORF  Open Reading Frame

PAINT  Promoter Analysis and Interaction Network Toolset

PIC  Pre-Initiation Complex

PKA  Protein Kinase A

PTM  Post-Translational Modification

PWM  Positional Weight Matrix
ROC  Receiver Operating Characteristic
RIPK  Receptor-Interacting Protein Kinase
RNA  Ribonucleic Acid
RNAPII  RNA Polymerase II
SELEX  Systematic Evolution of Ligands by Exponential Enrichment
SGD  Saccharomyces Genome Database
SST  Somatostatin
STAT  Signal Transducer and Activator of Transcription
TF  Transcription Factor
TNF  Tumor Necrosis Factor
TNFR  Tumor Necrosis Factor Receptor
TPP  Trans-Proteomic Pipeline
TSS  Transcription Start Site
UPS  Universal Proteomics Standard
Bibliography


174


[83] Z. He, F. Tang, S. Ermakova, M. Li, Q. Zhao, Y. Cho, W. Ma, H. Choi, A. M. Bode, C. S. Yang, and Z. Dong. Fyn is a novel target of (-)-epigallocatechin...


[92] Y. Q. Huang, J. J. Li, and S. Karpatkin. Thrombin inhibits tumor cell growth in association with up-regulation of p21(waf/cip1) and caspases via a p53-


[137] I. Mayordomo, F. Estruch, and P. Sanz. Convergence of the target of rapamycin and the snf1 protein kinase pathways in the regulation of the sub-


187


