

# **Mechanistic Bayesian Networks for Integrating Knowledge and Data to Unravel Biological Complexity**

by

**Abhik D. Shah**

A dissertation submitted in partial fulfillment  
of the requirements for the degree of  
Doctor of Philosophy  
(Bioinformatics)  
in The University of Michigan  
2011

Doctoral Committee:

Assistant Professor Peter J. Woolf, Co-Chair  
Professor Brian D. Athey, Co-Chair  
Professor Alfred O. Hero III  
Assistant Professor Yongqun He  
Assistant Professor Venkateshwar Keshamouni

© Abhik D. Shah

---

All Rights Reserved

2011

To my parents, Deepak and Usha Shah, who moved across the globe and made untold sacrifices so their children could learn and make a better life for themselves.  
Thank you mom and dad.

# Acknowledgments

”It’s about building a Seiko for a million people rather than a Rolex for a few”. With those words during our first meeting, Peter Woolf explained the perspective behind systems biology and his engineering mindset. His words immediately resonated with my own thoughts. Since then, Peter has been a great mentor and friend and has shaped both my professional and personal growth. His ideas and endless enthusiasm have influenced every part of this thesis and have been invaluable.

I would also like to thank Brian Athey and my committee for teaching me to think and communicate like a scientist. I’ve realized that a lot of learning takes place outside the classroom or the lab – it’s through conversations and by watching and listening that I’ve learned much about the attitude and hard work required to make scientific progress.

The Bioinformatics office staff, especially Yuri and Julia, have been incredibly patient with late registrations, incomplete forms and much more. Without them, us students would be completely lost.

Finally, I would like to thank my family and friends in Ann Arbor, BRC and beyond. Whether it’s listening to me go on and on about Bayesian networks without showing their boredom, helping me deal with the stress of graduate work or simply being who they are, their support has been invaluable though not always acknowledged.

# Table of Contents

<b>Dedication</b> . . . . .	ii
<b>Acknowledgments</b> . . . . .	iii
<b>List of Tables</b> . . . . .	vii
<b>List of Figures</b> . . . . .	viii
<b>Abstract</b> . . . . .	ix
<b>Chapter 1 Introduction</b> . . . . .	1
1.1 Background and motivation . . . . .	1
1.2 Current challenges . . . . .	2
1.2.1 General interaction networks . . . . .	2
1.2.2 Experimental data . . . . .	2
1.2.3 Regulatory and signaling relationships . . . . .	4
1.3 Bayesian networks . . . . .	4
1.3.1 Advantages of Bayesian networks . . . . .	4
1.3.2 Limitations of Bayesian networks . . . . .	5
1.4 Mechanistic Bayesian networks . . . . .	7
1.4.1 MBN templates . . . . .	8
1.4.2 MBN scoring metric . . . . .	8
1.4.3 MBN applications . . . . .	8
1.5 Pathway targets . . . . .	9
1.5.1 Sonic hedgehog pathway . . . . .	9
1.5.2 Prior-work: transcriptional profiling . . . . .	10
1.5.3 Prior-work: expression correlation . . . . .	11
1.5.4 Prior-work: model-based approach . . . . .	11
1.5.5 MBN solution . . . . .	12
1.6 Active pathways and subnetworks . . . . .	12
1.6.1 Epithelial-mesenchymal transition . . . . .	12
1.6.2 Prior-work: pathway analysis and enrichment . . . . .	13
1.6.3 Prior-work: seed expansion . . . . .	14
1.6.4 Prior-work: subnetwork identification . . . . .	14
1.6.5 MBN solution . . . . .	15

1.7	Contributions of this Thesis . . . . .	16
<b>Chapter 2</b>	<b>Bayesian Network Theory and Software . . . . .</b>	<b>18</b>
2.1	Bayesian statistics and notations . . . . .	18
2.2	Model representation . . . . .	20
2.3	Learning models . . . . .	22
2.3.1	Hand-built structure and parameters . . . . .	22
2.3.2	Learning parameters from data . . . . .	23
2.3.3	Learning structure and parameters from data . . . . .	23
2.4	Scoring metrics . . . . .	24
2.4.1	Posterior probability of a model . . . . .	25
2.4.2	BD scoring metric . . . . .	25
2.4.3	Discretizing data . . . . .	28
2.4.4	Handling interventions . . . . .	29
2.4.5	Handling missing values and hidden variables . . . . .	30
2.5	Network search algorithms . . . . .	31
2.6	MBN templates . . . . .	34
2.7	Software implementation . . . . .	35
2.7.1	Structure learning . . . . .	35
2.7.2	Convenience and scalability . . . . .	36
2.7.3	HTML report . . . . .	37
2.7.4	Extensive documentation . . . . .	41
2.7.5	Development process . . . . .	41
2.7.6	Related software . . . . .	41
2.8	Conclusions . . . . .	42
<b>Chapter 3</b>	<b>Identifying Pathway Targets . . . . .</b>	<b>43</b>
3.1	Sonic hedgehog pathway . . . . .	43
3.2	Methods . . . . .	45
3.2.1	Mechanistic Bayesian networks: theory and definition . . . . .	45
3.2.2	Mechanistic Bayesian network templates . . . . .	46
3.2.3	Maximum entropy discretization of the data . . . . .	47
3.2.4	Calculating the posterior probability of a model . . . . .	47
3.2.5	BD scoring metric . . . . .	48
3.2.6	Calculating the marginal likelihood with missing data . . . . .	48
3.2.7	Calculating the marginal likelihood with interventions . . . . .	49
3.2.8	Assessing model significance with bootstrapping based p-values . . . . .	49
3.2.9	Using MBN templates to define and identify shh targets . . . . .	49
3.2.10	Experimental data . . . . .	50
3.2.11	Data preprocessing . . . . .	50
3.2.12	MBN template analysis . . . . .	51
3.3	Results and discussion . . . . .	52
3.3.1	Known Shh targets not identified by MBN . . . . .	53
3.3.2	Comparison with non-Bayesian bioinformatics techniques . . . . .	55
3.3.3	Comparison with standard BN modeling of the Shh pathway . . . . .	56

3.3.4	Extension to larger MBN templates	57
3.3.5	Extension of MBN to other data types	57
3.4	Conclusion	58
<b>Chapter 4</b>	<b>Determining Context-specific Subnetworks</b>	<b>59</b>
4.1	Introduction	59
4.2	Methods	62
4.2.1	Theoretical model: bipartite MBN model	62
4.2.2	Theoretical model: interaction score	63
4.2.3	Theoretical model: assessing significance by bootstrapping	63
4.2.4	Theoretical model: target and downstream gene sets	63
4.2.5	The netshadow pipeline	64
4.2.6	Netshadow: querying prior knowledgebases	64
4.2.7	Netshadow: scoring models	65
4.2.8	Netshadow: weighted subnetwork	65
4.2.9	Netshadow: results visualization in cytoscape	65
4.2.10	Synthetic dataset	66
4.2.11	Experimental dataset and pre-processing	66
4.3	Results	66
4.3.1	Simulated study with synthetic data	67
4.3.2	Netshadow analysis of EMT interactome	67
4.3.3	Significant protein-protein interactions	67
4.3.4	Significant protein hubs	67
4.3.5	Enriched processes, components, and pathways	72
4.4	Discussion	73
4.4.1	Netshadow identifies a core subnetwork underlying emt	74
4.4.2	Netshadow identifies relevant linear and nonlinear interactions	74
4.4.3	Netshadow identifies interactions missed by topological analysis	74
4.5	Conclusion	75
<b>Chapter 5</b>	<b>Conclusions and Future Directions</b>	<b>76</b>
5.1	Mechanistic Bayesian networks	76
5.1.1	Python environment for Bayesian learning	77
5.1.2	MBN for identifying pathway targets	78
5.1.3	MBN for subnetwork identification	78
5.2	Extensions to MBN	79
5.2.1	Identifying Shh coregulators and pathway crosstalk	80
5.2.2	Limitations of MBN templates	80
5.2.3	Increasing the expressiveness of MBN templates	81
5.2.4	Generating hypotheses	81
5.2.5	Cloud computing	82
5.3	A future MBN workflow	84
<b>Bibliography</b>		<b>85</b>

# List of Tables

## Table

2.1	Number of BN models . . . . .	32
2.2	Comparing BN structure learning software . . . . .	41
3.1	Shh knockout experimental design . . . . .	50
3.2	Results from the MBN analysis . . . . .	52
3.3	Results from the sequential BN analysis . . . . .	55
3.4	Results from the parallel BN analysis . . . . .	56
4.1	Top 20 interactions in the EMT subnetwork . . . . .	69
4.2	Top 20 hubs in the EMT subnetwork . . . . .	70
4.3	Top 20 cellular components in the EMT subnetwork. . . . .	71
4.4	Top 20 biological processes in the EMT subnetwork. . . . .	73
4.5	Top 10 KEGG pathways in the EMT subnetwork . . . . .	75



# List of Figures

## Figure

1.1	HPRD interaction network . . . . .	3
1.2	Mechanistic Bayesian network workflow . . . . .	7
1.3	Mechanistic Bayesian network overview . . . . .	17
2.1	Example Bayesian network . . . . .	21
2.2	Effect of Dirichlet parameter priors . . . . .	26
2.3	Effect of discretization . . . . .	29
2.4	Local maximum in network search space . . . . .	33
2.5	Example MBN templates . . . . .	34
2.6	Using PEBL with a script and configuration file . . . . .	36
2.7	HTML report: summary statistics . . . . .	38
2.8	HTML report: top scoring networks . . . . .	39
2.9	HTML report: consensus networks . . . . .	40
3.1	Different representations of the Shh pathway . . . . .	44
3.2	Relative expression of putative Shh targets . . . . .	54
4.1	Netshadow example with synthetic data . . . . .	62
4.2	Netshadow analysis pipeline . . . . .	64
4.3	Extracting the EMT subnetwork from the global interactome . . . . .	68
4.4	Genes downstream from the interaction between ATXN1 and ANP32A . . . . .	72
5.1	Other uses of MBN templates . . . . .	80
5.2	A more expressive Shh template . . . . .	82
5.3	Future MBN workflow . . . . .	83

# Abstract

The determination of how protein interactions affect gene regulation is an important problem in systems biology. By identifying quantitative relationships between the interactome and transcriptome in complex pathologies, we can better characterize dysfunctional pathways, generate further hypotheses and identify potential targets for therapeutic interventions. This thesis develops methods and software for elucidating biological networks consistent with known mechanisms using Bayesian networks (BN) and high-throughput datasets in a novel methodology termed Mechanistic Bayesian networks (MBN).

This thesis contributes new algorithms for data pre-processing and evaluating hidden variable models that are implemented in PEBL, an open-source library for MBN modeling with features unmatched by other software. Due to its ease of use and extensibility, PEBL allows one to run large, distributed analyses using cloud-computing platforms.

MBN are used to identify the targets of the Sonic hedgehog signaling pathway that is implicated in development and cancer progression. The use of hidden variable models and the ability of BN to capture nonlinear, combinatorial and stochastic relationships identifies known and novel targets that are more biologically meaningful and outperforms other BN and non-BN methods. The approach developed is useful for identifying pathway targets, upstream regulators or, more generally, to identify additional components of partially-characterized topologies.

MBN are next applied to identify subnetworks of the global interactome that govern gene expression during the epithelial-mesenchymal transition (EMT), a developmental process implicated in cancer metastasis. By modeling the effects of a protein interaction on downstream genes, a scoring metric was developed that quantifies the relevance of interactions to EMT. Application of the method to a cell-line lung cancer dataset identifies a core subnetwork that recapitulates EMT biology and makes predictions about protein interactions and their targets. Because the method does not rely on differential expression or the co-regulation assumptions, it is equally useful for microRNA-target, protein-DNA or

mixed-interaction networks.

The methods and software in this thesis are generally applicable to problems in elucidating interactions among variables using partially characterized knowledge and noisy high-dimensional datasets and furthers state-of-the-art BN methods by identifying results consistent with both known mechanisms and statistical relationships in data.

# Chapter 1

## Introduction

### 1.1 Background and motivation

A common challenge in bioinformatics is the integration of diverse datatypes to elucidate mechanisms in complex biological systems. In this thesis, I present Mechanistic Bayesian networks as a novel method for identifying molecular entities and interactions that are consistent with both existing biological knowledge and statistical relationships in high-throughput experimental data. Although the methods developed in the thesis are general and can be applied to many problems in high-dimensional domains under uncertainty, they are demonstrated on two biological processes of scientific and clinical importance.

Many of the phenotypes and pathologies we are interested in understanding emerge not from a single gene but a complex network of space-, time- and context-dependent interactions among a large number of biological molecules of different types (Jenkins, 2009; Kalluri, 2009). The systems approach to biology contends that components of the cellular network do not function the same in isolation and need to be characterized within the context of either the entire network or a suitable subset (Sauer et al., 2007). Accordingly, we have begun to map individual *horizontal* layers of the cellular network, leading to the so-called *omes* that characterize the entire set of molecules or interactions of one type (Greenbaum et al., 2001). For example, the transcriptome is the set of all mRNA (Velculescu et al., 1997) and the protein interactome is the set of all observed or computationally-predicted protein-protein interactions (Sanchez et al., 1999). Understanding the mechanisms underlying a specific pathology requires identification of interactions within and between the omes in biologically-relevant contexts.

## 1.2 Current challenges

Sources of information useful in identifying biologically-relevant networks can be classified into two broad categories: general interaction networks and context-specific experimental data. Both sources possess characteristics that hamper the identification of relevant interactions. Furthermore, the types of regulatory and signaling relationships known to occur among biomolecules pose challenges to some analytical methods. In this section, the relevant characteristics of interaction networks, experimental data and biomolecular relationships are briefly described.

### 1.2.1 General interaction networks

Although much is known about biological networks, the existing knowledge is generic. Examples include interactions observed in noisy high-throughput assays such as yeast two-hybrid (Young, 1998), low-throughput methods such as co-immunoprecipitation (Moresco et al., 2010) and those manually extracted from literature (Keshava Prasad et al., 2009). Interactome databases are constructed by aggregating observations and predictions from heterogeneous celltypes and experimental conditions and include many false-positive and false-negative results (Hakes et al., 2008). A typical interactome diagram is shown in figure 1.1. It depicts the Human Protein Reference Database (HPRD) interactome as manually extracted from 19,924 journal articles representing many biological and experimental conditions (Keshava Prasad et al., 2009). The interaction network does not identify which nodes and edges are relevant to a specific phenotype and is too large for human evaluation. Although knowledge-driven techniques such as network analysis can identify important nodes and edges based on topological measures such as centrality and betweenness (Barabasi and Oltvai, 2004), they cannot determine the relevance to any specific biological context and can only consider the topology of the network and not the concentrations or activities of entities.

### 1.2.2 Experimental data

Experimental data, on the other hand, are generated from specific experiments with known conditions but are noisy and incomplete. Even large datasets are sparse – they include tens of thousand of variables but only a few hundred samples. For example, the largest single dataset in the Gene Expression Omnibus (GEO) (Barrett et al., 2009), a popular public data repository, includes 41,000 genes but only 202 samples (Noble et al., 2008). Furthermore, if

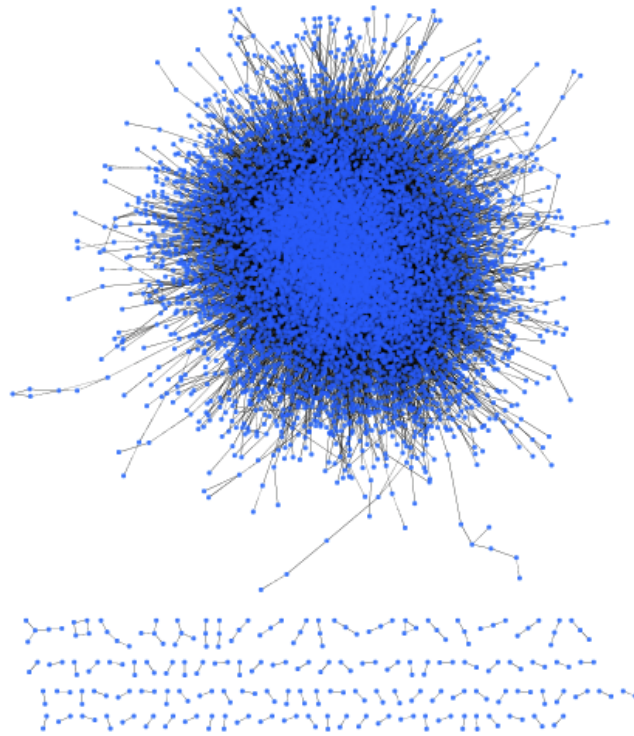


Figure 1.1: **Human protein reference database (HPRD) interaction network.** The HPRD protein interaction network includes 36,633 interactions between 9,217 proteins manually extracted from 19,924 journal articles and includes many biological and experimental conditions. The network is not specific to any condition or pathology and does not identify important nodes or edges. Visually, the network appears as a *hairball network* and is too large for human evaluation. The scale of the network necessitates further quantitative analysis to identify portions relevant to specific conditions.

---

a study contains only two conditions, determining the differences between them can identify relevant entities and interactions. Typical studies, however, include many conditions with unknown dependencies between them. For example, each of the 10 largest datasets in GEO contains multiple factors such as disease state, tissue, external stimuli and perturbations. Although there exist methods to calculate differential expression from data with multiple conditions ([Draghici, 2003](#)), they do not take full advantage of the data. Analyzing the specific pattern of changes between conditions rather than just the magnitude or significance of change, can identify relationships among genes. Furthermore, most studies only include a few types of assays and many relevant measurements and interactions such as protein concentration or protein activity remain unobserved.

### 1.2.3 Regulatory and signaling relationships

Finally, many regulatory and signaling relationships involve more than two entities and are known to be combinatorial, multimodal, nonlinear and stochastic. With existing knowledge that is generic and data that are noisy, sparse and incomplete, it is difficult to identify specific mechanisms. But identification of these signaling and regulatory relationships via quantitative and predictive models is crucial for generating further hypotheses and developing interventional therapies. Although statistical methods such as clustering and functional enrichment can suggest broad themes in experimental data, they cannot make predictions about specific molecules and their interactions. How can we identify the context-specific physical interactions that can lead to further hypotheses and drug targets?

## 1.3 Bayesian networks

Bayesian networks (BN) are a class of probabilistic graphical models that have emerged as a popular solution for inferring biological networks (Needham et al., 2007). BN and related theory and algorithms are described in detail in chapter 2; here, I give a brief overview of their advantages and limitations. A BN consists of *nodes* that represent biological entities such as mRNA and proteins and *edges* that represent statistical dependence between nodes.

### 1.3.1 Advantages of Bayesian networks

A BN is both explanatory and predictive: the network is similar to the qualitative pathway and network diagrams familiar to biologists but also includes a quantitative component that can be used to make predictions about the effects of interventions. BN models can learn linear, nonlinear, combinatorial and stochastic relationships between variables and can be trained and evaluated using noisy data. Not only can BN algorithms infer model parameters from noisy data, there exist algorithms to learn the model structure itself directly from data. For these reasons, BN have been used for the *de novo* inference of different types of biological networks using a variety of data types such as gene expression and protein phosphorylation states (Friedman et al., 2000; Friedman, 2004; Smith et al., 2002; Husmeier, 2003; Woolf et al., 2005; Imoto et al., 2003b; Yoo et al., 2002; Yu et al., 2004; Sachs et al., 2005). Furthermore, BN learning algorithms can integrate experimental data with both knowledge about interactions and circumstantial evidence such as colocalization and coessentiality data (Gevaert et al., 2007; Imoto et al., 2003a; Jansen et al., 2003; Werhli

and Husmeier, 2007).

One reason for the popularity of BN models in bioinformatics is that the network models resemble the qualitative pathway and network diagrams familiar to biologists. Based on this resemblance, most uses of BN in bioinformatics have been formulated as structure learning problems. One uses a scoring metric and search algorithm to find BN models most consistent with collected knowledge and data. Either the best-scoring model or a consensus network constructed from top-scoring networks is then accepted as the correct biological network. Unlike clustering methods that identify sets of similar entities or enrichment methods that identify broad themes in data, BN models identify directed relationships between entities and can help identify causal mechanisms.

### 1.3.2 Limitations of Bayesian networks

Although structure-learning has been used successfully to elucidate the relationships between genes, the approach has some limitations that are described here.

**Super-exponential search space.** First, the search space for BN models grows super-exponentially with the number of variables and the search problem has been proven to be NP-Hard (Chickering et al., 1994). For example, given 25 variables, there are  $2.7 \times 10^{111}$  possible BN models; as a comparison, estimates for the number of atoms in the universe range between  $10^{72}$  and  $10^{97}$ . Even after significantly filtering the variables in a typical microarray assay with tens of thousands of genes, the problem of identifying the most likely BN model is not solvable. This necessitates the use of search heuristics that tend to find high-scoring models but offer no guarantees. Thus, given a non-trivial structure learning problem, even with large computational clusters and grids, one can identify good solutions but is never sure whether the optimal solution has been found.

**Statistical vs physical interactions.** Second, the edges in a BN model identify statistical relationships between variables but make no guarantee that these correspond to actual physical interactions between biological molecules. The most common use of BN structure learning in bioinformatics has been to infer Gene Regulatory Networks (GRN) from microarray data (Friedman et al., 2000; Husmeier, 2003; Imoto et al., 2003b; Yu et al., 2004). An edge in a GRN implies that the parent gene regulates the child gene but does not describe the series of proteomic interactions that mediate the regulation. This is because proteins and their interactions are not observed during most experiments and thus not included in the



modeling process. In some works, existing knowledge about protein-protein interactions (PPI) have been used to bias the BN search towards models consistent with known interactions. PPI, however, are modeled as edges between nodes representing mRNA, making the implicit assumption that mRNA and protein expression are correlated (Imoto et al., 2003a; Djebbari and Quackenbush, 2008). This assumption, however, has been found to be true for permanent protein complexes but untrue for transient complexes, signaling interactions and the interactome in general (Bhardwaj and Lu, 2005; Jansen et al., 2002; Xulvi-Brunet and Li, 2010). For this reason, GRNs induced from gene expression data and PPI have shown poor correspondence with known pathways. For example, Djebbari and Quackenbush (2008) integrated microarray data for genes in the KEGG cell cycle pathway with literature derived PPI but modeled PPI as between mRNA nodes. The resulting BN models at various bootstrap-derived confidence thresholds were then compared to the true KEGG pathway. Even at the most stringent confidence thresholds, the percentage of true positives ( $PPV = \text{True Positives} / [\text{True} + \text{False Positives}]$ ) was around 50%. Although the resulting BN models identified significant statistical relationships between genes, the relationships did not always correspond to physical interactions.

Although proteins are not measured in most studies, in theory, BN can be used to model unobserved concentrations and states. A BN can contain *hidden nodes* to represent variables for which we do not have data. By integrating or sampling over the possible values for the hidden nodes, a BN model can still be scored for its consistency with existing knowledge and data. Hidden nodes, however, have not been used in a structure-learning problem to represent unobserved proteins because the sampling required to accommodate them further increases the amount of computation required for identifying high-scoring networks.

**Broadly-posed learning problem.** Finally, BN structure learning aims to find the network model that best explains the existing knowledge and observed data. The resulting models often suffer from the same problem as the large interactome diagrams: they are *hairball networks* too large for human evaluation. This is because the problem posed – of finding a BN structure that best fits knowledge and data – is too broad and does not take advantage of what is already known.

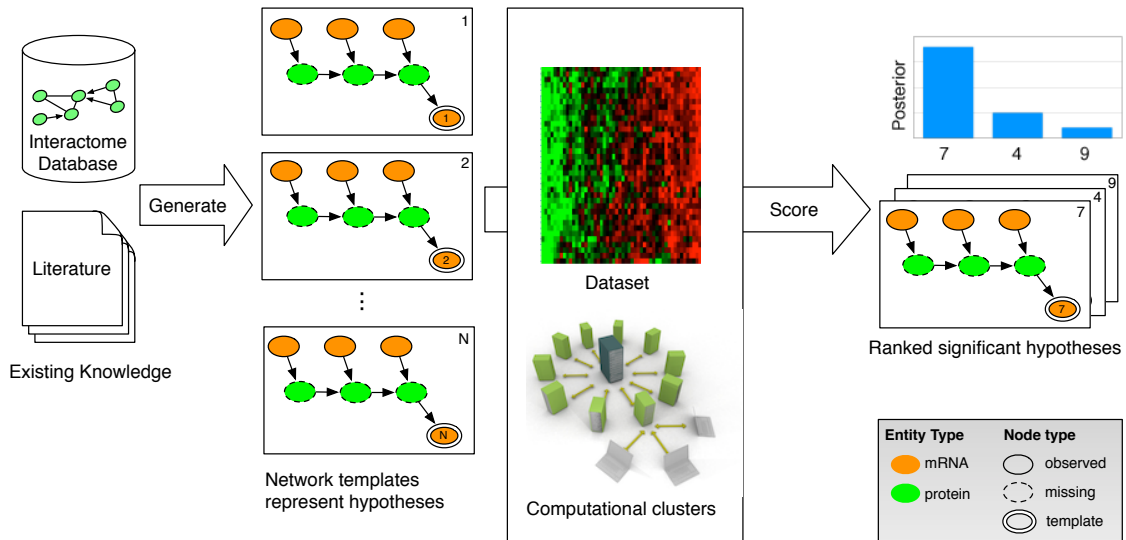


Figure 1.2: **Mechanistic Bayesian network workflow.** In a MBN analysis, one uses existing knowledge to create a set of hypotheses that are represented by a MBN template. Bayesian methods are used to assess the fit of each hypothesis to experimental data to identify models that are consistent with both existing knowledge and experimental data.

## 1.4 Mechanistic Bayesian networks

This thesis presents Mechanistic Bayesian networks (MBN), a novel approach for creating biologically-meaningful BN models by using a more targeted method than the usual structure-learning approach. Rather than searching for unconstrained BN models that best fit data, one poses specific queries about biological mechanisms in terms of network topologies. Existing partial knowledge about the topology is then integrated with context-specific experimental data to answer the query. The narrowly-posed query and existing knowledge define a constrained search space of possible answers and this smaller search space leads to two key benefits. First, known mechanisms can be more realistically modeled using computationally-intensive algorithms like Gibbs sampling. This allows, for example, for proteomic interactions to be modeled as edges between hidden nodes rather than as between mRNA nodes. Second, the search space can be exhaustively enumerated, obviating the need for search heuristics that make only asymptotic guarantees about finding the optimal solution.

### 1.4.1 MBN templates

The constrained search space is defined using a novel representation called MBN templates. A template defines a combinatorial space of BN models by mixing regular BN nodes corresponding to observed or unobserved variables with template nodes that are instantiated with variables from the dataset based on specified selection criteria. A template is thus instantiated into a set of BN models, each corresponding to one hypothesis for the query posed. Each model is scored using the scoring metric described below and finally, statistical significance is assessed via bootstrapping. Figure 1.2 depicts a typical MBN workflow; the details of each step are described in detail in chapter 2.

### 1.4.2 MBN scoring metric

BN models can be constructed with both continuous and discrete variables and with different conditional probability distributions (CPD) and prior distributions for parameters. In this thesis, I use BN with discrete variables, multinomial CPD and Dirichlet parameter priors and score models with the Bayesian Dirichlet (BD) scoring metric shown in equation 1.1. The rationale for these choices is explained in detail in chapter 2.

$$BD = \prod_{i=1}^n \prod_{j=1}^{q_i} \frac{\Gamma(\alpha_{ij})}{\Gamma(\alpha_{ij} + N_{ij})} \prod_{k=1}^{r_i} \frac{\Gamma(\alpha_{ijk} + N_{ijk})}{\Gamma(\alpha_{ijk})} \quad (1.1)$$

- $\Gamma(\bullet)$  is the Gamma function
- $n$  is the number of nodes
- $r_i$  is the arity of node  $i$
- $q_i$  is the arity of  $\pi_i$  (the parent set of node  $i$ )
- $N_{ij}$  is the number of samples where  $\pi_i$  is in configuration  $j$
- $N_{ijk}$  is the number of samples where  $\pi_i$  is in configuration  $j$  and the node has value  $k$
- $\alpha_{ij}$  and  $\alpha_{ijk}$  are the prior counts corresponding to  $N_{ij}$  and  $N_{ijk}$

### 1.4.3 MBN applications

In sections 1.5 and 1.6, I describe two common systems biology problems involving the relationships between the interactome and transcriptome and show how the use of Bayesian methods allows me to avoid the disadvantages faced by other techniques. Additionally, by formulating problems that are more constrained than general BN structure-learning, the

search space can be exhaustively enumerated and the models can encode existing knowledge more realistically via the use of hidden nodes or a more realistic encoding of known dependencies.

## 1.5 Pathway targets

A common bioinformatics problem is the determination of the downstream regulatory targets of a signaling pathway. Upon presentation of an external stimuli, often in the form of a ligand that binds a receptor on the cell surface, a series of protein interactions transduce the signal, resulting in changes in gene expression. Canonical pathways are abstractions of the portions of the cellular networks that tend to be activated together as part of a modular process. Identifying the targets of a pathway can provide insight into the effects of a pathway and can help in identifying pathways that are dysregulated in pathologies.

In this section, I first describe the Sonic hedgehog (Shh) signaling pathway as an illustration of typical difficulties in identifying pathway targets and then describe various existing methods for identifying pathway targets and discuss their advantages and limitations. Finally, I describe a MBN solution to the problem that avoids many of the limitations of other methods.

### 1.5.1 Sonic hedgehog pathway

To illustrate the difficulties in identifying pathway target, consider the Sonic hedgehog (Shh) pathway. Although the Shh pathway is discussed in further detail in chapter 3, I briefly describe the characteristics of the pathway that make identification of targets difficult.

Shh is a signaling protein that is critical in development and implicated in multiple cancers (Jenkins, 2009). Upon reception of the Shh ligand, a signaling cascade transduces the signal to the Gli transcription factors that then regulate Shh-target genes. A cell's transcriptional response to the Shh signal depends on the Shh dosage, the cell type, past or concurrent reception of other signals and on the extracellular microenvironment (Ingham and McMahon, 2001; Dessaud et al., 2007). The Shh pathway interacts with many other pathways including Bmp (Ohkubo et al., 2002), retinoic acid (Riddle et al., 1993; Kondo et al., 2005), Wnt (Iwatsuki et al., 2007), Ras (Pasca di Magliano et al., 2006), and Notch (Wall et al., 2005). Furthermore, in some contexts, Shh signaling varies from the canonical

pathway that is commonly studied (Jenkins, 2009). Identifying the transcriptional targets of Shh is a critical first step towards understanding its role in normal development and can suggest therapies for pathologies that implicate a dysfunctional Shh pathway. The context-dependent response and crosstalk with other pathways, however, make this determination difficult.

## 1.5.2 Prior-work: transcriptional profiling

The approach used for most of the large-scale pathway target identification described in literature is to determine differential gene expression between two or more samples from studies carefully designed to isolate the effects of the pathway of interest. For the Shh pathway, this has included inducing the pathway (Yu et al., 2009), inhibiting pathway members (Xu et al., 2006; Shi et al., 2010), using cell lines with known pathway dysfunctions (Yoon et al., 2002, 2009) or using mutants (McGlenn et al., 2005; Shah et al., 2009). To isolate the effects of one pathway, most studies are performed *ex vivo*. In all cases, a statistical model and threshold are chosen to determine the list of differentially expressed genes that are then accepted as pathway targets.

Although the determination of differentially expressed genes has been successful in identifying many pathway targets, it suffers from three limitations and difficulties. First, the reliance on controlled experiments to minimize the influence of other pathways makes it difficult to provide a biologically-relevant context. For example, Zhao et al. (2003) could only corroborate 37% of the previously-identified *in vitro* MyoD targets in their *in vivo* assay. Second, a recent study by Parikh et al. (2010) noted that determining downstream targets "depends crucially on the nature of the experiments" and even with experiments carefully designed to only perturb a single pathway, they could not delineate the targets of two tightly-coupled pathways when using differential expression to identify pathway targets. When attempting to build a database to identify causal pathways given a set of differentially expressed genes, they were forced to combine the MAPK and PI3K pathways into one because they couldn't identify a significant number of non-overlapping targets for each individual pathway. Additionally, among the 11 pathways they considered, they found at least 50% overlap in target genes in 3 cases. Third, the use of differential expression to infer changes in biological activity is problematic. While there are many statistical methods to calculate differential expression using noisy data (Jiang et al., 2008), they all aim to quantify the transcriptional activity of genes. Not all targets, however, will exhibit large expression changes and the magnitude of change cannot be used as proxy for importance

of target. Transcription factors (TF), for example, are often biologically-important targets of signaling pathways because they transduce the regulatory influence of the pathway onto many other genes; their expression change, however, is typically low because they are more often regulated post-transcriptionally (Yamaguchi et al., 2007).

### 1.5.3 Prior-work: expression correlation

One method of minimizing the influence of other pathways and identifying low expression-change targets is to identify genes whose expression correlates with the TFs known to be the effectors of the pathway; this approach, however, has some limitations. First, although Parikh et al. (2010) used this method to identify the targets of some pathways, they note that transcriptional regulators are only known for a limited number of pathways. Additionally, many pathways have multiple transcriptional regulators that collectively exert combinatorial control over the targets. For example, the Shh pathway in vertebrates has three canonical transcriptional regulators: Gli1, Gli2 and Gli3 (Ingham and McMahon, 2001). Each TF has different but overlapping sets of targets and exert their regulatory influence in a cooperative and combinatorial manner leading to the so-called *Gli code* (Ruiz i Altaba et al., 2007). Such a relationship will not be identified by linear correlation. Third, many TFs including the Gli proteins, are post-translationally targeted by multiple pathways via mediators that remain largely unknown (Ingham and McMahon, 2001; Stecca and Ruiz, 2010; Katoh and Katoh, 2009); thus, even their activity is not an indication of the activity of a single pathway.

### 1.5.4 Prior-work: model-based approach

What we seek are genes whose expression is a consequence of the events in the pathway of interest. Although we do not observe their activity experimentally, we have prior knowledge about some of these interactions. By constructing a predictive model of the pathway, we can test each candidate target by adding it to the model and assessing how well its predicted expression matches the observed data. The MetaReg modeling framework has been used in this model-based approach to identify the targets of signaling pathways in yeast (Gat-Viks and Shamir, 2007; Szczurek et al., 2009). It's applicability to mammalian signaling pathways, however, remains unknown for two reasons. First, MetaReg assumes that the true relationship among the signaling ligand, pathway members and targets can be described using deterministic logical signaling and regulatory functions. Second, these functions must either be user-supplied or can be automatically chosen from a small set of pre-defined functions intended to be biologically meaningful. Szczurek et al. limited their analysis to

a set of six functions with a maximum of one input variable. These two reasons make the method problematic for analysis in higher organisms that are known to have signaling and regulatory functions that are nonlinear and stochastic.

### **1.5.5 MBN solution**

In chapter 3, I describe a MBN template based solution for identifying the targets of the Shh pathway. The early signaling events of the canonical Shh pathway are used to build a template in which hidden nodes represent unobserved signaling proteins and their interactions. The template node is the target gene downstream of this partial pathway and is instantiated by every variable in the gene-expression dataset. The MBN approach outperforms other Bayesian and non-Bayesian bioinformatics methods in identifying known targets and in making biologically-meaningful predictions.

## **1.6 Active pathways and subnetworks**

A complementary problem to the one described in the previous section is that of identifying subsets of the protein interactome relevant to the transcriptional activity observed during a complex biological process. The determination of active pathways and subnetworks is a common bioinformatics approach for analyzing transcriptional data. Often, it is the next step after basic statistical analysis of gene expression data because it can characterize observations in terms of existing knowledge and indicate further targets for study ([Draghici, 2003](#); [Curtis et al., 2005](#)). A method that better identifies the specific mechanisms involved promises to have significant impact on many aspects of basic biological and clinical research.

In this section, I first describe the epithelial-mesenchymal transition (EMT), a critical step in many developmental processes and one that is implicated in cancer metastasis. Next, I describe various methods for identifying pathways and subnetworks relevant to observed transcriptional activity. Finally, I describe the MBN solution that avoids many of the limitations of other methods.

### **1.6.1 Epithelial-mesenchymal transition**

Many biological processes involve interactions among multiple pathways and they function differently in different biological contexts. Identifying active pathways and subnetworks in

such processes is difficult and requires the integration of general interaction networks and context-specific experimental data.

One such process is the epithelial-mesenchymal transition (EMT) that, like the Shh pathway, is involved in both development and cancer progression (Yang and Weinberg, 2008). During EMT, a fully-differentiated epithelial cell undergoes a transformation to a mesenchymal phenotype. This is a critical step in cancer metastasis as it allows a localized tumor cell to free itself from neighboring cells and the basement membrane as a first step towards establishing distant tumors (Keshamouni and Schiemann, 2009). Elucidating the biological processes underlying EMT is a necessary step in understanding how non-stem-cells can undergo such change in phenotype; determining specific mechanisms rather than abstract pathways or statistical dependencies will assist in identifying potential targets for therapeutic interventions. Identifying the mechanisms underlying EMT is difficult, however, because the process requires the coordinated activities of multiple pathways and because the extent of similarities between EMT in different developmental and oncogenic contexts is unknown (Kalluri, 2009).

## 1.6.2 Prior-work: pathway analysis and enrichment

The most common approach for this task is the family of pathway analysis methods. The simplest methods, often implemented in general purpose bioinformatics software, color or resize a protein in a network diagram based on the gene's differential expression (Weniger et al., 2007; Dahlquist et al., 2002; Mlecnik et al., 2005; Shannon et al., 2003). These methods, however, force the user to synthesize the results by visual inspection. Other methods use enrichment analysis based on statistical models such as the hypergeometric distribution to identify pathways (and other abstract concepts that map to a set of gene) represented in a user-supplied gene list more often than expected by chance. Over-representation analysis methods identify pathways enriched in a list of differentially expressed genes (Huang da et al., 2009b; Sartor et al., 2010) whereas functional class scoring methods like the popular Gene Set Enrichment Analysis identify pathways that map to genes near the top of a gene list ordered by differential expression (Subramanian et al., 2005; Tian et al., 2005). Unfortunately, both classes of methods condense a pathway into a gene set, thereby discarding all topological information. A newer class of enrichment methods transform a pathway into a weighted gene set using some measure of topological importance. Measures such as perturbation factor (Draghici et al., 2007), pathway connectivity index (Gao and Wang, 2007) and topology impact score (Liu et al., 2006) have been devised to capture a gene's



topological importance or influence. In these methods, an upstream member of a pathway or a hub node with many connections would be weighted higher than a downstream member because, based on the topology, it is expected to have more impact on the rest of the pathway.

Although enrichment methods have become popular for exploratory data analysis – a recent review includes 68 different tools (Huang da et al., 2009a) – the methods, whether they consider the specific topologies of pathways or not, can only identify previously-annotated pathways. Furthermore, even in pre-annotated pathways, they do not identify specific interactions. Two other classes of methods query the global interactome rather than pre-annotated pathways to find subnetworks relevant to experimental data. They are described in the two following sections.

### **1.6.3 Prior-work: seed expansion**

Seed-expansion methods query an interactome database for interactions between differentially expressed genes; the resulting *seed* interactions most likely won't include all genes in the list or form a connected network. Thus, they use graph algorithms such as Dijkstra's algorithm for shortest path (Dijkstra, 1959) or Yen's algorithm for k shortest paths (Yen, 1970) to expand the seed network by adding other proteins and interactions. (Campanaro et al., 2007; Wachi et al., 2005; Cabusora et al., 2005). The resulting networks show the differentially expressed genes in their topological context.

### **1.6.4 Prior-work: subnetwork identification**

Whereas seed-expansion methods only use experimental data to identify the seed interactions, subnetwork identification methods search for subnetworks with either high internal similarity or high collective differential expression in the data. There are two classes of subnetwork identification methods. Node-based methods define a score for network nodes as a function of differential expression. Edge-based methods, on the other hand, define a score for network edges as a function of the gene expression similarity of the nodes connected by the edges. In both cases, the network score is defined as a function of the the node or edge scores and a search heuristic is used to find high-scoring subnetworks. Ideker et al. (2002) were the first to formulate this problem and proved that the search problem is NP-Hard. They provided a node-based score and a simulated annealing search heuristic for finding a high-scoring, though not provably-optimal, subnetwork. Other node-based scores have been proposed and used with greedy heuristics (Breitling et al., 2004; Rajagopalan and

Agarwal, 2005; Nacu et al., 2007; Ulitsky et al., 2008) and linear programming (Dittrich et al., 2008; Qiu et al., 2009). Similarly, edge-based methods have been proposed more recently with both simulated annealing (Guo et al., 2007) and greedy search heuristics (Ulitsky and Shamir, 2007).

The node and edge based subnetwork identification methods described above are problematic for the following four reasons. (1) Node-based scoring methods use differential gene expression as an indicator of protein activity but this ignores the effects of post-transcriptional and post-translational regulation. Indeed, empirical studies have found poor correlation between mRNA and protein abundance (Gygi et al., 1999; Griffin et al., 2002; Anderson and Seilhamer, 1997; Chen et al., 2002; Tian et al., 2004) and between mRNA and protein activity (Glanemann et al., 2003). (2) Even if gene expression is accepted as an acceptable proxy for protein activity, node-based methods assume that an interaction between two active proteins is active in the condition represented by the experimental data. This, however, ignores the fact that global interactomes are constructed from heterogeneous sources. (3) Although edge-based methods avoid the assumptions made by node-based methods, they assume that interacting proteins are co-regulated and would exhibit expression similarity. Although some studies have identified gene expression correlation between interacting proteins (Hahn et al., 2005), Bhardwaj and Lu (2005) found that gene expression of interacting proteins is correlated in *E. coli* but not in yeast, mouse or human. Jansen et al. (2002) found gene expression correlation between interacting proteins in yeast for a few permanent complexes but not in transient interactions or for the global interaction network in general. Furthermore, a recent study not only confirmed the lack of correlation but also observed that co-expression and protein networks have different structural properties such as degree distribution, network diameter and shortest path length (Xulvi-Brunet and Li, 2010). (4) Both node and edge based methods define network scores based on local measures (of node or edge relevance) that requires search in a combinatorial space of subnetworks and necessitates the use of heuristics.

### 1.6.5 MBN solution

Rather than using local measures of relevance (differential expression for nodes or expression similarity for edges), in chapter 4, I use a MBN approach to identify context-specific protein subnetworks. I first characterize the transcriptional activity during EMT by identifying a set of differentially expressed genes. I then construct a MBN template in which two template nodes corresponding to every edge in a global interaction network regulate this set

of active genes. The resulting scoring metric for interactions relies on the expression profiles of the two interacting proteins and a set of differentially expressed genes and is no longer a local measure. Thus, rather than searching for high-scoring subnetworks, I can score each interactions individually and accept all above a threshold. The resulting EMT subnetwork recapitulates known EMT biology and makes predictions about specific interactions and the genes they influence.

## 1.7 Contributions of this Thesis

In this thesis, I present Mechanistic Bayesian networks (MBN), a set of novel Bayesian methods and software for integrating knowledge and data that encode known mechanisms more realistically and thus uncover biological networks that are more biologically meaningful. MBN allow me to avoid some of the limitations of other methods: (1) using mRNA expression as an indicator of protein activity; (2) using the co-expression assumption that states that two proteins that interact are coregulated and thus have similar expression profiles; (3) assuming linear correlation between variables. Additionally, by formulating targeted problems, I avoid the large space of models that hinder BN structure-learning problems and can use hidden nodes to model known mechanisms more realistically.

In chapter 2, I describe the underlying theory for Bayesian networks and show how the choices for CPD and parameter priors and the use of hidden nodes lead to more realistic modeling of known mechanisms. Specifically, I describe novel methods for discretization and hidden-node sampling. I also describe MBN templates and an open-source implementation for MBN modeling that includes features unmatched by other commercial or open-source software.

In chapter 3, I use a MBN template to encode the early events of the Shh signaling cascade and determine the likely downstream transcriptional targets. The results include known targets and novel predictions. I demonstrate how the MBN encoding achieves results that are more meaningful than both the use of differential expression and other BN methods of encoding known relationships.

In chapter 4, I apply MBN templates to the problem of identifying context-specific subnetworks of the global interactome responsible for the observed transcriptional changes during a complex biological process. The MBN-based edge-scoring metric is non-local and

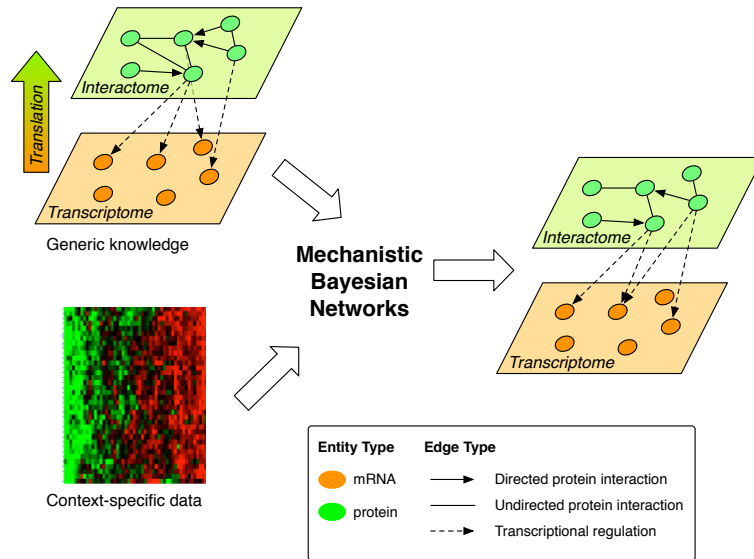


Figure 1.3: **Mechanistic Bayesian network overview.** MBN integrate existing knowledge that is generic with experimental data from specific experiments to identify context-specific cellular networks.

sidesteps the NP-Hard search space in other formulations of this problem. Furthermore, it neither relies on differential expression nor co-expression to identify relevant protein interactions. The resulting EMT subnetwork recapitulates known processes and functions and, furthermore, it makes biologically meaningful predictions about relevant interactions that would be missed by other methods.

The methods and software developed in this thesis are applicable to a large class of problems in biology and other fields where the integration of existing knowledge is both potentially useful and difficult. I apply the methods to systems of critical biological and clinical relevance and show how the proper integration of existing knowledge and experimental data can identify results that are consistent with both known mechanisms and statistical relationships in experimental data. The thesis contributes novel methods for uncovering mechanisms in complex systems and presents a strategy for a more realistic modeling of existing knowledge. The work presented here has resulted in two publications ([Shah and Woolf, 2009](#); [Shah et al., 2009](#)) and an additional manuscript currently in press.

# Chapter 2

## Bayesian Network Theory and Software

In chapter 1, Mechanistic Bayesian networks (MBN) were presented as a Bayesian network (BN) based method for solving common bioinformatics problems involving generic knowledge and noisy and incomplete experimental data. Briefly, existing knowledge is represented by a scaffold BN model and candidate hypothesis are tested by making structural changes to the models and assessing fit to data. In this chapter, I first describe the theory underlying Bayesian networks and then describe the Python Environment for Bayesian Learning (PEBL), a software library for learning BN models that offers features unmatched by other libraries.

### 2.1 Bayesian statistics and notations

Bayesian statistics is a mathematical system for describing uncertainties using probabilities. Unlike the frequentist approach, probabilities do not represent the number of occurrences of an event in repeated trials; instead, they represent a rational observers beliefs in certain propositions. The Bayesian approach is subjective: we begin with *prior* beliefs and update them using data to get *posterior* beliefs which can be used as priors for further updating or to make inferences. In the following section, I briefly describe terms and notations relevant to the exposition of Bayesian statistics.

**Random variables** Capital letters are used to denote random variables (or sets of random variables) and lowercase letters to denote specific values for those variables.

**Probabilities**  $P(A)$  refers to the probability distribution for the random variable  $A$  and  $P(A = a)$  refers to the probability of  $A$  taking on the specific value  $a$ . When the context is clear,  $P(a)$  is used to mean  $P(A = a)$ .

**Conditional Probability** The conditional probability  $P(A|B)$  is the probability of some event A, given the occurrence of another event B or, in the subjective interpretation, the belief about A after knowing the value or distribution for B. It is read probability of A under condition B or more simply as probability of A given B.

**Joint Probability** The joint probability  $P(A, B)$  is the probability of both events occurring together. Note that  $P(A, B)$  is a probability distribution whereas  $P(A = a, B = b)$  is a probability density.

**Marginalization**  $P(A)$  is called the marginal or unconditioned probability of A. For any variable B,  $P(A)$  can be calculated as the sum of all the joint probabilities of A and B. For example, if B can take on three values i, j and k:

$$P(A) = P(A, B = i) + P(A, B = j) + P(A, B = k)$$

Marginalization thus allows us to remove variables from consideration by summing (or integrating, in general) the joint probabilities of multiple variables over all possible values of the variable to be marginalized.

**Statistical Independence** A and B are said to be independent if  $P(A) = P(A|B)$ . That is, if the information derived from observing B does not alter our beliefs about A, the two variables are independent.

**Conditional Independence** A is said to be conditionally independent of B given C if:

$$P(A) \neq P(A|B) \text{ and}$$

$$P(A, C) = P(A, C|B)$$

That is, A is normally dependent on B but is independent of B when conditioned on C. Informally, one can say that any information in B that is relevant to predicting A is also contained in C.

**Bayes Rule** Formulated by Rev. Thomas Bayes, the theorem shows the relation between a

conditional probability and its inverse:

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)} \quad (2.1)$$

- $P(A|B)$  is the posterior distribution. It is our belief about A after observing B.
- $P(B|A)$  is the inverse conditional probability, also called the likelihood.
- $P(A)$  is the prior distribution. It is our belief about A before observing B.
- $P(B)$  is the marginal distribution for B.

## 2.2 Model representation

Analytical methods must decide how to represent data, relationships in the data and the learned output. Representation includes both mathematical constructs such as probability distributions and visual constructs such as the use of oval and arrows to indicate relationships between variables.

A Bayesian network includes both qualitative and quantitative representations. Qualitatively, a BN depicts the dependency structure between variables as a directed acyclical graph (DAG). A graph is a set of variables (also called nodes or vertices) and a set of pairwise relations (also called edges, arcs, or arrows) between them. In a directed graph, each edge has a direction; an edge is said to go from one node to another and the edge  $A \rightarrow B$  is not equivalent to  $B \rightarrow A$ . A DAG is a directed graph without any directed cycles (also called loops).

Formally, a BN encodes conditional independence relationships and specifies the decomposition of the joint probability distribution of all variables into lower-dimensional conditional distributions. The set of nodes that have an edge to a specific node are called the parents of that node and denoted by  $\pi_x$ . For example, in Figure 2.1,  $\pi_C = \{A, B\}$ . In a BN, a node is independent of all other nodes once conditioned on its parent set. Thus, the structure of a BN specifies how to decompose the joint distribution into lower-dimensional conditional distributions with fewer parameters that are easier to estimate from data. For example, given the BN in Figure 2.1:

$$P(A, B, C, D, E, F) = P(A)P(B)P(C|A, B)P(D|C)P(E|C)P(F|E) \quad (2.2)$$

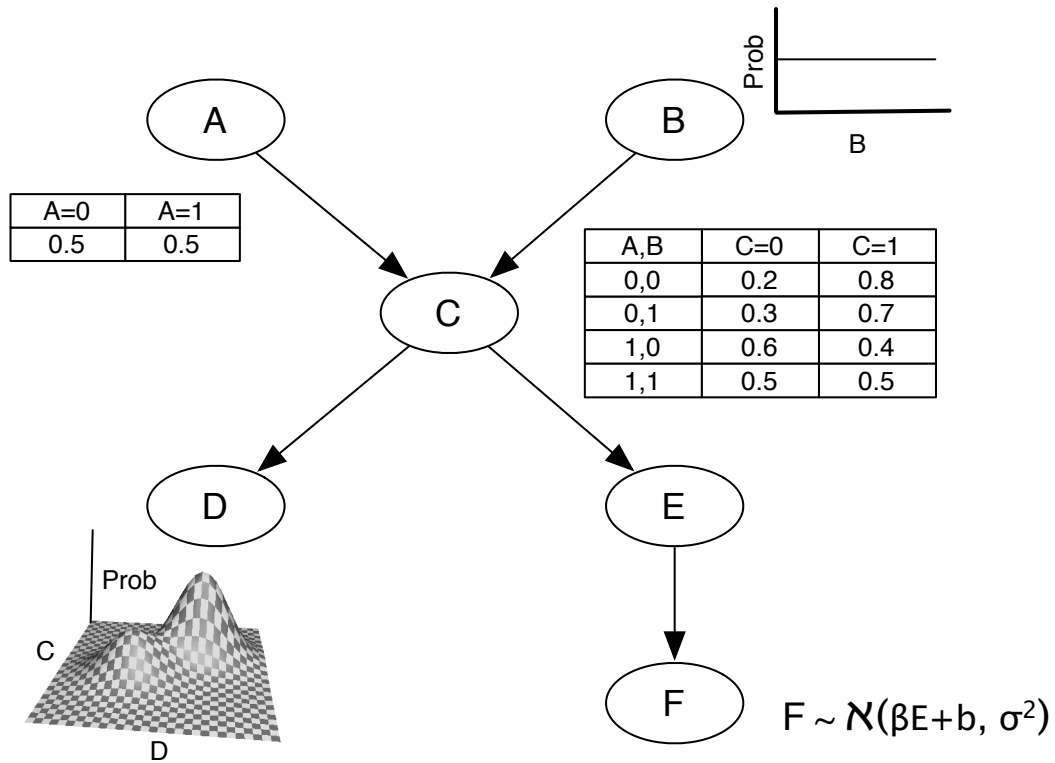


Figure 2.1: **Example Bayesian network.** This figure shows an example Bayesian Network with different forms of conditional probability distributions (CPD). Node A has a multinomial uniform distribution in the form of a conditional probability table (CPT), node B has a continuous uniform distribution, node C has a multinomial CPT, node D shows a non-parametric density estimation based CPD and node F shows a linear Gaussian CPD. Note that this example is not realistic: node B is modeled as both a continuous variable (at node B's CPD) and as discrete (at node C's CPT). In this thesis, models use multinomial distributions (as shown for nodes A and C). The structure of the network also encodes conditional independence relationship and allows for the decomposition of the joint probability as shown in Equation 2.2

The qualitative structure of BN works well for use in systems biology. Although not strictly causal, the edges in a BN are roughly causal relationships and BN have been used successfully in many causal domains. Additionally, the graphical structure of a BN maps fairly well to the pathway diagrams common in biology.

A BN is also a quantitative model that can be used to quantify causal relationships in data or to make predictions. Each node in the network includes a conditional probability distribution (CPD) relating the values of the node to the values of its parent set. Given specific values or distributions for the parent set, one can calculate the distribution for the



child node. This inference can be propagated through the network and many exact and approximate algorithms exist for inferring the distribution for a subset of variables given the observed values or distributions for other variables (Pearl, 1997; Lauritzen and Spiegelhalter, 1988).

Common choices for CPD include the continuous, linear Gaussian distribution; the discrete multinomial distribution; and non-parametric kernel densities. The linear Gaussian CPD can use continuous data such as gene expression profiles but enforce a linear relationship between a node and its parents; empirically, they do not perform well with nonlinear relationships common in biological systems (Friedman, 2004). Non-parametric kernel density methods do not assume any functional form but are computationally expensive because any calculation using the CPD requires iterating over all data points; in a non-parametric method, the data are not reduced to a smaller set of parameters. The multinomial CPD requires discrete data but can model nonlinear, multi-modal and logical relationships and is computationally tractable. MBN use the multinomial CPD for this reason but make certain modifications described later.

## 2.3 Learning models

With BN, we seek to learn graphical models from data that help us understand the causal structure of the underlying biological system. BN are typically constructed in one of three ways depending on the problem domain and availability of data:

1. hand-built structure and parameters.
2. hand-built structure with parameters learned from data.
3. structure and parameters learned from data.

I briefly describe usage 1 and 2 and then focus on 3 because that corresponds to most problems in systems biology.

### 2.3.1 Hand-built structure and parameters

A hand-built model is one that is constructed by a person rather than learned or inferred from data. Some of the earliest uses of BN were in expert systems used for diagnostic purposes. For example, consider a medical diagnostic system used to infer likely causes for observed symptoms. A domain expert – a doctor or medical researcher – would specify the relevant

causes and symptoms, lay out the causal structure between them and specify the conditional probabilities involved. Then, given a set of symptoms and observations (say, a patient's age, blood pressure and allergies), a software package could infer probabilities for likely causes. Without a Bayesian model, experts would have to estimate  $P(CAUSE|SYMPTOMS)$  for every possible combination of symptoms a patient could present. With a Bayesian model, they need to estimate  $P(SYMPTOMS|CAUSE)$  and use the Bayes rule to identify causes with high posterior probability. It was soon discovered, however, that although experts were successful at identifying relevant variables and laying out the qualitative causal structure between causes and effects, assigning specific probabilities to these causal relationships was still too difficult.

### **2.3.2 Learning parameters from data**

Based on the realization that people find it easy to describe causal structures qualitatively but not quantitatively, later expert systems hand-built the BN structure using experts but learned parameters from data. The model structure provides a framework for interpreting the data. Once a structure is specified, the conditional independences encoded in the network ensure that the CPD at a node depends only on the data for that node and its parent set. Learning the parameters amounts to fitting data to a specific probability distribution and can be accomplished using any regression method appropriate for the CPD.

Because the multinomial distribution used by MBN is discrete, it is represented by a conditional probability table (CPT). The CPT has a column for each possible value of the child node and rows for every combination of values for the parent set. The parameters for the multinomial CPT are just the observed counts in the data for every combination of parent set values and child node value. The CPT at each node can then be used to infer the child node distribution given the parent values.

### **2.3.3 Learning structure and parameters from data**

Although specifying the qualitative causal structure is easy in some domains such as medical diagnostics, it is currently not possible for systems biology. There is no expert that knows the underlying network of gene regulation or protein-protein interactions, for example. In such cases, both the network structure and parameters must be learned from data. Typically, in systems biology, we are not interested in building BN for inferential purposes but rather to understand the causal structure underlying biological systems. We are primarily inter-

ested in learning the qualitative structure of a BN and use data to select between different models. There are two main approaches to learning structure and parameters from data: constraint-based and score-based algorithms.

Constraint-based methods use statistical tests to accept or reject every possible edge of a network. The exact method is to use statistical tests to check for conditional independence of a pair of variables conditioned on every subset of the remaining variables. This is a NP-Hard problem and cannot be solved exactly (Chickering et al., 1994). Although several heuristics exist for approximate solutions (Margaritis, 2003; Tsamardinou et al., 2003, 2006), constraint-based methods have not proven to be popular in systems biology.

Score-based methods use a scoring metric for networks and a search algorithm to find high scoring networks. The space of networks is super-exponential and the search problem is NP-Hard (Chickering et al., 1994). For example, the number of network structures possible for a 25-node network is larger than the estimated number of atoms in the universe. Fortunately, there exist many search heuristics that have been used successfully in other fields and they can be adapted to search the space of BN structures. An additional benefit of score-based methods is that many commonly used search algorithms are *embarrassingly parallelizable*; that is, they can be implemented to run in parallel very easily. In the following sections, I describe the scoring metric and search algorithms used by BN.

## 2.4 Scoring metrics

When searching through the space of BN models, we need a way of comparing different models to identify the best one. A scoring metric uses data to assign a number to each model to indicate how correct we think it is. Correct, in this context, refers to how well the model matches the observed data and any known knowledge. More specifically, in the Bayesian paradigm, it is the posterior probability of the model: our prior belief in the model updated using observed data.

In this section, I first describe the theoretical problem of calculating the posterior probability of a model. I then describe the Bayesian Dirichlet (BD) scoring metric that uses a set of assumptions to simplify the calculation of the posterior and finally, describe the modifications I make to BD to score MBN.

### 2.4.1 Posterior probability of a model

The posterior probability is calculated using the Bayes rule 2.1. With data  $D$ , model structures  $M$  and model parameters  $\theta$ , we have:

$$P(M|D) = \frac{P(D|M)P(M)}{P(D)} \quad (2.3a)$$

$$\text{where } P(D|M) = \int P(D|M, \theta)P(\theta) d\theta \quad (2.3b)$$

- $P(M)$  is the structural prior; It is our prior beliefs over models. The probabilities can come from other sources of data or knowledge or can be uniform.
- $P(\theta)$  is the parameter prior for a specific model and is typically a uniform distribution.
- $P(D|M)$  is the marginal likelihood; it is calculated by marginalizing over model parameters.
- $P(D)$  is the probability of observing the data. It is a scaling term and ensures that the posterior is a true probability distribution (that it sums to 1.0).

We begin with  $P(M)$ , the prior belief about model structures. Often, this is a uniform distribution indicating that, before seeing the data, all models are equally likely to be correct. For each model, we calculate the posterior belief  $P(M|D)$  by multiplying the prior with the marginal likelihood of data given model,  $P(D|M)$ . This term assess the likelihood of the data being generated by a physical process described by the model. As described above, we calculate the likelihood by marginalizing over all possible parameter values (and their probability).

### 2.4.2 BD scoring metric

When calculating the posterior probability of a model, we must keep in mind that the data is a random sample from the underlying physical system. Another random sample (say, another set of microarray experiments on the same biological samples) would be slightly different – due to stochasticities in the physical system and in the measurement technologies – and would result in slightly different CPT. It is said that the CPT (and resulting BN models) *overfit* the idiosyncrasies in the training data and are not generalizable to the underlying physical system. This is a common problem when using machine-learning techniques with high-dimensional data and requires the use of regularization methods. With multinomial

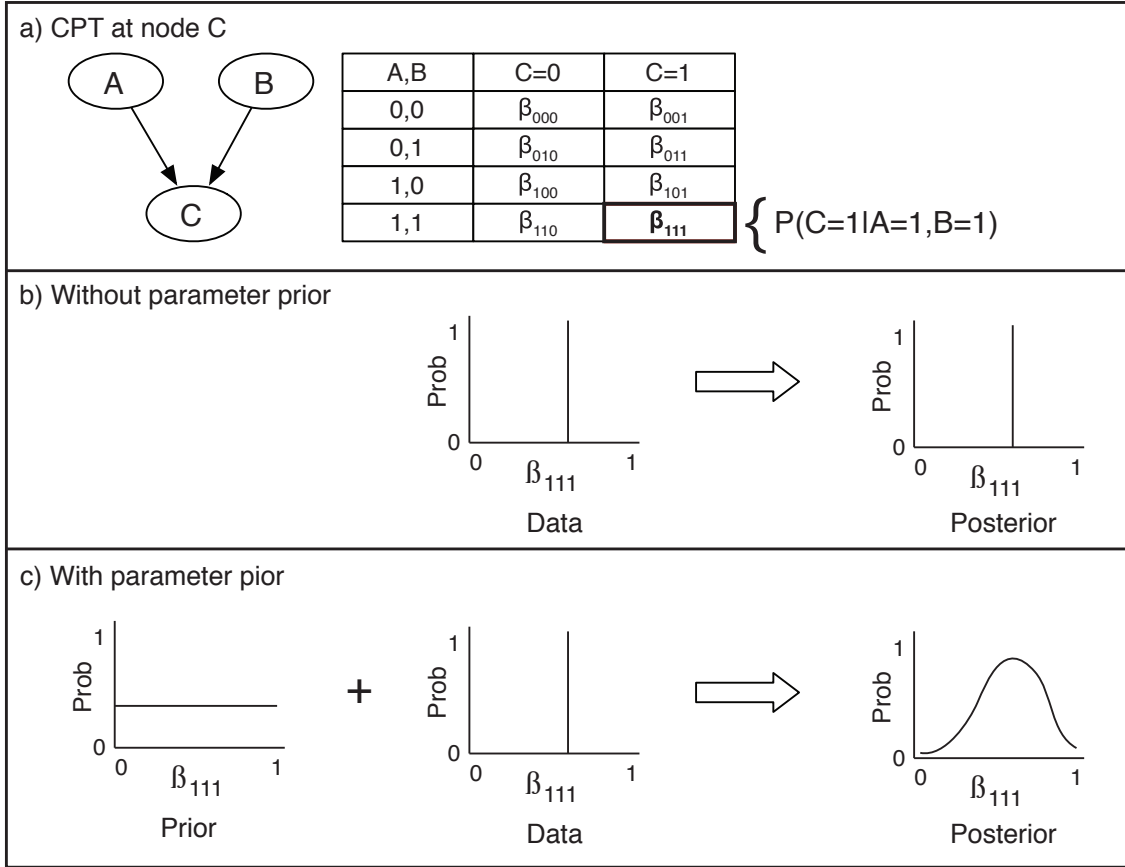


Figure 2.2: **Effect of Dirichlet parameter priors.** (a) In the network show,  $P(C|A,B)$  is encoded by a multinomial conditional probability table (CPT) in which  $\beta_{111}$  represents the number of data sample where  $A,B,C = 1,1,1$ . (b) Without a prior distribution for parameters, our posterior belief is based only on the counts in the observed data. (c) Because the observed data are stochastic samples from an underlying physical system, we use a prior distribution for  $\beta_{111}$  to avoid *overfitting* the specific idiosyncrasies of the sample. In this case, we use a Uniform distribution, indicating that before seeing the data,  $\beta_{111}$  can be any value between 0 and 1. Updating the prior with the data leads to a posterior with *shoulders* around the estimate from data.

CPD, a common regularization solution is to use Dirichlet distributions as the parameter prior. The Dirichlet distribution is the conjugate prior for the multinomial likelihood. This means that a Dirichlet prior and multinomial likelihood lead to a Dirichlet posterior. Rather than a specific value for each model parameter, we infer a probability distribution. Informally, one can think of this as adding *shoulders* to the parameter point-estimates learned from data (Figure 2.2).

To calculate the marginal likelihood requires integrating over the Dirichlet parameter

distributions. Using four assumptions listed below, the calculation can be simplified and leads to the Bayesian Dirichlet (BD) scoring metric (Cooper and Herskovits, 1992; Heckerman et al., 1995). The assumptions required to derive BD are:

1. All data are discrete.
2. Given a BN structure, data instances are independent.
3. Data contains no missing values.
4. We use a uniform prior over model parameters.

The BD scoring metric has three desirable properties for scoring BN models:

1. *global parameter independence* – parameters associated with each variable in a network are independent
2. *local parameter independence* – parameters associated with each states of the parents of a variable are independent
3. *parameter modularity* – if a variables has the same parents in two different networks, the associated parameters in both networks are the same

One property that is often desired is that of *likelihood equivalence* for BN structures in a Markov equivalence class. A Markov equivalence class is a set of BN models that encode the same conditional independencies. Likelihood equivalence states that if the data used to assess models are not the result of experimental interventions, they should not help in differentiating between BN models within an equivalence class – that is, the likelihood  $P(Data|Model)$  should be the same for all models in the class.

Although the BD metric is not likelihood equivalent, a related metric called the Bayesian Dirichlet equivalent (BDe) is likelihood equivalent. BDe is calculated as:

$$BDe = \prod_{i=1}^n \prod_{j=0}^{q_i} \frac{\Gamma(N'_{ij})}{\Gamma(N'_{ij} + N_{ij})} \prod_{k=1}^{r_i} \frac{\Gamma(N'_{ijk} + N_{ijk})}{\Gamma(N'_{ijk})} \quad (2.4)$$

- $\Gamma(\bullet)$  is the Gamma function
- $n$  is the number of nodes
- $r_i$  is the arity of node  $i$
- $q_i$  is the arity of  $\pi_i$  (the parent set of node  $i$ )
- $N_{ij}$  is the number of samples where  $\pi_i$  is in configuration  $j$
- $N_{ijk}$  is the number of sampler where  $\pi_i$  is in configuration  $j$  and the node has value  $k$
- $N'$  is the *equivalent sample size* for the prior network
- $N'_{ij}$  and  $N'_{ijk}$  are counts from a user-specified prior network

In equation 2.4,  $N'_{ijk}$  and  $N'_{ij}$  are the exponents for the prior Dirichlet distributions. Although the BDe metric is likelihood equivalent, it requires user-specified prior network to calculate  $N'_{ijk}$  and  $N'_{ij}$ . This is often not feasible in practice. Setting  $N'_{ijk} = 1$  and  $N'_{ij} = r_i$  leads to the BD metric. With BD, rather than using a prior network to calculate Dirichlet priors, we set all Dirichlet exponents to 1, making the assumption that all parameters are equally likely; that is, we believe that all relationships are equally likely before observing the data.


Although BD is easier to use in practice than BDe, we would still like to retain the likelihood equivalence property. In the next section, I show how the discretization used by MBN ensures likelihood equivalence with the BD metric without the use of a prior network for calculating Dirichlet parameter priors.

### 2.4.3 Discretizing data

BD is based on Dirichlet priors and multinomial likelihood and requires discrete data. Data taking on continuous values must be binned into a finite number of bins. There are many ways of discretizing data but MBN use a maximum-entropy method. When used with gene expression data, each discretization method represents an implicit model of how continuous expression values map to discrete functional states. According to the principle of maximum entropy (Jaynes, 1957), given a set of distributions that meet required constraints, the one that best represents the current state of knowledge is the one with largest entropy. In the case of discretization, the only constraint is the number of bins. Essentially, the maximum-entropy discretization is the least biased method of mapping continuous values to discrete states.

Accordingly, rather than creating equal-sized bins or binning based on a specific distribution such as the commonly-used Gaussian, we discretize so that each bin has the same number of values. This has two important consequences. First, variables have the maximum entropy given the number of bins; this minimizes the loss of information inherent in any discretization. Second, each variable has the same entropy. This eliminates scoring bias due to different variable entropies and ensures likelihood equivalence as shown in following example.

Suppose we generate data for two variables, A and B, where  $B = \ln(A)$  (or conversely,  $B = e^A$ ). As shown in figure 2.3, there are three possible Bayesian network models: 1) A

True Model:  $B = \ln(A)$    
 Data: 12 samples based on true model.




Discretization	entropy(A)	entropy(B)			
			$P(A)P(B)$	$P(A)P(B A)$	$P(A B)P(B)$
Max-ent	2.48 bits	2.48 bits	.0053	.50	.50
Equal-width	2.48 bits	0.96 bits	.042	.37	.58

Figure 2.3: **Effect of discretization.** Discretization that results in different entropies for variables introduces bias in the scoring (and structure learning) methods. To the best of my knowledge, this property of maximum entropy discretization has not previously been described in literature.

and B are independent, 2) A is the parent of B and 3) B is the parent of A. Given a uniform prior distribution over network structures, models 2 and 3 should have equal posterior probability because each describe the data equally well. The posterior probability for model 1 should approach 0.0 as the number of data samples increases. This is the case for the maximum-entropy discretization. When the variable entropies differ, however, as is the case in equal-width binning, the network with B as a parent scores much better.

By using a maximum-entropy discretization, MBN minimize information loss inherent in any discretization and ensure likelihood equivalence with out the use of a prior network. In the next sections, I show how BD is altered for data that are the result of experimental interventions and when data contain missing values or hidden variables.

## 2.4.4 Handling interventions

Given a node, the BD scoring metric approximates the conditional probability of the child node given its parents. Intuitively, it quantifies the effect that the parents have on the child node in the data. Some data, however, are not just passive snapshots of the underlying system. They are the result of specific external interventions. We must take care when learning CPT for variables that have been intervened upon. Consider the example of using siRNA silencing for gene knockouts. If gene A was silenced, then the expression value 0 for variable A is the result of the intervention and not of the parent nodes values; we must ensure that we do not ascribe the cause for the child nodes 0 value to the values of its parents. I follow the approach used by Yoo et al.: when constructing the CPT for a node, we omit any data instances that are the result of an intervention on that variable (Yoo et al., 2002).



## 2.4.5 Handling missing values and hidden variables

Calculating the BD score relies upon having no missing values for any variables, a restriction too strict for typical systems biology data. Data common in systems biology has both missing values (from scratches on a microarray, for example) and hidden variables (protein concentrations not measured, for example). With both missing values and hidden variables, we can still calculate the posterior probability of a network. The exact solution is to treat the missing values as a random variable and marginalize over all possible completions of the partially observed data. With all possible complete data  $D'$  given the partially observed data  $D$ , model  $M$  and model parameters  $\theta$ :

$$P(M|D) = \iint P(D'|M, \theta)P(\theta)P(D') d\theta dD'$$

The number of possible complete data is the product of the number of possible values for each missing value. Exact integration over all possible complete data is only feasible with trivial number of missing values; for most cases, I use a Monte Carlo Markov Chain (MCMC) called Gibbs sampling ([Geman and Geman, 1993](#)) to approximate the integral. Pseudocode for the application of Gibbs sampling to scoring BN models with missing values or hidden nodes is shown in [algorithm 1](#).

For MBN, I use a modified Gibbs sampler restricted to the space of complete data where all variables are maximum-entropy discretized ([Shah et al., 2009](#)). This ensures that the marginal probability – the case when a node has no parents – is the same for all nodes, regardless of whether the variable is fully observed, partially observed or hidden. This avoids the scoring bias discussed in [section 2.4.3](#). The maximum-entropy Gibbs sampler is similar to the one in [algorithm 1](#) except for two differences:

1. **Line marked 1:** Rather than initializing missing values with random values, we initialized them with random selection from the set of values that ensure that the variable is maximum-entropy discretized.
2. **Lines marked 2-4:** Rather than scoring the network by replacing a datum with all possible values, we swap the value with another for the same variable ( $Data[var][i] \leftrightarrow Data[var][j]$  where  $i \neq j$ ), thus ensuring that we only sample missing value completions that are consistent with maximum-entropy discretization for all variables.

```

Input:
  • N: Bayesian network
  • D: data matrix
  • missing: list of indices in D with missing values
  • arity: list of variable arities
  • max_iters: number of Gibbs sampling iterations
  • burnin_iters: number of iterations for burnin

Output: score for network

1 InitializeMissingValues() // using random values
  scores ← []
  while  $i < \text{max\_iters}$  do
    foreach (variable, sample) in missing do
2       scorelist ← []
3       for value := 0 to arity[variable] do
4         scorelist[value] = ScoreWithData(N, D, variable, sample, value)
        end
        chosen ← ChooseValue(scorelist) // using score as probability
        D[variable][sample] ← chosen
        scores[i] ← scorelist[chosen]
        i ← i + 1
    end
  end
return average(scores[burnin_iters:])

```

**Algorithm 1:** Gibbs sampling for scoring BN with missing values

## 2.5 Network search algorithms

The number of possible BN models grows super-exponentially with the number of variables. The number of BN models with  $n$  variables is given by the recursive equation 2.5 (Robinson, 1977). Table 2.1 lists the number of BN given different number of variables.

$$f(n) = \sum_{i=1}^n (-1)^{i+1} \binom{n}{i} 2^{i(n-1)} f(n-i) \quad (2.5)$$

The large space of BN models makes exhaustive enumeration intractable for non-trivial structure-learning problems. Thus, once we have selected a scoring metric, either the standard BD or BD with the modification described above, we need a method for searching for high-scoring models. Two commonly used search methods for BN are the greedy

Variables	BN Models
2	3
3	25
4	543
5	29,281
6	3,781,503
7	$1.1 \times 10^9$
10	$4.2 \times 10^{18}$
20	$2.3 \times 10^{72}$
25	$2.7 \times 10^{111}$
50	$7.2 \times 10^{424}$
100	$1.1 \times 10^{1631}$
500	$3.2 \times 10^{38602}$

Table 2.1: **Number of BN models** The number of BN models grows super-exponentially with the number of variables and necessitates the use of search heuristics for non-trivial problems.

---

hill-climbing and simulated annealing (SA) (Kirkpatrick et al., 1983) algorithms that are popular in many fields for non-convex search and optimization problems. I first describe hill-climbing and then describe simulated annealing as an extension to hill-climbing.

Although the space of BN models is high-dimensional and discrete, when describing search algorithms, it is helpful to visualize it as a 3D landscape with each point corresponding to a BN model and the altitude of the point corresponding to the models score. Each point has many neighboring points: all BN models that are one *edit* distance away. These are all networks that can be generated from an existing network by 1) adding an edge, 2) removing an existing edge or 3) reversing an existing edge. Both algorithms begin with an initial seed network that is either randomly generated or the best found by another method; this corresponds to either selecting a random point in the landscape or the highest peak found previously. The algorithms then explore the landscape, one step at a time, in search of the highest point.

At each iteration, hill-climbing selects a random neighbor and accepts it as the new *current* network if its score is higher. The algorithm is considered *greedy* because, at each iteration, it only considers neighboring networks and implicitly assumes that following a neighbor with a higher score will eventually lead to the global highest-scoring network. A common problem for greedy algorithms is that they get stuck on local maxima. These

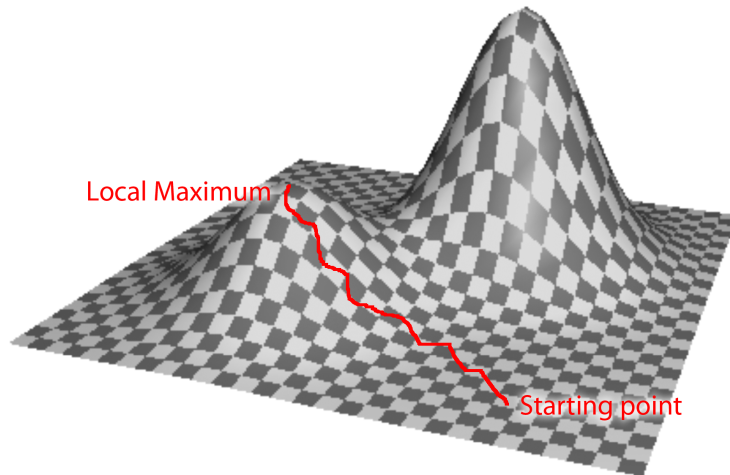


Figure 2.4: **Local maximum in network search space.** The high-dimensional, discrete space of networks is visualized as a continuous 3D landscape to illustrate the problems with local maxima. Each 2D point corresponds to a network and the height is its posterior probability (or value of some scoring metric). Because all points neighboring the indicated local maximum have a lower score, a greedy search algorithm converges with the assumption that it has found the globally maximum solution. One solution is to run the algorithm using multiple random starting points. Another solution is to introduce random jumps into the search algorithm via a method such as the one used in Simulated Annealing.

---

are minor peaks in the landscape; because all neighboring points are *below*, the algorithm believes that it has found the highest peak and thus the best BN model (Figure 2.4). Due to the susceptibility to local maxima, one typically runs hill-climbing multiple times with different initial seeds.

Another solution for avoiding local maxima is to use the simulated annealing (SA) algorithm. SA is based on the metallurgical process of annealing, a technique involving heating and gradual cooling of a metal to increase crystal size and overall strength. The heat causes the metal atoms to become unstuck from their initial positions and randomly explore states of higher energy. The slow cooling gives them more chances of finding lower energy states than their initial one. Similarly, SA begins with an initial temperature parameter that slowly decreases over many iterations according to a cooling schedule. At each iteration, a neighboring network is chosen with a probability that is a function of both the networks score and the SA temperature. At higher temperatures, SA is more likely to choose a network with a lower score than the *current* network and the search is similar to a random walk. As the temperature cools, this likelihood of choosing a worse network diminishes and at temperature 0, SA is equivalent to hill-climbing and only chooses better

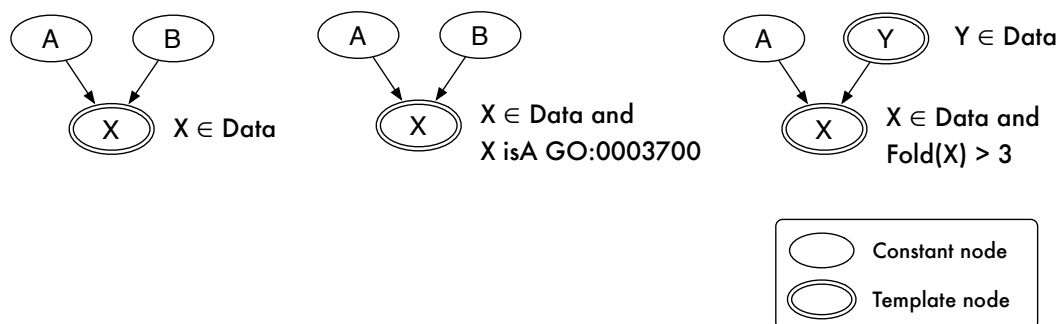


Figure 2.5: **Example MBN templates.** Templates encode topological and semantic constraints to define a subspace of BN models. Template nodes specify selection criteria that can use annotations such as Gene Ontology terms or properties of the data (such as gene expression fold change).

networks. Although SA is also a greedy algorithm, this allowance for choosing worse networks during part of the search saves SA from being stuck on local maxima; this feature, however, depends crucially on the cooling schedule which cannot be determine *a priori* and needs to be adjusted empirically for each problem.

## 2.6 MBN templates

As shown in table 2.1, the number of BN given variables grows super-exponentially. For non-trivial problems, even with large computational resources, one can only sample a small subset of the space of all BN models. With MBN, I formulate problems that are more specific than the general structure-learning problem. This allows me to exhaustively score all possible models. I present MBN templates as a way to describe a constrained space of BN models. A template looks like a regular BN – has nodes and edges and is required to be a DAG – with one crucial difference. A template includes two types of nodes: constant and template nodes. A constant node is a regular node corresponding to either one variable in the dataset or an unobserved variable while a template node is a holding place for variables in the dataset. Template nodes specify selection criteria that are used to generate specific instantiations of the template. Each instantiation is a regular BN that can be scored using existing methods. A template is a convenient way to describe and visualize constrained search spaces where all models have the same structure but different node identities. Figure 2.5 shows three examples of MBN templates and the use of selection criteria.

## 2.7 Software implementation

Bayesian statistics in general and the BN formalism specifically have many properties that are helpful in using high-throughput data to uncover mechanisms underlying complex biological phenomena. In this chapter, however, we identified several pitfalls in the way BN have been typically used in systems biology and have offered MBN as a set of modifications to address those issues. Unfortunately, when I conceived of MBN, there was no software package that met all the requirements:

- specification of soft and hard priors for integrating biological knowledge
- maximum-entropy discretization
- altered BD for interventional data
- BD with Gibbs sampling for data with missing values and hidden variables
- easy specification of custom algorithms for structure learning in constrained subspaces
- ability to use clusters, grids and cloud platforms

I developed the Python Environment for Bayesian Learning (PEBL) to meet these requirements. More importantly, PEBL is designed in a modular manner that makes it easy to extend and customize for different BN-related problems. In the following sections, I describe some notable features of PEBL that make it easy to learn and use MBN models.

### 2.7.1 Structure learning

PEBL can load data from tab-delimited text files with continuous, discrete and class variables and can perform maximum entropy discretization. Data collected following an intervention is important for determining causality but requires an altered scoring procedure (Pe'er et al., 2001; Sachs et al., 2002). PEBL uses the BD metric for scoring networks and handles interventional data using the method described by Yoo et al. (Yoo et al., 2002).

PEBL can handle missing values and hidden variables using exact marginalization and Gibbs sampling (Heckerman, 1998). The Gibbs sampler can be resumed from a previously suspended state, allowing for interactive inspection of preliminary results or a manual strategy for determining satisfactory convergence.

A key strength of Bayesian analysis is the ability to use prior knowledge. PEBL supports structural priors over edges specified as 'hard' constraints or 'soft' energy matrices (Imoto et al., 2003a) and arbitrary constraints specified as Python functions or lambda expressions.

<pre> from pebl import data, result from pebl.learner.greedy import GreedyLearner from pebl.taskcontroller.xgrid import XgridController  dataset = data.fromfile('mydata.txt') runner = XgridController('grid.com', 'pass') learners = [GreedyLearner(dataset) for i in range(10)] results = runner.run(learners) result.merge(results).tohtml('./myoutput')</pre>	<pre> [data] filename = mydata.txt [learner] type = greedy.GreedyLearner numtasks = 10 [taskcontroller] type=xgrid.XgridController [xgrid] controller = grid.com password = pass [result] output = ./myoutput</pre>
--	---

(a) Python script

(b) PEBL configuration file

Figure 2.6: Two ways of using PEBL: with a Python script and a configuration file. Both methods create 10 greedy learners with default parameters and run them on an Apple Xgid. The Python script can be typed in an interactive shell, run as a script or included as part of a larger application.

---

PEBL includes greedy hill-climbing and simulated annealing learners and makes writing custom learners easy. Efficient implementation of learners requires careful programming to eliminate redundant computation. PEBL provides components to alter, score and rollback changes to BN in a simple, transactional manner and with these, efficient learners look remarkably similar to pseudocode.

### 2.7.2 Convenience and scalability

PEBL includes both a library and a command line application. It aims for a balance between ease of use, extensibility and performance. The majority of PEBL is written in Python, a dynamically-typed programming language that runs on all major operating systems. Critical sections use the Numpy library ([Ascher et al., 2001](#)) for high-performance matrix operations and custom extensions written in ANSI C for portability and speed. The amount of time to learn networks depends on many factors including the dataset, the learning algorithm and hardware specifications. As an example, with a dataset of 10 variables and 200 samples, 100,000 iterations of the greedy learner takes approximately 10s on a 2.4 GHz Intel Core 2 Duo CPU.

PEBL's use of Python makes it suitable for both programmers and domain experts.

Python provides interactive shells and notebook interfaces and includes an extensive standard library and many third-party packages. It has a strong presence in the scientific computing community (Oliphant, 2007). Figure 2.6 shows a script and configuration file example that showcase the ease of using PEBL.

While many tasks related to Bayesian learning are embarrassingly parallel in theory, few software packages take advantage of it. PEBL can execute learning tasks in parallel over multiple processors or CPU cores, an Apple Xgrid<sup>1</sup>, an IPython cluster<sup>2</sup> or the Amazon EC2 platform<sup>3</sup>. The EC2 platform is especially attractive for scientists because it allows one to rent processing power on an on-demand basis and execute PEBL tasks on them.

PEBL can run in parallel not only built-in learners but also custom written ones. The source code for the custom learners need not be installed on every machine. PEBL will transparently package the source code and push it out to every machine executing a learning task. This feature along with the ease of writing custom learners means that researchers can easily code up new algorithms and run them on distributed platforms with minimal overhead.

With appropriate configuration settings and the use of parallel execution, PEBL can be used for large learning tasks. Although PEBL has been tested successfully with datasets with 10000 variables and samples, BN structure learning is a known NP-Hard problem (Chickering et al., 1994) and analysis using datasets with more than a few hundred variables is likely to result in poor results due to poor coverage of the search space.

### 2.7.3 HTML report

The results of a PEBL analysis are available as regular python objects, serialized to a text file that can be loaded in any python interpreter or as a HTML report. The HTML report is generated as a set of standalone files that do not require a webserver. The report includes three tabs that 1) summarize the approximated posterior and show some metadata about the learning tasks; 2) show the top 10 networks in the posterior; and 3) show the consensus network at different confidence thresholds. Screenshots of the three tabs in HTML report are shown in figures 2.7, 2.8 and 2.9 respectively.

---

<sup>1</sup>Grid computing solution by Apple, Inc. <http://www.apple.com/server/macosx/technology/xgrid.html>

<sup>2</sup>Cluster of Python interpreters. <http://ipython.scipy.org>

<sup>3</sup>An pay-per-use, on-demand computing platform by Amazon, Inc. <http://aws.amazon.com>



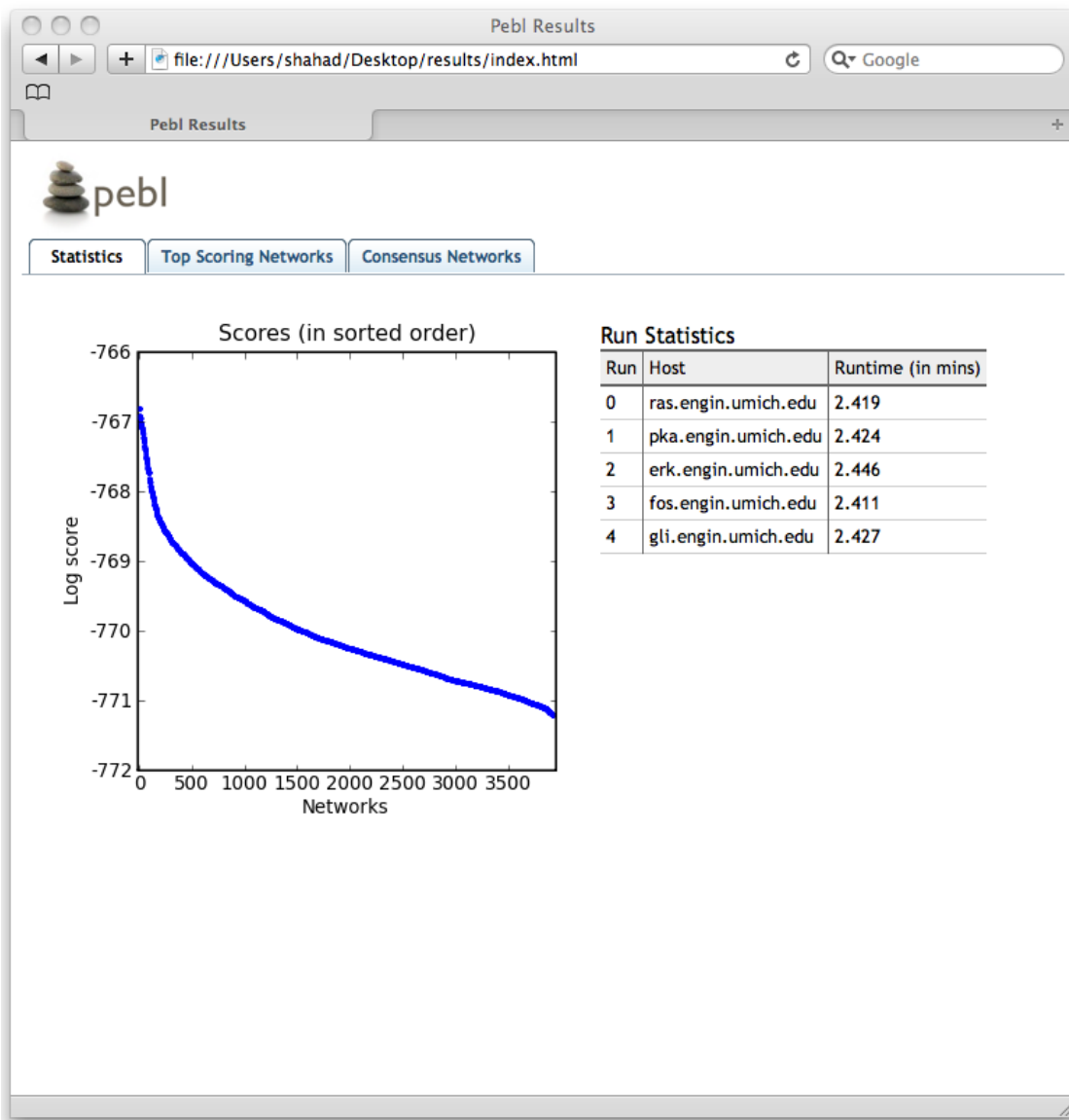


Figure 2.7: **HTML report: summary statistics.** This screenshot shows the summary statistics provided in the HTML report. The plot on the left shows the scores for the top networks found and the table on the right shows the machines where each task was executed and the runtime. This particular report is from a test run using the Apple Xgrid distributed computing platform. Note that each learning task was limited to few iterations and thus finished quickly. In an actual, distributed analysis, each run would last for hours.

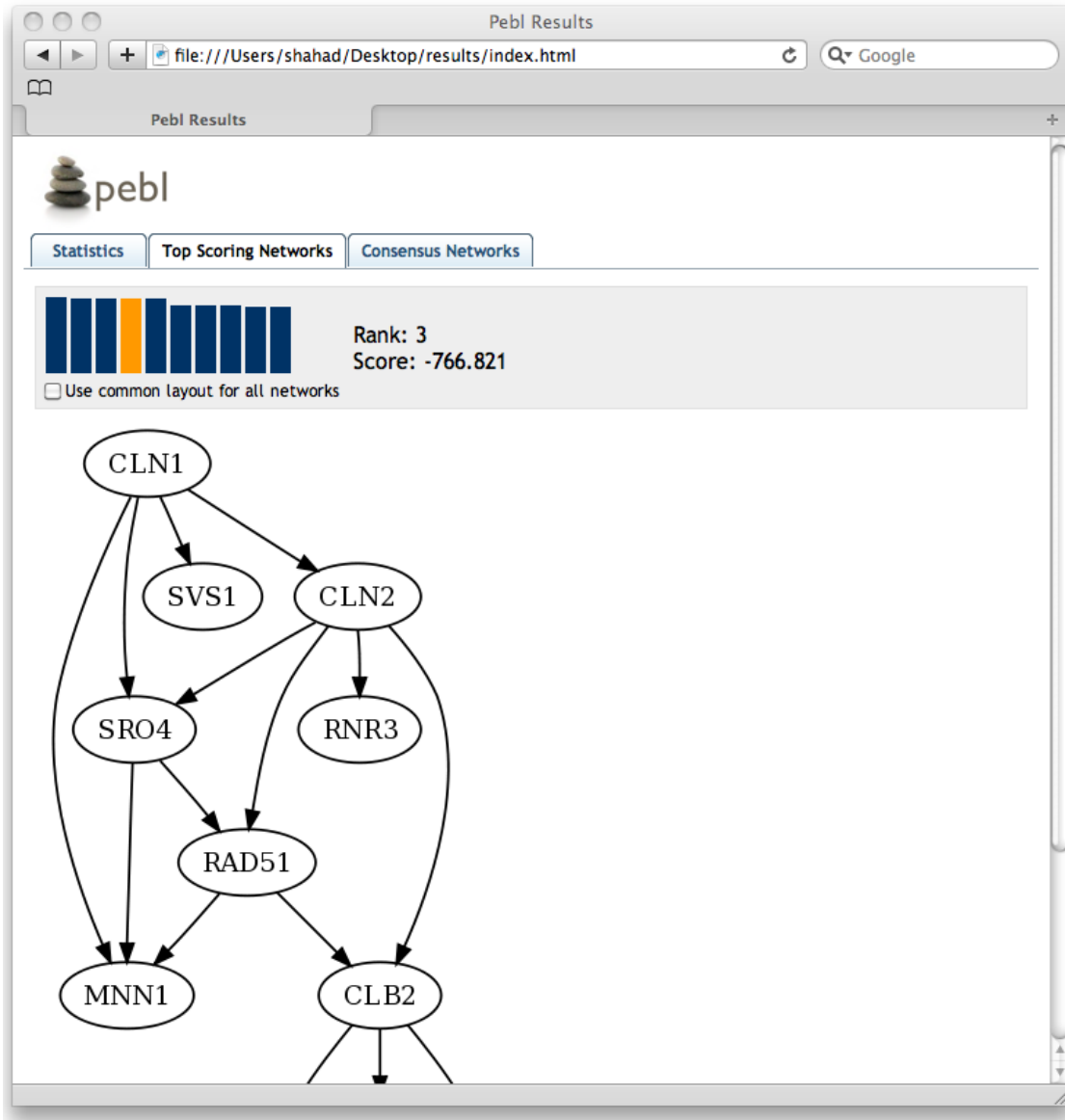


Figure 2.8: **HTML report: top scoring networks.** This screenshot shows a network viewer that lets one explore the top-scoring networks found during the analysis. The clickable bar graph shows the relative scores for the best networks and allows one to choose the network to view.

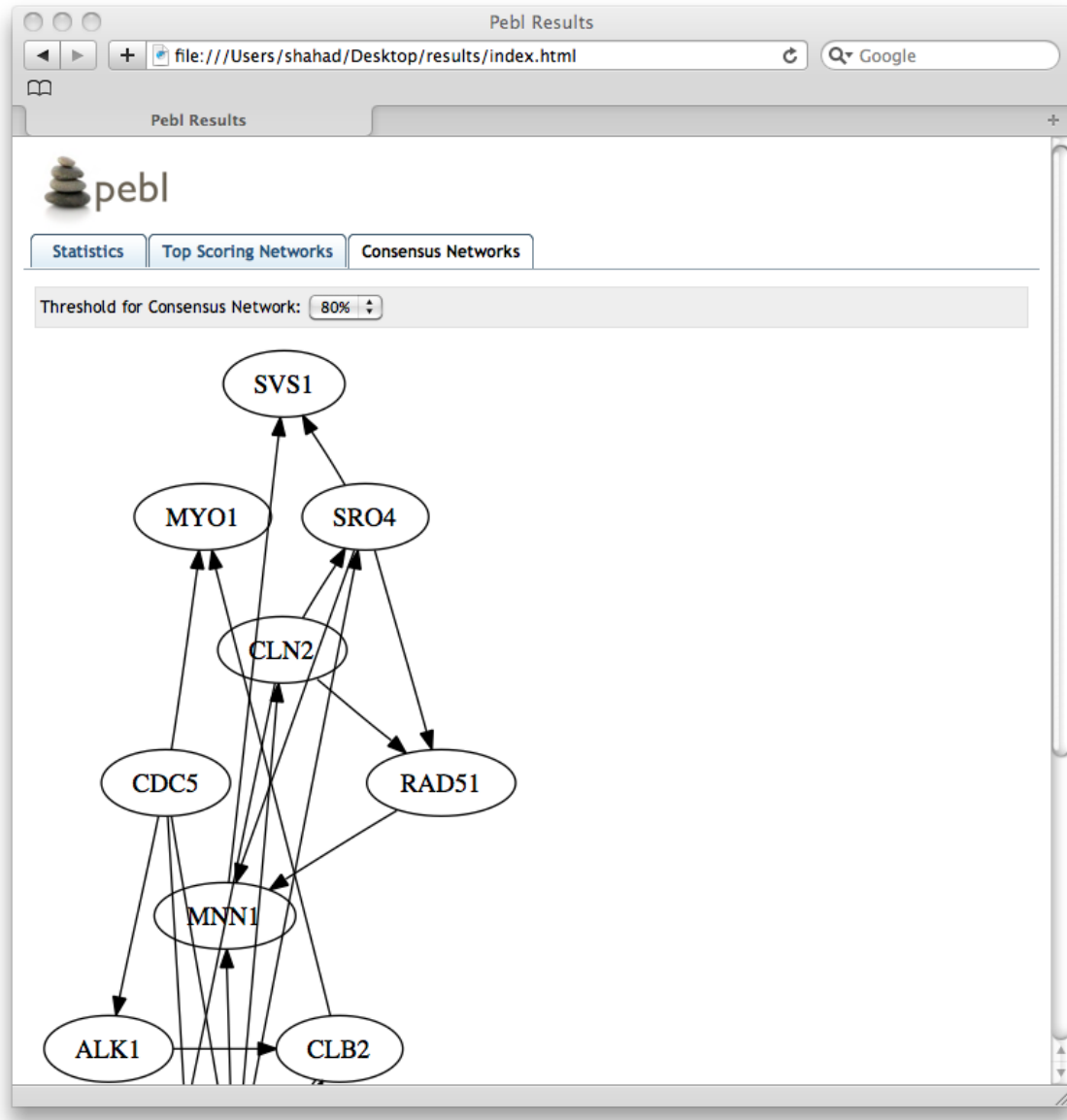


Figure 2.9: **HTML report: consensus networks.** This screenshot shows a network viewer that lets one view consensus networks at various confidence thresholds. A consensus network is created by including all edges with posterior probability above the given threshold.

	<b>BANJO</b>	<b>BNT</b>	<b>Causal Explorer</b>	<b>Deal</b>	<b>LibB</b>	<b>PEBL</b>
Latest Version	2.0.1	1.04	1.4	1.2-25	2.1	0.9.10
License	Academic <sup>1</sup>	GPL	Academic <sup>1</sup>	GPL	Academic <sup>1</sup>	MIT
Scripting Language	Matlab <sup>2</sup>	Matlab	Matlab	R	N/A	Python
Application	Yes	No	No	No	Yes	Yes
Interventional Data	No	Yes	No	No	No	Yes
DBN	Yes	Yes	No	No	No	No
Structural Priors	Yes <sup>3</sup>	No	No	No	No	Yes
Missing Data	No	Yes	No	No	Yes	Yes
Parallel Execution	No	No	No	No	No	Yes

<sup>1</sup> Custom academic, non-commercial license; not OSI approved.

<sup>2</sup> Via a Matlab-Java bridge.

<sup>3</sup> Only constraints/hard-priors supported.

Table 2.2: Comparing the features of popular Bayesian network structure learning software.

## 2.7.4 Extensive documentation

PEBL includes multiple forms of documentation including an installation guide, a tutorial with many code examples, API and configuration parameter references and a developer’s guide for programmers interested in extending PEBL.

## 2.7.5 Development process

The benefits of open source software derive not just from the freedoms afforded by the software license but also from the open and collaborative development model. PEBL’s source code repository and issue tracker are hosted at Google Code and freely available to all. Additionally, PEBL includes 216 automated unit tests and mandates that every source code submission and resolved error be accompanied with tests.

## 2.7.6 Related software

While there are many software tools for working with BN, most focus on parameter learning and inference rather than structure learning. Of the few tools for structure learning, few are open-source and none provide the set of features included in PEBL. As shown in Table 2.2, the ability to handle interventional data, model with missing values and hidden variables, use soft and arbitrary priors and exploit parallel platforms are unique to PEBL. PEBL, however, does not currently provide any features for inference or learning Dynamic Bayesian Networks (DBN). Despite its use of optimized matrix libraries and custom C extension modules, PEBL can be an order of magnitude or more slower than software written in Java

or C/C++; the ability to use a wider range of data and priors, the parallel processing features and the ease-of-use, however, should make it an attractive option for many users.

## 2.8 Conclusions

BN have many features that make them attractive for analyzing biological data. They can integrate generic knowledge and noisy data and can identify relationships between variables that are linear, nonlinear, multimodal and stochastic. In this thesis, I present Mechanistic Bayesian networks (MBN) as the use of BN in narrowly-formulated problems. This has allowed me to better represent existing knowledge using techniques like the use of hidden nodes that are computationally intractable for large structure learning problems.

I developed PEBL as a response to the requirements for MBN learning. Even in constrained problems, scoring BN models with hidden nodes requires the use of computational clusters. PEBL thus has support for running algorithms on multiple distributed computing platforms. Furthermore, PEBL is easy to use for non-expert programmers and complex analysis can be represented by simple scripts. As a result, PEBL is currently being used by various users in both bioinformatics and other fields. In the following two chapters, I demonstrate how the theory and software described in this chapter are used to solve difficult problems in relating the interactome and transcriptome.

# Chapter 3

## Identifying Pathway Targets

In chapter 1, I introduced Mechanistic Bayesian networks (MBN) as a way of using Bayesian networks (BN) in a non-structure-learning paradigm to solve targeted problems. By formulating specific problems, the space of possible models is severely constrained making the use of hidden nodes to represent unobserved proteins tractable even for non trivial problems. In chapter 2, I described how the use of multinomial conditional probability distributions (CPD) and Dirichlet parameter priors allow a BN to model linear, nonlinear, combinatorial and stochastic relationships without overfitting data. I also presented the Python Environment for Bayesian Learning (PEBL), a software library for MBN analysis. In this chapter, I apply MBN to the problem of identifying the downstream targets of a signaling pathway. I take advantage of PEBL's ability to score BN models with hidden nodes and to run analysis on distributing computing platforms.

In the following sections, I first describe the Sonic hedgehog (Shh) signaling pathway, the methods used to identify the downstream targets of Shh and then show results of the MBN analysis on an experimental data from a mouse model knockout study. I demonstrate the MBN's ability to identify known Shh targets that would be missed by other BN and non-BN methods.

### 3.1 Sonic hedgehog pathway

The Shh pathway plays a central role in organismal development and the progression of some cancers (Michaud and Yoder, 2006). Because of its central role, the Shh pathway is well studied providing us with an ideal test case to validate our MBN approach. The details of Shh are reviewed in detail elsewhere (McMahon, 2000; Ingham and McMahon, 2001; Rubin and de Sauvage, 2006), but here we will summarize the early steps of the pathway that we will use in this work. Shh is a secreted protein that acts as both a short-range contact-dependant factor and as a long-range diffusible morphogen. The Shh ligand

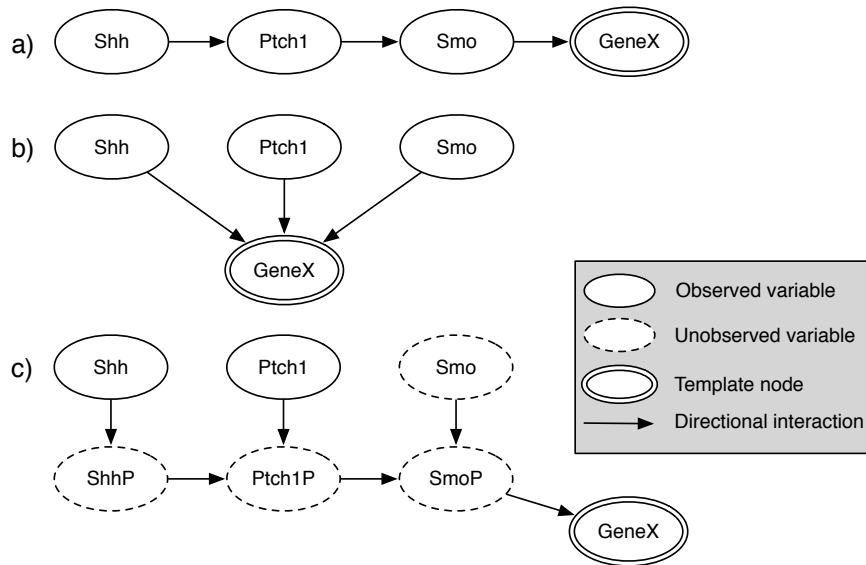


Figure 3.1: **Different representation of the Shh pathway.** (a) A sequential BN model of the pathway that omits protein activity, (b) a parallel BN model of the pathway, and (c) an MBN model of the pathway. The ovals Shh, Ptch1 and Smo represent mRNA measurements, while the ovals, ShhP, Ptch1P and SmoP, represent proteins. The oval GeneX indicates a candidate downstream expression target of the pathway. Dotted ovals represent entities that are known in the mechanism but are not observed, while unbroken ovals represent experimentally observed variables. Arrows between nodes represent a directional interaction but do not specify the functional form.

binds its canonical receptor Patched1 (Ptch1), which releases its inhibition of a second membrane-bound protein, Smoothed (Smo). Derepression of Smo in turn activates a signaling cascade inside the cell, eventually activating the Gli transcription factors and regulating the expression of a variety of genes. While some of the downstream targets of the pathway are known, many remain unknown. The initial steps of the pathway are shown in Figure 3.1(c) with the terminal node GeneX representing a putative downstream target. Note that in this cascade, the Shh, Ptch1, and Smo proteins can all directly or indirectly affect the target, although in different ways and to different degrees. Figure 3.1(a) and 3.1(b) show more abstracted representations of the pathway.

Due to its dose-dependant effect and role in development, the Shh signal requires strict spatio-temporal control. The pathway's known targets include pathway components, promoters and antagonists, thus regulating the effects of the Shh ligand over time. Cell surface proteins that promote Shh signaling, for example, are initially expressed in Shh-responsive cells, sensitizing cells to even low levels of the Shh ligand. As the level of Shh signaling

increases, downregulation of positive Shh components such as Gas1 and upregulation of negative components such as Ptch1, Ptch2, and Hhip that sequester the ligand ensure a tight control over Shh signaling (Allen et al., 2007). Additionally, Shh is known to cooperate with or antagonize other pathways such as Bmp (Ohkubo et al., 2002), retinoic acid (Riddle et al., 1993; Kondo et al., 2005), Wnt (Iwatsuki et al., 2007), Ras (Pasca di Magliano et al., 2006), and Notch (Wall et al., 2005) in a time-, dose-, and spatially-dependant manner. These factors further complicate the identification of pathway targets.

The Shh pathway raises a more general problem: given gene expression data from multiple samples, tissue and organs under different genetic knockout conditions, how can we identify downstream targets of a given pathway? Due to the interactions between different pathways and multiple cascades between ligand reception and eventual transcriptional regulation, it is not clear what constitutes a downstream target of a specific pathway. Common analysis approaches include significance testing between samples, differential expression, and clustering (Draghici, 2003). While these approaches are widely used and helpful, they fail to incorporate knowledge of the underlying pathway. When the pathway information is used, it is done while ignoring fundamental biological knowledge such as the central dogma. DNA, mRNA, and proteins are conflated into one variable and specific interactions such as protein-protein or protein-DNA are implicitly assumed to be detectable via gene expression data. To circumvent this problem, we present a novel mechanistic Bayesian network approach that more closely respects the meaning of both the pathway and the expression data.

## 3.2 Methods

In the following sections we describe the theoretical underpinnings of the MBN framework and provide a method for evaluating the significance of a result. Next we describe how we tested the MBN method using gene expression data from gene knockout mouse models to identify targets of the Shh pathway.

### 3.2.1 Mechanistic Bayesian networks: theory and definition

Mechanistic Bayesian networks represent a subset of a general class of analysis tools called Bayesian Networks (BN). A BN is a probabilistic graphical model that encodes dependencies between variables in a compact and descriptive manner. A BN can be represented as a



graph with nodes representing variables and edges representing dependencies or relationships between the variables. Mathematically, each node describes a conditional probability distribution (CPD) that quantitatively models the relationships between a node and its parent nodes. Note that an edge (an arrow) between two variables indicates a directed relationship but does not specify the functional form of the relationship. Said another way, an edge in a Bayesian network does not indicate activation, inhibition or any other specific function. This interpretation of an edge differs from the usual definition of an edge in a signaling pathway, where edges are often assumed to have a defined effect. The advantage of the more broad edge definition is that the relationships between nodes in a Bayesian network can be more complex and include functions that are nonlinear, multimodal, or logical.

BNs have been used successfully to model complex phenomenon in many fields and in systems biology, in particular, to model gene-regulatory networks, protein-interaction networks, signaling networks and to integrate heterogeneous biological data ([Djebbari and Quackenbush, 2008](#); [Friedman, 2004](#); [Friedman et al., 2000](#); [Jansen et al., 2003](#); [Woolf et al., 2005](#)). Methods for training and learning BNs are well established ([Cooper and Herskovits, 1992](#); [Heckerman, 1998](#)) and are available in a large number of software applications ([Korb and Nicholson, 2003](#)).

In contrast to the more general Bayesian network, a Mechanistic Bayesian Network (MBN) adheres more closely to known biological mechanisms by differentiating between mRNA transcripts and proteins, and by including pathway-based structural constraints. Because most studies do not measure both protein and gene expression, unknown quantities in an MBN are treated as unobserved, latent variables. By creating models that more closely resemble the known mechanism, MBN effectively incorporate in additional information that is not available in the experimental data alone.

### **3.2.2 Mechanistic Bayesian network templates**

When analyzing a biological system, users often want to generate and rank a set of entities that match certain constraints, such as all downstream targets of the Shh pathway or all proteins that participate in the crosstalk between the Shh and Wnt pathways. While it is possible to devise specific sets of constraints and methods for each case, we propose a generic MBN based template approach. Each set of constraints is expressed as an MBN template from which specific MBN are instantiated and evaluated against an experimental dataset. An MBN template is composed of two types of nodes: constant and template nodes.

A constant node is a regular node corresponding to either one variable in the dataset or an unobserved variable while a template node is a holding place for variables in the dataset. Each instantiation of the template is a regular MBN that can be scored using existing BN methods.

### 3.2.3 Maximum entropy discretization of the data

Before MBN can be applied to a dataset, the data must be discretized into a finite set of bins. While there are many ways to bin data, we use the Maximum Entropy principle to derive our discretization scheme. The principle states that the distribution that maximizes the information entropy is the true distribution given testable information (Jaynes, 1957). Accordingly, we bin our data such that each bin contains the same number of data points as this maximizes the entropy of the distribution. There is no theory prescribing the optimal number of bins for any given dataset or analytical method; increasing the number of bin decreases the information loss incurred during discretization but also increases the number of parameters and thus decreases the statistical power of the analysis given the same data samples. Most modelers working with systems biology data have arrived at three bins as a suitable compromise between information loss and statistical power (Djebbari and Quackenbush, 2008; Friedman et al., 2000; Yu et al., 2004)

### 3.2.4 Calculating the posterior probability of a model

Once the data are discretized, a particular model topology can be scored as the posterior probability of a model given data, given by:

$$P(M|D) = \frac{P(D|M)P(M)}{P(D)} \quad (3.1)$$

- $P(M|D)$  is the posterior probability of the BN or MBN model  $M$  given the data  $D$ . Said another way, it is our belief that the model  $M$  is correct after having observed the data  $D$ .
- $P(M)$  is the prior probability of the model, that is, our belief that the model  $M$  is correct before having observed the data  $D$ . The prior probability allows us to integrate model probabilities computed using other methods and data.
- $P(D|M)$  is the likelihood that the data was generated from the model  $M$ .  $P(D|M)$  is calculated as the likelihood after marginalizing over all model parameters given a network structure.
- $P(D)$  is a scaling factor that is usually ignored.

In the MBN framework, a Bayesian network is modeled using a multinomial representation with Dirichlet priors for the relationships between variables. This representation is convenient in that it is relatively agnostic to functional forms and has a convenient closed form solution, the Bayesian Dirichlet (BD) metric, described below.

### 3.2.5 BD scoring metric

The MBN approach uses the BD scoring metric (Cooper and Herskovits, 1992; Heckerman et al., 1995) to calculate the marginal likelihood of a dataset given a model. The BD metric is a closed form solution to the marginal likelihood for a multinomial Bayesian network model with Dirichlet priors. The BD metric is expressed as:

$$P(D|M) = \prod_{i=1}^n \prod_{j=1}^{q_i} \frac{\Gamma(\alpha_{ij})}{\Gamma(\alpha_{ij} + N_{ij})} \prod_{k=1}^{r_i} \frac{\Gamma(\alpha_{ijk} + N_{ijk})}{\Gamma(\alpha_{ijk})} \quad (3.2)$$

- $\Gamma(\bullet)$  is the Gamma function
- $n$  is the number of nodes
- $r_i$  is the arity of node  $i$
- $q_i$  is the arity of  $\pi_i$  (the parent set of node  $i$ )
- $N_{ij}$  is the number of samples where  $\pi_i$  is in configuration  $j$
- $N_{ijk}$  is the number of sampler where  $\pi_i$  is in configuration  $j$  and the node has value  $k$
- $\alpha_{ij}$  and  $\alpha_{ijk}$  are the prior counts corresponding to  $N_{ij}$  and  $N_{ijk}$  respectively.

A full derivation of the BD metric is described elsewhere (Heckerman et al., 1995).

In practice, the BD metric is represented in log-space for computational convenience. Because of this log transformation, all BD scores will be negative.

### 3.2.6 Calculating the marginal likelihood with missing data

If we have some missing values or latent variables in the dataset, we can use an altered method for computing the marginal likelihood. Because the likelihood is no longer fully factorizable into the product of the probabilities for each variable, we must calculate the marginal likelihood given every potential completion of the missing data. The simplest method is to marginalize over all possible sets of values for the missing data and take the average. However, this exact enumeration approach is impractical for non-trivial cases and so a heuristic must be used.

In this work we will use a modified Gibbs sampling ([Heckerman, 1998](#)) approach to approximate the marginal likelihood. We alter the method to only sample missing data completions that result in a maximum entropy discretization for all variables. This maximum entropy requirement ensures that  $p(x_i) = p(x_j)$  for all sets of variables and eliminates bias due to differential discretization for observed and hidden variables.

### **3.2.7 Calculating the marginal likelihood with interventions**

When calculating the marginal likelihood, data from interventions such as genetic knockouts are handled differently. If a particular data sample is the result of an intervention on a set of variables, the values for those variables no longer depend on their parent sets. They are, instead, arbitrarily set as the result of the intervention. Accordingly, the specific variable samples are ignored when constructing the multinomial table for any node that was intervened upon. ([Yoo et al., 2002](#))

### **3.2.8 Assessing model significance with bootstrapping based p-values**

To evaluate the significance of an MBN prediction, we calculate a p-value for each MBN result using nonparametric bootstrapping ([Efron and Tibshirani, 1994](#)). In this approach, we generate a large number of MBN models with template nodes replaced by randomly generated variables with the same discretization schemes as the observed variables. The scores for these MBN models are used as the null model against which MBN scores are compared to determine p-values. Due to the cost of scoring each bootstrap sample, we can only resolve p-values down to .001 and cannot correct for multiple hypothesis testing. We do note here that correction for multiple testing would lower the significance of each result.

### **3.2.9 Using MBN templates to define and identify shh targets**

To determine the downstream targets of the Shh pathway, we created an MBN template based on the canonical pathway as described in literature with a terminal downstream template node as shown in [Figure 3.1\(c\)](#). Specific MBN models are instantiated from the template by replacing the template node (the candidate target gene) with a specific gene from the dataset. Thus, each instantiated MBN model is a separate hypothesis that can be evaluated against the experimental data.

Developmental stage and location	wt	Shh <sup>-/-</sup>	Ptch1 <sup>-/-</sup>	Smo <sup>-/-</sup>
Embryonic day 8.5, whole embryo	3		3	3
Embryonic day 8.75, whole embryo	3		3	3
Embryonic dat 10.5, head	4	4		
Embryonic dat 10.5, limb bud	6	6		
Embryonic dat 10.5, trunk	3	3		

Table 3.1: **Shh knockout experimental design.** Experimental design for gathering gene expression data from a range of developmental stages, locations, and genetic backgrounds. Columns indicate the genetic background of the mouse model, rows indicate the developmental stage and location of the samples and the numbers within the table cell indicate the number of samples assayed.

### 3.2.10 Experimental data

To test our approach, we assembled data from three mouse models with gene knockouts in *Shh*, *Ptch1*, and *Smo* described in more detail elsewhere (Tenzen et al., 2006). Briefly, samples from different embryonic tissues at varying developmental stages were assayed using the U74v2 Affymetrix microarrays to determine the expression profile. Not all combinations of developmental stages and genetic backgrounds were available due to prenatal lethality for *Ptch1* and *Smo*. Samples assayed include 6-8- somite-stage (approximately 8.5 days post fertilization) with wildtype, *Smo*<sup>-/-</sup>, and *Ptch1*<sup>-/-</sup> backgrounds; 10-13 somite-stage (approximately 8.75 days post fertilization) with wildtype, *Smo*<sup>-/-</sup>, and *Ptch1*<sup>-/-</sup> backgrounds; 10.5 days post fertilization embryo samples from head, trunk, and limb bud with wildtype and *Shh*<sup>-/-</sup> backgrounds. The experimental design is summarized in Table 3.1.

### 3.2.11 Data preprocessing

The raw gene expression data (.CEL files) were processed using RMA (Irizarry et al., 2003) as implemented in the Bioconductor Affy package (Gautier et al., 2004). The data were annotated using updated probeset definitions (.CDF files) provided by the Microarray Lab at the University of Michigan (Dai et al., 2005).

Data were discretized into 3 bins using the Maximum Entropy discretization scheme described above. Because the data contain samples from varying developmental stages and genes with significant expression variation between stages, genes were discretized separately for each tissue type. This discretization scheme allowed the software to more easily identify relevant patterns within a tissue type. Gene were knocked out by altering the sequence to

create inactive protein. Thus, for a knockout on a particular gene, the gene's expression was left unaltered but the protein expression was set to 0 and the value was marked as an intervention so it could be handled differently by the scoring procedure.

### 3.2.12 MBN template analysis

To test the MBN template system using the Shh dataset, we developed a template that matched the topology shown in Figure 3.1(c). In this template, we search for a downstream target node, GeneX, that responds to the Shh signaling cascade. In this search, we assume that each gene has an equal prior probability of being the target node, and all relationships are scored using the BD score and a Gibbs sampler, described above. Using this template, we generated 6299 candidate MBN models – one for each candidate target gene that showed significant differential expression. Due to the custom CDF annotation, Smo was not present on the chip, thus we modeled the Smo mRNA expression as a hidden node. No protein expression was observed in this experiment, so nodes representing protein concentrations or activities were also modeled as hidden. Rather than selecting the number of iterations for the Gibbs sampler a priori, we calculated the posterior distribution  $P(M|D)$  with 150,000 iterations, then with 300,000 iterations. Although the sampler had largely converged at this point as evidenced by the observation that the top 25 results did not change significantly, we resumed the Gibbs's sampler for the 300,000 run and sampled another 300,000 iterations for a total of 600,000 iterations per MBN model. The computation took approximately 5.6 minutes per MBN for a total of approximately 12 hours on a 48-node compute grid. P-values were calculated with bootstrapping describe above using 1054 MBN models to represent the null distribution. All code and data required to replicate the analysis are included in the supplemental material. The code can be easily modified to run similar analysis on different dataset using different MBN templates.

In addition, we also tested two alternative Bayesian approaches shown in Figure 3.1(a) and 3.1(b). These approaches use topologies that are simpler than the MBN approach in Figure 3.1(c), but less closely adhere to the known mechanism of the biochemistry of the pathway.

### 3.3 Results and discussion

In the following sections we show the results from the MBN analysis of the Shh knockout dataset and discuss the biological plausibility for each finding. Next we discuss expected hedgehog pathway target genes that were not identified by the MBN and a comparison with other techniques.

Gene	Bayesian Score	P-value	Shh Correlation	Ptch1 Correlation	Citations
Gas1 Growth arrest specific 1	-226.34	< .001	-0.016	-0.57	(Allen et al., 2007)
Gli1 GLI-Kruppel family member	-226.69	< .001	0.61	0.74	(Sheng et al., 2006)
Ptch2 Patched homolog 2	-229.17	< .001	0.54	0.69	(Motoyama et al., 1998)
Gtpbp4 GTP binding protein 4	-229.44	< .001	-0.28	0.35	
Mig12 Mid1 interacting protein 1	-229.61	< .001	0.2	0.61	
Msx1 Homeo box, msh-like 1	-229.87	< .001	0.03	-0.56	(Bok et al., 2007)
Crabp2 Cellular retinoic acid binding protein II	-229.97	< .001	-0.05	-0.46	
Has2 Hyaluronan synthase 2	-231.03	< .001	-0.08	0.46	
Foxd1 Forkhead box D1	-231.61	< .001	0.29	0.43	(Jeong et al., 2004)
Ak1 Adenylate kinase 1	-232.22	< .001	0.16	-0.01	

Table 3.2: **Results from the MBN analysis (model in Figure 3.1(c))** This table shows the top 10 predicted downstream targets from the MBN analysis. The columns specify the gene name and a short description; the Bayesian score as described in Methods; the resulting p-value for the Bayesian score calculated using a bootstrap method as described in Methods; a linear correlation to the Shh transcript; a linear correlation to the Ptch1 transcript; and a literature citation, if available, that specifies that the gene is indeed a true downstream target of the Shh pathway.

The predicted target genes from the MBN analysis are shown in Table 3.2. The top scoring hits include many known Shh targets as described below. The top hit, Gas1, has been shown to be a negative target of Shh signaling whereas knockout and subsequent gain-of-function studies have shown Gas1 to be a positive component of the Shh pathway

that acts synergistically with Ptch1 to bind the hedgehog ligand (Allen et al., 2007). The second target, Gli1, is one of the three Gli transcription factors that regulate Shh target genes and is often used as a canonical readout of Shh activity. Shh signaling is known to both induce Gli1 expression and also regulate its nuclear accumulation and therefore its activity (Sheng et al., 2006). Ptch2 is a homolog of Ptch1 but is expressed in different cells. Like Ptch1, Ptch2 is known to be transcriptionally modulated by Shh signaling (Motoyama et al., 1998). Knockout studies have shown that Shh signaling is required for expression of Msx1, a transcriptional repressor with a putative role in limb-pattern formation (Bok et al., 2007). Foxc2 and Foxd1 have been shown to be upregulated by Shh signaling in null, conditional, and constitutively active Smo mutant backgrounds (Jeong et al., 2004).

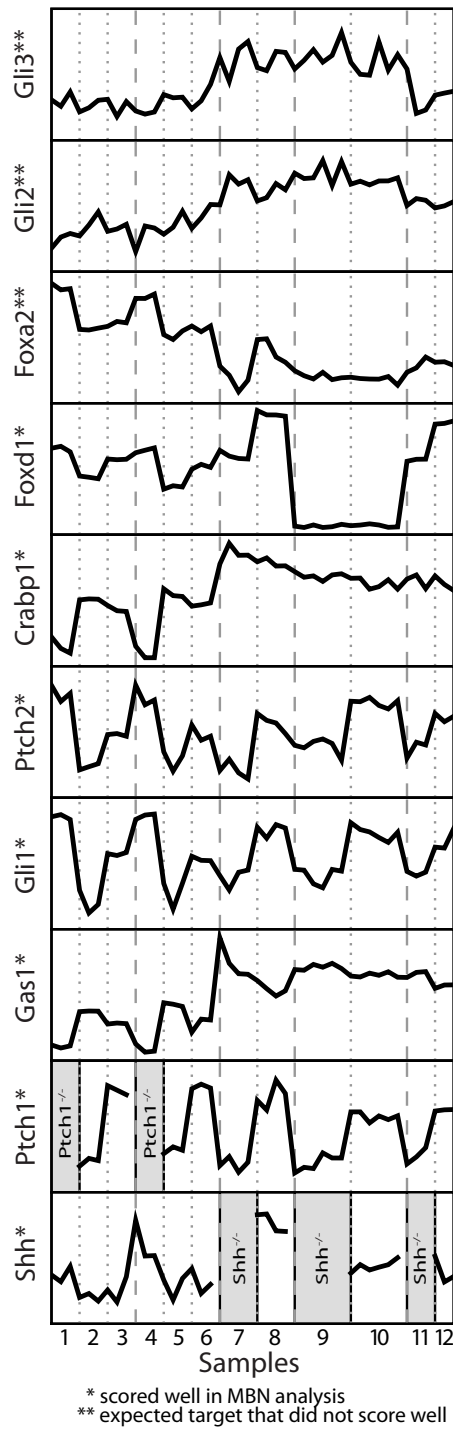
Beyond these known Shh-related genes, some of the highly ranked results appear to be novel putative Shh targets. Mig12 binds the Opitz syndrome gene Mid1 and the complex is thought to stabilize microtubules (Berti et al., 2004). Mid1 is repressed by Shh (Granata and Quaderi, 2003) and this is partially borne out by our gene expression results which show a positive influence from Shh and a negative influence from Ptch1. Interestingly though, our analysis shows Mid1 to be a poor target. Mig12 is not a known Shh target but is a significant result in our analysis and appears to be upregulated by the Shh pathway.

Both cellular retinoic acid binding proteins, Crabp1 and Crabp2, score well in our analysis but are not known Shh targets. In some developmental processes, retinoic acid (RA) has been shown to work synergistically with Shh to create the zone of polarizing activity, (Riddle et al., 1993) for example, and in adult pluripotent stem cells (Kondo et al., 2005). Crabp binds RA with high affinity and is thought to spatially modulate the effect of RA by adjusting the concentration of RA reaching the nucleus (Maden et al., 1988). One interpretation of the MBN result is that that Shh modulates the effects of RA by regulating Crabp.

### **3.3.1 Known Shh targets not identified by MBN**

Our results, however, do fail to identify some known and expected targets. For example, one might expect the genes Foxa2, Gli2, Gli3, Ihh and Dhh to be identified by the MBN as these are often listed as targets of the Shh pathway, but in our analysis they did not score near the top. The MBN template used in this analysis (Figure 3.1) detects relationships between a candidate target and Shh and Ptch1, constrained by the information bottleneck introduced by the hidden protein nodes. When we examine the relationship between the





**Sample groups:**

- |   |                                     |
|---|-------------------------------------|
| 1. X8 somite, Ptch1 <sup>-/-</sup> (3)  | 7. Head, Shh <sup>-/-</sup> (4)     |
| 2. X8 somite, Smo <sup>-/-</sup> (3)    | 8. Head, wt (4)                     |
| 3. X8 somite, wt (3)                    | 9. Limb bud, Shh <sup>-/-</sup> (6) |
| 4. X13 somite, Ptch1 <sup>-/-</sup> (3) | 10. Limb bud, wt (6)                |
| 5. X13 somite, Smo <sup>-/-</sup> (3)   | 11. Trunk, Shh <sup>-/-</sup> (3)   |
| 6. X13 somite, wt (3)                   | 12. Trunk, wt (3)                   |

Figure 3.2: From these expression patterns one can see that the profiles for Gas1 and Gli1 closely follow changes in Shh and Ptch1 expression. Gli2 and Gli3 show higher activation in the adult tissues as compared to the somite samples but, even in the adult tissues, their expression pattern does not correspond to the knockout state of Shh. Foxa2 expression shows a strong response to the Ptch1 knockouts in the somite samples and with Shh knockouts in the adult head tissues but there is no pattern in the adult limb and trunk tissues. While the patterns in Gli2, Gli3, Foxa2 are significant, the patterns either do not coincide with patterns in the early steps of the Shh pathway (Shh, Ptch1, Smo) indicating that there may be other genes involved in their regulatory control or the patterns do not hold over all tissues. Accordingly, those genes do not score highly in the MBN method.

expected but poorly scoring genes and Shh and Ptch1 (Figure 3.2), we find no consistent pattern of differential expression. It is clear that the expression does not change significantly or consistently with the pattern of change for Shh or Ptch1. Genes identified as targets by the MBN such as Gas1, Gli1, Ptch2, Crabp1 and Foxd1, show clear relationships that can be assessed by eye.

### 3.3.2 Comparison with non-Bayesian bioinformatics techniques

The results we obtain using MBN are different than those from standard techniques for determining downstream pathway targets. Whereas clustering, differential expression and significance testing are, in general, unable to identify nonlinear and multimodal relationships, Bayesian methods such as MBN do not have this limitation. Bayesian methods learn based on complex patterns of change rather than the magnitude of change or a simple linear correlation.

Gene	Description	Bayesian Score	Citations
Ak1	Adenylate kinase 1	-80.3	
Ubc	Ubiquitin C	-81.84	
Pdcd4	Programmed cell death 4	-81.9	
Nckap1	NCK-associated protein 1	-81.9	
Crabp2	Cellular retinoic acid binding protein II	-82.07	
Ntn1	Netrin 1	-82.6	( <a href="#">Dakubo et al., 2008</a> )
Oprs1	Opioid receptor, sigma 1	-82.7	
Fgf8	Fibroblast growth factor 8	-82.84	( <a href="#">Ohkubo et al., 2002</a> )
Nme6	Expressed in non-metastatic cells 6, protein	-83.1	
Gtf3c5	General transcription factor IIIC, polypeptide 5	-83.1	

Table 3.3: **Results from the sequential BN analysis (model in Figure 3.1(a)).** This table shows the top 10 predicted downstream targets from the sequential BN analysis. The columns specify the gene name and short description; the Bayesian score as described in Methods; and a literature citation, if available, that specifies that the gene is indeed a true downstream target of the Shh pathway.

---

When we use differential expression as an indicator of significance, for example, we find none of the canonical targets near the top of the list (full list available in the supplemental materials). Gli1, the commonly used indicator of Shh activity, ranks 629th in terms of fold-change, but 2nd with respect to MBN analysis. This difference is expected, however, because only small changes in gene expression are required to modulate the activities of transcription factors and pathway components that participate in feedback loops with the

Gene	Description	Bayesian Score	Citations
Foxc2	Forkhead box C2	-115.16	(Jeong et al., 2004)
Gli1	GLI-Kruppel family member GLI1	-116.19	(Sheng et al., 2006)
Ntn1	Netrin 1	-117.24	(Dakubo et al., 2008)
Ptch2	Patched homolog 2	-117.24	(Motoyama et al., 1998)
Gspt1	G1 to S phase transition 1	-117.39	
Mid1ip1	Mid1 interacting protein 1	-117.39	
B4galt3	UDP-Gal:betaGlcNAc beta 1,4-galactosyltransferase, polypeptide 3	-117.51	
Cklfsf8	Chemokine-like factor super family 8	-118.05	
Ptprz1	Protein tyrosine phosphatase, receptor type Z, polypeptide 1	-118.2	
Dctn3	Dynactin 3	-118.42	

Table 3.4: **Results from the parallel BN analysis (model in Figure 3.1(b))** This table shows the top 10 predicted downstream targets from the sequential BN analysis. The columns specify the gene name and short description; the Bayesian score as described in Methods; and a literature citation, if available, that specifies that the gene is indeed a true downstream target of the Shh pathway.

Shh ligand. We obtain slightly better but still poor results when identifying targets based on a linear correlation to either Shh or Ptch1 gene expression. This difficulty with clustering can be seen in Table 3.2. Clustering methods suffer from similar failures as linear correlation and fail to take into account known information about the initial steps of the pathway.

### 3.3.3 Comparison with standard BN modeling of the Shh pathway

To isolate the effect of the MBN template over more standard BN approaches, we ran a direct comparison between the approaches using the three topologies shown in Figure 3.1. The simplest translation of the Shh pathway, shown in Figure 3.1(a), is to interpret the qualitative diagram as a BN and replace all arrows with a Bayesian edge to produce a sequential model. Although the topology looks most similar to a biochemical diagram, for a Bayesian network the topology indicates that only Smo influences the target gene. An alternative topology is shown in Figure 3.1(b). In this parallel topology, the target gene is influenced by all of the genes in the pathway. However, this topology does not capture the unobserved proteomic effects, nor does the topology distinguish between the relative position of Shh, Ptch, and Smo in the signaling pathway.

The 10 best-scoring results from the sequential BN, parallel BN and MBN models are

shown in tables 3.3, 3.4 and 3.2, respectively. When we compare the results from the three topologies shown in Figure 3.1, we find that the target genes predicted by the MBN topology (Figure 3.1(c)) best match what is known about the pathway. If we compare the number of known targets identified in the top 10 targets predicted by each method we find the MBN predicts 5/10, the parallel topology predicts 4/10, and the sequential topology predicts 2/10. In the MBN results, Gas1, Gli1 and Ptch2 rank 1st, 2nd, and 3rd respectively and are all well-known Shh targets, while in the parallel model these genes rank 11th, 2nd and 3rd respectively, and in the sequential model these genes rank 24th, 20th and 21st respectively. One reason for this discrepancy is that the BN models impose fewer constraints on the candidate gene which would produce more false positives – a result in line with what we see. The full lists of target genes for each topology is provided in the supplementary materials.

### 3.3.4 Extension to larger MBN templates

While we used a small template pathway in this work as a proof of concept, the method can be applied to larger pathways in a similar way. However, large numbers of hidden nodes increase the computational requirements and may make the analysis intractable using the Gibbs sampler used in this work. The MBN template algorithm’s runtime complexity is  $O(nm^2h^2)$  where  $n$  is the number of candidate genes,  $m$  is the number of data samples and  $h$  is the number of hidden or latent variables. Because the limiting step is the Gibbs sampling’s  $O(m^2h^2)$  runtime complexity, we could use a more complex but efficient heuristic sampling method such as variational learning (Beal and Ghahramani, 2006). As currently implemented, MBN templates should be limited to models with a small number of hidden nodes.

### 3.3.5 Extension of MBN to other data types

Although this study focused on gene expression data, the MBN template method can be used with any type of observed data that can be represented in the pathway. If a molecular entity can be represented in an MBN template while being faithful to the underlying biochemical mechanisms, data regarding the molecule’s concentration or activity can be used. Possible other measurements that could be used with a similar MBN approach include protein expression, kinase activity, and miRNA expression. Furthermore, MBN templates can be used to integrate observations for multiple data types assuming that the measurements are made on a common set of samples.

## 3.4 Conclusion

We have shown how MBN can be used to integrate gene expression data and known topological information to uncover mechanistic details about complex pathways. Although we have shown only one example with finding downstream targets of the Shh pathway, a similar approach could be used on other pathway topologies, data types, and to identify targets at different locations in the pathway. In this example, the MBN method provided better target predictions than other methods, due in part because the topology assumed by the MBN is closer to the known biochemical mechanism.

In chapter 4, I use MBN in a problem complementary to the one discussed in this chapter: given observed transcriptional activity, I identify the subset of the protein interactome that best explains it. Due to the size of the interactome, the use of hidden nodes is no longer tractable; I thus choose a different network architecture to make the models as mechanistically realistic as possible.

# Chapter 4

## Determining Context-specific Subnetworks

In chapter 1, I introduced Mechanistic Bayesian networks (MBN) as a way of using Bayesian networks (BN) in a non-structure-learning paradigm to solve targeted problems. In chapter 2, I described how the use of multinomial conditional probability distributions (CPD) and Dirichlet parameter priors allow a BN to model linear, nonlinear, combinatorial and stochastic relationships without overfitting data. I also presented the Python Environment for Bayesian Learning (PEBL), a software library for MBN analysis. In this chapter, I apply MBN to the problem of identifying the subset of the protein interactome that best explain observed transcriptional activity during a complex biological process. In chapter 3, the use of hidden nodes to represent unobserved proteins allowed me to better encode existing knowledge. Accordingly, the results were more biologically meaningful and had better agreement with previously described results. The large size of the protein interactome makes the use of hidden nodes unfeasible for this problem. I use a different approach to model existing knowledge while being faithful to known mechanisms. By testing for statistical dependence along the flow of information rather than between precursors of interacting proteins, I derive a scoring metric that identifies condition-specific protein interactions.

### 4.1 Introduction

Many biological phenomena are driven by interactions among large numbers of cellular entities working in concert. Mapping these interactions is critical to defining the processes that govern cellular processes. Typically, two classes of molecular data are used to map a biological network: experimental observation of concentrations and states, and prior knowledge about the interactions involved. Experimental data, when used alone, ignore what is known about the interactions and processes involved. On the other hand, prior knowledge in bioinformatics databases is aggregated from heterogeneous sources representing a wide variety of experimental and physiological conditions; not all interactions are relevant to any specific condition or process. We and others have suggested that integrating both

experimental and prior data should yield more relevant and interpretable biological maps (Campanaro et al., 2007; Ideker et al., 2002; Liu et al., 2007; Shah et al., 2009; Wachi et al., 2005).

In this work, we introduce Netshadow, a Bayesian method for integrating prior mechanistic knowledge networks with experimental observations to identify parts of the global interactome relevant to a specific biological condition or process. We demonstrate the method on both a synthetic dataset and an experimental dataset from a TGF- $\beta$ -induced epithelial-to-mesenchymal transition (EMT) study. With the EMT dataset, we integrate a literature-extracted protein-protein interaction (PPI) network with gene expression data to yield a weighted subnetwork that is mechanistically consistent with both known interactions and the observed gene expression patterns.

Several studies have previously generated condition-specific subnetworks. Campanaro et al. and Wachi et al. generated subnetworks from proteins that exhibited differential gene expression beyond a threshold (Campanaro et al., 2007; Wachi et al., 2005). Ideker et al. and Liu et al. devised algorithms for identifying active subnetworks and high-scoring networks, respectively. Both methods identified connected regions of the global interactome that collectively exhibit significant differential gene expression for a particular condition even when not every protein in the subnetwork was differentially expressed (Liu et al., 2007; Ideker et al., 2002). Ulitsky et al. introduced an algorithm for identifying jointly connected active subnetworks, that is, subnetworks that exhibit high average internal similarity (Ulitsky and Shamir, 2007).

The prior works described above have relied on one of two assumptions: 1) that proteins that are important to a condition show significant differential mRNA expression and/or 2) that proteins that interact are co-regulated and thus have correlated gene expression profiles. Although both assumptions have led to useful algorithms, they are problematic for the following reasons. First, using mRNA abundance to quantify protein activity ignores the effects of microRNA, mRNA stability, and post-translational regulation; indeed, empirical studies have found poor correlation between mRNA and protein abundance (Gygi et al., 1999; Griffin et al., 2002; Anderson and Seilhamer, 1997; Chen et al., 2002; Tian et al., 2004).

Second, proteins that interact often don't have correlated gene expression. Although some studies have identified gene expression correlation between interacting proteins (Hahn et al.,

2005), Bhardwaj et al. (Bhardwaj and Lu, 2005) found that gene expression of interacting proteins is correlated in E. coli but not in yeast, mouse or human. Jansen et al. (Jansen et al., 2002) found gene expression correlation between interacting proteins in yeast for a few permanent complexes but not in transient interactions or for the global interaction network in general. Furthermore, a recent study by Xulvi-Brunet et al. not only confirmed the lack of correlation but also observed that co-expression and PPI networks have different structural properties such as degree distribution, network diameter and shortest path length (Xulvi-Brunet and Li, 2010).

Departing from the differential expression and co-expression assumptions, Netshadow uses a mechanistically-inspired approach instead. Netshadow identifies physiologically relevant physical interactions by the effect of that interaction on the expression of other genes. To identify relevant interactions, we first create a set of differentially expressed genes, called the target gene set, to represent the activity observed in a biological condition or process. Next, we use the target gene set as a bait to quantify the relevance of each edge of a global interaction network to the biological condition with a Bayesian model in which the interactors indirectly regulate the target genes. Finally, we generate the condition-specific weighted subnetwork by pruning insignificant edges.

As an example, consider the hypothetical scenario in Figure 4.1 that is used in the simulated study presented in the results. In the hypothetical cellular process, proteins A and B form a complex that regulates genes e, f and g whereas proteins C and D also form a complex but have no downstream effect. From an experimental study, we know that the process is characterized by differential expression of genes e, f and g but we have no measurements regarding protein interactions or complexes. From a global interactome database, we know that interactions A–B and C–D have been previously reported but don't know whether they are physiologically relevant in our biological context. Using Netshadow, we create a target gene set with genes e, f and g and score models in which A–B and C–D affect the target genes. Note that the specific mechanics underlying the direct or indirect regulation do not matter as long as the interaction leads to the eventual regulation of some genes. For example, rather than forming a complex with B, A could have activated B by a post-translation modification such as phosphorylation and the activated B could initiate a signaling cascade that eventually regulates genes e, f and g. Also note that, in our model, the relationship between the genes need not be a linear or pair-wise correlation. To our knowledge, we were the first to use such a mechanistically-inspired method to identify interactions relevant to a biological condition or process.



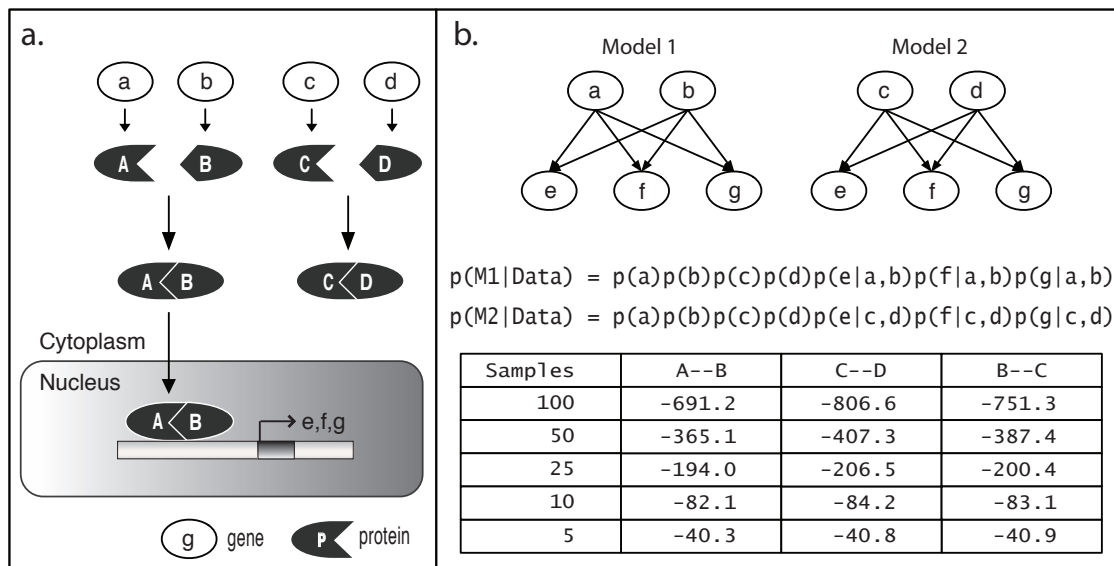


Figure 4.1: **Netshadow example with synthetic data.** The figure shows the hypothetical scenario that is used for the simulated study presented in Results. The left panel depicts a pathway in which proteins A and B form a complex that regulates genes e, f, and g whereas proteins C and D also form a complex but have no downstream effects. Synthetic gene expression data are generated from the pathway with nonlinear relationships and added noise (described in Methods). The right panel shows two bipartite BN models corresponding to the interactions A–B and C–D and the interactions scores with different number of data samples. The true interaction, A–B, outscores the rest with every data sample size.

## 4.2 Methods

In this section, we describe the theoretical model used in our method, derive the interaction scoring metric, and describe a bootstrap-based approach for assessing significance. Next, we describe the python-language implementation of this method. Finally, we present the synthetic and experimental datasets and the pre-processing steps used with the data.

### 4.2.1 Theoretical model: bipartite MBN model

We are interested in scoring interactions based on how well they explain the gene expression activity observed in our experimental dataset. We treat each candidate interaction as a hypothesis and assess how well the experimental data support that hypothesis. Each interaction is modeled by a bipartite Bayesian network with edges from each interactor to each target gene, as shown in Figure 4.1(b). The model depicts the direct or indirect regulation of the target genes by the product of interaction. While BN are described in detail elsewhere

(Heckerman, 1998), we note some salient points here. Edges in a BN encode conditional independence relationships: the child node is independent of all variables except its parent nodes. Knowing the values for the parent nodes decreases the uncertainty about the child node and thus the parent nodes are said to explain the child nodes. It is important to note that edges in a BN do not encode any specific functional form; they don't specify activation or inactivation, rather, just a probabilistic relationship.

#### **4.2.2 Theoretical model: interaction score**

The score for an interaction is the posterior probability of the underlying BN model, which is given by the Bayes rule  $P(M|D) = P(D|M)P(M)/P(D)$ .  $P(D|M)$  is the likelihood of the data given the model after marginalizing over all model parameters; it is calculated using the BD scoring metric (Heckerman, 1998). The likelihood quantifies how well the data agree with the hypothesis encoded by the BN.  $P(M)$  is the prior belief in the model; it can be used to integrate interaction confidence calculated using other methods and datasets.  $P(D)$ , the probability of the data, is calculated by marginalizing over all models and serves as a scaling term to ensure that  $P(M|D)$  is a true probability distribution.

#### **4.2.3 Theoretical model: assessing significance by bootstrapping**

A common challenge for many machine-learning methods applied to biological data is over-fitting. Typical high-throughput datasets contain few samples but a large number of variables; microarrays assay tens of thousands of genes but studies typically contain less than a hundred samples. The small sample size does not normally support statistically significant conclusions. To overcome this problem, we use a non-parametric bootstrapping procedure (Efron and Tibshirani, 1994) to assess the significance of each interaction. We first sample interaction scores, with replacement, from the list of all scored interactions. Next, rather than assume a specific distribution for the scores, we calculate the p-value for a given interaction by counting the number of bootstrap samples with score greater than that for the current interaction. We repeat this 10,000 times to calculate the average p-value.

#### **4.2.4 Theoretical model: target and downstream gene sets**

The target gene set includes the top differentially expressed genes in the dataset. Not every gene in the set, however, will be downstream of any specific interaction. We use a simplified

version of the Netshadow model to identify true downstream genes: each potential downstream gene from the target set is scored using a BN with edges from each interactor to the gene. A bootstrap method is used to calculate the p-value and genes with  $p < .05$  are reported as downstream for the interaction.

#### 4.2.5 The netshadow pipeline

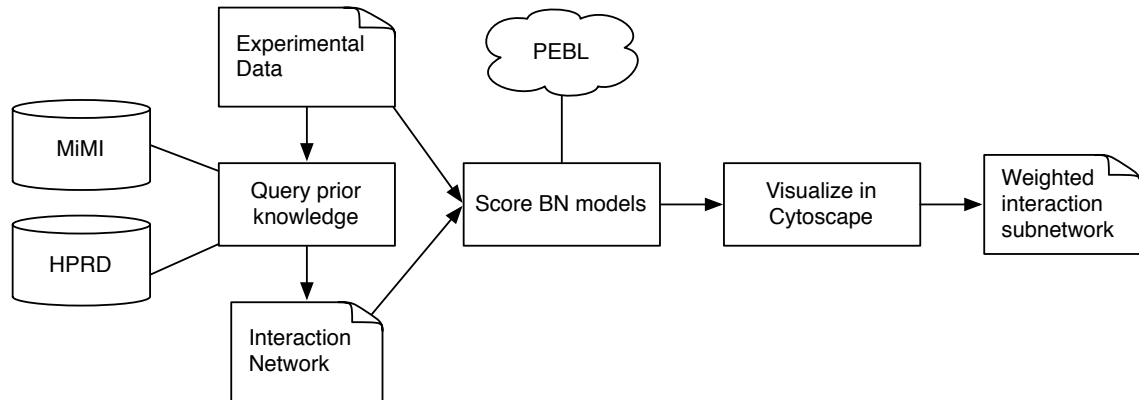


Figure 4.2: **Netshadow analysis pipeline.** The input to Netshadow is the experimental gene expression dataset. First, all genes in the dataset are used to query multiple sources of prior knowledge to generate the prior interaction network. Next, the top differentially expressed genes are used to create the target gene set and all edges from the prior network are scored to generate the Netshadow weighted subnetwork. Finally, the subnetwork is exported to Cytoscape for visualization.

We have implemented the Netshadow method as a python-language application. We first give a brief overview of the analysis pipeline (Figure 4.2) and then describe each part in detail. The input to Netshadow is the experimental gene expression dataset. First, all genes in the dataset are used to query multiple sources of prior knowledge to generate the prior interaction network. Next, the top differentially expressed genes are used to create the target gene set and all edges from the prior network are scored to generate the Netshadow weighted subnetwork. Finally, the subnetwork is exported for visualization and further analysis.

#### 4.2.6 Netshadow: querying prior knowledgebases

Netshadow can query and use results from multiple sources of prior knowledge. The Human Protein Resource Database (HPRD) is a manually-curated database of PPI extracted from literature (Keshava Prasad et al., 2009). When using HPRD, Netshadow uses Release 8, which contains 36,633 PPI (excluding homo-dimers) among 7,773 proteins. The Michigan

Molecular Interactions (MiMI) database provides PPI and pathway information integrated from multiple sources with complete provenance information (Tarcea et al., 2009). When using MiMI, Netshadow uses the r2 version with 154,850 PPI (excluding homo-dimers) among 10,806 proteins.

#### **4.2.7 Netshadow: scoring models**

Netshadow first discretizes data into 3 bins using a maximum-entropy scheme as described elsewhere (Shah et al., 2009). It uses our open-source Bayesian network software PEBL (Shah and Woolf, 2009) for scoring BN models and takes advantage of several parallel processing and cloud platforms.

#### **4.2.8 Netshadow: weighted subnetwork**

The interaction network generated by querying prior knowledgebases is pruned and then weighted by the interaction score for each edge. The interaction score (described in detail above) is the posterior probability of a bipartite Bayesian network modeling the direct or indirect regulation of the set of target genes by the interactorbootstrapping approach described above. Edges with  $p > .01$  are pruned and the remaining edges are weighted by their interaction score. The network is visualized with edge thickness and color proportional to its interaction score and node size proportional to its degree (number of edges connected to it).

#### **4.2.9 Netshadow: results visualization in cytoscape**

Cytoscape is an application for analyzing and visualizing networks with an emphasis on biological data (Shannon et al., 2003). Netshadow generates node and edge attribute files that can be loaded into Cytoscape to visualize the weighted networks using different criteria and for further integration and analysis. Cytoscape's VizMapper feature is used to visualize the weighted interaction network with differing node sizes and edge thickness to highlight significant interactions and interactors.

#### 4.2.10 Synthetic dataset

To test Netshadow on a known system, we generated synthetic data corresponding to the hypothetical scenario described in the introduction and depicted in Figure 4.1(a). The data were generated using a steady-state assumption and include nonlinear relationships with 20% Gaussian noise. Specifically, independent variables were sampled from a normal distribution and dependent variables were modeled with sigmoid functions plus a noise term. Noise was sampled from a normal distribution with mean 20% of the dependent variables mean. Scripts for generating and using synthetic data are included in the Supplementary Material.

#### 4.2.11 Experimental dataset and pre-processing

We assembled gene expression data from a previously described gene expression analysis of TGF- $\beta$ -induced EMT in A459, lung adenocarcinoma cell line (GEO ID: GSE17708) (Keshamouni et al., 2009; Sartor et al., 2010). Briefly, in this experiment A459 cells were stimulated with 5 ng/mL of TGF- $\beta$  to induce EMT and harvested at 0, 0.5, 1, 2, 4, 8, 16, 24 and 72 hours in 3 separate experiments, and the resulting RNA collected and assayed using Affymetrix HG-U133\_plus\_2 arrays. The data were analyzed using the Robust Multichip Average (RMA) (Bolstad et al., 2003) algorithm as implemented in the Bioconductor Affy package and annotated using updated probeset definitions (.CDF files) provided by the Microarray Lab at the University of Michigan (Dai et al., 2005). The custom CDF re-annotation discarded genes that no longer map to probesets with high confidence, thus decreasing the number of genes assayed to 16,713. Because our method relies on probabilistic relationships between gene expression profiles rather than differential expression, we did not need to filter the genes using a fold-change filter and, instead, selected all 16,713 genes from the assay to query prior knowledgebases. We used a threshold of *fold change* > 3.0 to select 485 genes for the target gene set. Note that differential expression is used to select the set of target genes that act as bait but not candidate interactors or interactions.

### 4.3 Results

We first show the results of a simulated study with synthetic data and then describe the results with the experimental gene expression datasets from the cancer cell-line EMT study.

### 4.3.1 Simulated study with synthetic data

We generated synthetic data as described in the Methods. Because the synthetic data were stochastically generated, for each test, we created 25 different datasets and report the average results. We also tested the effect of data size by generating datasets with 100, 50, 25, 10 and 5 samples. We compared the interaction score for three interactions: A–B, C–D and a hypothetical interaction B–C (Figure 4.1(b)). With every dataset size, the true interaction, A–B, outscores B–C and C–D although with few data samples, the differences in scores are smaller. Note that the scores for B–C are better than for C–D, indicating that a false interaction involving one interactor from a true interaction can score well, leading to potential false-positives in our final result.

### 4.3.2 Netshadow analysis of EMT interactome

We queried HPRD with all 16,713 genes in the experimental dataset to generate a prior interaction network with 28,275 PPI. Using a fold-change threshold of 3.0, we constructed a target gene set of 485 genes. We then scored each interaction in the 28,275 edge network and assessed edge confidence as described in Methods. To create the EMT subnetwork, we pruned all edges with  $p > .01$  and weighted remaining edges with their interaction score. The resulting subnetwork has 269 interactions between 301 proteins and includes hubs, large clusters and connected components.

### 4.3.3 Significant protein-protein interactions

Table 4.1 lists the top 20 interactions from the EMT subnetwork shown in Figure 4.3(c). The interactions are related to expected cellular processes such as cell cycle regulation, TGF- $\beta$  pathway, extracellular matrix modeling, adhesion, migration, differentiation, proliferation and cytoskeleton organization.

### 4.3.4 Significant protein hubs

In many different types of cellular networks, a few nodes with high connectivity, called hubs, act as the backbone and hold the network together by providing a path for interactions among many other nodes (Barabasi and Oltvai, 2004). Table 4.2 lists the top 20 hubs in the EMT subnetwork. Most of the hubs are well-known participants in EMT. FN1, CTNNB1, VIM, COL4A1, TGFB1 are known EMT markers (Zeisberg and Neilson, 2009) and 12 of

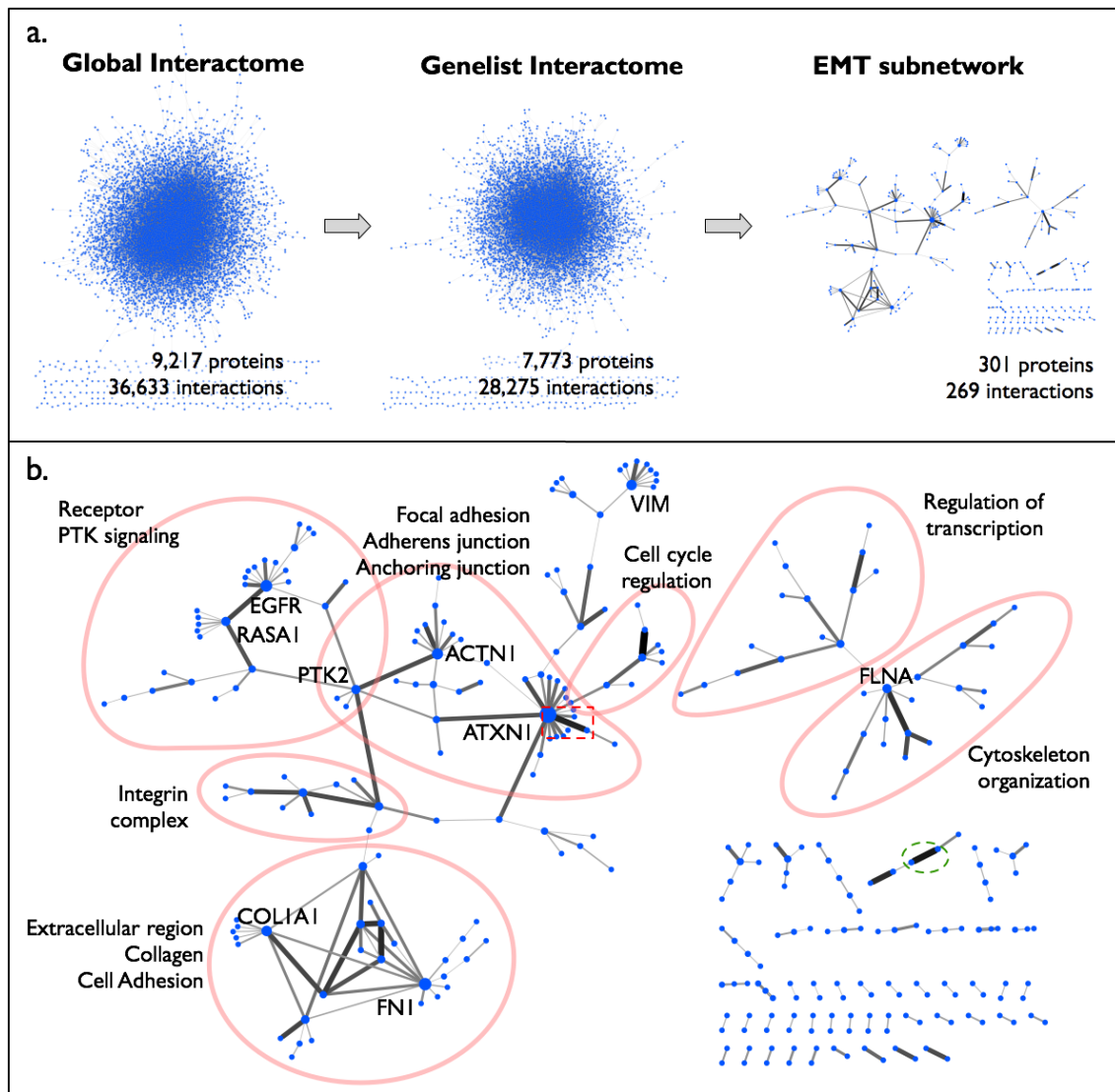


Figure 4.3: **Extracting the EMT subnetwork from the global interactome.** The top panel shows the simplification of the global interactome by successive application of the EMT gene expression data and the Netshadow algorithm. The networks depicted are (1) the global protein interaction network, (2) the subnetwork among proteins in the EMT dataset and (3) the Netshadow generated EMT subnetwork ( $p < .01$ ). The lower panel shows the EMT subnetwork network in detail. Node size is proportional to the nodes degree (number of interactions) and edge color and thickness are proportional to the interaction score. The network is annotated with enriched Gene Ontology terms and the 10 largest hub nodes. Two interactions that are discussed in the text are highlighted: ATXN1–ANP32A in red square and NEK6–SMURF2 in green circle.

Interaction	Score	P-value	Absolute Correlation	Correlation Rank	Annotation
CDKN1B – CCND3	-11944	< 1E-04	0.88	282	TGF- $\beta$ induced cell cycle regulation (Zhang et al., 1999, 2005; Toyoshima and Hunter, 1994)
NEK6 – SMURF2	-12016	1.00E-04	0.39	9134	TGF- $\beta$ induced cell cycle regulation (Yin et al., 2003; Ewing et al., 2007; Lin et al., 2000)
ANP32A – ATXN1	-12054	2.00E-04	0.66	2316	Adhesion-related gene expression repression (Cvetanovic et al., 2007; Matilla et al., 1997)
SERPINE2 – COL4A2	-12065	3.00E-04	0.92	110	ECM modeling (Donovan et al., 1994; Buchholz et al., 2003)
PAWR – RRAS2	-12082	3.00E-04	0.89	222	
FLNA – RALA	-12094	3.00E-04	0.75	1271	Migration (Ohta et al., 1999)
SERPINE2 – COL4A1	-12104	3.00E-04	0.96	26	ECM modeling (Donovan et al., 1994; Buchholz et al., 2003)
TGFBI – COL4A1	-12131	3.00E-04	0.96	14	Adhesion (Kim et al., 2002)
ARNTL – AHR	-12152	3.00E-04	0.73	1432	Proliferation-related gene expression (Hogenesch et al., 1997; Shimba et al., 2002)
JAG1 – THBS1	-12158	3.00E-04	0.89	212	Notch signaling (Aho, 2004)
PDLIM5 – ACTN1	-12165	3.00E-04	0.87	206	Cytoskeleton organization (Nakagawa et al., 2000)
TGFBI – COL1A1	-12166	3.00E-04	0.97	11	Adhesion (Kim et al., 2002; Billings et al., 2002)
EGFR – RASA1	-12167	3.00E-04	0.9	163	Proliferation (Serth et al., 1992; Liu and Pawson, 1991)
EXOC2 – RALA	-12179	4.00E-04	0.84	450	Cytoskeleton organization (Moskalenko et al., 2003; Sugihara et al., 2002)
TBC1D4 – NAV1	-12183	4.00E-04	0.75	1191	
EXOC1 – RALA	-12183	4.00E-04	0.91	129	Cytoskeleton organization (Moskalenko et al., 2003; Sugihara et al., 2002; Brymora et al., 2001)
ITGAV – TGFB1	-12184	5.00E-04	0.87	319	TGF- $\beta$ pathway (Mu et al., 2002)
TGFB1 – BMP2	-12184	5.00E-04	0.77	1050	Differentiation (Kim and Niyibizi, 2001)
SAT1 – CCDC90B	-12187	5.00E-04	0.82	592	
ATXN1 – RBM9	-12187	5.00E-04	0.86	344	Proliferation and differentiation (Lim et al., 2006)

Table 4.1: **Top 20 interactions in the EMT subnetwork.** The BN Score and p-value are calculated as described in the Methods. The fourth and fifth columns report the absolute value of the gene expression correlation between interacting proteins and the rank according to that correlation (where lower number indicates higher ranking). Note that Netshadow’s mechanistic model ranks interactions significantly differently than linear correlation.



Protein	Description	Degree	Fold Change	FC Rank
ATXN1	ataxin 1	19	2.65	648
FN1	fibronectin 1	12	3.88	210
EGFR	epidermal growth factor receptor	11	2.14	1083
ACTN1	actinin, alpha 1	10	2.62	668
COL1A1	collagen, type I, alpha 1	9	1.95	1320
VIM	vimentin	9	1.79	1569
FLNA	filamin A, alpha	7	2.23	987
PTK2	PTK2 protein tyrosine kinase 2	7	1.51	2207
RASA1	RAS p21 protein activator 1	7	1.57	2076
COL4A1	collagen, type IV, alpha 1	6	6.81	14
COL7A1	collagen, type VII, alpha 1	6	2.49	765
ITGAV	integrin, alpha V	6	2.27	952
TGFB1	transforming growth factor, beta 1	6	2.47	778
THBS1	thrombospondin 1	6	4.14	168
AR	androgen receptor	5	2.34	877
CCND3	cyclin D3	5	2.04	1196
COL4A2	collagen, type IV, alpha 2	5	6.7	17
CTNNB1	catenin, beta 1, 88kDa	4	1.81	1536
DUT	deoxyuridine triphosphatase	4	1.15	3799
EPHB2	EPH receptor B2	4	3.33	360

Table 4.2: **Top 20 hubs in the EMT subnetwork.** The protein hubs are sorted according to node degree (number of interactions). The fourth and fifth columns report the gene expression fold change and ranking based on that metric, respectively. Note that, in the EMT subnetwork, hub nodes are not generally differentially expressed.

the 20 largest hubs (PTK2, ACTN1, CTNNB1, COL1A1, COL4A1, COL4A2, CCND3, EGFR, FN1, FLNA, ITGAV, THBS1) reflect actin cytoskeleton reorganization, extracellular matrix remodeling, cell-cycle regulation and formation of Focal Adhesions which is required for acquiring a migratory and invasive phenotype during EMT (Keshamouni et al., 2006).

Interestingly, the largest hub, ATXN1, is not known to be involved in EMT. ATXN1 is known for its role in the neurodegenerative disease known as spinocerebral ataxia type 1. Though the precise function is not known, ATXN1 was shown to bind RNA *in vitro* (Yue et al., 2001) and demonstrated to have the ability to shuttle between the nucleus and cytoplasm (Irwin et al., 2005) and to modulate gene transcription (Okazawa et al., 2002; Tsai et al., 2004; Tsuda et al., 2005; Lam et al., 2006; Serra et al., 2006). In our results, its interaction partners include adhesion and adherens junction related proteins such as TRIP6, ITGB4 and ZYX and transcriptional repressors and modulators such as ZHX2, SPEN,

RBM9, ANP32A and UBE2I, consistent with its role as a transcription modulator. Among these, the ATXN1 interaction with ANP32A (red square in Figure 4.3(b)) is the third-highest scoring interaction in our result. ANP32A is a co-repressor that inhibits histone acetyltransferases, forms a complex with the transcriptional repressor E4F and modulates its activity; it is known that ATXN1 relieves this transcriptional repression by competitively binding ANP32A (Cvetanovic et al., 2007). Gene ontology analysis of the downstream genes for the ATXN1–ANP32A interaction (Figure 4.4) showed enrichment for the processes of cell adhesion and cell migration in up-regulated genes (ADAM19, GLIPR1, ALOX5AP, BEAN, PTPRF, PTPRK, RHOU, SLC26A2, SLC29A1 and DOCK4). The one down-regulated gene, HABP2 which binds hyaluronic acid in the ECM, is a known inhibitor of vascular smooth muscle cell proliferation and migration (Kannemeier et al., 2004). This suggests that the ATXN1–ANP32A interaction might play an important role in the regulation of cell adhesion and migration during TGF- $\beta$ -induced EMT and worth considering an experimental validation.

Cellular Component	GO ID	Count	P-value	Benjamini
basement membrane	GO:0005604	16	2.16E-11	7.69E-09
adherens junction	GO:0005912	20	1.05E-10	1.86E-08
extracellular region part	GO:0044421	49	3.05E-10	3.62E-08
basolateral plasma membrane	GO:0016323	22	3.70E-10	3.29E-08
anchoring junction	GO:0070161	20	6.11E-10	4.35E-08
extracellular matrix part	GO:0044420	17	7.21E-10	4.28E-08
focal adhesion	GO:0005925	15	7.70E-09	3.91E-07
extracellular matrix	GO:0031012	26	9.17E-09	4.08E-07
cell-substrate adherens junction	GO:0005924	15	1.26E-08	4.98E-07
Cytoskeleton	GO:0005856	57	2.40E-08	8.55E-07
cell-substrate junction	GO:0030055	15	2.84E-08	9.18E-07
extracellular space	GO:0005615	36	6.77E-08	2.01E-06
proteinaceous extracellular matrix	GO:0005578	23	1.84E-07	5.03E-06
integrin complex	GO:0008305	8	7.46E-07	1.90E-05
cytoplasmic membrane-bounded vesicle lumen	GO:0060205	9	1.21E-06	2.87E-05
cell projection	GO:0042995	34	1.24E-06	2.76E-05
extracellular region	GO:0005576	68	1.49E-06	3.11E-05
actin cytoskeleton	GO:0015629	20	1.67E-06	3.30E-05
vesicle lumen	GO:0031983	9	1.73E-06	3.24E-05
cytoplasmic vesicle part	GO:00044433	16	2.43E-06	4.32E-05

**Table 4.3: Top 20 enriched Gene Ontology cellular components in the EMT subnetwork.** The top 20 enriched cellular components in the EMT subnetwork include various cell-cell and cell-substrate adhesion junctions, the cytoskeleton and components of the extracellular space. The second column reports the number of genes in the Netshadow network that are annotated with a given Gene Ontology term; the p-value and Benjamini-corrected p-value are reported in the third and fourth columns, respectively.

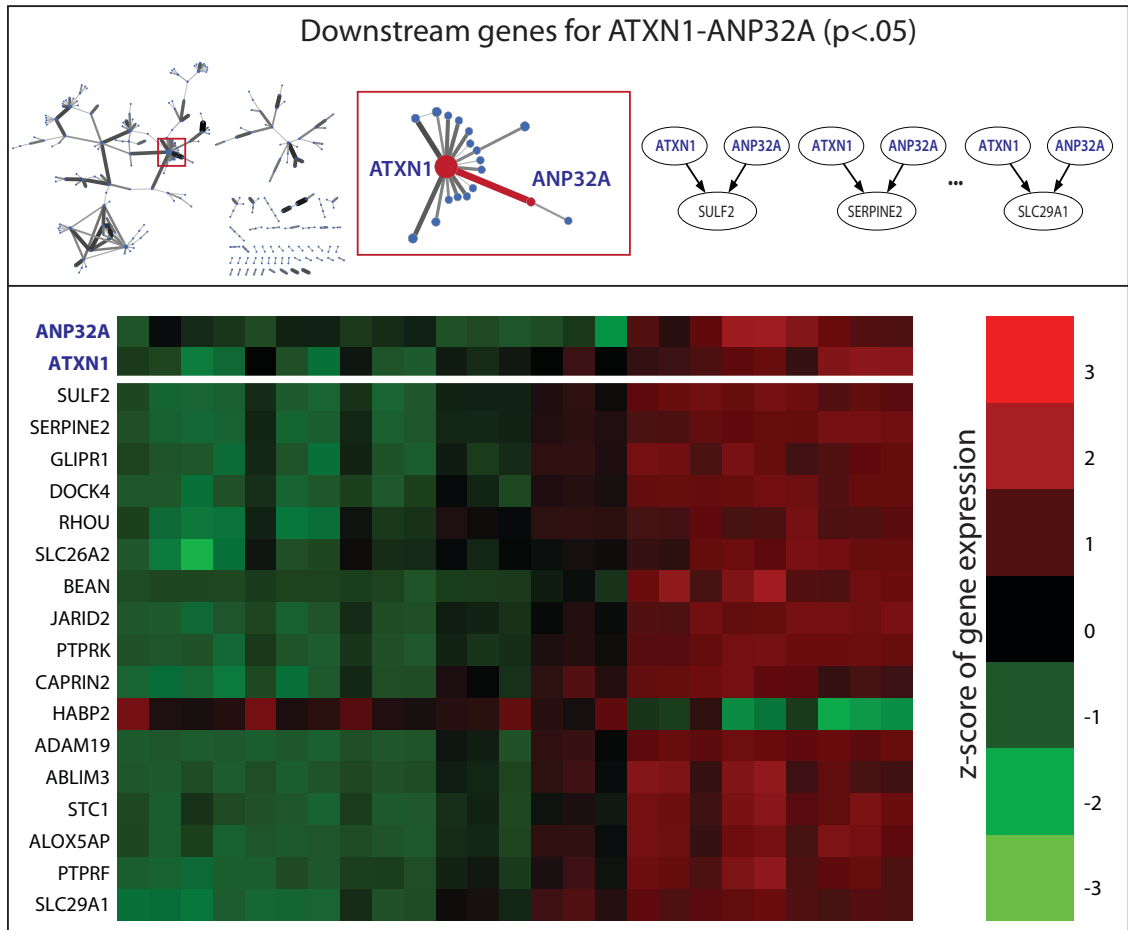


Figure 4.4: **Genes downstream from the interaction between ATXN1 and ANP32A.** The top panel shows the Netshadow generated EMT subnetwork; a close-up surrounding the interaction between ATXN1 and ANP32A; and the Bayesian networks used to rank the downstream genes. The heatmap in the lower panel shows the z-score of the gene expression profile of ATXN1, ANP32A and the downstream genes ( $p < .05$ ). The arrays are arranged according to time since TGF- $\beta$  induction.

### 4.3.5 Enriched processes, components, and pathways

The functions of proteins in the EMT subnetwork and the pathways in which they participate were assessed using the DAVID functional analysis tool (Huang da et al., 2009b). The top 20 biological processes, top 20 cellular components and top 10 KEGG pathways enriched in the list of proteins are shown in tables 4.4, 4.3 and 4.5, respectively. Enriched processes include those related to adhesion, motility, proliferation and apoptosis. The proteins constitute cellular components related to inter-cellular and cell-substrate junctions, the extracellular matrix, the cytoskeleton and vesicle transport. Not surprisingly, the proteins participate in

adhesion, inter-cellular junction and cytoskeleton related pathways; they also include known cancer-related gene expression patterns, further supporting the hypothesis that EMT is a critical step in cancer progression.

## 4.4 Discussion

In the following section, we discuss the results of the Netshadow analysis with the TGF- $\beta$ -induced EMT experimental dataset and compare with other methods.

Biological Process	GO ID	Count	P-value	Benjamini
cell adhesion	GO:0007155	53	4.53E-18	1.05E-14
biological adhesion	GO:0022610	53	4.81E-18	5.60E-15
cellular component movement	GO:0006928	40	1.08E-14	8.36E-12
localization of cell	GO:0051674	40	1.08E-14	8.36E-12
regulation of cell proliferation	GO:0042127	51	1.94E-14	1.13E-11
regulation of cell motion	GO:2000145	25	1.10E-13	5.10E-11
cell migration	GO:0016477	29	4.43E-13	1.72E-10
enzyme linked receptor protein signaling pathway	GO:0007167	31	1.96E-12	6.52E-10
response to wounding	GO:0009611	38	3.13E-12	9.11E-10
regulation of cell migration	GO:0030334	22	4.53E-12	1.17E-09
cell motility	GO:0048870	29	7.40E-12	1.72E-09
positive regulation of macromolecule metabolic process	GO:0010604	47	1.66E-11	3.51E-09
regulation of locomotion	GO:0040012	22	5.88E-11	1.14E-08
regulation of apoptosis	GO:0042981	45	1.21E-10	2.16E-08
regulation of programmed cell death	GO:0043067	45	1.65E-10	2.74E-08
regulation of cell death	GO:0010941	45	1.84E-10	2.86E-08
positive regulation of biosynthetic process	GO:0009891	41	2.62E-10	3.81E-08
positive regulation of cellular component movement	GO:0051272	16	2.86E-10	3.92E-08
cell-substrate adhesion	GO:0031589	16	3.32E-10	4.29E-08
positive regulation of cellular biosynthetic process	GO:0031328	40	7.08E-10	8.67E-08

Table 4.4: **Top 20 enriched Gene Ontology biological processes in the EMT subnetwork.** The top 20 enriched biological processes in the EMT subnetwork relate to adhesion, motion, proliferation and cell death, processes known to be involved in EMT. The second column reports the number of genes in the Netshadow network that are annotated with a given Gene Ontology term; the p-value and Benjamini-corrected p-value are reported in the third and fourth columns, respectively.

#### 4.4.1 Netshadow identifies a core subnetwork underlying emt

In general, global PPI network are difficult to interpret, as is shown in Figure 4.3(a). After a Netshadow analysis, we identify (at  $p < .01$ ) a smaller core subnetwork (Figure 4.3(b)) that recapitulates biological processes known to be involved in EMT: cell cycle regulation, TGF- $\beta$  pathway, cell adhesion, extracellular matrix degradation, cell migration, proliferation, Notch signaling, and cytoskeleton organization.

#### 4.4.2 Netshadow identifies relevant linear and nonlinear interactions

Table 1 lists the 20 highest scoring interactions in the EMT Netshadow subnetwork. The table also lists the Pearson correlation between the interacting proteins and their rank according to that correlation. Only two of the highest scoring interactions in Table 1 (TGFB1–COL1A1 and TGFB1–COL4A1) are in the top 20 interactions as ranked by interactor correlation, and only a third in the top 100 (SERPINE2–COL4A1). This is not surprising because several studies have observed low gene expression correlation between interacting proteins (Bhardwaj and Lu, 2005; Jansen et al., 2002; Xulvi-Brunet and Li, 2010).

Similarly, our method identifies important proteins that are ranked poorly by differential gene expression as measured by fold change (Table 2). Only two of the 20 proteins identified as important by our method (COL4A1 and COL4A2) would be ranked in the top 20 according to the differential gene expression.

#### 4.4.3 Netshadow identifies interactions missed by topological analysis

In weighted networks such as the one generated by Netshadow, each edge is scored using quantitative data. With unweighted networks, edge importance can be assessed using various topological measures that consider only the qualitative architecture of the network.

By integrating experimental data, Netshadow identifies physiologically relevant interactions that may or may not be topologically significant. By topologically significant we mean the interactions that involve hub nodes, are part of a cluster, or connect multiple clusters. As an example, consider the second-highest scoring interaction in our result, that between NEK6 and SMURF2 (Table 1). NEK6 is a serine/threonine kinase that is required for cell cycle progression and implicated in cancer cell transformation (Yin et al., 2003). SMURF2 is a Smad specific ubiquitin ligase that regulates the stability of Smad proteins

Pathways	KEGG ID	Count	P-value	Benjamini
Pathways in cancer	map05200	24	4.30E-08	3.31E-06
Regulation of actin cytoskeleton	map04810	13	6.16E-06	2.37E-04
ECM-receptor interaction	map04512	11	6.78E-06	1.74E-04
Leukocyte transendothelial migration	map04670	12	8.24E-06	1.59E-04
Adherens junction	map04520	10	1.19E-04	1.82E-03
Focal adhesion	map04510	10	5.00E-04	6.40E-03
p53 signaling pathway	map04115	9	1.28E-03	1.40E-02
Thyroid cancer	map05216	5	1.77E-03	1.69E-02
Bladder cancer	map05219	6	3.13E-03	2.64E-02
Pancreatic cancer	map05212	7	3.95E-03	3.00E-02

Table 4.5: **Top 10 KEGG pathways in the EMT subnetwork.** The top 10 KEGG pathways enriched in the EMT subnetwork are related to cytoskeleton remodeling; cell-cell and cell-ECM interactions; and several cancer-related pathways. The second, third and fourth columns list the overlap between the pathway and EMT subnetwork, the p-value for that overlap and that Benjamini-corrected p-value, respectively.

which are mediators of TGF- $\beta$  signaling (Lin et al., 2000). Additionally, the interaction has been observed in vivo and implicated in TGF--dependent dissolution of tight junctions during EMT (Ewing et al., 2007; Barrios-Rodiles et al., 2005). Topologically, however, the NEK6 and SMURF2 edge is uninteresting in the EMT subnetwork because neither NEK6 or SMURF2 are hub nodes nor do either interact with hub nodes. Additionally, the interaction is neither part of a cluster nor does it connect other clusters (blue circle in Figure 4.3(a)).

## 4.5 Conclusion

Although this analysis used a protein-protein interaction network and gene expression data, the Netshadow method can be used more generally to integrate mechanistic knowledge with high-throughput measurements. For example, it would be possible to create a subnetwork of relevant miRNA-target interactions using a miRNA interactions database and gene expression data.

# Chapter 5

## Conclusions and Future Directions

In this thesis, I undertook the challenge of creating analytical methods for integrating generic knowledge and noisy experimental high-throughput data to identify relationships between the interactome and transcriptome that are consistent with both known mechanisms and statistical relationships in the data. In this chapter, I summarize the key contributions of the thesis and offer suggestions for future directions.

### 5.1 Mechanistic Bayesian networks

Systems biology is based on the premise that an integration of more and different types of information combined with a global quantitative analysis of the cell will yield richer understanding than analysis of a few molecules in isolation or with one modality of information. In chapters 1 and 2, I presented Mechanistic Bayesian networks (MBN) as a modeling strategy and set of algorithms for constructing Bayesian network (BN) models that are consistent with mechanisms described in existing knowledge and statistical relationships in data.

To learn MBN models, I use modifications to the Bayesian Dirichlet equivalent (BDe) scoring metric to accommodate learning with data from interventions, data with missing values and to include hidden variables. In MBN analysis, I formulate problems more specific than the general structure-learning approach common in BN analysis of bioinformatics data. This results in a significantly constrained space of models that can be exhaustively enumerated. This constrained space of models also allows me to better represent mechanisms using methods that are unfeasible with the structure-learning approach. For example, in chapter 3, I used hidden nodes to represent unmeasured proteins and their interactions and in chapter 4, I used a mechanistically-faithful model to evaluate proteomic relationships.

### 5.1.1 Python environment for Bayesian learning

To learn MBN models required software with the following features that were lacking in extant Bayesian network and machine learning libraries and applications:

- specification of soft and hard priors for integrating biological knowledge
- maximum-entropy discretization
- altered BDe for interventional data
- BDe with Gibbs sampling for data with missing values and hidden variables
- easy specification of custom algorithms for structure learning in constrained subspaces
- ability to run on distributed clusters and cloud-computing platforms

In chapter 2, I described the development of PEBL, the Python Environment for Bayesian Learning. PEBL is an open-source python library to learn Bayesian and MBN models from data and knowledge. In addition to the novel MBN features, PEBL includes features useful for running large and repeatable analyses. PEBL can be used interactively via the python shell, be integrated into other scripts or application or be run as a stand-alone binary driven by a configuration file. PEBL can output its results as python objects or as a set of dynamic HTML files. PEBL offers an easy way to write custom learners that can search through a constrained space of network models. Finally, PEBL can transparently run all analysis on a single computer or on various clusters and cloud-computing services. This mix of features allows non-expert programmers to easily run large, distributed analyses driven by a configuration file or simple script. As an example of the simplicity of required programming, the analyses presented in chapters 3 and 4 are driven by short, easy-to-understand scripts.

The quality of academic software has been widely lamented as being far behind commercial software. All too often, the development process and resulting software do not meet the basic criteria of quality software development. Because the software is not well-documented, well-tested, easily-installable or configurable, it is of minimal utility to other researchers and exists as a record of past analysis rather than as a reusable tool. For all software development during this thesis, I made a concerted effort to follow the best practices of the open-source development community. PEBL was developed using publicly-accessible version control system, issue tracker and mailing lists. PEBL includes a tutorial, developer's guide, API reference and source-code comments. It also includes a suite of 216 automated tests to ensure correctness and guard against introducing errors during the development process. Finally, all source code is released under a liberal open-source license that allows for unrestricted academic, non-profit and commercial use.



### 5.1.2 MBN for identifying pathway targets

The Sonic Hedgehog pathway that is involved in normal organismal development and cancer progression is a good example of the complexities present in many biological pathways. The Shh ligand acts as both a short-range contact-dependant factor and a long-range diffusible morphogen. The effects of the pathway are time-, dose-, and context-dependent and the effector genes – Gli1, Gli2 and Gli3 – cooperate with or antagonize other pathways such as Bmp, retinoic acid, Wnt, Ras and Notch. Generally, due to the interactions with other pathways and multiple paths between ligand binding and eventual transcriptional regulation, it is not even clear what constitutes a downstream target of a particular pathway.

In chapter 3, I encoded the known early events in the Shh pathway as a MBN template – providing a mathematical definition for Shh target genes – and used the template to identify likely targets using a gene-knockout mouse-model dataset. The results included known Shh targets, recapitulated the downstream effects of the pathway and presented novel predictions. A comparison with other methods showed that the MBN method made better target predictions, due in part because the topology of the MBN template better matched the known biochemical mechanisms.

Although the specific analysis searched for downstream targets, MBN templates based on known partial qualitative topology can be used to identify likely members, effectors, upstream regulators or antagonists of a pathway or network. For larger templates, the computational complexity of the Gibbs sampling could make the analysis intractable. Other methods for learning with hidden nodes such as Variational learning ([Beal and Ghahramani, 2006](#)) should be explored for suitability within the MBN framework.

### 5.1.3 MBN for subnetwork identification

Interaction knowledgebases contain qualitative assertions aggregated from literature, past measurements and past analyses. They represent a high-level compilation of a large amount of information and are useful in understanding the mechanisms active in biological processes. These knowledgebases, however, represent different physiological and experimental conditions: different cell types, tissues, interventions and experiments. Not every interaction will be relevant to any specific biological process. Furthermore, the large size of the networks makes them inscrutable, leading to the well-known *hairball network* problem where a network is inscrutable to visual analysis.

In chapter 4, I used MBN to integrate interactome knowledge with experimental data to identify a core subnetwork relevant to the epithelial-mesenchymal transition (EMT). EMT is a process by which a fully-differentiated epithelial cell undergoes a phenotype change to become a mesenchymal cell. Like the Shh pathway studied in chapter 3, EMT is critical in both normal organismal development and cancer progression – it is required, for example, in growing limbs from a bud and for localized tumors to metastasize.

Typical subnetwork identification methods make one or both of the following assumptions: 1) that proteins important to a process or biological condition show significant differential mRNA expression and 2) that proteins that interact are co-regulated and thus have correlated gene expression profiles. Many studies have found that these assumptions are true only in limited cases and cannot be relied upon in the general case. Following the MBN paradigm, rather than making these assumptions, I created a MBN template in which the interactors in a candidate interaction regulate the expression of a set of potential target genes that were observed to be active in the EMT process. This faithful encoding of the known biochemical mechanisms allowed me to learn a much smaller, targeted subnetwork that recapitulated processes known to be involved in EMT.

Although gene expression data was used to identify the core subnetwork of an initial protein-protein interaction network, the method is more generally applicable. The initial network can be of other interaction types such as miRNA-mRNA or protein-DNA; or it can contain a mixture of interaction types. For each interaction type, one would need to create a scoring metric based on a MBN template appropriate for the mechanisms involved in the interaction.

## **5.2 Extensions to MBN**

In this thesis, MBN templates were used to identify the regulatory targets of a signaling pathway and the proteomic subnetwork responsible for observed transcriptional activity. In the following sections, the Sonic hedgehog (Shh) pathway is used as an example to demonstrate other uses for MBN templates, to identify their limitations and to suggest changes that allow for a more thorough integration of existing knowledge to identify mechanisms in the pathway.

## 5.2.1 Identifying Shh coregulators and pathway crosstalk

In chapter 3, a MBN template was used to identify the downstream targets of the Sonic hedgehog (Shh) signaling pathway. Figure 5.1 depicts two other templates that use existing knowledge about the Shh pathway to identify proteins that co-regulate the Shh target Gas1 and signaling pathways that crosstalk with Shh, respectively. MBN templates present a principled approach for representing queries about biological systems in terms of interaction topologies and for using known topological information to identify novel members.

## 5.2.2 Limitations of MBN templates

MBN templates are a way to represent a set of hypotheses, each specified by a BN model. A template is similar to a regular BN but has two types of nodes: constant and template nodes. Constant nodes are regular nodes that represent observed or unobserved variables. Template nodes are placeholders and replaced by specific variables in each instantiation of the template. All the hypotheses represented by a MBN template, thus, have the same structure but differ in the identity of some nodes. In chapter 3, the template node was instantiated by all variables in the dataset. In chapter 4, the template contained two template nodes representing interacting proteins; they were instantiated with protein interactions from the Human Protein Reference Database (HPRD) interactome. MBN templates as used in this thesis have three limitations:

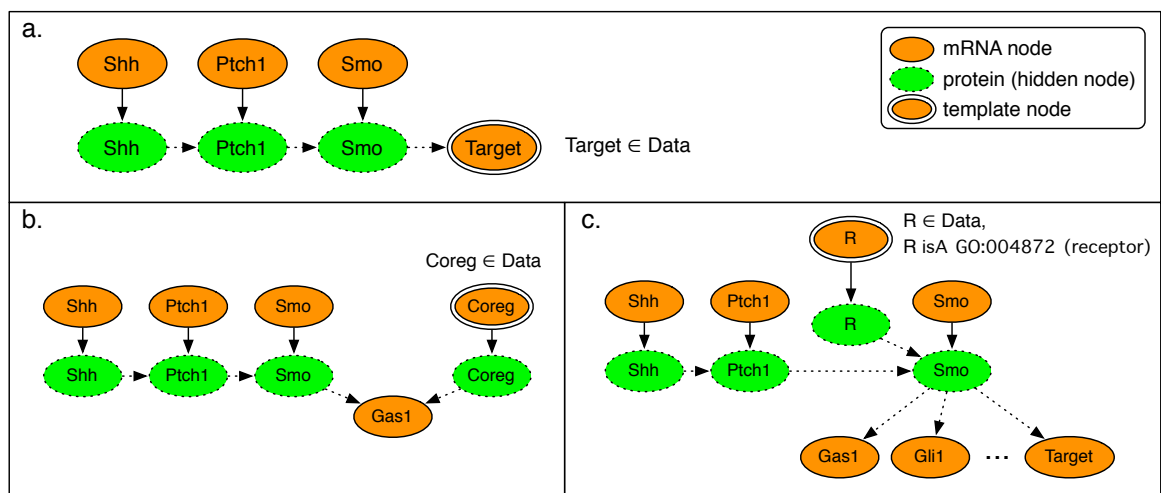


Figure 5.1: **Other uses of MBN templates.** Panel a depicts the template used to identify Sonic hedgehog (Shh) targets. Panels b and c show templates to identify co-regulators for the Shh target Gas1 and signaling pathways that might crosstalk with Shh, respectively.

1. Only node identity can vary between template instantiations.
2. MBN templates are constructed by hand.
3. All template instantiations have equal probability.

These three limitations mean that templates are only useful for visually representing a set of hypotheses but do not help in generating these hypotheses in the first place. Additionally, the types of hypotheses that can be represented is limited; hypotheses about different signaling paths from ligand binding to gene transcription, for example, cannot be represented by a template. Finally, although the models generated by a template are evaluated using Bayesian methods, templates do not offer any method for calculating prior probabilities.

### 5.2.3 Increasing the expressiveness of MBN templates

MBN templates, as used in this thesis, cannot describe arbitrary regions of the model search space because their structure is fixed and only node identities can vary. By using alternate Bayesian frameworks such as Multi-entity Bayesian networks (MEBN)([Laskey, 2006](#)), templates can have variable structure, node identities and prior probabilities. [Figure 5.2](#) shows a MEBN-based template. In [chapter 3](#), the template only used the canonical Shh pathway – those few early events of the pathway that are known with high certainty. With a MEBN-based template, however, one can exploit knowledge about alternate signaling pathways between Shh ligand reception and eventual target regulation. When scoring potential targets, we can use all available knowledge to calculate posterior probabilities.

### 5.2.4 Generating hypotheses

MBN templates represent a set of hypotheses for a given biological query. The templates in this thesis were hand-built after consulting relevant literature and bioinformatics databases. This manual approach is biased and will become increasingly difficult as the amount of biological knowledge increases. To fully utilize existing knowledge, we require an automated method for generating hypotheses.

Existing tools such as HyBrow ([Racunas et al., 2004](#)) and BioDeducta ([Shrager et al., 2007](#)) use first-order logic and semantic knowledgebases to identify networks in response to a user's query ("all paths from Shh to Gli proteins") and simple semantic constraints ("for proteins to interact, they must be expressed in the same cell type"). They do not use

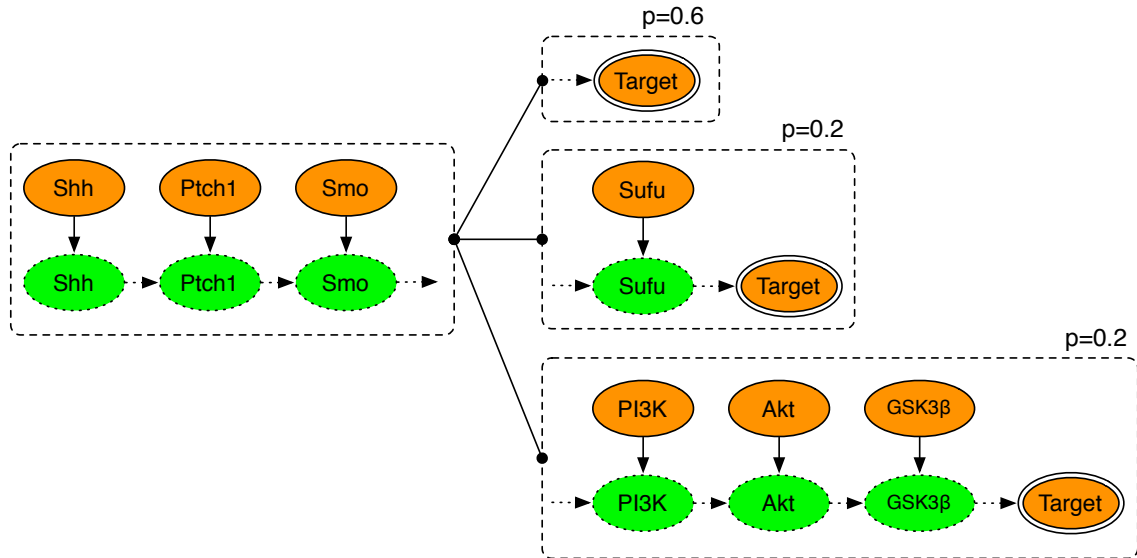


Figure 5.2: **A more expressive Shh template.** By defining templates using Bayesian frameworks such as Multi-entity Bayesian networks (MEBN), templates can describe any arbitrary region of the model search space by specifying variable structure, node identities and prior probabilities. This example depicts a more realistic encoding of the Shh pathway by including alternative paths between Shh ligand reception and target regulation.

experimental data to identify the context-specific variants. Their outputs, however, can be encoded as templates and further assessed with experimental datasets.

### 5.2.5 Cloud computing

The concepts and tools described in this thesis requires massive amounts of computing due to the large amounts of knowledge and data being analyzed and the large space of possible models that must be evaluated. High-performance computing is moving towards the use of shared-nothing systems based on commodity off-the-shelf hardware. This means that rather than using hardware specially designed for a particular application or with high-speed CPUs and interconnects, one uses standard commercially-available hardware connected via regular networking hardware and protocols. The ultimate expression of this idea is in the notion of cloud computing in which computing is treated like a utility not unlike electrical power or water. A user (or computer) requests some amount of computing resources from a provider and pays only for the specific usage; for many tasks, this is much cheaper and more convenient than purchasing and maintaining a dedicated cluster.

PEBL, the Bayesian and MBN software described in chapter 2 can run analysis on

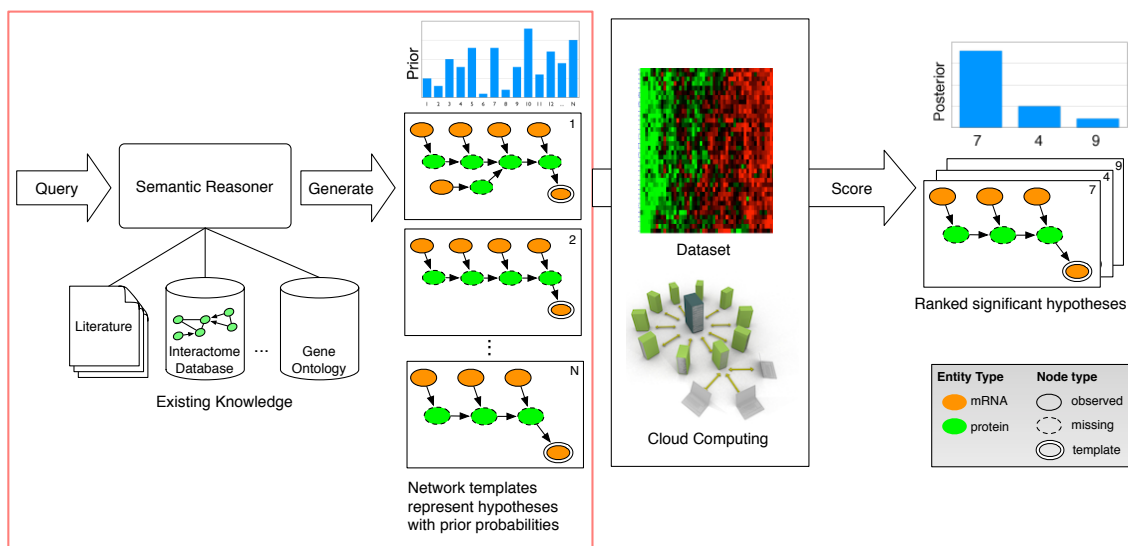


Figure 5.3: **Future MBN workflow.** Rather than creating the template by hand, one queries a semantic reasoner using a formal query language. The reasoner uses semantically-encoded knowledge to generate a template to represent all the possible answers – along with their knowledge-based prior probabilities – to the query. Experimental datasets are then used to calculate the context-specific posterior probabilities. The steps outlined in red are those that differ from the MBN workflow used in this thesis.

a single-core CPU, on multiple cores or CPUs, on Apple’s Xgrid, on an IPython cluster or on Amazon’s Elastic Cloud Computing (EC2) platform. A PEBL learner – whether included in PEBL or custom-written – can be configured to run on different platforms by changing a few configuration parameters. This has allowed me to write software that use our lab’s Xgrid but can be distributed to other researchers who may have access to other clusters. Furthermore, with the EC2 support, one can rent the required resources, making the code developed in chapters 3 and 4 reusable for labs without dedicated computing resources.

I refactored the cloud-computing parts of PEBL into a separate library called anyCloud that allows python programmers to integrate similar functionality into their software. With anyCloud, a python programmer architects his code as a series of *tasks* that are created and run using a specified interface. To handle the amount of knowledge, data and analysis required, anyCloud and similar tools will be crucial for transforming bioinformatics software to run on public or private clouds rather than being tied to a single machine or specific cluster and grid technologies.

### 5.3 A future MBN workflow

The pathologies we are interested in addressing are complex and involve many unknowns. To translate results from the lab to the clinic will require modeling at multiple scales and integration of even more knowledge and data. It appears that the exponential growth we have witnessed in the amount of available biological information and the ability of new measurement technologies to interrogate living systems will not be slowing for the foreseeable future. The amount of information (and the rapid pace of new information) is already beyond the synthesis and reasoning capacity of the human mind. Unless we can find automated means to synthesize available knowledge and integrate it with targeted experimental data, the availability of information will become a burden rather than a benefit.

The MBN methodology provides a principled approach for representing queries and existing knowledge about biological systems and for using experimental datasets to identify the condition-specific interaction among biomolecules that underlie complex pathologies. Figure 5.3 depicts a future MBN workflow. By using semantic knowledgebases, semantic reasoners, expressive MEBN-based templates and Bayesian learning algorithms running on the cloud, we can effectively utilize all available knowledge and data, however uncertain, to identify mechanisms in complex biological systems.

## Bibliography

- Aho, S. 2004. Soluble form of jagged1: unique product of epithelial keratinocytes and a regulator of keratinocyte differentiation, *J Cell Biochem*, 92(6), 1271–1281.
- Allen, B. L., T. Tenzen, and A. P. McMahon. 2007. The hedgehog-binding proteins gas1 and cdo cooperate to positively regulate shh signaling during mouse development, *Genes Dev*, 21(10), 1244–1257.
- Anderson, L. and J. Seilhamer. 1997. A comparison of selected mrna and protein abundances in human liver, *Electrophoresis*, 18(3-4), 533–537.
- Ascher, D., P. F. Dubois, K. Hinszen, J. Hugunin, and T. Oliphant. 2001. An open source project: Numerical python, Tech. rep., Lawrence Livermore National Laboratory.
- Barabasi, A. L. and Z. N. Oltvai. 2004. Network biology: understanding the cell's functional organization, *Nat Rev Genet*, 5(2), 101–113.
- Barrett, T., D. B. Troup, S. E. Wilhite, P. Ledoux, D. Rudnev, C. Evangelista, I. F. Kim, A. Soboleva, M. Tomashevsky, K. A. Marshall, K. H. Phillippy, P. M. Sherman, R. N. Muerter, and R. Edgar. 2009. Ncbi geo: archive for high-throughput functional genomic data, *Nucleic Acids Res*, 37(Database issue), D885–890.
- Barrios-Rodiles, M., K. R. Brown, B. Ozdamar, R. Bose, Z. Liu, R. S. Donovan, F. Shinjo, Y. Liu, J. Dembowy, I. W. Taylor, V. Luga, N. Przulj, M. Robinson, H. Suzuki, Y. Hayashizaki, I. Jurisica, and J. L. Wrana. 2005. High-throughput mapping of a dynamic signaling network in mammalian cells, *Science*, 307(5715), 1621–1625.
- Beal, M. J. and Z. Ghahramani. 2006. Variational bayesian learning of directed graphical models with hidden variables, *Bayesian Analysis*, 1(4), 793–832.
- Berti, C., B. Fontanella, R. Ferrentino, and G. Meroni. 2004. Mig12, a novel opitz syndrome gene product partner, is expressed in the embryonic ventral midline and co-operates with mid1 to bundle and stabilize microtubules, *BMC Cell Biol*, 5, 9.



- Bhardwaj, N. and H. Lu. 2005. Correlation between gene expression profiles and protein-protein interactions within and across genomes, *Bioinformatics*, 21(11), 2730–2738.
- Billings, P. C., J. C. Whitbeck, C. S. Adams, W. R. Abrams, A. J. Cohen, B. N. Engelsberg, P. S. Howard, and J. Rosenbloom. 2002. The transforming growth factor-beta-inducible matrix protein (beta)ig-h3 interacts with fibronectin, *J Biol Chem*, 277(31), 28003–28009.
- Bok, J., D. K. Dolson, P. Hill, U. Ruther, D. J. Epstein, and D. K. Wu. 2007. Opposing gradients of gli repressor and activators mediate shh signaling along the dorsoventral axis of the inner ear, *Development*, 134(9), 1713–1722.
- Bolstad, B. M., R. A. Irizarry, M. Astrand, and T. P. Speed. 2003. A comparison of normalization methods for high density oligonucleotide array data based on variance and bias, *Bioinformatics*, 19(2), 185–193.
- Breitling, R., A. Amtmann, and P. Herzyk. 2004. Graph-based iterative group analysis enhances microarray interpretation, *BMC Bioinformatics*, 5, 100.
- Brymora, A., V. A. Valova, M. R. Larsen, B. D. Roufogalis, and P. J. Robinson. 2001. The brain exocyst complex interacts with rala in a gtp-dependent manner: identification of a novel mammalian sec3 gene and a second sec15 gene, *J Biol Chem*, 276(32), 29792–29797.
- Buchholz, M., A. Biebl, A. Neesse, M. Wagner, T. Iwamura, G. Leder, G. Adler, and T. M. Gress. 2003. Serpine2 (protease nexin i) promotes extracellular matrix production and local invasion of pancreatic tumors in vivo, *Cancer Res*, 63(16), 4945–4951.
- Cabusora, L., E. Sutton, A. Fulmer, and C. V. Forst. 2005. Differential network expression during drug and stress response, *Bioinformatics*, 21(12), 2898–2905.
- Campanaro, S., S. Picelli, R. Torregrossa, L. Colluto, M. Ceol, D. Del Prete, A. D'Angelo, G. Valle, and F. Anglani. 2007. Genes involved in tgf beta1-driven epithelial-mesenchymal transition of renal epithelial cells are topologically related in the human interactome map, *BMC Genomics*, 8, 383.
- Chen, G., T. G. Gharib, C. C. Huang, J. M. Taylor, D. E. Misek, S. L. Kardia, T. J. Giordano, M. D. Iannettoni, M. B. Orringer, S. M. Hanash, and D. G. Beer. 2002. Discordant protein and mrna expression in lung adenocarcinomas, *Mol Cell Proteomics*, 1(4), 304–313.
- Chickering, David M., Dan Geiger, and David Heckerman. 1994. Learning bayesian networks is np-hard, Tech. rep., Microsoft Research.

- Cooper, Gregory and Edward Herskovits. 1992. A bayesian method for the induction of probabilistic networks from data, *Mach. Learn.*, 9(4), 309–347.
- Curtis, R. K., M. Oresic, and A. Vidal-Puig. 2005. Pathways to the analysis of microarray data, *Trends Biotechnol*, 23(8), 429–435.
- Cvetanovic, M., R. J. Rooney, J. J. Garcia, N. Toporovskaya, H. Y. Zoghbi, and P. Opal. 2007. The role of lanp and ataxin 1 in e4f-mediated transcriptional repression, *EMBO Rep*, 8(7), 671–677.
- Dahlquist, K. D., N. Salomonis, K. Vranizan, S. C. Lawlor, and B. R. Conklin. 2002. Genmapp, a new tool for viewing and analyzing microarray data on biological pathways, *Nat Genet*, 31(1), 19–20.
- Dai, M., P. Wang, A. D. Boyd, G. Kostov, B. Athey, E. G. Jones, W. E. Bunney, R. M. Myers, T. P. Speed, H. Akil, S. J. Watson, and F. Meng. 2005. Evolving gene/transcript definitions significantly alter the interpretation of genechip data, *Nucleic Acids Res*, 33(20), e175.
- Dakubo, Gabriel D., Shawn T. Beug, Chantal J. Mazerolle, Sherry Thurig, Yaping Wang, and Valerie A. Wallace. 2008. Control of glial precursor cell development in the mouse optic nerve by sonic hedgehog from retinal ganglion cells, *Brain Research*, 1228, 27–42.
- Dessaud, E., L. L. Yang, K. Hill, B. Cox, F. Ulloa, A. Ribeiro, A. Mynett, B. G. Novitch, and J. Briscoe. 2007. Interpretation of the sonic hedgehog morphogen gradient by a temporal adaptation mechanism, *Nature*, 450(7170), 717–720.
- Dijkstra, EW. 1959. A note on two problems in connection with graphs., *Num. Math.*, 1, 269–271.
- Dittrich, M. T., G. W. Klau, A. Rosenwald, T. Dandekar, and T. Muller. 2008. Identifying functional modules in protein-protein interaction networks: an integrated exact approach, *Bioinformatics*, 24(13), i223–231.
- Djebbari, A. and J. Quackenbush. 2008. Seeded bayesian networks: constructing genetic networks from microarray data, *BMC Syst Biol*, 2, 57.
- Donovan, F. M., P. J. Vaughan, and D. D. Cunningham. 1994. Regulation of protease nexin-1 target protease specificity by collagen type iv, *J Biol Chem*, 269(25), 17199–17205.
- Draghici, S., P. Khatri, A. L. Tarca, K. Amin, A. Done, C. Voichita, C. Georgescu, and R. Romero. 2007. A systems biology approach for pathway level analysis, *Genome Res*, 17(10), 1537–1545.

- Draghici, Sorin. 2003. *Data Analysis Tools for DNA Microarrays*, Chapman and Hall/CRC.
- Efron, Bradley and Robert Tibshirani. 1994. *An Introduction to the Bootstrap*, Chapman and Hall/CRC.
- Ewing, R. M., P. Chu, F. Elisma, H. Li, P. Taylor, S. Climie, L. McBroom-Cerajewski, M. D. Robinson, L. O'Connor, M. Li, R. Taylor, M. Dharsee, Y. Ho, A. Heilbut, L. Moore, S. Zhang, O. Ornatsky, Y. V. Bukhman, M. Ethier, Y. Sheng, J. Vasilescu, M. Abu-Farha, J. P. Lambert, H. S. Duewel, II Stewart, B. Kuehl, K. Hogue, K. Colwill, K. Gladwish, B. Muskat, R. Kinach, S. L. Adams, M. F. Moran, G. B. Morin, T. Topaloglou, and D. Figeys. 2007. Large-scale mapping of human protein-protein interactions by mass spectrometry, *Mol Syst Biol*, 3, 89.
- Friedman, N. 2004. Inferring cellular networks using probabilistic graphical models, *Science*, 303(5659), 799–805.
- Friedman, N., M. Linial, I. Nachman, and D. Pe'er. 2000. Using bayesian networks to analyze expression data, *J Comput Biol*, 7(3-4), 601–620.
- Gao, S. and X. Wang. 2007. Tappa: topological analysis of pathway phenotype association, *Bioinformatics*, 23(22), 3100–3102.
- Gat-Viks, I. and R. Shamir. 2007. Refinement and expansion of signaling pathways: the osmotic response network in yeast, *Genome Res*, 17(3), 358–367.
- Gautier, L., L. Cope, B. M. Bolstad, and R. A. Irizarry. 2004. affy-analysis of affymetrix genechip data at the probe level, *Bioinformatics*, 20(3), 307–315.
- Geman, Stuart and Donald Geman. 1993. Stochastic relaxation, gibbs distributions and the bayesian restoration of images, *Journal of Applied Statistics*, 20(5), 25–62.
- Gevaert, O., S. Van Vooren, and B. De Moor. 2007. A framework for elucidating regulatory networks based on prior information and expression data, *Ann N Y Acad Sci*, 1115, 240–248.
- Glanemann, C., A. Loos, N. Gorret, L. B. Willis, X. M. O'Brien, P. A. Lessard, and A. J. Sinskey. 2003. Disparity between changes in mrna abundance and enzyme activity in corynebacterium glutamicum: implications for dna microarray analysis, *Appl Microbiol Biotechnol*, 61(1), 61–68.
- Granata, A. and N. A. Quaderi. 2003. The opitz syndrome gene mid1 is essential for establishing asymmetric gene expression in hensens node, *Dev Biol*, 258(2), 397–405.

- Greenbaum, D., N. M. Luscombe, R. Jansen, J. Qian, and M. Gerstein. 2001. Interrelating different types of genomic data, from proteome to secretome: 'oming in on function, *Genome Res*, 11(9), 1463–1468.
- Griffin, T. J., S. P. Gygi, T. Ideker, B. Rist, J. Eng, L. Hood, and R. Aebersold. 2002. Complementary profiling of gene expression at the transcriptome and proteome levels in *saccharomyces cerevisiae*, *Mol Cell Proteomics*, 1(4), 323–333.
- Guo, Z., L. Wang, Y. Li, X. Gong, C. Yao, W. Ma, D. Wang, J. Zhu, M. Zhang, D. Yang, S. Rao, and J. Wang. 2007. Edge-based scoring and searching method for identifying condition-responsive protein-protein interaction sub-network, *Bioinformatics*, 23(16), 2121–2128.
- Gygi, S. P., Y. Rochon, B. R. Franza, and R. Aebersold. 1999. Correlation between protein and mrna abundance in yeast, *Mol Cell Biol*, 19(3), 1720–1730.
- Hahn, A., J. Rahnenfuhrer, P. Talwar, and T. Lengauer. 2005. Confirmation of human protein interaction data by human expression data, *BMC Bioinformatics*, 6, 112.
- Hakes, L., J. W. Pinney, D. L. Robertson, and S. C. Lovell. 2008. Protein-protein interaction networks and biology—what's the connection?, *Nat Biotechnol*, 26(1), 69–72.
- Heckerman, D. 1998. A tutorial on learning with bayesian networks, in *Learning in Graphical Models*, The MIT Press, 301–354.
- Heckerman, David, Dan Geiger, and David M. Chickering. 1995. Learning bayesian networks: The combination of knowledge and statistical data, *Machine Learning*, 20(3), 197–243.
- Hogenesch, J. B., W. K. Chan, V. H. Jackiw, R. C. Brown, Y. Z. Gu, M. Pray-Grant, G. H. Perdew, and C. A. Bradfield. 1997. Characterization of a subset of the basic-helix-loop-helix-pas superfamily that interacts with components of the dioxin signaling pathway, *J Biol Chem*, 272(13), 8581–8593.
- Huang da, W., B. T. Sherman, and R. A. Lempicki. 2009a. Bioinformatics enrichment tools: paths toward the comprehensive functional analysis of large gene lists, *Nucleic Acids Res*, 37(1), 1–13.
- Huang da, W., B. T. Sherman, and R. A. Lempicki. 2009b. Systematic and integrative analysis of large gene lists using david bioinformatics resources, *Nat Protoc*, 4(1), 44–57.

- Husmeier, D. 2003. Reverse engineering of genetic networks with bayesian networks, *Biochem Soc Trans*, 31(Pt 6), 1516–1518.
- Ideker, T., O. Ozier, B. Schwikowski, and A. F. Siegel. 2002. Discovering regulatory and signalling circuits in molecular interaction networks, *Bioinformatics*, 18 Suppl 1, S233–240.
- Imoto, S, T Higuchi, T Goto, K Tashiro, S Kuhara, and S Miyano. 2003a. Combining microarrays and biological knowledge for estimating gene networks via bayesian networks, in *Proceedings of the 2003 IEEE Bioinformatics Conference, 2003. CSB 2003*, 104–113.
- Imoto, S., S. Kim, T. Goto, S. Miyano, S. Aburatani, K. Tashiro, and S. Kuhara. 2003b. Bayesian network and nonparametric heteroscedastic regression for nonlinear modeling of genetic network, *J Bioinform Comput Biol*, 1(2), 231–252.
- Ingham, P. W. and A. P. McMahon. 2001. Hedgehog signaling in animal development: paradigms and principles, *Genes Dev*, 15(23), 3059–3087.
- Irizarry, R. A., B. Hobbs, F. Collin, Y. D. Beazer-Barclay, K. J. Antonellis, U. Scherf, and T. P. Speed. 2003. Exploration, normalization, and summaries of high density oligonucleotide array probe level data, *Biostatistics*, 4(2), 249–264.
- Irwin, S., M. Vandelft, D. Pinchev, J. L. Howell, J. Graczyk, H. T. Orr, and R. Truant. 2005. Rna association and nucleocytoplasmic shuttling by ataxin-1, *J Cell Sci*, 118(Pt 1), 233–242.
- Iwatsuki, K., H. X. Liu, A. Gronder, M. A. Singer, T. F. Lane, R. Grosschedl, C. M. Mistretta, and R. F. Margolskee. 2007. Wnt signaling interacts with shh to regulate taste papilla development, *Proc Natl Acad Sci U S A*, 104(7), 2253–2258.
- Jansen, R., D. Greenbaum, and M. Gerstein. 2002. Relating whole-genome expression data with protein-protein interactions, *Genome Res*, 12(1), 37–46.
- Jansen, R., H. Yu, D. Greenbaum, Y. Kluger, N. J. Krogan, S. Chung, A. Emili, M. Snyder, J. F. Greenblatt, and M. Gerstein. 2003. A bayesian networks approach for predicting protein-protein interactions from genomic data, *Science*, 302(5644), 449–453.
- Jaynes, E. T. 1957. Information theory and statistical mechanics. ii, *Physical Review*, 108(2), 171.
- Jenkins, D. 2009. Hedgehog signalling: emerging evidence for non-canonical pathways, *Cell Signal*, 21(7), 1023–1034.

- Jeong, J., J. Mao, T. Tenzen, A. H. Kottmann, and A. P. McMahon. 2004. Hedgehog signaling in the neural crest cells regulates the patterning and growth of facial primordia, *Genes Dev*, 18(8), 937–951.
- Jiang, N., L. J. Leach, X. Hu, E. Potokina, T. Jia, A. Druka, R. Waugh, M. J. Kearsey, and Z. W. Luo. 2008. Methods for evaluating gene expression from affymetrix microarray datasets, *BMC Bioinformatics*, 9, 284.
- Kalluri, R. 2009. Emt: when epithelial cells decide to become mesenchymal-like cells, *J Clin Invest*, 119(6), 1417–1419.
- Kannemeier, C., N. Al-Fakhri, K. T. Preissner, and S. M. Kanse. 2004. Factor vii-activating protease (fsap) inhibits growth factor-mediated cell proliferation and migration of vascular smooth muscle cells, *FASEB J*, 18(6), 728–730.
- Katoh, Y. and M. Katoh. 2009. Integrative genomic analyses on gli1: positive regulation of gli1 by hedgehog-gli, tgfbeta-smads, and rtk-pi3k-akt signals, and negative regulation of gli1 by notch-csl-hes/hey, and gpcr-gs-pka signals, *Int J Oncol*, 35(1), 187–192.
- Keshamouni, V. G., P. Jagtap, G. Michailidis, J. R. Strahler, R. Kuick, A. K. Reka, P. Pappoulas, R. Krishnapuram, A. Srirangam, T. J. Standiford, P. C. Andrews, and G. S. Omenn. 2009. Temporal quantitative proteomics by itraq 2d-lc-ms/ms and corresponding mrna expression analysis identify post-transcriptional modulation of actin-cytoskeleton regulators during tgfbeta-induced epithelial-mesenchymal transition, *J Proteome Res*, 8(1), 35–47.
- Keshamouni, V. G., G. Michailidis, C. S. Grasso, S. Anthwal, J. R. Strahler, A. Walker, D. A. Arenberg, R. C. Reddy, S. Akulapalli, V. J. Thannickal, T. J. Standiford, P. C. Andrews, and G. S. Omenn. 2006. Differential protein expression profiling by itraq-2dlc-ms/ms of lung cancer cells undergoing epithelial-mesenchymal transition reveals a migratory/invasive phenotype, *J Proteome Res*, 5(5), 1143–1154.
- Keshamouni, V. G. and W. P. Schieman. 2009. Epithelial-mesenchymal transition in tumor metastasis: a method to the madness, *Future Oncol*, 5(8), 1109–1111.
- Keshava Prasad, T. S., R. Goel, K. Kandasamy, S. Keerthikumar, S. Kumar, S. Mathivanan, D. Telikicherla, R. Raju, B. Shafreen, A. Venugopal, L. Balakrishnan, A. Marimuthu, S. Banerjee, D. S. Somanathan, A. Sebastian, S. Rani, S. Ray, C. J. Harrys Kishore, S. Kanth, M. Ahmed, M. K. Kashyap, R. Mohmood, Y. L. Ramachandra, V. Krishna, B. A. Rahiman, S. Mohan, P. Ranganathan, S. Ramabadran, R. Chaerkady, and A. Pandey.

2009. Human protein reference database—2009 update, *Nucleic Acids Res*, 37(Database issue), D767–772.
- Kim, J. E., R. W. Park, J. Y. Choi, Y. C. Bae, K. S. Kim, C. K. Joo, and I. S. Kim. 2002. Molecular properties of wild-type and mutant betaig-h3 proteins, *Invest Ophthalmol Vis Sci*, 43(3), 656–661.
- Kim, M. K. and C. Niyibizi. 2001. Interaction of tgf-beta1 and rhbmp-2 on human bone marrow stromal cells cultured in collagen gel matrix, *Yonsei Med J*, 42(3), 338–344.
- Kirkpatrick, S., Jr. Gelatt, C. D., and M. P. Vecchi. 1983. Optimization by simulated annealing, *Science*, 220(4598), 671–680.
- Kondo, T., S. A. Johnson, M. C. Yoder, R. Romand, and E. Hashino. 2005. Sonic hedgehog and retinoic acid synergistically promote sensory fate specification from bone marrow-derived pluripotent stem cells, *Proc Natl Acad Sci U S A*, 102(13), 4789–4794.
- Korb, Kevin and Ann Nicholson. 2003. *Bayesian Artificial Intelligence*, Chapman and Hall/CRC.
- Lam, Y. C., A. B. Bowman, P. Jafar-Nejad, J. Lim, R. Richman, J. D. Fryer, E. D. Hyun, L. A. Duvick, H. T. Orr, J. Botas, and H. Y. Zoghbi. 2006. Ataxin-1 interacts with the repressor capicua in its native complex to cause sca1 neuropathology, *Cell*, 127(7), 1335–1347.
- Laskey, Kathryn B. 2006. Mebn: A logic for open-world probabilistic reasoning, Tech. rep., George Mason University.
- Lauritzen, S. L. and D. J. Spiegelhalter. 1988. Local computations with probabilities on graphical structures and their application to expert systems, *Journal of the Royal Statistical Society. Series B (Methodological)*, 50(2).
- Lim, J., T. Hao, C. Shaw, A. J. Patel, G. Szabo, J. F. Rual, C. J. Fisk, N. Li, A. Smolyar, D. E. Hill, A. L. Barabasi, M. Vidal, and H. Y. Zoghbi. 2006. A protein-protein interaction network for human inherited ataxias and disorders of purkinje cell degeneration, *Cell*, 125(4), 801–814.
- Lin, X., M. Liang, and X. H. Feng. 2000. Smurf2 is a ubiquitin e3 ligase mediating proteasome-dependent degradation of smad2 in transforming growth factor-beta signaling, *J Biol Chem*, 275(47), 36818–36822.

- Liu, C. C., W. S. Chen, C. C. Lin, H. C. Liu, H. Y. Chen, P. C. Yang, P. C. Chang, and J. J. Chen. 2006. Topology-based cancer classification and related pathway mining using microarray data, *Nucleic Acids Res*, 34(14), 4069–4080.
- Liu, M., A. Liberzon, S. W. Kong, W. R. Lai, P. J. Park, I. S. Kohane, and S. Kasif. 2007. Network-based analysis of affected biological processes in type 2 diabetes models, *PLoS Genet*, 3(6), e96.
- Liu, X. Q. and T. Pawson. 1991. The epidermal growth factor receptor phosphorylates gtpase-activating protein (gap) at tyr-460, adjacent to the gap sh2 domains, *Mol Cell Biol*, 11(5), 2511–2516.
- Maden, M., D. E. Ong, D. Summerbell, and F. Chytil. 1988. Spatial distribution of cellular protein binding to retinoic acid in the chick limb bud, *Nature*, 335(6192), 733–735.
- Margaritis, D. 2003. *Learning Bayesian Network Model Structure from Data.*, Ph.D. thesis.
- Matilla, A., B. T. Koshy, C. J. Cummings, T. Isobe, H. T. Orr, and H. Y. Zoghbi. 1997. The cerebellar leucine-rich acidic nuclear protein interacts with ataxin-1, *Nature*, 389(6654), 974–978.
- McGlenn, E., K. L. van Bueren, S. Fiorenza, R. Mo, A. M. Poh, A. Forrest, M. B. Soares, F. Bonaldo Mde, S. Grimmond, C. C. Hui, B. Wainwright, and C. Wicking. 2005. Pax9 and jagged1 act downstream of gli3 in vertebrate limb development, *Mech Dev*, 122(11), 1218–1233.
- McMahon, A. P. 2000. More surprises in the hedgehog signaling pathway, *Cell*, 100(2), 185–188.
- Michaud, E. J. and B. K. Yoder. 2006. The primary cilium in cell signaling and cancer, *Cancer Res*, 66(13), 6463–6467.
- Mlecnik, B., M. Scheideler, H. Hackl, J. Hartler, F. Sanchez-Cabo, and Z. Trajanoski. 2005. Pathwayexplorer: web service for visualizing high-throughput expression data on biological pathways, *Nucleic Acids Res*, 33(Web Server issue), W633–637.
- Moresco, J. J., P. C. Carvalho, and 3rd Yates, J. R. 2010. Identifying components of protein complexes in *c. elegans* using co-immunoprecipitation and mass spectrometry, *J Proteomics*, 73(11), 2198–2204.



- Moskalenko, S., C. Tong, C. Rosse, G. Mirey, E. Formstecher, L. Daviet, J. Camonis, and M. A. White. 2003. Ral gtpases regulate exocyst assembly through dual subunit interactions, *J Biol Chem*, 278(51), 51743–51748.
- Motoyama, J., T. Takabatake, K. Takeshima, and C. Hui. 1998. Ptch2, a second mouse patched gene is co-expressed with sonic hedgehog, *Nat Genet*, 18(2), 104–106.
- Mu, D., S. Cambier, L. Fjellbirkeland, J. L. Baron, J. S. Munger, H. Kawakatsu, D. Sheppard, V. C. Broaddus, and S. L. Nishimura. 2002. The integrin alpha(v)beta8 mediates epithelial homeostasis through mt1-mmp-dependent activation of tgf-beta1, *J Cell Biol*, 157(3), 493–507.
- Nacu, S., R. Critchley-Thorne, P. Lee, and S. Holmes. 2007. Gene expression network analysis and applications to immunology, *Bioinformatics*, 23(7), 850–858.
- Nakagawa, N., M. Hoshijima, M. Oyasu, N. Saito, K. Tanizawa, and S. Kuroda. 2000. Enh, containing pdz and lim domains, heart/skeletal muscle-specific protein, associates with cytoskeletal proteins through the pdz domain, *Biochem Biophys Res Commun*, 272(2), 505–512.
- Needham, C. J., J. R. Bradford, A. J. Bulpitt, and D. R. Westhead. 2007. A primer on learning in bayesian networks for computational biology, *PLoS Comput Biol*, 3(8), e129.
- Noble, C. L., A. R. Abbas, J. Cornelius, C. W. Lees, G. T. Ho, K. Toy, Z. Modrusan, N. Pal, F. Zhong, S. Chalasani, H. Clark, I. D. Arnott, I. D. Penman, J. Satsangi, and L. Diehl. 2008. Regional variation in gene expression in the healthy colon is dysregulated in ulcerative colitis, *Gut*, 57(10), 1398–1405.
- Ohkubo, Y., C. Chiang, and J. L. Rubenstein. 2002. Coordinate regulation and synergistic actions of bmp4, shh and fgf8 in the rostral prosencephalon regulate morphogenesis of the telencephalic and optic vesicles, *Neuroscience*, 111(1), 1–17.
- Ohta, Y., N. Suzuki, S. Nakamura, J. H. Hartwig, and T. P. Stossel. 1999. The small gtpase rala targets filamin to induce filopodia, *Proc Natl Acad Sci U S A*, 96(5), 2122–2128.
- Okazawa, H., T. Rich, A. Chang, X. Lin, M. Waragai, M. Kajikawa, Y. Enokido, A. Komuro, S. Kato, M. Shibata, H. Hatanaka, M. M. Mouradian, M. Sudol, and I. Kanazawa. 2002. Interaction between mutant ataxin-1 and pqbp-1 affects transcription and cell death, *Neuron*, 34(5), 701–713.

- Oliphant, T. E. 2007. Python for scientific computing, *Computing in Science & Engineering*, 10–20.
- Parikh, J. R., B. Klinger, Y. Xia, J. A. Marto, and N. Bluthgen. 2010. Discovering causal signaling pathways through gene-expression patterns, *Nucleic Acids Res*, 38 Suppl, W109–117.
- Pasca di Magliano, M., S. Sekine, A. Ermilov, J. Ferris, A. A. Dlugosz, and M. Hebrok. 2006. Hedgehog/ras interactions regulate early stages of pancreatic cancer, *Genes Dev*, 20(22), 3161–3173.
- Pearl, Judea. 1997. *Probabilistic reasoning in intelligent systems: networks of plausible inference*, Morgan Kaufmann.
- Pe'er, D., A. Regev, G. Elidan, and N. Friedman. 2001. Inferring subnetworks from perturbed expression profiles, *Bioinformatics*, 1(1), 1–9.
- Qiu, Y. Q., S. Zhang, X. S. Zhang, and L. Chen. 2009. Identifying differentially expressed pathways via a mixed integer linear programming model, *IET Syst Biol*, 3(6), 475–486.
- Racunas, S. A., N. H. Shah, I. Albert, and N. V. Fedoroff. 2004. Hybrow: a prototype system for computer-aided hypothesis evaluation, *Bioinformatics*, 20 Suppl 1, i257–264.
- Rajagopalan, D. and P. Agarwal. 2005. Inferring pathways from gene lists using a literature-derived network of biological relationships, *Bioinformatics*, 21(6), 788–793.
- Riddle, R. D., R. L. Johnson, E. Laufer, and C. Tabin. 1993. Sonic hedgehog mediates the polarizing activity of the zpa, *Cell*, 75(7), 1401–1416.
- Robinson, R. 1977. Counting unlabeled acyclic digraphs, in *Combinatorial Mathematics V*, Springer Berlin / Heidelberg, vol. 622 of *Lecture Notes in Mathematics*, 28–43.
- Rubin, L. L. and F. J. de Sauvage. 2006. Targeting the hedgehog pathway in cancer, *Nat Rev Drug Discov*, 5(12), 1026–1033.
- Ruiz i Altaba, A., C. Mas, and B. Stecca. 2007. The gli code: an information nexus regulating cell fate, stemness and cancer, *Trends Cell Biol*, 17(9), 438–447.
- Sachs, K., D. Gifford, T. Jaakkola, P. Sorger, and D. Lauffenburger. 2002. Bayesian network approach to cell signaling pathway modeling, *Science's STKE*.

- Sachs, K., O. Perez, D. Pe'er, D. A. Lauffenburger, and G. P. Nolan. 2005. Causal protein-signaling networks derived from multiparameter single-cell data, *Science*, 308(5721), 523–529.
- Sanchez, C., C. Lachaize, F. Janody, B. Bellon, L. Roder, J. Euzenat, F. Rechenmann, and B. Jacq. 1999. Grasping at molecular interactions and genetic networks in drosophila melanogaster using flynets, an internet database, *Nucleic Acids Res*, 27(1), 89–94.
- Sartor, M. A., V. Mahavisno, V. G. Keshamouni, J. Cavalcoli, Z. Wright, A. Karnovsky, R. Kuick, H. V. Jagadish, B. Mirel, T. Weymouth, B. Athey, and G. S. Omenn. 2010. Conceptgen: a gene set enrichment and gene set relation mapping tool, *Bioinformatics*, 26(4), 456–463.
- Sauer, U., M. Heinemann, and N. Zamboni. 2007. Genetics. getting closer to the whole picture, *Science*, 316(5824), 550–551.
- Serra, H. G., L. Duvick, T. Zu, K. Carlson, S. Stevens, N. Jorgensen, A. Lysholm, E. Burright, H. Y. Zoghbi, H. B. Clark, J. M. Andresen, and H. T. Orr. 2006. Roralpha-mediated purkinje cell development determines disease severity in adult sca1 mice, *Cell*, 127(4), 697–708.
- Serth, J., W. Weber, M. Frech, A. Wittinghofer, and A. Pingoud. 1992. Binding of the h-ras p21 gtpase activating protein by the activated epidermal growth factor receptor leads to inhibition of the p21 gtpase activity in vitro, *Biochemistry*, 31(28), 6361–6365.
- Shah, A., T. Tenzen, A. P. McMahon, and P. J. Woolf. 2009. Using mechanistic bayesian networks to identify downstream targets of the sonic hedgehog pathway, *BMC Bioinformatics*, 10, 433.
- Shah, Abhik and Peter Woolf. 2009. Python environment for bayesian learning: Inferring the structure of bayesian networks from knowledge and data, *J. Mach. Learn. Res.*, 10, 159–162.
- Shannon, P., A. Markiel, O. Ozier, N. S. Baliga, J. T. Wang, D. Ramage, N. Amin, B. Schwikowski, and T. Ideker. 2003. Cytoscape: a software environment for integrated models of biomolecular interaction networks, *Genome Res*, 13(11), 2498–2504.
- Sheng, T., S. Chi, X. Zhang, and J. Xie. 2006. Regulation of gli1 localization by the camp/protein kinase a signaling axis through a site near the nuclear localization signal, *J Biol Chem*, 281(1), 9–12.

- Shi, Ting, Tapati Mazumdar, Jennifer DeVecchio, Zhong-Hui Duan, Akwasi Agyeman, Mohammad Aziz, and Janet A. Houghton. 2010. cDNA microarray gene expression profiling of hedgehog signaling pathway inhibition in human colon cancer cells, *PLoS ONE*, 5(10), e13054.
- Shimba, S., K. Komiyama, I. Moro, and M. Tezuka. 2002. Overexpression of the aryl hydrocarbon receptor (ahr) accelerates the cell proliferation of a549 cells, *J Biochem*, 132(5), 795–802.
- Shrager, J., R. Waldinger, M. Stickel, and J. P. Massar. 2007. Deductive biocomputing, *PLoS ONE*, 2(4), e339.
- Smith, V. A., E. D. Jarvis, and A. J. Hartemink. 2002. Evaluating functional network inference using simulations of complex biological systems, *Bioinformatics*, 18 Suppl 1, S216–224.
- Stecca, B. and I. Altaba A. Ruiz. 2010. Context-dependent regulation of the gli code in cancer by hedgehog and non-hedgehog signals, *J Mol Cell Biol*, 2(2), 84–95.
- Subramanian, A., P. Tamayo, V. K. Mootha, S. Mukherjee, B. L. Ebert, M. A. Gillette, A. Paulovich, S. L. Pomeroy, T. R. Golub, E. S. Lander, and J. P. Mesirov. 2005. Gene set enrichment analysis: a knowledge-based approach for interpreting genome-wide expression profiles, *Proc Natl Acad Sci U S A*, 102(43), 15545–15550.
- Sugihara, K., S. Asano, K. Tanaka, A. Iwamatsu, K. Okawa, and Y. Ohta. 2002. The exocyst complex binds the small GTPase RalA to mediate filopodia formation, *Nat Cell Biol*, 4(1), 73–78.
- Szczurek, E., I. Gat-Viks, J. Tiuryn, and M. Vingron. 2009. Elucidating regulatory mechanisms downstream of a signaling pathway using informative experiments, *Mol Syst Biol*, 5, 287.
- Tarcea, V. G., T. Weymouth, A. Ade, A. Bookvich, J. Gao, V. Mahavisno, Z. Wright, A. Chapman, M. Jayapandian, A. Ozgur, Y. Tian, J. Cavalcoli, B. Mirel, J. Patel, D. Radev, B. Athey, D. States, and H. V. Jagadish. 2009. Michigan molecular interactions r2: from interacting proteins to pathways, *Nucleic Acids Res*, 37(Database issue), D642–646.
- Tenzen, T., B. L. Allen, F. Cole, J. S. Kang, R. S. Krauss, and A. P. McMahon. 2006. The cell surface membrane proteins Cdo and Boc are components and targets of the hedgehog signaling pathway and feedback network in mice, *Dev Cell*, 10(5), 647–656.

- Tian, L., S. A. Greenberg, S. W. Kong, J. Altschuler, I. S. Kohane, and P. J. Park. 2005. Discovering statistically significant pathways in expression profiling studies, *Proc Natl Acad Sci U S A*, 102(38), 13544–13549.
- Tian, Q., S. B. Stepaniants, M. Mao, L. Weng, M. C. Feetham, M. J. Doyle, E. C. Yi, H. Dai, V. Thorsson, J. Eng, D. Goodlett, J. P. Berger, B. Gunter, P. S. Linseley, R. B. Stoughton, R. Aebersold, S. J. Collins, W. A. Hanlon, and L. E. Hood. 2004. Integrated genomic and proteomic analyses of gene expression in mammalian cells, *Mol Cell Proteomics*, 3(10), 960–969.
- Toyoshima, H. and T. Hunter. 1994. p27, a novel inhibitor of g1 cyclin-cdk protein kinase activity, is related to p21, *Cell*, 78(1), 67–74.
- Tsai, C. C., H. Y. Kao, A. Mizutani, E. Banayo, H. Rajan, M. McKeown, and R. M. Evans. 2004. Ataxin 1, a scal neurodegenerative disorder protein, is functionally linked to the silencing mediator of retinoid and thyroid hormone receptors, *Proc Natl Acad Sci U S A*, 101(12), 4047–4052.
- Tsamardinos, I., C. Aliferis, and A. Statnikov. 2003. Algorithms for large scale markov blanket discovery.
- Tsamardinos, Ioannis, Laura Brown, and Constantin Aliferis. 2006. The max-min hill-climbing bayesian network structure learning algorithm, *Machine Learning*, 65, 31–78.
- Tsuda, H., H. Jafar-Nejad, A. J. Patel, Y. Sun, H. K. Chen, M. F. Rose, K. J. Venken, J. Botas, H. T. Orr, H. J. Bellen, and H. Y. Zoghbi. 2005. The axh domain of ataxin-1 mediates neurodegeneration through its interaction with gfi-1/senseless proteins, *Cell*, 122(4), 633–644.
- Ulitsky, I. and R. Shamir. 2007. Identification of functional modules using network topology and high-throughput data, *BMC Syst Biol*, 1, 8.
- Ulitsky, Igor, Richard M. Karp, and Ron Shamir. 2008. Detecting disease-specific dysregulated pathways via analysis of clinical expression profiles.
- Velculescu, V. E., L. Zhang, W. Zhou, J. Vogelstein, M. A. Basrai, Jr. Bassett, D. E., P. Hieter, B. Vogelstein, and K. W. Kinzler. 1997. Characterization of the yeast transcriptome, *Cell*, 88(2), 243–251.
- Wachi, S., K. Yoneda, and R. Wu. 2005. Interactome-transcriptome analysis reveals the high centrality of genes differentially expressed in lung cancer tissues, *Bioinformatics*, 21(23), 4205–4208.

- Wall, D., Y. Wang, and V. Wallace. 2005. Interaction between the shh and notch signaling pathways in retinal development, *Invest. Ophthalmol. Vis. Sci.*, 46(5), 586.
- Weniger, M., J. C. Engelmann, and J. Schultz. 2007. Genome expression pathway analysis tool—analysis and visualization of microarray gene expression data under genomic, proteomic and metabolic context, *BMC Bioinformatics*, 8, 179.
- Werhli, A. V. and D. Husmeier. 2007. Reconstructing gene regulatory networks with bayesian networks by combining expression data with multiple sources of prior knowledge, *Stat Appl Genet Mol Biol*, 6, Article15.
- Woolf, P. J., W. Prudhomme, L. Daheron, G. Q. Daley, and D. A. Lauffenburger. 2005. Bayesian analysis of signaling networks governing embryonic stem cell fate decisions, *Bioinformatics*, 21(6), 741–753.
- Xu, J., B. P. Srinivas, S. Y. Tay, A. Mak, X. Yu, S. G. Lee, H. Yang, K. R. Govindarajan, B. Leong, G. Bourque, S. Mathavan, and S. Roy. 2006. Genomewide expression profiling in the zebrafish embryo identifies target genes regulated by hedgehog signaling during vertebrate development, *Genetics*, 174(2), 735–752.
- Xulvi-Brunet, R. and H. Li. 2010. Co-expression networks: graph properties and topological comparisons, *Bioinformatics*, 26(2), 205–214.
- Yamaguchi, R., M. Yamamoto, S. Imoto, M. Nagasaki, R. Yoshida, K. Tsuiji, A. Ishige, H. Asou, K. Watanabe, and S. Miyano. 2007. Identification of activated transcription factors from microarray gene expression data of kampo medicine-treated mice, *Genome Inform*, 18, 119–129.
- Yang, J. and R. A. Weinberg. 2008. Epithelial-mesenchymal transition: at the crossroads of development and tumor metastasis, *Dev Cell*, 14(6), 818–829.
- Yen, JY. 1970. An algorithm for finding shortest routes from all source nodes to a given destination in general network, *Quart. Appl. Math.*, 27, 526–530.
- Yin, M. J., L. Shao, D. Voehringer, T. Smeal, and B. Jallal. 2003. The serine/threonine kinase nek6 is required for cell cycle progression through mitosis, *J Biol Chem*, 278(52), 52454–52460.
- Yoo, C., V. Thorsson, and G. F. Cooper. 2002. Discovery of causal relationships in a gene-regulation pathway from a mixture of experimental and observational dna microarray data, *Pac Symp Biocomput*, 498–509.

- Yoon, J. W., R. Gilbertson, S. Iannaccone, P. Iannaccone, and D. Walterhouse. 2009. Defining a role for sonic hedgehog pathway activation in desmoplastic medulloblastoma by identifying gli1 target genes, *Int J Cancer*, 124(1), 109–119.
- Yoon, J. W., Y. Kita, D. J. Frank, R. R. Majewski, B. A. Konicek, M. A. Nobrega, H. Jacob, D. Walterhouse, and P. Iannaccone. 2002. Gene expression profiling leads to identification of gli1-binding elements in target genes and a role for multiple downstream pathways in gli1-induced cell transformation, *J Biol Chem*, 277(7), 5548–5555.
- Young, K. H. 1998. Yeast two-hybrid: so many interactions, (in) so little time, *Biol Reprod*, 58(2), 302–311.
- Yu, J., V. A. Smith, P. P. Wang, A. J. Hartemink, and E. D. Jarvis. 2004. Advances to bayesian network inference for generating causal networks from observational biological data, *Bioinformatics*, 20(18), 3594–3603.
- Yu, M., J. Gipp, J. W. Yoon, P. Iannaccone, D. Walterhouse, and W. Bushman. 2009. Sonic hedgehog-responsive genes in the fetal prostate, *J Biol Chem*, 284(9), 5620–5629.
- Yue, S., H. G. Serra, H. Y. Zoghbi, and H. T. Orr. 2001. The spinocerebellar ataxia type 1 protein, ataxin-1, has rna-binding activity that is inversely affected by the length of its polyglutamine tract, *Hum Mol Genet*, 10(1), 25–30.
- Zeisberg, M. and E. G. Neilson. 2009. Biomarkers for epithelial-mesenchymal transitions, *J Clin Invest*, 119(6), 1429–1437.
- Zhang, Q., X. Wang, and D. J. Wolgemuth. 1999. Developmentally regulated expression of cyclin d3 and its potential in vivo interacting proteins during murine gametogenesis, *Endocrinology*, 140(6), 2790–2800.
- Zhang, W., D. Bergamaschi, B. Jin, and X. Lu. 2005. Posttranslational modifications of p27kip1 determine its binding specificity to different cyclins and cyclin-dependent kinases in vivo, *Blood*, 105(9), 3691–3698.
- Zhao, P., J. Seo, Z. Wang, Y. Wang, B. Shneiderman, and E. P. Hoffman. 2003. In vivo filtering of in vitro expression data reveals myod targets, *C R Biol*, 326(10-11), 1049–1065.