21

Computational Methods for Learning Bayesian Networks from High-Throughput Biological Data

Bradley M. Broom

Department of Biostatistics and Applied Mathematics,
University of Texas M.D. Anderson Cancer Center,
1515 Holcombe Blvd, Houston, TX 77030
E-mail: bmbroom@mdanderson.org

Devika Subramanian

Department of Computer Science,
Rice University,
6100 Main St., Houston, TX 77005
E-mail: devika@rice.edu

Abstract

Data from high-throughput technologies, such as gene expression microarrays, promise to yield insight into the nature of the cellular processes that have been disrupted by disease, thus improving our understanding of the disease and hastening the discovery of effective new treatments. Most of the analysis thus far has focused on identifying differential measurements, which form the basis of biomarker discovery. However, merely listing differentially expressed genes or gene products is not sufficient to explain the molecular basis of disease. Consequently, there is increasing interest in extracting more information from available data in the form of biologically meaningful relationships between the quantities being measured. The holy grail of such techniques is the robust identification of causal models of disease from data.

The goal of this chapter is to survey computational learning methods that extract models of altered interactions that lead to and occur in the diseased state. Our focus is on methods that represent biological processes as Bayesian networks and that learn these networks from experimental measurements of cellular activity. Specifically, we will survey computational methods for learning Bayesian networks from high-throughput biological data.

21.1 Introduction

Many diseases, especially cancers, involve the disruption or deregulation of many cellular processes. It is hoped that high-throughput technologies, such as gene expression microarrays—which provide a snapshot of the level of gene transcription occurring in a cell, for many thousands of genes—will yield insight into the nature of the affected processes, improve our understanding of the disease, and hasten the discovery of effective new treatments.

However, merely identifying differential measurements is not enough. For instance, numerous studies [1, 26, 8, 16, 31, 10, 24, 5, 2, 20, 23, 25, 32] have identified hundreds of genes that are differentially expressed between tumor cells corresponding to prostate cancer of different grades and normal prostate cells. Yet our detailed understanding of the genetic and/or regulatory events that lead to the initiation and progression of prostate cancer remains incomplete. There is increasing evidence that disease progression in complex diseases, especially solid tumors, does not arise from an individual molecule or gene, but from complex interactions between a cell's numerous constituents and its environment [4].

Consequently, in addition to making inferences about the differential expression of individual measurements, such as gene and protein expression levels, there is increasing interest in learning the underlying relationships between the quantities being measured. For example, we might hope to obtain a network of dependencies between the differentially expressed genes. Such a network can help distinguish root causes from downstream effects of a cellular process disruption. The goal of this chapter is to survey computational learning methods for elucidating from high-throughput experimental measurements, insights into the nature of the altered interactions that lead to and occur in the diseased state.

Biological interaction networks (often called pathways) can be represented at several levels of abstraction ranging from network models which emphasize the fundamental components (genes and metabolic products) and connections between them (the L1 models as defined in [17]), to detailed differential equation models of the kinetics of specific reactions (the L2 models) [18, 17]. The choice of abstraction level is generally a function of the biological problem being addressed and the type and quantity of data available. For instance, models based on differential equations have been used for detailed modeling of specific molecular interactions when time series data for the concentrations of the various molecular components involved is available. Boolean networks approximate gene expression values as binary variables that are either on or off, and represent gene interactions as Boolean functions. Approaches based on Bayesian networks [12] and their generalizations allow representations of multiple valued discrete values as well as continuous quantities. Bayesian networks have a solid formal foundation in probability theory and naturally support reasoning about incomplete and noisy data. They have been used in a wide variety of models generated from gene expression data including [13, 28, 3, 29, 15, 27].

The following section introduces Bayesian networks and their use in modeling biological processes. Section 21.3 describes challenges that arise in learning Bayesian networks from high-throughput data. Section 21.4 presents methods for addressing these challenges. A complete example of structure learning from expression data is presented in Section 21.5. Section 21.6 concludes the chapter with a summary of the state-of-the art, as well as open questions in the area.

21.2 Bayesian Networks

Bayesian networks are a compact graphical representation of the joint probability distribution over a set of random variables, $X_1, \ldots, X_n$. A variable X_i can represent the mRNA expression level of a gene, or expression level of a protein, or the activity level of a signaling molecule. Typically, continuous expression levels are discretized into two or more categories; e.g., on/off for signaling activity and high/medium/low for enzyme levels, by the selection of appropriate thresholds. A Bayesian network specification has two components:

(i) a directed acyclic graph $G = (V, E)$ with a node set V corresponding to the random variables $X_1, \ldots, X_n$, and edge set E on these nodes. The edges reflect conditional independence assumptions made. A node is conditionally independent of all other nodes given its parents in the network.

(ii) a set θ of conditional probability distributions for each node in the graph G. These probability distributions are local, and are of the form $P(X_i|Parents_G(X_i))$.

The two components (G, θ) specify a unique distribution on the random variables $X_1, \ldots, X_n$.

$$P(X_1, \ldots, X_n) = \prod_{i=1}^{n} P(X_i | Parents_G(X_i))$$

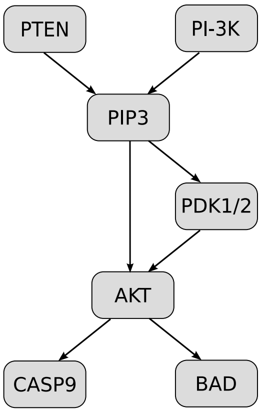
Thus, unlike purely qualitative network models, Bayesian networks contain quantitative information in the form of conditional probabilities of variables given their parents in the network.

As an example, consider the portion of the *PI3K/PTEN/AKT* signalling pathway implicated in androgen-independent prostatic adenocarcinoma [14]. *PI3* kinase generates the potent phospholipid *PIP3* in the absence of *PTEN. PIP3* is absent in quiescent cells, but is significantly upregulated following stimulation by growth and survival factors. PIP3 recruits *AKT* (a proto-oncoprotein), which is activated by two kinases: *PDK1* and *PDK2.* Once activated, *AKT* suppresses apoptosis by phosphorylating and inactivating the pro-apoptotic proteins *caspase-9* and *BAD.* A Bayesian network representation of this pathway is shown in Figure 21.1. The edges in the network topology mirror the causal mechanisms in this pathway, and the conditional probability distributions associated with the nodes reflect our understanding of how the individual components of the pathway work. The network topology makes a number of conditional independence assumptions explicit. For instance, the *AKT* levels are a function of its parents in the network: *PIP3* and *PDK12.* The influence of other nodes on the levels of *AKT* are mediated through their effects on the levels of its parents. Thus,

$$\begin{aligned} P(AKT = high|\ PIP3, PDKI2, PTEN, PI3K, CASP9, BAD) = \\ P(AKT = high|PIP3, PDK12) \end{aligned}$$

We say that *AKT* is conditionally independent of all other nodes in the network, given its parent nodes. The quantitative network parameters model the underlying processes. The first conditional probability table $Pr(PIP3 = high|PTEN, PI3K)$ represents a stochastic process which turns on the levels of *PIP3* when *PTEN* is underexpressed and there is plenty of PI3K available in the cell. The levels of *caspase-9* and *BAD* respond directly to the level of *AKT. AKT* is activated by high levels of *PIP3* and *PDK12.*

Bayesian networks are representations of the full joint distributions on their nodes. In our example network which has 7 boolean nodes, we would need $2^7 - 1 = 127$ parameters to fully specify the distribution. Our factored representation only requires 14 parameters. We can derive any probability of interest from this model. For example, using standard



PTEN	PI3K	P(PIP3=high)
low	low	0.1
low	high	0.95
high	low	0.05
high	high	0.07

PIP3	PDK*	P(AKT=high)
low	low	0.05
low	high	0.05
high	low	0.05
high	high	0.95

AKT	P(CASP9=high)	P(BAD=high)
low	0.05	0.05
high	0.95	0.95

Fig. 21.1. The PI3K/PTEN/AKT signalling pathway represented as a Bayesian network.

Bayesian network inference we can calculate that *BAD* levels are high when *PTEN* is suppressed and *PIP3* levels are high, and low otherwise. Note that having elevated levels of *PTEN* is not sufficient to raise levels of *BAD* unless *PIP3* is also present.

$$\begin{aligned} P(BAD = high|PTEN = low, PIP3 = high) &= 0.824 \\ P(BAD = high|PTEN = low, PIP3 = low) &= 0.090 \\ P(BAD = high|PTEN = high) &= 0.138 \end{aligned}$$

We now turn to the problem of learning such networks from available data on cellular activity.

21.3 Learning Bayesian networks

To learn a Bayesian network on variables $X_1, \ldots X_n$, we start with M measurements of these n variables in a data set

$$D = \{(X_1(1), \ldots X_n(1)), \ldots, (X_1(M), \ldots, X_n(M))\}$$

These measurements can be obtained from a variety of sources. One source is flow cytometry, where each measurement $(X_1(i), \ldots, X_n(i))$ is a set of n phosphorylated protein expression levels measured simultaneously in an individual cell. Flow cytometry easily yields thousands of data points ($M \approx 10000$). Using such data, Sachs et al. [27] automatically derived most of the traditionally described signaling relations among 11 ($n = 11$) signaling components in human immune system cells.

The data most commonly available at the current time are gene expression microarrays that simultaneously measure the level of mRNA transcription for tens of thousands of genes. For such data sets, $n \approx 12,500$, while the number of independent measurements M is very small; e.g., $M = 100$ would be a fairly large study. This chapter concentrates on the issues raised by this kind of data, but many of the principles involved apply to other data types also.

The problem of learning Bayesian networks from data has been studied extensively over the past decade [22, 13]. Most approaches define a hypothesis space of potential network models and use the data to find ones that are most likely given the data. Learning a network model on a set $X_1, \ldots, X_n$ of variables entails inferring the graph of dependencies between them, as well as the parameters θ consisting of the local conditional probabilities: $P(X_i|Parents_G(X_i))$. If the graph structure G is known, the parameters θ can be estimated from the available data D by maximum likelihood estimation if M is large, or by Bayesian estimation, if M is small and priors on the parameter vector θ are available. The likelihood of the data D given parameter vector θ for a known structure G is:

$$L(\theta; D) = P(D|\theta) = \prod_{i=1}^{M} P(X_1(i), \ldots, X_n(i))|\theta)$$

For maximum likelihood estimation, we estimate parameters ${\theta^*}_{ML}$ such that

$${\theta^*}_{ML} = argmax_{\theta} L(\theta; D)$$

When the number of samples M is small, the above approach tends to overfit the model parameters to the available data. Bayesian methods reduce overfitting by representing and using available knowledge about the parameters in the form of a prior distribution $P(\theta)$. For example, if we had knowledge about how levels of AKT impacted the levels of *caspase-9*, then we can generate a prior distribution such as $P(AKT = high|caspase\_9 = high) = 0.8$ and $P(AKT = high|caspase\_9 = low) = 0.2$. The data D then serves to update the prior $P(\theta)$ to yield the posterior probability distribution $P(\theta|D)$. By Bayes Rule,

$$P(\theta|D) = \frac{P(D|\theta)P(\theta)}{P(D)}$$

Then, the estimated parameter $\theta^*{}_{MAP}$ is:

$$\theta^*{}_{MAP} = argmax_\theta P(D|\theta)P(\theta)$$

Since $P(D)$ is independent of θ, it is treated as normalizing constant, and the scoring function is simply the product of the likelihood of the data given the parameter vector θ and the prior $P(\theta)$. When the prior distribution is a Dirichlet distribution, which is a conjugate prior, the posterior distribution $P(\theta|D)$ can be easily computed in closed form.

Learning the structure G of the Bayesian network from data is a very challenging problem. The most common approach to discovering the structure of Bayesian networks from data is to define a space of graph models to consider, and then set up a scoring function that evaluates how well a model explains the available data. Then, an optimization algorithm is used to search for the highest-scoring model. The scoring function is the logarithm of the posterior probability of the network structure given the data

$$Score(G; D) = logP(G|D) = logP(D|G) + logP(G)$$

where $P(D|G) = \int_\theta P(D|G,\theta)P(\theta|G)d\theta$. We average over all parameters θ associated with a graph structure G.

Learning a Bayesian network that provably maximizes the above scoring function is NP-hard [6]. Thus, learning optimal Bayesian networks for high-throughput datasets is computationally infeasible.

21.4 Algorithms for learning Bayesian networks

There are three main challenges in designing learning algorithms for Bayesian networks. First, the number of potential networks in n variables is super-exponential in n. The first challenge, therefore, is to choose an appropriate subset of variables to include in the model. The second challenge is to devise and use good approximation algorithms for guiding search toward biologically plausible solutions consistent with the data.

The second property that makes learning networks difficult is the very small number, M, of samples. Any error associated with each sample is significant, and could lead to erroneous network structures being learnt. Further, when the number of available samples is limited, the data is not sufficient to uniquely identify a structure. In fact, there may an exponential number of networks with the same score with respect to the available data. Enumerating them is itself infeasible. Extracting common structural properties of a set of high scoring networks, which is a form of model averaging, is usually employed for learning the graph structure of a domain. The third challenge is therefore to devise and use robust methodologies to identify particular network structures.

Another consequence of the small number of samples is that nodes can reliably have at most two or three parents. There is simply insufficient data from which to reliably learn the conditional probability distribution of any nodes with more parents. The natural representations for many biological networks would contain many variables with more parents, so this is a severe limitation that raises challenging issues for determining the biological meaning and validity of the computed networks.

The remainder of this section will address in detail each of these challenges.

21.4.1 Node Selection

In *ab initio* construction of gene regulatory networks, a starter set of differentially expressed genes obtained from a pre-processing phase (such as by clustering or correlational analyses followed by thresholding on p-values) is used [13, 15, 28]. A particular danger of the *ab initio* approach is that it may select several highly correlated variables. Because there are so few samples, this is inappropriate, since the learnt network will then more likely correspond to noise in the data than to a functional relationship between the genes involved. Nevertheless, while learning Bayesian networks we clearly expect some correlations, so how can we select an appropriate subset of genes from the thousands of potential candidates in a typical high-throughput experiment? One approach is to consider variables that occur on specific pathways of interest, as in [9]. Tools such as Cytoscape [30] and GenMAPP [7] make this selection easy to perform.

21.4.2 Computational Complexity

Since learning optimal Bayesian networks from data is NP-hard, most networks of practical interest are much too large for exact methods to be feasible. Thus considerable research is devoted to finding good approximations to optimal networks. Even so, many of these approximation methods are themselves computationally infeasible for problems involving several thousand genes.

21.4.2.1 Greedy Hill Climbing Algorithm

A standard algorithm for finding an approximately optimal network is to start with a candidate network (such as the network with no edges), and consider a set of potential modifications, such as the addition, removal, or reversal of an edge between nodes (subject to the acyclic constraint). The scoring function, $P(G|D)$, is evaluated for every modified network. The highest-scoring modified network becomes the current candidate, and the process is repeated until no modified network scores higher than the current candidate. A well-known limitation of this algorithm is that it may become trapped in a local maximum, so it is usual to restart the hill-climbing process a number of times from random permutations of the best candidate, and to keep the network that scores highest overall.

The major computational cost of this algorithm is evaluating the scoring function for each potential candidate network. However, the overall score is composed of the scores for each node given its parents in the network. Many of these node configurations will be shared by many of the networks considered, so the computation can be made more efficient by caching the scores for these nodes. Even so, computing the score for each node, given its parents, for all combinations of parents considered by the algorithm is the major computational expense.

21.4.2.2 Sparse Candidate Algorithm

To reduce the computational expense of learning an approximately optimal Bayesian network containing a few hundred variables, Friedman et al. [13] introduced the *sparse candidate algorithm*, in which the potential parents of a node in the network are initially limited to the k nodes with which it is most highly correlated. Any such nodes that do not appear as parents in the learned network are replaced by the next most highly correlated nodes, and the entire process is repeated. If k is much less than the number n of nodes, the total search space is reduced and, more importantly, far fewer parent configurations must be evaluated for each variable (at most 2^k instead of 2^n). This approximation algorithm was used in [12] to reconstruct portions of the yeast cell cycle from expression data obtained at different points in the cell cycle. Each measurement was treated as an independent sample, and an additional node representing the cell cycle phase was introduced as a mandatory root node in the network. Using no prior biological knowledge or constraints, the method identified several important subnetworks of interactions.

21.4.3 Identifying Robust Network Features

Although gene expression microarrays measure the expression levels of many thousands of genes simultaneously, a typical study includes at most a few hundred different samples, which is far too few to reliably reconstruct a unique network model. In fact, it is not unusual for an exponential number of different networks on a given set of variables to have very similar high scores! To circumvent this fundamental limitation on the amount of data needed to learn network structures with high confidence, it is often more appropriate to learn the probabilities of specific network features, such as edges. Specifically, the probability of a network feature f given the data D is obtained by summing the probabilities of all graphs in which the feature occurs: $\sum_G P(G|D)f(G)$ where $f(G)$ is one if the feature is present in graph G, or zero if not. Note that the resulting set of network features is not necessarily a Bayesian network; for instance, an edge between two commonly connected variables need not always, or even predominately, occur in the same direction.

A simple approach for estimating this probability is to learn a large number of approximately optimal, but different, networks from the data, and then count those network features (such as edges between variables) that are common to these high scoring networks.

Friedman and Koller [11] describe an efficient Bayesian approach for estimating the probability of network features across all high-scoring networks. They introduce a total order between nodes: only nodes that occur before a node can be parents of that node. They show that the probability of a network feature due to all graph structures consistent with a specific fixed order of variables can be computed efficiently. The total probability of each network feature is then computed by using MCMC to integrate over all possible orders. MCMC over the space of orders instead of directly over the space of Bayesian networks converges to the stationary distribution of the Markov chain much faster, since the space of orders is much smaller and much less peaked than the space of Bayesian networks. Friedman and Koller showed that MCMC over orders converges at least ten times faster than MCMC over Bayesian networks (some of which did not converge within the limits on the number of iterations). Koivisto and Sood [19] modify this approach by using an efficient exponential algorithm to sum over all possible orders, which is computationally feasible for networks of up to about 30 nodes.

A significant issue with both approaches is that the robust features identified are not necessarily biological in origin. As mentioned above, gene expression data typically contains many highly correlated genes. We "identified" significant gene interactions amongst a set of moderately correlated candidate genes by extracting those edges common to hundreds of high-scoring Bayesian networks learnt from the discretized gene expression data using the sparse candidate algorithm. However, when we modified the discretization method, the set of edges obtained changed drastically. Since many of the genes were reasonably correlated, we believe that many of the edges common to the high-scoring networks are merely artifacts of the discretization.

To overcome this problem it is not sufficient merely to require that the genes included in the network not occur in the same cluster, since typical clustering methods exclude all but the most highly correlated genes to reduce the number of false positives identified. Consequently, many of the excluded genes are still sufficiently correlated to cause problems. This is a significant issue for applying these methods to genetic pathways, since genes within the same (or even a closely related) pathway are expected to be highly correlated.

21.4.4 Incorporating Known Biological Information

A complementary approach for overcoming the problems created by the small number of samples is to incorporate known biological information into the network learning process, such as by incrementally adding additional genes into the network using existing knowledge about gene interactions. Segal et al. [29] have combined gene expression data and promoter sequence data to identify transcriptional modules in *Saccharomyces cerevisiae*. Bar-Joseph et al. [3] combined genome-wide location data with the gene-expression data to obtain insights into regulatory networks for the same organism.

Another pathway-centric line of work is exemplified by that of Mamitsuka and Okuno [21]. They observe that current metabolic interaction maps imply many possible metabolic pathways, only some of which are biologically active. By synthesizing genetic pathways from the interactions described in these metabolic interaction maps and evaluating their likelihood using existing protein class information and gene expression microarray data, they were able to identify specific biologically active pathways.

Koivisto and Sood [19] describe an extension of their feature estimation algorithm that allows biological information to constrain the possible variable orders, thus making exact structure discovery feasible for larger networks.

Sachs et al. [27] incorporated interventional data into their network learning algorithm. By directly perturbing the phosphoralation states of measured molecules, they were able to infer more strongly whether one molecule was upstream, downstream, or neither of other molecules with which it was correlated.

21.4.5 Biological Relevance and Validation

It is tempting to think that Bayesian networks can naturally represent the modularity found in biological pathways, with the parent-child relationship implying causality. For instance, Sachs et al. [27] used the directionality of parent-child relationships to encode event cascades in signaling networks. A common misgiving is that feedback loops in pathways cannot be represented by acyclic Bayesian networks, but a temporal extension called dynamic Bayesian networks [33] enables the loop to be represented by unrolling it over time.

In practice, however, the interpretation of a network structure derived by computational methods is not trivial. The parent-child relation in a derived network need not be causal. Even if causal, an edge between variables in a Bayesian network does not imply a direct biological mechanism. There may be a number of intermediate variables between the linked variables that have not been included in the network being modeled. Studies that recreate a known network, perhaps with some new interactions, from high-throughput data, are easily interpretable. It remains a challenge to understand the biological significance of a network learnt *de novo* from biological data, especially gene expression data for which the only realistic prospect is to learn network features.

21.5 Example: Learning Robust Features from Data

To illustrate the learning of robust network features from data, we will learn the likely links in a Bayesian network for a subset of the publicly available prostate cancer data obtained by Singh et al. [31]. This data set consists of 102 Affymetrix U95Av2 gene expression arrays from prostate samples (50 normals and 52 tumors). The Affymetrix CEL files were processed using Bioconductor to obtain numeric gene expression values for over 12000 genes.

All gene expression values were individually discretized into three values (low, medium, and high) by determining the two cutoff points that would maximize a weighted average of the gene's self-information (I_s) and its mutual information (I_m) with the sample classification, specifically $0.425I_s + I_m$. The weight of 0.425 was chosen empirically because it appears to balance the tendency to create small discretization ranges that are probably overfitting the data, with the creation of ranges with a uniform number of members.

From this database of discretized genes, we selected probesets indicative of 13 genes known to be associated with Glutathione metabolism. Reduced Glutathione (GSH) is an important cellular tripeptide that plays a vital role in the degradation of toxic cellular compounds. GSTM5, GSTP1, GGTL4, GGTLA1 catalyze the conjugation of GSH with toxic cellular compounds and its subsequent decomposition. Reformation of reduced glutathione from its oxidized form involves GSR, GPX4, IDH1, and IDH2, while GSS, GCLC, and GCLM are responsible for the de novo synthesis of GSH from its three constituent peptides. The de novo synthesis rate is controlled by GCLC, whose expression is determined by transcription factors KEAP1 and NFE2L2.

We learnt 500 different high-scoring Bayesian networks from this discretized data using the sparse candidate algorithm with a limit of at most three parents per node. Duplicate networks were excluded by storing generated networks in a database, and excluding previously encountered networks from the search space. The scores of the high-scoring networks were recorded and checked to verify there was no trend.

The network shown in figure 21.2 was obtained by recording an edge between nodes if there was an edge between those nodes—in either direction—in at least 60% of the high-scoring networks. The most frequently occuring edges occur in more then 99% of the networks and are thick black, and get progressively lighter with reduced frequency (90% black, 80% dark grey, 70% medium grey, 60% light grey). Edges between positively correlated nodes are terminated with a solid arrow if the edge occurs in the same direction at least 90% of the time, with an open arrow if the edge occurs in the same direction between 60% and 90% of the time, or nothing if neither direction is dominant. Edges between inversely correlated nodes are terminated with a solid box (at least 90% directionality), an open box (between 60% and 90%), or a bar (less than 60%). Edges between nodes without a simple linear correlation are terminated in a solid star (and all occur in the same direction at least 90% of the time).

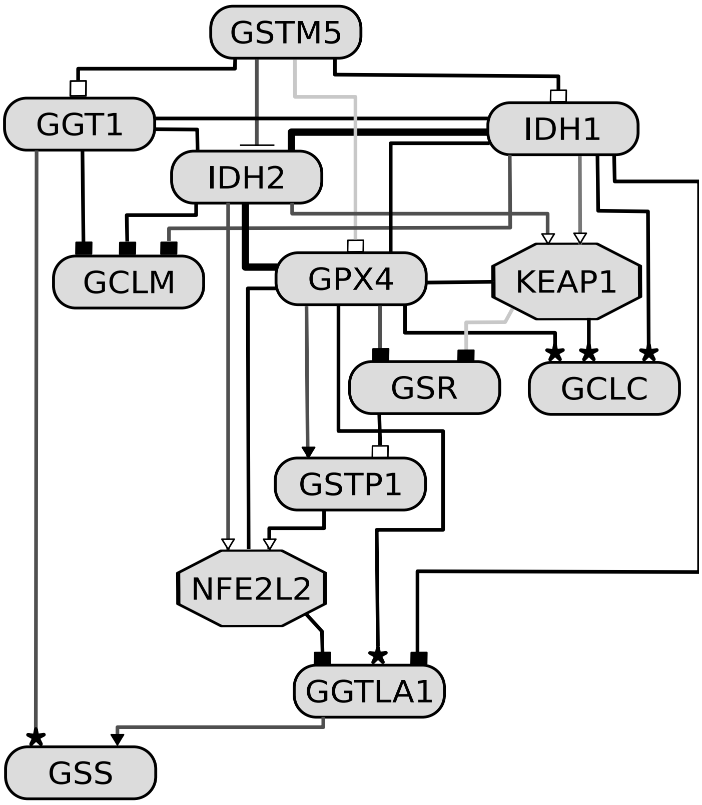


Fig. 21.2. Frequent edge network between thirteen genes involved in Glutathione synthesis, reclamation, and conjugation. Edges denote commonly occurring edges in 500 high-scoring Bayesian networks learnt from the data, with edge color corresponding to edge frequency, and edge termination denoting positive, negative, or non-monotonic correlations, as detailed in the text.

Deriving a biological explanation of these links is challenging. However, by employing what is known about the biological interactions between these genes and their products, we can interpret the significance of the edges that occur in this network, and thereby gain an understanding of the types of biological inferences that could be made from networks of this kind.

From this network, it is apparent that the levels of IDH1, IDH2, and GPX4, which are all involved in the conversion of oxidized Glutathione to GSH, are highly correlated. The absence of arrows on the edges between these nodes suggests that this data is insufficient to separate the nodes in this clique. Node GGT1 is also closely correlated with IDH1 and IDH2.

Interestingly, GSR, which catalyzes conversion of oxidized Glutathione to GSH, is inversely correlated with GPX4 (and IDH1, IDH2). The two genes that mediate conjugation of GSH with toxins, GSTP1 and GSTM5, occur in different parts of this network, with GSTP1 correlated to GPX4 and GSTM5 inversely correlated with the whole clique of IDH1, IDH2, GPX4, and GGT1.

In the de-novo synthesis of GSH, the final, but not rate limiting, step is catalyzed by GSS, whose expression depends most closely on that of GGT1 and GGTLA1, which both occur in the downstream degradation of GSH after its conjugation with toxins.

The rate limiting step in the de-novo synthesis of GSH is controlled by a dimer of the GCLC and GCLM proteins, with GCLC expression controlling the rate. GCLC is expressed when the NFE2L2 protein migrates from the cytoplasm to the nucleus, and this migration is triggered by KEAP1 expression. Thus, KEAP1 expression is the effective regulator of GCLC expression, and so it is reasonable that the figure shows KEAP1 and not NFE2L2 as a parent of GCLC. The edges from the parents of GCLC to GCLC terminate in a star because GCLC expression is highest for medium levels of each parent. High-levels of KEAP1 also correlate with reduced conversion of oxidized Glutathione into GSH via GSR. Consequently, for the highest-levels of KEAP1 expression, the levels of GSH would be low since it is being neither synthesized nor reclaimed.

As exemplified by the relations between KEAP1, NFE2L2, and GCLC, the edges in this diagram do not (necessarily) represent direct biological interactions at the protein-protein, protein-DNA, or DNA-DNA levels, but represent a higher level of effective interactions. For instance, expression of GSS, a key gene in GSH synthesis, is highly correlated with its downstream degradation by GGT1 and GGTLA1, even though there is no direct biological interaction. Thus, interaction networks learnt via Bayesian networks show promise for modeling the effective high-level mechanisms that control the underlying molecular interaction pathways.

21.6 Conclusion

This chapter has surveyed computational learning methods for elucidating Bayesian network models, or at least robust features of such models, from high-throughput experimental measurements, such gene expression data. The challenges imposed by the large number of variables but the small number of sample points were described, and a variety of computational strategies for addressing these challenges were outlined. To date, Bayesian networks have been successfully inferred for microarray data from yeast and for flow cytometry data from human immune system cells, but not for gene expression data from mammalian or oncological sources. Computational inference of Bayesian network structures from high-throughput data is difficult, but new computational methods are making it feasible to automatically deduce robust interactions between variables. The application of these methods to high-throughput biological data sets will help us to understand the nature of the altered biological interactions that lead to and occur in many diseases.

Bibliography

[1] C. Abate-Shen and M. M. Shen, *Molecular genetics of prostate cancer*, Genes Dev., 14 (2000), pp. 2410–2434.

[2] V. J. Assikis, K. A. Do, S. Wen, X. Wang, J. H. Cho-Vega, S. Brisbay, R. Lopez, C. J. Logothetis, P. Troncoso, C. N. Papandreou, and T. J. McDonnell, *Clinical and biomarker correlates of androgen-independent, locally aggressive prostate cancer with limited metastatic potential*, Clin. Cancer Res., (2004).

[3] Z. Bar-Joseph, G. K. Gerber, T. I. Lee, N. J. Rinaldi, J. Y. Yoo, F. Robert, D. B. Gordon, E. Fraenkel, T. S. Jaakkola, R. A. Young, and D. K. Gifford, *Computational discovery of gene modules and regulatory networks*, Nature Biotechnology, 21 (2003), pp. 1337–1342.

[4] A.-L. Barabasi and Z. N. Oltvai, *Network biology: understanding the cell's functional organization*, Nature Reviews Genetics, 5 (2004), pp. 101–112.

[5] J. H. Cho-Vega, P. Troncoso, K. A. Do, C. Rago, X. Wang, S. Tsavachidis, L. J. Medeiros, K. Spurgers, C. Logothetis, and T. J. McDonnell, *Combined laser capture microdissection and serial analysis of gene expression from human tissue samples*, Mod. Pathol., (2004).

[6] G. Cooper and E. Herskovitz, *A bayesian method for the induction of probabilistic networks from data*, Machine Learning, 9 (1992).

[7] K. D. Dahlquist, N. Salomonis, K. Vranizan, S. C. lawlor, and B. R. Conklin, *Genmapp, a new tool for viewing and analyzing microarray data on biological pathways*, Nature Genetics, 31 (2002), pp. 19–20.

[8] S. M. Dhanasekaran, D. G. R. R. Barrette, R. Shah, S. Varambally, K. Kurachi, K. Pienta, and A. M. Chinnaiyan, *Delineation*

of prognostic biomarkers in prostate cancer, Nature, 412 (2001), pp. 822–826.

[9] S. Draghici, P. Khatri, R. P. Martins, G. C. Ostermeier, and S. A. Krawetz, *Global functional profiling of gene expression*, Genomics, 81 (2003), pp. 98–104.

[10] L. E. Eder, J. Bektic, G. Baartsch, and H. Klocker, *Genes differentially expressed in prostate cancer*, BJU International, 93 (2004), pp. 1151–1155.

[11] N. Friedman and D. Koller, *Being bayesian about network structure: A bayesian approach to structure discovery in bayesian networks*, Machine Learning, 50 (2003), pp. 95–126.

[12] N. Friedman, M. Linial, I. Nachman, and D. Pe'er, *Using Bayesian networks to analyze expression data*, Journal of Computational Biology, 7 (2000), pp. 601–620.

[13] N. Friedman, I. Nachman, and D. Pe'er, *Learning Bayesian network structure from massive datasets: the "sparse candidate" algorithm*, in Proc. Fifteenth Conference on Uncertainty in Artificial Intelligence (UAI '99), H. Dubios and K. Laskey, eds., Morgan Kaufmann, 1999, pp. 206–215.

[14] J. R. Graff, *Emerging targets in the akt pathway for the treatment of androgen-independent prostatic adenocarcinoma*, Expert Opinion in Therapeutic Targets, 6 (2002), pp. 103–113.

[15] A. Hartemink, D. K. Gifford, and T. Jaakola, *Using graphical models and genomic expression data to statistically validate models of genetic regulatory networks*, in Proceedings of the 6th Pacific Symposium on Biocomputing, 2001, pp. 422–33.

[16] R. Henrique and C. Jeronimo, *Molecular detection of prostate cancer: A role for GSTP1 hypermethylation*, Eur Urol., 46 (2004), pp. 660–669.

[17] T. Ideker and D. Lauffenberger, *Building with a scaffold: emerging strategies for high to low-level cellular modeling*, Trends in Biotechnology, 21 (2003).

[18] H. D. Jong, *Modeling and simulation of genetic regulatory systems: A literature review*, Journal of Computational Biology, 9 (2002), pp. 67–103.

[19] M. Koivisto and K. Sood, *Exact bayesian structure discovery in bayesian networks*, Journal of Machine Learning Research, 5 (2004), pp. 549–573.

[20] P. K. Majumder, P. G. Febbo, R. Bikoff, R. Berger, Q. Xue, L. M. McMahon, J. Manola, J. Brugarolas, T. J. McDonnell, T. R. Golub, M. Loda, H. A. Lane, and W. R. Sellers, *mTOR inhibition reverses Akt-dependent prostate intraepithelial neoplasia through regulation of apoptotic and HIF-1-dependent pathways*, Nature Medicine, 10 (2004), pp. 594–601.

[21] H. Mamitsuka and Y. Okuno, *A hierarchical mixture of markov models for finding biologically active metabolic paths using gene expression and protein classes*, in Proc. 2004 IEEE Computational Systems Bioinformatics Conference, IEEE, 2004, pp. 341–352.

[22] J. Pearl, *Probabilistic reasoning in intelligent systems*, Morgan Kauffman, 1988.

[23] L. L. Pisters, C. A. Pettaway, P. Troncoso, T. J. McDonnell, L. C. Stephens, C. G. Wood, K. A. Do, S. M. Brisbay, X. Wang, E. A. Hossan, R. B. Evans, C. Soto, M. G. Jacobson, K. Parker,

J. A. Merritt, M. S. Steiner, and C. J. Logothetis, *Evidence that transfer of functional p53 protein results in increased apoptosis in prostate cancer*, Clin. Cancer Res., 10 (2004), pp. 2587–93.

[24] D. R. Rhodes, T. R. Barrette, M. A. Rubin, D. Ghosh, and A. M. Chinnaiyan, *Meta-analysis of miroarrays: interstudy valication of gene expression profiles reveals pathway dysregulation in prostate cancer*, Cancer Research, 62 (2002), pp. 4427–4433.

[25] C. J. Rosser, M. Tanaka, L. L. Pisters, N. Tanaka, L. B. Levy, D. C. Hoover, H. B. Grossman, T. J. McDonnell, D. A. Kuban, and R. E. Meyn, *Adenoviral-mediated pten transgene expression sensitizes bcl-2-expressing prostate cancer cells to radiation*, Cancer Gene Ther., 11 (2004), pp. 273–9.

[26] E. Ruijter and et. al., *Molecular genetics and epidemiology of prostate carcinoma*, Endocr. Rev., 20 (1999), pp. 22–45.

[27] K. Sachs, O. Perez, D. Pe'er, D. A. Lauffenburger, and G. P. Nolan, *Causal protein-signaling networks derived from multiparameter single-cell data*, Science, 308 (2005), pp. 523–529.

[28] E. Segal, N. Friedman, D. Koller, and A. Regev, *A module map showing conditional activity of expression modules in cancer*, Nature Genetics, 36 (2004), pp. 1090–8.

[29] E. Segal, R. Yelensky, and D. Koller, *Geneome-wide discovery of transcriptional modules from DNA sequence and gene expression*, Bioinformatics, 19 (2003), pp. i273–i282.

[30] P. Shannon, A. Markiel, O. Ozier, N. S. Baliga, J. T. Wang, D. Ramage, N. Amin, B. Schwikowski, and T. Ideker, *Cytoscape: a software environment for integrated models of biomolecular interaction networks*, Genome Res., 13 (2003), pp. 2498–504.

[31] D. Singh, P. G. Febbo, K. Riss, D. G. Jackson, J. Manola, C. Ladd, P. Tamayo, A. A. Renshaw, A. V. D'Amico, J. P. Richie, E. S. Lander, M. Loda, P. W. Kantoff, T. R. Golub, and W. R. Sellers, *Gene expression correlates of clinical prostate cancer behavior*, Cancer Cell, 1 (2002), pp. 203–209.

[32] K. B. Spurgers, K. R. Coombes, R. E. Meyn, D. L. Gold, C. J. Logothetis, T. J. Johnson, and T. J. McDonnell, *A comprehensive assessment of p53-responsive genes following adenoviral-p53 gene transfer in bcl-2-expressing prostate cancer cells*, Oncogene, 23 (2004), pp. 1712–23.

[33] M. Zou and S. D. Conzen, *A new dynamic bayesian network (dbn) approach for identifying gene regulatory networks from time course microarray data*, Bioinformatics, 21 (2005), pp. 71–9.