BIOINFORMATICS

Vol. 1 no. 1 2001
Pages 1–9

Inferring Subnetworks from Perturbed Expression Profiles

Dana Pe'er[1], Aviv Regev[2,3], Gal Elidan[1] and Nir Friedman[1]

[1]School of Computer Science & Engineering, Hebrew University, Jerusalem, 91904, Israel, [2]Department of Cell Research and Immunology, Life Sciences Faculty, Tel Aviv University, Tel Aviv, 69978, Israel and [3]Department of Computer Science and Applied Mathematics, Weizmann Institute of Science, Rehovot, 76100, Israel

ABSTRACT

Genome-wide expression profiles of genetic mutants provide a wide variety of measurements of cellular responses to perturbations. Typical analysis of such data identifies genes affected by perturbation and uses clustering to group genes of similar function. In this paper we discover a finer structure of interactions between genes, such as causality, mediation, activation, and inhibition by using a Bayesian network framework. We extend this framework to correctly handle perturbations, and to identify significant substructures of interacting genes. We apply this method to expression data of *S. cerevisiae* mutants and uncover a variety of structured metabolic, signaling and regulatory pathways.
Contact: danab@cs.huji.ac.il

INTRODUCTION

Integrated molecular pathways consisting of interacting proteins, genes, and small molecules underlie the major functions of living cells. These include signal transduction and processing, regulation of gene expression and metabolism. Genome wide expression profiles allow us to gain insight into these processes. In order to obtain a wide variety of profiles, reflecting different active pathways, various perturbations and treatments are employed. Perturbation by mutation of specific genes serves a dual purpose, providing a rich variety of different profiles, while allowing us to compare a wild type profile with a mutant one and to determine the molecular effect or function of the mutated gene.

Two recent studies use such an experimental design, employing different types of analysis. Holstege et al. (1998) compare mutant and wild type profiles to identify sets of "downstream" genes whose expression is affected by a specific mutation. Hughes et al. (2000) use clustering to group either genes with correlated expression in different mutant strains or entire mutant profiles. Valuable biological insight can be gained by both approaches.

In this paper, we strive to answer questions that deal with finer structure. For example, is the effect of a mutated gene on a target gene direct, or is it mediated by other genes? Which genes mediate the interactions within a cluster of genes or between clusters? What is the nature of the interaction between genes (e.g does gene A inhibit gene B)?

To infer such finer relations from perturbed gene expression profiles[†] we use the framework of Friedman et al. (2000). In this framework, we treat the measured expression level of each gene as a random variable and regulatory interactions as probabilistic dependencies between random variables. Friedman *et al.* use *nonparametric bootstrap* to estimate the confidence of *features* of Bayesian networks learned from expression profiles. This allows them to identify pairwise relations of high confidence such as: "Genes A and B closely interact".

We extend this framework in four ways. First, we adapt and extend recent results on learning with interventions (Cooper and Yoo, 1999) to handle genetic mutations. Second, we devise new, better suited, methods for discretizing the data prior to analysis. Third, we define and learn new features: mediator, activator and inhibitor. Finally, we describe how to use features to construct *substructures* of strong statistical significance.

The resulting method comprises the following steps. We start by discretizing the data. Then, we apply bootstrap analysis to learn an ensemble of networks which represent potential models of the interactions between genes. We use this ensemble to extract features involving relationships between pairs and triplets of genes with high statistical confidence. We then identify statistically significant subnetworks which contain several high-confidence features. These subnetworks capture a strong statistical signal in the expression profile that often reflects a coherent cellular process.

As a case study, we apply our framework for the analysis of the Rosetta Compendium of expression profiles from *Saccharomyces cerevisiae* (Hughes et al., 2000).

[†] We stress that any attempt to perform this task is limited to learning relations that are represented in mRNA expression data. For example, post-translational regulation may often be missed.

BAYESIAN NETWORK ANALYSIS OF EXPRESSION DATA

Probabilistic Modeling of Gene Expression

Measurements of gene expression involve noise arising from the measurement technology, the experimental procedures, and the underlying stochastic biological processes. Thus, we treat gene expression as a probabilistic process, and represent the expression level of each gene as a *random variable*. The joint distribution over the set of all genes reflects the distribution of cell "states" and how these affect transcript levels. Our ultimate goal is to estimate and understand the structure of this distribution.[‡]

Most standard methods for analyzing gene expression focus on pairwise relations between genes, such as correlation. However, biological interaction is seldom this simple, and often includes chains of mediators between two genes. By going beyond pairwise relations and exploring multi-variable interactions, we can infer more about the structure of the relationship between genes. In particular, we focus on *conditional independence*. For example, if X and Y are co-regulated by Z then, while Y correlates with X, it might be that given the value of Z, Y becomes independent of X. In this case, we say that Z *separates* between X and Y. In general, such a separator can be a set of variables.

Bayesian Networks

A *Bayesian network* over a set $\mathbf{X} = \{X_1, \ldots, X_n\}$ is a representation of a joint probability distribution over $\mathbf{X}$. This representation consists of a *directed acyclic graph* (DAG) G whose vertices correspond to the random variables $X_1, \ldots, X_n$, and a parameterization which describes a conditional distribution for each variable given its immediate parents in G.

The graph G represents conditional independence properties of the distribution. These are the *Markov Independencies*: Each variable X_i is independent of its non-descendants, given its parents in G. A distribution that satisfies these independencies can be decomposed into the *product form*

$$P(X_1, \ldots, X_n) = \prod_{i=1}^{n} P(X_i | \mathbf{Pa}_i^G), \tag{1}$$

where $\mathbf{Pa}_i^G$ is the set of parents of X_i in G. The parameterization component of the network describes the conditional distributions $P(X_i|\mathbf{Pa}_i^G)$. Thus, the network represents the unique distribution.

The Markov independencies represented by G often imply other conditional independencies. We can determine whether G implies that X and Y are independent given $\mathbf{Z}$ by using *d-separation* (Pearl, 1988). This is a simple graph theoretic criteria on the structure of the graph G.

Two DAGs may imply exactly the same set of independencies. For example, consider graphs $X \rightarrow Y$ and $X \leftarrow Y$ over two variables X and Y. Both graphs imply that X and Y are not independent. In such a situation, we say that the two graphs are *equivalent*.

The notion of equivalence is crucial, since when we examine observations from a distribution, we cannot distinguish between equivalent graphs. Thus, we want to find the common properties of *equivalence classes* of DAGs. Pearl and Verma (1991) show that equivalent graphs have the same underlying undirected graph but might disagree on the direction of some of the arcs. Moreover, they show that an equivalence class of network structures can be uniquely represented by a *partially directed graph* (PDAG), where a directed edge $X \rightarrow Y$ denotes that all members of the equivalence class contain the arc $X \rightarrow Y$; an undirected edge X—Y denotes that some members of the class contain the arc $X \rightarrow Y$, while others contain the arc $Y \rightarrow X$.

Learning Bayesian Networks

Given a *training set* $D = \{\mathbf{x}[1], \ldots, \mathbf{x}[M]\}$ of independent samples from an unknown distribution $P(\mathbf{X})$, we want to estimate this distribution by a network G. The common approach to this problem is to introduce a statistically motivated scoring function that evaluates each network with respect to the training data, and to search for the optimal network according to this score (Heckerman, 1998). A popular score based on Bayesian reasoning, scores candidate graphs G by their posterior probability given the data (see (Heckerman et al., 1995) for a complete description). We define the score $S(G : D)$ to be proportional to $P(G \mid D)$. An important characteristic of this score is that when the data is complete (no missing values) the score is *decomposable*:

$$\mathrm{S}(G : D) = \sum_i S_{local}(X_i, \mathbf{Pa}_i^G : D) \tag{2}$$

The contribution of each variable X_i to the total score depends only on the values of X_i and $\mathbf{Pa}_i^G$ in the training instances.

$$S_{local}(X_i, \mathbf{U} : D) = \log P(\mathbf{Pa}_i = \mathbf{U}) + \log \int \prod_m P(X_i[m] \mid \mathbf{U}[m], \theta) dP(\theta).$$

The first term is the prior probability assigned to the choice of the set $\mathbf{U}$ as the parents of X_i. The second term measures the probability of the data, when we integrate over the possible parameterizations (θ) of the

‡ We use the following notation in the remainder of the paper. We use capital letters, such as X, Y, Z, for variable names. Sets of variables are denoted by boldface capital letters $\mathbf{X}, \mathbf{Y}, \mathbf{Z}$.

conditional distribution. These local contributions for each variable can be computed using a closed form equation (see (Heckerman, 1998) for details).

MODELING PERTURBATIONS INTO BAYESIAN NETWORKS

Ideal Interventions

Above we assumed that each training instance was sampled from the underlying distribution. This does not apply in genetic mutation experiments. For instance, by knocking out gene X, we replace the original molecular control on X's expression (its parents) by an external one. Thus, any consequent measurement (in which X's value is constantly set to 0) will not teach us anything about X's conditional distribution on its parents. Modeling such interventions for learning Bayesian networks involves two issues: the score function and the definition of equivalence.

Recall that the score of a DAG G, given a data set D, decomposes into a product of entities that depend on the conditional distributions $P(X|\mathbf{Pa}_X^G)$. Suppose that in a certain sample, we intervene by fixing the value of $X_i[m]$. In this sample, it is clear that we should not take into account $P(X_i[m] \mid \mathbf{Pa}_i[m])$, as the value of X_i in the sample does not depend on this distribution. However, if our intervention only modified the value of X_i, all others variables were sampled from their respective conditional distributions. We call such manipulations *ideal interventions* (Pearl, 2000) and treat their score as follows: If we let *Int*(m) denote the set of variables that were intervened in the m'th sample, then the modified local score is

$$S_{local}(X_i, \mathbf{U} : D) = \log P(\mathbf{Pa}_i = \mathbf{U}) + \log \int \prod_{m, X_i \notin \mathit{Int}(m)} P(X_i[m] \mid \mathbf{U}[m], \theta) dP(\theta).$$

See (Cooper and Yoo, 1999) for more details on this score.

This score is no longer *structure equivalent*, i.e., the score of two equivalent graphs, G and G' is no longer guaranteed to be the same. This should be expected, as interventions help us determine the direction of causality. We say that G and G' are *intervention equivalent* given interventions $I \subseteq \{X_1, \ldots, X_n\}$, if they receive the same score given a data set D where *Int*$(m) \subseteq I$, for all m. This notion of equivalence is more restrictive, and thus more edges in the PDAG will be directed. These include, but are not limited to, all edges entering or leaving an intervened variable X. We modified the procedure for constructing a PDAG representation from a DAG (Chickering, 1995) to fit our new equivalence relation. Due to space restrictions, we omit the technical details.

Modeling Perturbations

We distinguish between two types of perturbations in gene expression data. The first type includes gene deletion and over-expression. Both imply a direct change to the expression level of the mutated gene. Formally, the random variable corresponding to this level is deterministically assigned a specific value. We model such mutations as ideal interventions, as described above.

The second class of perturbations includes temperature sensitive and kinetic mutations (Holstege et al., 1998)) as well as external treatments (e.g. environmental stress (Gasch et al., 2000)). These perturbations do not directly determine an expression level of a specific gene, and thus cannot be modeled as ideal interventions. However, since they have an important effect on the expression level of many genes, their occurrence in a given sample should be indicated. We add *indicator variables* to our domain, one for each treatment type. We constrain such variables to be roots i.e. no other variables can be their parents in the network.

ZOOMING IN: IDENTIFYING FEATURES

Potential Features

We now focus on the following question: Can we elucidate the nature of interaction between two genes? We learn from the perturbed gene expression profiles a Bayesian network G and construct its corresponding PDAG U_G (taking into account the patterns of interventions) Assuming that G correctly captures the dependencies in the domain, we consider several types of "queries" or "features" that can be identified from G and U_G.

Markov and Edge Relations To find if there is a direct interaction between X and Y we can query our network whether X and Y are *Markov neighbors*. Markov neighbors are variables that are not separated by any other measured variable in the domain. They include parent-child relations (one gene regulating another), or spouse relations (two genes that co-regulate a third). Since our domain consists of many variables which are not modeled into our network (e.g. protein activation), many of the resulting Markov neighbors represent sibling relations i.e. two genes which are regulated by a third variable, not modeled by the network (Elidan et al., 2001). When neither of these situations occur, the network implies that the interaction between X and Y is indirect.

We can query whether the edge $X \rightarrow Y$ appears in U_G. Recall that this implies that X and Y are Markov neighbors (parent-child type) and that the edge between them is directed in all networks in the equivalence class of G. The existence of such a directed edge suggests that X is a direct *cause* of Y.[§]

[§] To reach causal conclusions from a Bayesian Network few assumptions

Separators When X and Y are indirectly dependent, we can ask what factors *mediate* this dependence. In the simple case, a single variable Z, separates X and Y. For example, the edges $X \rightarrow Z \rightarrow Y$ or the undirected edges X—Z—Y appear in U_G. In the former case, X affects Z, which in turn affects Y; while in the latter, Z might be a common cause of both X and Y.

In more complex cases, X and Y may be more distant in the graph structure (e.g Z is a common grandparent of both X and Y) and there might be more than one variable that mediates their interaction (e.g X is parent of Z_1 and Z_2, who in turn are both parents of Y). In these cases we must employ a global approach, searching for variables $\mathbf{Z}$, such that Y is independent of X given $\mathbf{Z}$ in the network. In such a situation, we say that $\mathbf{Z}$ explains all the dependencies between the two variables.

We can test such dependencies using d-separation. More precisely, to check that two variables X and Y are independent given Z, we need to check that no path between X and Y can "pass" information when the value of Z is known (See Pearl (1988) for the precise definition). Testing for d-separation between two variables requires $O(n)$ time, where n is the number of variables. Computing d-separation for every pair of variables in the network is thus in the order of $O(n^3)$ with a relatively large coefficient. For a large domain, this calculation is time and memory consuming. We note, however, that when two variables are far from each other in the network, the dependence between them is significantly diminished. Thus, in practice we check for d-separation between variables along paths of limited length.

Activation and Inhibition When X is a parent of Y, we can gain understanding of X's effect on Y. Here we are interested only in the conditional distribution $P(Y \mid \mathbf{Pa}_Y)$. Let $\mathbf{U} = \mathbf{Pa}_Y - \{X\}$. Intuitively, if $P(Y = 1 \mid X, \mathbf{u})$ increases when X transitions from -1 to 0 and then to 1 and $\mathbf{u}$ is held fixed, we say that X *activates* Y. Since all other direct influences on Y have been kept at the same state, the change in X is the explanation to the change in Y. Similarly, if $P(Y = -1 \mid X, \mathbf{u})$ increases, then X *inhibits* Y. Our current strict criterion requires that X activates/inhibits Y for every set of values $\mathbf{u}$ of $\mathbf{U}$. Less naive approaches that soften this requirement are under study.

Feature Confidence

Above we assumed that the network G correctly represents the interactions in the underlying domain. How reasonable is this assumption? If we have a sufficiently large number of samples, we can be (almost) certain that the network we learn is a good model of the data. However, given only a small number of training instances, there may be many models that explain the data almost equally well. Such models can have qualitatively very different structures. We do not have confidence that one network is an accurate description of the biological domain.

Therefore, instead of querying a single structure, we can examine the *posterior* probability of the feature given the data. Formally, we consider the distribution of *features*. A feature of a network is a property such as "$X \rightarrow Y$ is in the network" or "$\mathbf{Z}$ d-separates X from Y in the network". We define the feature using an indicator function $f(G)$ that has the value 1 when G satisfies the feature and value 0 otherwise. The posterior probability of a feature is

$$P(f(G) \mid D) = \sum_G f(G) P(G \mid D). \quad (3)$$

This probability reflects our *confidence* in the feature f.

A naive way of calculating Eq. 3, is by enumerating all high scoring networks. Unfortunately, the number of such networks can be exponential in the number of variables, so exact computation of the posterior probability is impractical. Instead, we can estimate this posterior by sampling representative networks, and then estimating the fraction that contain the feature of interest. We can generate such networks using non-parametric bootstrap (Friedman et al., 1999) or using more exact but costly MCMC simulations (Friedman and Koller, 2001). Friedman et al. (2000) evaluate the bootstrap approach in simulated data that matches the distributions observed in gene expression data. They note that the rate of false negatives is high. Thus, the fact that we do not detect high confidence for a feature, does not mean it does not exist, but rather that the data does not strongly support this feature.

RECONSTRUCTING SIGNIFICANT SUBNETWORKS

Using the methods of previous section we assign confidence to features (e.g., all Markov edges). We then estimate which confidence levels to consider significant[¶] (0.75 in the experiments below), and focus on these significant features. As we show below, important insights can be gained from such features. Nonetheless, this approach suffers from two drawbacks. First, it is limited to examining the relations between two or three genes. Second, it can be overly cautious discarding correct features whose confidence is below our threshold. We suggest to overcome these drawbacks by identifying *subnetworks* of high confidence. This allows us to both

must be made. See (Pearl, 2000; Cooper and Glymour, 1999) regarding the connection between Bayesian networks and causality, and (Friedman et al., 2000) for a discussion of these connections in the context of gene expression.

[¶]This can be done by *simulation studies*, where we generate training sets from a known network, and by *randomization studies*, where we permute the expression of genes across experiments to create a data set where all genes are independent of each other Friedman et al. (1999, 2000).

broaden our viewpoint and gain confidence about features that when isolated are not significant.

For now, we focus on Markov pairs, as these can be more reliably reconstructed using our methods (Friedman et al., 1999). Our hypothesis is that if we can find a subnetwork that contains a *concentration* of Markov pairs with relatively high confidence, then our estimate of edges and other features in this region is more reliable. Indeed, such subnetworks often correspond to biologically meaningful relations between genes. While our a full-scale network is currently of insufficient quality, statistically significant sub-networks can reconstructed.

Naive Approach

A naive approach for finding subnetworks is as follows. We start by selecting a threshold t_s of significant confidence. We can then construct a graph over variables, with an edge between X and Y if this Markov pair is significant (beyond t_s). In this graph we find maximal connected components. Each non-trivial component (one that contains more than three variables) is considered a *seed* of a subnetwork. We expand each seed by adding variables that are related to some variable(s) in the seed by a Markov pair with confidence above t', where $t' < t_s$ is an additional parameter. In the experiments below, we use $t_s = 0.75$ and $t' = 0.5$.

While the results found by the naive procedure make biological sense, there are two drawbacks to this approach. First, there is no measure of quality for the resulting networks. Second, the seed is symmetrically expanded by inclusion of all directly related variables. This overlooks variables which are quite significantly, though indirectly, related to the seed.

Score-based Approach

We propose to evaluate the significance of a subnetwork in terms of the concentration of it features. We start by estimating the probability of observing Markov pairs at different confidence levels. Let $F(c)$ be the probability that random Markov pair (X, Y) has confidence $C(X, Y)$ of at least *c*. We estimate this probability by computing the observed fraction of Markov pairs with confidence c and higher among the $\binom{n}{2}$ possible pairs in our domain.

Consider a subnetwork that contains the variables $\mathbf{U}$ with Markov pairs $e_1, \ldots, e_l$ that have confidences $c_1 \geq c_2 \geq \cdots \geq c_l \geq t_0$, respectively, where t_0 is a threshold we choose in advance. (In our experiments, we set $t_0 = 0.5$.) We want to evaluate the significance of the existence of these edges among the variables in $\mathbf{U}$. We do so, by bounding the expected number of similar subnetworks we expect to find under a null-hypothesis model. This model assumes each edge confidence is sampled independently from the same distribution F we observed in our data.

The probability of sampling the observed confidence levels or higher for the particular edges $e_1, \ldots, e_l$ is $\prod_i F(c_i)$. Thus, under the null hypothesis, the probability of a subnetwork over $\mathbf{U}$ having confidence levels better than $c_1, \ldots, c_l$ is at most $\binom{K}{l} \prod_i F(c_i)$ where $k = |\mathbf{U}|$ and $K = \binom{k}{2}$. Since we search for a similar substructure over all possible subsets of size k of $\{X_1, \ldots, X_n\}$, the expected number of occurrences of such a subnetwork overall is at most

$$\binom{n}{k} \binom{K}{l} \prod_i F(c_i)$$

Thus, we evaluate potential subnetworks by a score that bounds their *E-value* from above under the null hypothesis.

To find subnetworks with optimal scores, we employ a greedy hill-climbing search. This search starts with candidate seeds, which are triplets of variables connected by at least two high scoring edges. At each step of the search we consider adding or removing a single variable to the "current" subnetwork. We then select the operation that leads to the best score. Once we reach a local optimum, we add it to a pool of subnetworks. We repeat this search from all potential seeds, and then return the subnetworks that have E-value better (smaller) than a prespecified threshold (we use e^{-5}).

As a sanity check, we tried the procedure on randomized data. We reshuffled the original data-set, thus eliminating genuine dependencies between variables. We constructed subnetworks from the randomized data and validated that none of resulting subnetworks scored above our threshold.

DISCRETIZING GENE EXPRESSION DATA

Due to noisy experimental procedures and measurement techniques, gene expression data must be handled with care to ensure successful application of analysis methods. Friedman et al. (2000) consider two strategies for treating gene expression measurements. In the first strategy, they discretize the expression levels to several discrete states (e.g., "under-expressed", "baseline", and "over-expressed) using a fixed discretization policy (e.g., 2-fold change from control). In the second approach, they use the actual measurements, and model dependencies with a linear regression model. As they show, the first strategy is sensitive to the discretization procedure, and the second one is heavily biased toward linear dependencies.

In this paper, we introduce a new, adaptive, discretization procedure that *learns* for each gene the distribution of expression values in each state. In particular, this procedure takes into account the gene-specific variation, and uses it to differentiate which expression levels significantly deviate from the baseline expression of the gene.

Our basic assumption is that each gene can be in few discrete functional expression states, which relate to its activity. Thus, it is convenient to model the expression level of the gene in different experiments as samples from a mixture of normal distributions, where each normal component corresponds to a specific state. Once we estimate such a mixture model, we can classify each expression level to the most likely mixture component and get a discretization. We use standard k-means clustering to estimate such a mixture.

We face two issues. First, how many states of the gene actually appear in the data? For some genes, only two states (say, "baseline" and "over-expressed") are actually present in the data. Second, what initial start classification should we use for k-means? Both choices are crucial to get a sensible discretization.

To deal with both issues, we adopt the following strategy. We use measurements of expression levels in repeated wild-type experiments (i.e., measurements of expression without perturbations) to estimate the distribution of the gene's expression level in its baseline state. We then consider each measurement of the gene in the perturbed samples, and determine whether it is significantly over-expressed or under-expressed with respect to its distribution in the control experiments. Based on the outcome of this test, we place the measurement in the appropriate classification for the initialization of k-means. The number of mixtures we learn with k-means is then the number of classes in this initial assignment that contain some measurements. We then run k-means from the initial point, and use the classification it determines as the discretization for the gene.

The only question that remains is how to identify significant changes in expression levels. We employ a Bayesian procedure to estimate the posterior probability over the mean and variance for a given gene (DeGroot, 1989) in the control experiments, and then test the probability that the treated sample came from the same distribution. For lack of space we omit the technical details.

RESULTS

The Rosetta Inpharmatics Compendium (Hughes et al., 2000) is a reference dataset compiled of 300 full-genome expression profiles obtained from 276 deletion mutants, 11 tetracyclin regulatable alleles of essential genes, and 13 chemically treated *S. cerevisiae* cultures, each compared to a baseline wild type or mock-treated culture. We chose a subset of 565 genes which included the mutated genes and genes which showed significant change in at least 4 profiles. We used their 63 wild-type verses wild-type measurements to estimate the baseline distribution provided to our discretization procedure. The feature confidences were computed using a 100-fold bootstrap learning procedure. Each network requires approximately 1 hour CPU using an Intel III 600mhz processor with a 1 gigabytes RAM.

We have developed *Pathway Explorer* a visualization tool for the resulting subnetworks. The network is displayed as graph in which extensive local information is associated with the undirected and directed edges. We stress that no prior biological knowledge was used by our learning procedure when reconstructing the networks. The full annotated results can be viewed using Pathway Explorer at our web site: http://www.cs.huji.ac.il/labs/compbio/ismb01. Here we focus on several examples that highlight the validity and power of our approach.

Pairwise Relations

Biological analysis of individual Markov pair relations indicates that many are supported by previous findings, and represent either a known biochemical or regulatory interaction, a shared common regulator, or a functional link. Strikingly, the Pearson correlation coefficient between approximately a third of these "proof-of-principle" gene pairs was lower than 0.7. Our method is capable of discovering such relations because of the *context specific* nature in which it handles the data. There are many biological processes that occur only under specific conditions. Correlation "misses" such interactions, which are only apparent in part of the samples. (Scores for features are presented in the following format: (Confidence, Pearson correlation) for each such pair.) Two such "proof of principle" Markov pairs are, Phosphoribosylaminoimidazole carboxylase (*ADE2*) and Phosphoribosylamidoimidazole-succinocarboxamide synthase (*ADE1*) (0.797, 0.518), which catalyze the sixth and seventh steps in the de novo purine biosynthesis pathway, respectively; and *SST2*, a (negative) regulator of the mating signaling pathway and *STE6*, the membrane transporter responsible for the export of the "a" mating factor (0.914, 0.677).

Even pair-wise relations alone succeed in providing new biological insight. For example, we studied an edge relation (0.914, 0.162) from *ESC4*, a protein involved in chromatin silencing to *KU70*, a key component of the DNA non-homologous double strand break DNA repair mechanism. This is a previously unknown link, yet we supply strong supporting evidence from the literature. First, other chromatin silencing genes (SIR2, 3, and 4) are necessary together with KU70 and KU80 for DNA end joining [W1].[‖] Second, *ESC4* contains 6 BRCT domains, that are known to occur predominantly in proteins involved in cell cycle checkpoint functions responsive to DNA damage [W2]. Together, these facts

[‖] The notations [Wn] relate to additional citations appearing at our web site.

clearly support both a functional association between the two proteins and a regulatory directed interaction (from *ESC4* to *KU70*) assigning a new (putative) regulatory function to *ESC4* in double strand break repair. Note, that a *ku70* mutant strain is included in the compendium data, while *ESC4* had not been mutated. This illustrates how our treatment of mutations aids in inferring causal relations in a counter intuitive direction. While typical analysis can only find the effect of a mutation, we find a causal source (in wild-type strains) of a mutated gene.

Separator Relations

In this section we provide an illustration of the capability of separator triplets to explain away dependencies, providing an enhanced insight into the underlying molecular architecture of pathways. First, we consider three genes each appearing in several undirected separator triplet relations. All three genes are well known mediators of transcriptional responses, and the genes they separate share functional roles and regulation patterns, consistent with the separator serving as a common regulator.

The first gene, *KAR4*, is a mating transcriptional regulator of karyogamy (nuclear fusion) genes, which is known to pair with the mating transcription factor Ste12p to activate genes required for nuclear fusion [W3]. *KAR4* separates several pairs of cell fusion genes (e.g. *AGA1* and *FUS1*). The second gene, *SLT2*, encodes the MAP kinase of the cell wall integrity (low osmolarity) pathway, which post-translationally activates (by phosphorylation) the transcription factors *Rlm1p* and *Swi4/6* which in turn activate low osmolarity response genes [W4]. *SLT2* separates several pairs of cell membrane and cell wall proteins (e.g. *YSP1*) as well as previously uncharacterized one (e.g. *SRL3*). In addition, an activation relation was detected between *SLT2* and *YSP1* which is consistent with *SLT2*'s known regulatory effect. The third gene, *SST2*, is a post-translational negative regulator of the G-protein in the mating signaling pathway [W5]. *SST2* separates the mating response genes *TEC1* and STE6. Moreover, a directed inhibition edge was discovered from *SST2* to *STE6*, consistent with *SST2*'s known inhibitory role in the mating pathway.

We conclude that in all three cases, our inference has reconstructed the regulatory role in the correct molecular and functional context, revealing both transcriptional and post-translational regulators. Furthermore, since previously uncharacterized genes participated in some of these interactions (e.g. *SRL3* in *SLT2*, *YNL276W* in *KAR4*) we could assign them putative functions, probably as effectors, in cell wall integrity and cell fusion, respectively.

The power of separator relations at identifying indirect dependencies can be appreciated to the fullest when examining d-separator relations (X–Z–Y). The main difference between Markov-triplet and d-separator relations, is that the mediating gene is itself not necessarily in a direct Markov relation with the two genes it separates. For computational efficiency we computed only singleton separators appearing in paths of length at most 6. Strikingly, in 35 of the resulting 120 interactions, the mediating (Z) gene was either a transcriptional regulator or signaling molecule.** Such molecules were considerably less frequent in the X and Y positions (only 18 and 11 interactions respectively). These results are consistent with a co-regulatory role to the mediating gene.

The co-regulatory role of mediators is further emphasized by numerous examples. Each of the general transcriptional regulator *SIN3* and the GTP-binding signaling protein *RHO1* occupy the mediator position in several d-separator relations (5 and 6, respectively). In each case, a combination of statistical and biological evidence indicates a co-regulatory role for these proteins. For example, *RHO1* mediates interactions between effector proteins that affect the level of free glucose in the cell (glycogen phosphorylase *GPH1*, hexokinase 1 *HXK1*, the hexose transporter *HXT6* and α 1,4 glucan branching enzyme *GLC3*). In some cases, the Markov relation between the two "extreme" is high (0.97, 0.89), providing further support for a co-regulatory role for *RHO1*. This novel finding identifies a new regulatory spectrum for *RHO1* and is consistent with its activation of β 1,3 glucan synthase (which utilizes UDP-glucose as a substrate). It also allows us to assign a putative role to a protein of unknown function, *YJL161W*, which appeared in one of the d-separator relations.

In some cases, d-separator relations provide support for the regulatory role of putative transcription factors and signaling molecules. For example, YPR015C, which has two zinc finger motifs, appears as a mediator between two uncharacterized genes with a high Markov score (X-Y 0.92). In other cases, such relations identify functional links that were not directly identified by pair or triplet relations. For instance, no Markov relation was identified between the two signaling proteins from the mating pathway *STE5* and *STE11*. They were identified, however, within a significantly scored (0.5) d-separated triplet, consistent with their shared functional role and physical interaction (STE5 is a scaffold protein which complexes with STE11 and other signaling proteins). These results indicate the importance of d-separator relations in the identification of indirect relations, in particular those involving common regulatory and signaling molecules.

**The 8 transcriptional regulators at position Z include general repressors (*ISW1(2 relations), TOP1(1), SIN3(5)*), specific transcription factors (*MTH1(1), RGT1(1), IMP2(1)*), and putative transcriptional regulators (*YFL052W(1), YPR015C(1)*). The 7 signaling molecules are *KSS1(1), MFA2(3), RAS1(6), RHO1(6), STE11(1), TFS1(3), YKL161C(2).*

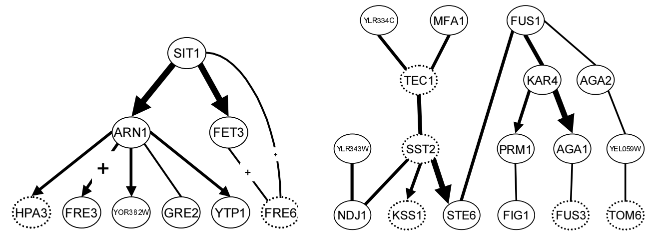

Fig. 1. Two subnetworks that visualize features discovered. (a) Iron homeostasis (b) Mating response. The width of the arc corresponds to the confidence of the feature. The edges are directed only when there is high confidence in its orientation. Nodes circled with a dashed line correspond to genes which have been mutated in some of the samples. Arcs marked by a + sign are activators, size corresponds to confidence of feature. Due to space limitations, the iron homeostasis pathway is not discussed here.

Subnetwork analysis

The full power of our approach becomes apparent when exploring subnetworks. We applied our naive approach to constructing subnetworks (see above). In order to obtain subnetworks that represent a coherent biological processes, we then merged subnetworks whose genes are known to be related to the same biological process. This resulted in 6 well-structured subnetworks, interleaved with additional higher and lower confidence relations. Each of the subnetworks represents a coherent molecular response: mating response, low osmolarity cell wall integrity pathway, stationary phase response, iron homeostasis, amino acid metabolism along with mitochondrial function, and citrate metabolism (two are depicted in Figure 1, all available at our website). Of 87 top scoring Markov pairs, 61 were appeared[††] within these subnetworks.

Our score based approach to constructing subnetworks produced 5 highly significant networks, capturing 4 of the 6 partially hand-crafted networks. The subnetworks resulting from this method are usually larger and structured in an almost modular fashion. They typically are composed of tighter subnetworks (usually around a high degree mediating gene) connected through high confidence edges to other such tight subnetworks. Interestingly, most of the high degree variables in these networks correspond to known regulatory genes.

While Hughes et al. (2000) identify some of these responses (amino acid metabolism, iron homeostasis and mating) using clustering, our reconstructed networks provide a much richer context for both regulatory and functional analysis. For example, they describe a large cluster of genes associated with amino acid metabolism. In our network, we can discern at least three finer structures with high confidence. The first involves the genes *ARG1*, *ARG3* and *ARG5*, all part of the urea cycle (and its close periphery), which are known to be transcriptionally co-regulated [W6,7]. The second, for sulfate metabolism which further decomposes into two branches: one of sulfate transporters (*SUL1* and *SUL2*) and the other of sulfate assimilation (*MET3*, *MET14*, and *MET22*). The common separator for these branches is the *MET10* gene. The third and major part of the network interleaves various enzymes for amino acid metabolism (e.g. *HIS4*, *HIS5*, *LEU4*, *ILV2* and *ARG4*) with mitochondrial proteins, most prominently transporters and carriers (e.g. *BAT1*, *OAC1*, and *YHM1*). A regulatory link has been found between the general amino acid response and mitochondrial function [W8]. Thus, a large group of genes, which by correlation alone would be simply clustered together, can be organized in clear functional networks.

We use the mating response subnetwork, shown in Figure 1(b), to illustrate the power of our method to reconstruct a coherent biological tale and raise novel biological hypotheses. We discern two distinct branches , one for cell fusion and the other for outgoing mating signaling. According to our network, the cell fusion response branch is mediated by the *KAR4* gene (see above), and includes several known cell membrane fusion genes (*FUS1*, *AGA1*, *AGA2*, *PRM1* and *FIG1*) [W9,10,11,12] as well as two genes previously unassociated with this process (*TOM6* and *YEL059W*). The multitude of high confidence relations strongly suggests a putative role to *KAR4* not only regulation of nuclear fusion but also regulation of cell membrane fusion.

Another branch in this network is directed from the mating signaling pathway regulator *SST2* (above). Since an *SST2* mutant has been incorporated in the compendium we could determine edge direction, and identify *SST2* as a prime regulator of several other genes (*TEC1*, *STE6*, *MFA1*) previously shown to be transcriptionally regulated by the mating pathway [W13,14,15]. The regulatory link from *SST2* to *KSS1* is intriguing as the two share an interaction with *MPT5*, a multicopy suppressor of transcript specific regulators of mRNA degradation in yeast [W16,17], but *KSS1* was not previously associated with the mating pathway, but rather with the (related) filamentous invasive growth response.

Some puzzling discrepancies exist in our network. The first is the absence of the main transcription factor of the pathway, *STE12*. This may be due either to loss of information by our discretization procedure or to our bias to reduce the number of false positive interactions. The second, is the marginal position of the pathway's MAP

[††]An additional 16 relations could be explained as individual interacting gene pairs or triplets, and only 10 relations currently remain unassociated or unexplained.

kinase, *FUS3*. This may be due to positive feedback, rendering *FUS3* both an activator and an activation target. However, despite the knockout mutation in FUS3 we have failed to identify directed regulation. We believe that larger number of repetitions for each mutation will enhance our framework's capabilities to discover such regulatory relations.

DISCUSSION AND FUTURE WORK

In this paper we extend the framework of Friedman et al. (2000). We integrated into this framework a new discretization procedure and a principled way for learning with a mixture of observational and interventional data. We examined new types of features that can be uncovered using our analysis method. Last but not least, we presented automated methods of integrating these features into structures representing biological processes. We applied these tools to analyze the Compendium data of *S. cerevisiae* mutations (Hughes et al., 2000).

This analysis illustrates the differences between our techniques and clustering methods. On the one hand, we are able to discover inter-cluster interactions between weakly correlated genes. On the other hand, we can uncover finer intra-cluster structure among correlated genes. This assists us in understanding the roles of genes within a richer context and in assigning them putative novel functions. The use of the Pathway Explorer greatly facilitates such biological exploration. Both regulatory, metabolic and signaling components are identified, showing the potential of our approach to uncover the three major types of molecular networks. We stress that our approach cannot recover all interactions. Instead, we attempt to provide the biologist with a number of highly promising hypotheses.

The primary contribution of this paper is an automated methodology for finding significant subnetworks of interacting genes. These are shown to related to known biological pathways. Still, the issue of uncovering biological pathways from gene expression data remains major challenge. A crucial issue is how to use prior biological knowledge to improve the quality of analysis and increase the number of novel interactions detected.

Additional directions for exploration include better reconstruction of causal structure. Our analysis mostly found mediators that were common ancestors. Yet, we seldom found intermediate steps in causal chains. This is partially due to the lack of repeated measurements of each genetic mutant. Nevertheless, it poses a serious challenge for methods of analysis. A related problem is identifying latent factors (e.g., co-regulators) that interact with several observed genes (Elidan et al., 2001).

Acknowledgments

The authors are grateful to Michal Chur, Rani Nelken, Matan Ninio, Itsik Pe'er and Eran Segal for comments on drafts of this paper and useful discussions. This work was supported in part by Israel Science Foundation (ISF) grant 244/99, Israeli Ministry of Science grant 2008-1-99, and an ISF infrastructure grant. D. Pe'er was supported by an Eshkol Fellowship. A. Regev was supported by the Colton Foundation. N. Friedman was supported by an Alon Fellowship.

REFERENCES

Chickering, D. M. (1995). A transformational characterization of equivalent Bayesian network structures. In *UAI'95*, pp. 87–98.

Cooper, G. and C. Glymour (1999). *Computation, Causation, and Discovery*.

Cooper, G. and C. Yoo (1999). Causal discovery from a mixture of experimental and observational data. In *UAI'99*, pp. 116–125.

DeGroot, M. H. (1989). *Probability and Statistics*.

Elidan, G., N. Lotner, N. Friedman, and D. Koller (2001). Discovering hidden variables: A structure-based approach. In *NIPS'00*.

Friedman, N., M. Goldszmidt, and A. Wyner (1999). Data analysis with Bayesian networks: A bootstrap approach. In *UAI '99*, pp. 206–215.

Friedman, N. and D. Koller (2001). Being Bayesian about Bayesian network structure: A Bayesian approach to structure discovery in Bayesian networks. *Machine Learning*. To appear. Earlier version appeared in UAI'00.

Friedman, N., M. Linial, I. Nachman, and D. Pe'er (2000). Using Bayesian networks to analyze expression data. *J. Comp. Bio. 7*, 601–620.

Gasch, A. P., P. T. Spellman, C. M. Kao, O. Carmel-Harel, M. B. Eisen, G. Storz, D. Botstein, and P. O. Brown (2000). Genomic expression program in the response of yeast cells to environmental changes. *Mol. Bio. Cell 11*, 4241–4257.

Heckerman, D. (1998). A tutorial on learning with Bayesian networks. In *Learning in Graphical Models*. Kluwer.

Heckerman, D., D. Geiger, and D. M. Chickering (1995). Learning Bayesian networks: The combination of knowledge and statistical data. *Machine Learning 20*, 197–243.

Holstege, F. C., E. G. Jennings, J. J. Wyrick, T. I. Lee, C. J. Hengartner, M. R. Green, T. R. Golub, E. S. Lander, and R. A. Young (1998). Dissecting the regulatory circuitry of a eukaryotic genome. *Cell 95*(5), 717–28.

Hughes, T. R., M. J. Marton, A. R. Jones, C. J. Roberts, R. Stoughton, C. D. Armour, H. A. Bennett, E. Coffey, H. Dai, Y. D. He, M. J. Kidd, A. M. King, M. R. Meyer, D. Slade, P. Y. Lum, S. B. Stepaniants, D. D. Shoemaker, D. Gachotte, K. Chakraburtty, J. Simon, M. Bard, and S. H. Friend (2000). Functional discovery via a compendium of expression profiles. *Cell 102*(1), 109–26.

Pearl, J. (1988). *Probabilistic Reasoning in Intelligent Systems*.

Pearl, J. (2000). *Causality: Models, Reasoning, and Inference*.

Pearl, J. and T. S. Verma (1991). A theory of inferred causation. In *KR'91*, pp. 441–452.