

Large-scale regulatory network analysis from microarray data: modified Bayesian network learning and association rule mining

Zan Huang ^{a,*}, Jiexun Li ^b, Hua Su ^b, George S. Watts ^c, Hsinchun Chen ^b

^a Department of Supply Chain and Information Systems, Smeal College of Business, The Pennsylvania State University, University Park, PA 16802, United States

^b Department of Management Information Systems, Eller College of Business and Public Administration, The University of Arizona, Tucson, AZ 85721, United States

^c Arizona Cancer Center and Department of Pharmacology and Toxicology, The University of Arizona, Tucson, AZ 85724, United States

Available online 11 April 2006

Abstract

We present two algorithms for learning *large-scale* gene regulatory networks from microarray data: a modified information-theory-based Bayesian network algorithm and a modified association rule algorithm. Simulation-based evaluation using six datasets indicated that both algorithms outperformed their unmodified counterparts, especially when analyzing large numbers of genes. Both algorithms learned about 20% (50% if directionality and relation type were not considered) of the relations in the actual models. In our empirical evaluation based on two real datasets, domain experts evaluated subsets of learned relations with high confidence and identified 20–30% to be “interesting” or “maybe interesting” as potential experiment hypotheses.

© 2006 Elsevier B.V. All rights reserved.

Keywords: Genetic regulatory networks; Microarray; Bayesian networks; Association rules

1. Introduction

Recent advances in microarray technologies have made it possible to routinely measure the expression levels of tens or even hundreds of thousands of genes simultaneously. Such high-throughput experimental data have initiated much recent research on large-scale gene expression data analysis. Various data mining techniques (e.g., clustering and classification) have been employed to uncover the biological functions of genes from micro-

array data [2,18,20]. Recently, these techniques have included a reverse engineering approach to extracting gene regulatory networks from microarray data in order to reveal the structure of the transcriptional gene regulation processes.

The general purpose of gene regulatory network analysis is to extract pronounced gene regulatory features (e.g., activation and inhibition) by examining gene expression patterns. Changes of expression levels of genes across different samples provide information that allows reverse engineering techniques to construct the network of regulatory relations among those genes. Many studies have shown that these learned networks have the potential to help researchers propose and evaluate new hypotheses in basic research of genetic regulatory process [17,19,40]. Such data-driven

* Corresponding author. 419 Business Building, University Park, PA 16802. Tel.: +1 814 863 1940.

E-mail addresses: zanhuang@psu.edu (Z. Huang), jiexun@eller.arizona.edu (J. Li), hus@eller.arizona.edu (H. Su), gwatts@azcc.arizona.edu (G.S. Watts), hchen@eller.arizona.edu (H. Chen).

regulatory network analysis would eventually lead to better understanding of the complex genetic regulatory process, which has important implications in the pharmaceutical industry and many other biomedical fields.

Many components in regulatory networks are involved in pathological processes such as diabetes and cancer [35]. Dissection of regulatory networks provide a better understanding of the mechanism of these diseases and will help in the development of diagnosis methods, selection of gene therapy candidates, and elucidation of drug targets. For instance, the strategy for new cancer drug searching has shifted from finding chemicals that kill tumor cells towards identifying molecular targets that underlie cell transformation [54]. The latter approach relies on deeper mechanistic understanding of the regulatory processes and holds promise to discover more effective and safer drugs. Gene regulatory network analysis, although not directly helping researchers in identifying molecular markers for disease diagnosis and providing drug target, provides new insights and hypotheses in basic genetic research to understand the mechanism of gene regulation pathways and ultimately achieve a holistic understanding of the genetic regulatory process, which provides the foundation for applications in genetic disease diagnosis and drug design.

Previous studies have proposed various network learning approaches: pair-wise comparison [3,10], differential equation estimation [58], and Bayesian network learning [19,38], among others. A common problem has been that only a relatively small number of genes were included into the network. This was mainly due to the inherent dimensionality problem in microarray data, which usually contain insufficient sample measurements of a large number of genes to capture the complex interactions of the biological systems. On the other hand, knowledge of the overall structures of the gene network has been acknowledged to be invaluable for a complete mechanistic understanding of the complex roles of individual genes and their interactions [33]. Many recent studies have been targeted at deriving the large-scale or even complete gene networks using heterogeneous functional genomics data, including protein direct physical interaction data (such as yeast two-hybrid assays) [26], gene synthetic lethal screens [51], biomedical literature text [28], and functional linkages derived from comparative genomics methods [42] as well as gene expression data [7,25,50]. Along with these recent developments and computational needs, scalable techniques need to be developed to derive gene regulatory networks that include large numbers of genes from the gene expression data.

In this paper, we present two scalable gene regulatory network learning algorithms: a *modified information-theory-based Bayesian network algorithm* and a *modified association rule mining algorithm*. The high dimensionality nature and limited sample size of the currently available gene expression data make large-scale regulatory network learning from such data a difficult problem both theoretically and computationally. Both algorithms presented in this paper are heuristic in nature. In these algorithms we intend to construct a *rough causal network* from the small-sample-size data for a relatively large number of variables, as opposed to the formal graphical model that describes the exact statistical dependency relationships among the set of variables as in typical Bayesian network or probabilistic graphic model literature [22,41]. For the modified Bayesian network algorithm we have relaxed certain theoretical restrictions to make possible efficient learning of large-scale graphical models from high-dimensional small-sample-size data. The modified association rule mining algorithm identifies individual genetic regulatory relationships without an overall assessment of all the derived relationships as a network. Both algorithms are limited and the learned networks are not guaranteed to be *the* correct underlying graphical model. However, both algorithms can utilize the limited data to derive genetic networks involving large numbers of genes efficiently. Our hope is that the resulting rough genetic regulatory networks can still reveal the overall structure of the underlying regulatory networks and identify interesting relationships to assist biomedical researchers in more effective utilization of the available microarray data for potential valuable insights and findings. In order to empirically assess the performances of the proposed algorithms, we conducted simulation-based evaluation and empirical evaluation studies using a real human gene expression dataset. We present the technical details of the two modified algorithms and the evaluation results in this paper.

The remainder of the paper is organized as follows. Section 2 reviews the literature on existing approaches to gene regulatory network learning and their limitations. Section 3 addresses representation and modeling issues for learning the gene networks from microarray data. We present in Section 4 the algorithmic details of the modified information-theory-based Bayesian network learning algorithm and the modified association rule mining algorithm. Section 5 presents the simulation-based and empirical evaluation studies. We conclude the paper in Section 6 by summarizing contributions and future directions.

2. Literature review

Gene regulatory network analysis is aimed at identifying regulatory relationships among large numbers of genes that form a network representation of the underlying regulatory processes. In such analysis, gene interactions are viewed as signaling processes that form a complex feedback network. Information for the construction and maintenance of this signaling system is stored in the genome. The DNA sequence codes for the structure and molecular dynamics of RNA and proteins in turn determine biochemical recognition of the signaling processes. The regulatory molecules that control the expression of genes are themselves products of other genes [34]. Thus, genes turn each other *on* and *off* within a proximal genetic network of transcriptional regulators [46] that provides a partial picture of the complex biological processes. Many other factors (e.g., proteins and metabolites) that also play important roles in this signaling system are largely hidden from observation. However, a genetic network that summarizes the aggregated effects reflected in gene expression patterns still reveals valuable information about the underlying processes.

One traditional approach to identifying gene regulatory relationships is the “knockout” experiment, in which the gene expression level of a particular gene is lowered while all other conditions are kept constant. Differences in gene expression levels of those other genes are used to infer underlying regulatory relations. Usually, this approach can reliably uncover regulatory relationships among a small number of genes but fails to scale up to study regulatory networks consisting of hundreds of genes because of the sheer number of experimental manipulations needed to reveal a complete regulatory network. Recent development of microarray technologies has made it possible to routinely measure expression levels of tens or hundreds of thousands of genes simultaneously. Such development has enabled researchers to apply reverse engineering approaches on observational expression data to identify large-scale gene regulatory network structures.

The earliest proposed models for learning gene regulatory networks from microarray data were *discrete models*. Several studies proposed to construct Boolean regulatory networks in which the gene expression levels were represented as 0 (not expressed) or 1 (expressed) [30,34]. These models are based on the assumption that biological networks can be represented by binary, synchronously updating switching networks. However, large amounts of information might be lost during binary discretization. Many continuous models have been proposed recently to fully utilize the information contained in gene expression data. Wessels et al. have categorized

existing *continuous models* of gene regulatory networks into pair-wise networks and weighted sum networks [58] *Pair-wise networks* are constructed with pair-wise comparison methods based on relationships between pairs of genes. Two examples of such methods are the Correlation Metric Construction (CMC) [3] and Activation/Inhibition Networks [10] Such approaches ignore the fact that the expression level of one gene is governed by combined actions of multiple other genes and thus cannot reveal a complete regulatory structure. *Weighted sum network* models represent a gene expression level as a weighted sum of the expression levels of its upstream genes. Wessels et al. used a generalized differential or difference equation (the difference equation is shown in (1)) to characterize this class of models [58]

$$X_i[t + 1] = R_i \cdot (\sum_{j=1}^J W_{ij} \cdot X_j[t] + \sum_{k=1}^K V_{ik} \cdot U_k + B_i) - \lambda_i \cdot X_i[t] \quad (1)$$

where g refers to the regulation-expression (activation) function, $X_i[t]$ represents the expression of gene i at time instance t , R_i represents the rate constant of gene i , W_{ij} represents the strength of influence of gene j on gene i , $U_k[t]$ represents k th external input at time instance t , V_{ik} represents the influence of the k th external input on gene i , B_i represents the base expression level of gene i , and λ_i represents the degradation constant of gene i .

A variety of approaches have been proposed to estimate the parameters of such differential or difference equations, including Recurrent Hopfield networks [37] linear programming [10] simulated annealing [36] genetic algorithms [55] and linear regression [16,45,56] When a large number of genes are incorporated into the model and a relatively small number of samples are available, a dimensionality problem inevitably arises.

Murphy and Mian [38] and Friedman et al. [19] have initiated the use of Bayesian network models to represent and learn gene regulatory networks from microarray data. Bayesian network models can include most previously proposed discrete and continuous models as special cases and allow hidden variables as well as incorporation of prior knowledge into learning. This approach recently has attracted substantial research interest and has become the major approach for regulatory network learning [15,21,25,40,44] Many of these recent studies explored using Bayesian networks learning framework to combine microarray data and existing knowledge as well as other types of genomic data to learn more accurate regulatory network models.

Previous gene regulatory network studies have been limited to modeling relations among only a small set of genes, mainly due to the dimensionality problem of the

microarray data, i.e., the number of available samples of measurements is typically much smaller than the number of genes that can be measured, restricting the analytical models to only include a small subset of the measured genes. For example, Mjolsness et al. [36] used 8 samples of 27 genes, Wahde et al. [55] used 28 samples of 65 genes, and Someren et al. [45] used 18 samples of 45 genes and 14 samples of 113 genes. However, gene interactions form a complex signaling system in which the behavior of any single gene may be affected by many other genes. Since study with a limited number of genes typically reveals only an incomplete local structure of the regulatory network, it is desirable to infer regulatory networks having large numbers of genes in order to capture complete regulatory processes. A study by Friedman et al. [19] is one of the few that have analyzed large numbers of genes (800 genes were included). In their study, the goal was to identify pronounced gene regulatory relations rather than to specify the exact regulatory network structure.

The poor scalability of most existing network analysis techniques has limited the empirical value of gene regulatory network analysis. We present in this paper two algorithms designed for learning large-scale gene regulatory networks from microarray data: a modified information-theory-based Bayesian network learning algorithm and a modified association rule mining algorithm.

3. Learning large-scale gene regulatory networks

We represent the expression level of a particular gene as a random variable. For each gene in the microarray chip, we can obtain the expression level of a test sample and a reference sample. The log ratio between the test and reference expression levels is commonly used to measure the gene expression level. Thus the input data for our analysis are in fact relative expression levels rather than the absolute expression levels. The gene expression data contained in one chip constitute one sample of observation. We represent a microarray dataset consisting of N genes and M samples as two $M \times N$ expression measurement matrices, the test expression matrix $T = \{T_{ij}\}$ and reference expression matrix $R = \{R_{ij}\}$, where T_{ij} and R_{ij} represent the test and reference expression levels of the i th gene in the j th sample, respectively. The relative expression data used for regulatory network analysis can be represented by an $M \times N$ relative expression matrix (or simply expression matrix) $E = [53]$ where $E_{ij} = \log(T_{ij}/R_{ij})$.

To allow the proposed algorithms to analyze large-scale genetic regulatory networks using small samples of data, discretized gene expression levels were employed.

In particular, following the commonly adopted expression data discretization method [8,19,27,52] that was also confirmed by our collaborating domain scientists, we performed ternary discretization and transformed the continuous expression levels E_{ij} into discrete expression levels X_{ij} taking values of 1, 0, and -1 as in (2).

$$X_{ij} = \begin{cases} +1, & \text{when } E_{ij} \geq +\Theta, \text{ over-expressed,} \\ -1, & \text{when } E_{ij} \leq -\Theta, \text{ under-expressed,} \\ 0, & \text{otherwise} \end{cases} \quad (2)$$

where Θ is a predefined threshold. We believe that this ternary discretization method preserves the important biological meaning of the expression data and at the same time gives relatively simple data representation to support efficient computation. We define the discrete expression level of gene j as a discrete random variable X_j , whose observed values in the M samples are given by the column vector $(X_{1j}, \dots, X_{Mj})'$. Our proposed genetic regulatory network learning algorithms operate on such an expression data representation.

In this section, we describe the algorithmic details of the two proposed gene regulatory network learning techniques: a modified Bayesian networks learning algorithm (MBN algorithm) and a modified association rule mining algorithm (MAR algorithm). Bayesian networks have been widely applied to infer causal relationships among random variables and to generate causal network structures. Many recent studies have applied Bayesian networks in regulatory network analysis, mostly for small-scale networks. We customized an efficient Bayesian networks learning algorithm to accommodate the high dimensionality and small sample size characteristics of the microarray data. The basic idea is to relax some theoretical restrictions to generate a rough causal network. The association rule mining algorithm has been traditionally applied for market basket analysis, with specific focus on individual rules rather than network structure. We customized it to accommodate ternary gene expression levels in the microarray data and consolidate individual rules into a rough causal network structure. We assess the effectiveness of both algorithms for learning regulatory networks from microarray data using both the simulated and real microarray data.

3.1. Bayesian networks

3.1.1. Bayesian networks background

Bayesian networks are one type of probabilistic graphical model in which vertices represent random variables and the absence of an edge between two vertices

represents conditional independence. The Bayesian network representation was first introduced by Wright [61] and was reinvented later under various names, such as “causal network,” “belief network,” and “influence diagram.” Bayesian networks have been frequently applied in many real-world applications, including diagnosis, forecasting, automated vision, sensor fusion, and manufacturing control [23]. Consider a finite set $V = \{V_1, \dots, V_N\}$ of random variables. In our context, the discrete expression level of a gene is treated as a random variable. Bayesian network representation contains two components: a directed acyclic graph (DAG) G whose vertices correspond to random variables, and conditional probability distributions of the random variables, given its dependent variables (parents) in G . The graph G specifies the dependency relationships among variables and encodes the *Markov Assumption*: Each variable V_i is independent of its non-descendants, given its parents in G .

We use an example adopted from [19] to illustrate the basic idea. Given a Bayesian network specified in Fig. 1 for 5 genes: $A, B, C, D,$ and E , this structure specifies the parents for genes $B, D,$ and C : $\text{Pa}(B) = \{A, E\}$, $\text{Pa}(D) = \{A\}$, $\text{Pa}(C) = \{B\}$, where $\text{Pa}(V)$ represents the parent vertex set for vertex V . It also implies several conditional independency relationships: $I(A; E), I(B; D|A), I(C; A, D, E|B), I(D; B, C, E|A),$ and $I(E; A, D)$, where $I(X; Y|Z)$ denotes that X and Y are conditionally independent given Z . In our context, it can be interpreted that when genes in Z are at fixed expression levels, expression levels of genes in X do not give any information on the expression levels of genes in Y and vice versa. Once G is specified for a set of genes, we can interpret a directional edge from X to Y in G as a statement that X is the “cause” of Y , or the expression level of X has an effect on the expression level of Y .

3.1.2. Modified Bayesian network algorithm (MBN algorithm)

There are two general approaches to learning Bayesian networks from data: the search and scoring

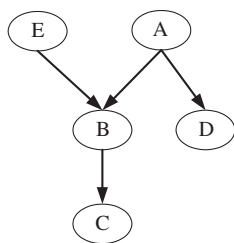


Fig. 1. A simple example of Bayesian network. (a) Human dataset—MBN algorithm. (b) Human dataset—MAR algorithm. (c) Yeast dataset—MBN algorithm. (d) Yeast dataset—MAR algorithm.

methods and dependency analysis methods [11,60]. With the first approach, the learning problem is viewed as searching for a structure that best fits the data. Different scoring methods have been applied to determine the fit between the network structure and the data, including Bayesian scoring, entropy-based, and minimum description length, among others. Because search and scoring methods are NP-hard [12], many search heuristics have been adopted. Previous studies of gene regulatory network learning using Bayesian networks typically employed the heuristic-based search and scoring methods [19,38]. The dependency analysis approach tries to discover from data the dependencies among variables and then to use these dependencies to construct the network structure [48,49,57]. Different forms of conditional independency (CI) tests have been used to measure the dependency relationships.

The dependency analysis approach is generally more efficient than the search and scoring approach for sparse networks (the number of edges in the graph is relatively small) [11]. The disadvantage of the dependency analysis approach is that it has a more restrictive assumption regarding the probability distribution of the data. We customized a recently developed algorithm of this approach, the information-theory-based learning algorithm proposed by Cheng et al. [11] which extends Chow and Liu’s tree construction algorithm [14] to perform Bayesian network learning using a three-phase procedure. Although a recent theoretical analysis [13] has pointed out problems with a fundamental assumption of the algorithm, the monotone DAG faithfulness, for inferring exact network models from data, empirical evaluations have shown that this algorithm generates reasonably good approximate network models in many applications [9,11]. We briefly describe the three major phases of the algorithm and our customization for analyzing microarray data. Readers are referred to [11] for additional technical details.

The algorithm employs mutual information (MI) and conditional mutual information (CMI) to measure the volume of information flow. The MI and CMI measures between vertices X_i and X_j are defined below.

$$I(X_i, X_j) = \sum_{x_i, x_j} P(x_i, x_j) \log \frac{P(x_i, x_j)}{P(x_i)P(x_j)} \quad (3)$$

$$I(X_i, X_j|C) = \sum_{x_i, x_j, c} P(x_i, x_j, c) \log \frac{P(x_i, x_j|c)}{P(x_i|c)P(x_j|c)} \quad (4)$$

where x_i denotes the potential values of X_i ($-1, 0,$ and 1 in our context) and C is a set of dependent vertices

referred to as a condition set. The CMI measure in (4) is used for the conditional independence (CI) test in this algorithm. We next describe the algorithm in three phases.

- *Phase I: Drafting.* A draft network is constructed based on the MI measures. A threshold ε is specified and the pairs of genes that have a MI measure larger than ε are selected to form candidate edges. The candidate edges are added into the draft network in the order of descending MI measures, under the condition that the draft network is maintained as a singly connected graph (a graph without loops). At the end of this phase, the draft network contains exactly $N-1$ edges, where N is the number of genes.
- *Phase II: Thickening.* The algorithm determines whether an unconnected pair of vertices can be separated (direction dependent separation or d-separated [41] by its condition sets based on the draft network structure. For a pair of unconnected genes $\langle X_i, X_j \rangle$, two condition sets, C_i and C_j , are formed by including the neighbors of the two genes in the draft network. CMI measures of $\langle X_i, X_j \rangle$ are computed using C_i and C_j as well as their subsets as the condition sets. If any of these CMI measures is below the threshold ε , the pair is deemed to be separated, otherwise an edge is added between the pair. In this phase edges can be added but cannot be removed (thus the name thickening phase). The thickening process assures that the resulting network contains all the edges of the underlying dependency model. This comprehensive property holds when the underlying graph model is monotone DAG-faithful [11].
- *Phase III: Thinning.* Each edge of the graph obtained in Phase II is examined using the CI tests to determine whether the two vertices of the edge can be d-separated with the edge being removed. For an edge connecting X_i and X_j in the current network, the edge is first removed temporarily and the same CI test procedure in Phase II is performed to determine whether the pair is d-separated. If they are separated, the edge is permanently removed. If the pair is not separated, an extended CI test procedure is performed. This extended CI test procedure extends the condition sets C_i and C_j to not only include the direct neighbors of X_i and X_j but also their second-order neighbors (neighbors of their neighbors). If the pair is now separated with respect to the extended CI test, the edge is permanently removed; otherwise the edge is kept in the network. In this

phase the edges can only be removed (thus the name thinning phase). After the thinning phase, the resulting network is guaranteed to contain exactly the same edges as those in the underlying model under the monotone DAG-faithfulness assumption [11]. An edge orientation procedure is also performed to determine the directions of the edges in the learned network. The number of edges whose directions can be determined depends on the specific network structure and expression patterns. Typically there are a fraction of the edges whose direction can not be determined.

The most significant advantage of this algorithm is that, unlike all other practicable dependency analysis-based algorithms, it can avoid the exponential complexity of CI tests. The algorithm requires $O(N^4)$ number of CI tests when learning Bayesian networks with completely unknown network structures.

Our modification of Cheng's Bayesian network learning algorithm for large-scale regulatory network analysis mainly involved Phases II and III of the algorithm. In the thickening phase (Phase II), based on the assumption that the regulatory networks are relatively sparse as well as on the computational efficiency consideration, we limited the maximum number of edges added in this phase to be k times of the total number of vertices. Because of this limitation the modified algorithm does not assure that all the edges of the underlying model are included. Researchers could set the parameter k based on domain knowledge regarding the sparsity of the underlying regulatory network to assure reasonable coverage.

In the thinning phase (Phase III), we modified the algorithm by relaxing the criterion for d-separation to accommodate the small sample size relative to the number of variables in the data. Because of the small sample size a relatively large condition set typically gives a zero or unreliable CMI measure [11] resulting in a small number of edges kept in the network. This problem is most serious when performing the extended CI tests, in which direct neighbors as well as second-order neighbors are included into the condition sets. We relaxed the d-separation assessment by not performing the extended CI test. Due to this relaxation, the resulting network model is not guaranteed to only include edges in the underlying true network model.

After the network structure has been learned, we use a simple heuristic to determine the biological meanings of the edges. For an edge that points from gene X to gene Y , we calculate the activation and inhibition

measures as defined in (5) and (6) to determine the relation type.

Activation Measure

$$= \frac{P(Y = 1, X = 1) + P(Y = -1, X = -1)}{P(X = 1 \text{ or } X = -1)} \quad (5)$$

Inhibition Measure

$$= \frac{P(Y = -1, X = 1) + P(Y = 1, X = -1)}{P(X = 1 \text{ or } X = -1)} \quad (6)$$

The activation and inhibition measures are consistent with domain experts' intuitions. For example, if we always observe that gene Y is over (under)-expressed whenever gene X is over (under)-expressed, we expect an activation influence from X to Y corresponding to an activation measure of 1. Edges with activation (inhibition) measures larger than a specified threshold (0.5 in our current experiments) are labeled as activation (inhibition) relations. The resulting Bayesian network model may contain three types of gene relations: activation, inhibition, and dependency (when the edge direction cannot be determined).

For all relations included into the final network, we treat the MI measure between the two genes as a measure of relation strength. When examining a learned regulatory network, researchers can assess the networks with different confidence levels by setting varying values of minimum relation strength.

3.2. Association rule mining

3.2.1. Association rule mining background

Association rule mining was originally proposed for market basket analysis to study consumer purchasing patterns in retail stores [1]. In recent years, it has also been applied in other areas such as customer relationship management [5], image processing [43] and network communications [59]. An association rule is a relationship of the form $A \Rightarrow B$, where A is the antecedent item set and B is the consequent item set. The rule $A \Rightarrow B$ holds in the transaction set D with *confidence* c if $c\%$ of transactions in D that contain A also contain B . The rule $A \Rightarrow B$ has *support* s if $s\%$ of transactions in D contain both A and B . The goal of association rule mining is to find all the rules that have support and confidence greater than user-specified thresholds. Association rule mining can be used to extract genetic regulatory relationships based on conditional dependency. Association rules not only capture the correlation between genes, but also provide the direction of relationships. Some initial work of using association rules in molecular classifica-

tion and gene regulatory relation extraction has been reported in [6,8,29,31,32,39].

3.2.2. Modified association rule mining algorithm (MAR algorithm)

Classic association rule mining algorithms can handle only Boolean data. Becquet et al. applied binary discretization to gene expression data and set values less than or equal to a threshold to 0 and all values greater than the threshold to 1 [6]. Such an approach may lead to serious information loss because binary representation fails to capture the overall gene expression distribution. Alternatively, a ternary discretization approach was used in [8,52] to convert each gene expression to one of three levels, i.e. under-expressed, normal, or over-expressed. For each gene, three Boolean variables were used to represent the three expression levels respectively. This approach captures gene expression distribution more completely than binary discretization but the increased number of variables to be processed may lead to larger numbers of potential association rules, thereby increasing the difficulty in interpreting and analyzing the rules extracted. It may also reduce the computational efficiency of the mining process.

We propose to use one single variable to represent expression level of each gene, i.e. for gene X , an expression level can be represented as under-expressed ($X=-1$), normal ($X=0$) or over-expressed ($X=1$), respectively. Based on such a ternary discretization, we modified the classic association rule mining algorithm to extract gene activation and inhibition relations from microarray data.

According to biological interpretations, the activation and inhibition relations between two genes, X and Y , can be respectively defined as follows:

- (a) X activates Y ($X \xrightarrow{+} Y$): IF X is over-expressed ($X=1$), THEN Y is over-expressed ($Y=1$); IF X is under-expressed ($X=-1$), THEN Y is under-expressed ($Y=-1$)
- (b) X inhibits Y ($X \xrightarrow{-} Y$): IF X is over-expressed ($X=1$), THEN Y is under-expressed ($Y=-1$); IF X is under-expressed ($X=-1$), THEN Y is over-expressed ($Y=1$).

Based on these notions, the support and confidence measures of association rules are redefined. Given a gene expression matrix E , the *support* of a single gene X is defined in (7):

$$\text{support}(X) = \frac{\|X = 1 \text{ OR } X = -1\|}{|E|} \quad (7)$$

where $\|X=1 \text{ or } X=-1\|$ is the number of samples in E where $X=1$ or -1 ; $|E|$ is the number of samples in E .

For two genes, X and Y , two new measures of support^+ and support^- are defined in (8) and (9), respectively:

$$\begin{aligned} \text{support}^+(XY) &= \frac{\|X=1 \text{ AND } Y=1\| + \|X=-1 \text{ AND } Y=-1\|}{|E|} \end{aligned} \quad (8)$$

$$\begin{aligned} \text{support}^-(XY) &= \frac{\|X=1 \text{ AND } Y=-1\| + \|X=-1 \text{ AND } Y=1\|}{|E|} \end{aligned} \quad (9)$$

In essence the two support measures are consistent with the activation and inhibition measures (5) and (6) introduced before. The confidence^+ of an activation rule in form of “ $X \xrightarrow{+} Y$ ” is defined in (10):

$$\text{confidence}^+(X \xrightarrow{+} Y) = \frac{\text{support}^+(XY)}{\text{support}(X)} \quad (10)$$

Similarly, the confidence^- of an inhibition rule in the form of “ $X \xrightarrow{-} Y$ ” is defined in (11):

$$\text{confidence}^-(X \xrightarrow{-} Y) = \frac{\text{support}^-(XY)}{\text{support}(X)} \quad (11)$$

This modified association rule mining algorithm first generates all rules (both activation and inhibition) that have *support* and *confidence* equal to or greater than a user-specified minimum support (called *minsups*) and minimum confidence (called *minconf*), respectively. When assembling the individual rules (genetic relations) into a gene network, a resolution procedure is performed. Note that it is possible for relations with reversed directions to both appear in the learned relations ($X \xrightarrow{+} Y$ and $Y \xrightarrow{+} X$). These two relations with reversed directions both have the same support measures. We include the one with the higher confidence measure into the regulatory network. If both relations also have the same confidence measure, we include an undirected relation into the gene network.

We use an example to illustrate the basic idea behind our method, given expression values of gene X

Table 1
Expression values of gene X and Y

Gene	1	2	3	4	5	6
X	-4.0285	-4.7155	-5.057	1.2113	-1.7731	1.0751
Y	-1.5454	-3.3322	-3.7306	0.4040	1.6306	1.3292

Table 2
Transformed expression levels of gene X and Y

Gene	1	2	3	4	5	6
X	-1	-1	-1	1	-1	1
Y	-1	-1	-1	0	1	1

and Y in six samples (in Table 1), $\text{minsups}=20\%$, and $\text{minconf}=60\%$.

By applying ternary discretization (threshold= ± 1), we transformed data as in Table 2:

Based on our definition of *support* and *confidence*, for rule “ $X \xrightarrow{+} Y$ ”, we have $\text{support}(X)=100\%$, $\text{support}^+(XY)=66.67\% > 20\%$, and $\text{confidence}^+(X \xrightarrow{+} Y)=66.67\% > 60\%$. The rule $X \xrightarrow{+} Y$ is thus extracted as an activation rule.

To be consistent with the MBN algorithm, we use the confidence measure of the rules included into the final network as the genetic relation strength. Presumably, the most reliable genetic relations reflected in the microarray data should have higher confidence measures.

4. Evaluation

In this section, we present two types of evaluation studies: a simulation-based evaluation and an empirical evaluation. In the simulation-based evaluation, we generated expression levels of a set of genes based on pre-defined gene regulatory network models. Models learned from such simulated data were compared with the pre-defined models to determine the ability of our algorithms to uncover the true structure. In the empirical evaluation, we generated gene regulatory networks using gene expression datasets from real experiments including a *Saccharomyces cerevisiae* dataset and a *Homo sapiens* dataset.

4.1. Simulation-based evaluation

The simulation approach is frequently used to validate network-learning techniques (e.g., [11,58]). With this approach, a network model is pre-defined based either on existing knowledge in an application domain or on a randomly generated structure. Values of the variables of a sample of simulated observations are generated based on this network model. The inferential power of the network learning techniques, in terms of ability to uncover underlying structures, can then be assessed by comparing the model learned from the simulated data with the pre-defined true model.

The simulation approach has been recently employed to evaluate different approaches to extracting gene regulatory networks. Wessels et al. conducted a comprehensive evaluation of various network extraction techniques using simulated gene expression data [58]. Their study focused on time-series models and a small number of genes (15 genes) were included in the datasets. Their main conclusion was that most existing approaches had surprisingly low inferential power.

We incorporated characteristics of real gene expression data into the simulation process. With these simulated gene expression data we could take into consideration the major limitations of the experimental gene expression data (e.g., the limited number of samples and noise and measurement errors) and evaluate the practical usefulness of the network learning techniques in deriving gene regulatory networks from realistic microarray data. The usefulness was interpreted as the extent to which the results can help researchers propose new hypotheses that may eventually lead to new discoveries on gene regulatory relations. A series of network accuracy measures will be introduced later to assess the usefulness of the network results.

The advantage of using simulation data for evaluation is that the true network model is known. In contrast, large portions of the true network model are unknown when using real experimental data, which makes it difficult to determine the quality of a learned network.

4.1.1. Model and data simulation

We used the Bayesian network model to represent a gene regulatory network. Thus regulatory relations among genes were described by a directional acyclic graph, in which a directional edge represents the regulatory influence of parent (upstream) genes on child (downstream) genes. Therefore, the expression level of a gene was assumed to be dependent on expression levels of its parent genes. The choice of the Bayesian network model was mainly based on the stochastic nature of the gene regulation process. Another advantage of the Bayesian network model is that it does not require time-series data for learning purpose, which makes it applicable to most microarray datasets.

When generating gene regulatory models, the following key factors were considered:

- *Number of genes*: the number of genes might largely affect the performances of the learning techniques.
- *Number of source genes*: the source genes are the genes corresponding to the root vertices in a network model. In reality, researchers typically introduce treatments that may drastically change the expression levels of specific genes and observe changes in expression levels of other genes to infer gene regulatory relations. The source genes correspond to the genes that are directly affected by such treatments. Since typical gene expression dataset comprises several groups of observations under different treatments, the network model behind the observational data may contain several source genes.
- *Density of the network model*: The density of a network model is typically defined as the number of edges in the network over the total number of possible edges. Density of the network significantly affects the network learning performance. We used a fan-out factor (defined as the maximum number of decedents or downstream genes of one gene) to control the network density. Because gene regulatory network is generally believed to be sparse [4] we set a relatively small fan-out factor.
- *Gene regulatory relation types*: Two types of edges were included into regulatory models to represent the activation and inhibition relations. We provide simple interpretations of the biological meaning of the gene activation and inhibition relations using conditional probabilities. Consider a simple relation $X \rightarrow Y$ and assume there are no other parent (upstream) genes of gene Y . The ideal activation and inhibition relations can be described by the conditional probabilities in Table 3 (with 0 noise level).
- *Noise level*: We introduced a noise level p into the conditional probabilities (as presented in Table 3) to capture uncertainties in gene expression levels. This noise factor aggregates many limitations of gene expression data, such as measurement errors, ignored factors that affect gene expression levels, etc.
- *Aggregation of effects of multiple parent genes*: The formula we used to derive the probabilities of the

Table 3
Conditional probabilities of activation and inhibition relations (noise level of p)

	$\frac{x}{y}$	$\frac{1}{1}$	$\frac{1}{0}$	$\frac{1}{-1}$	$\frac{0}{1}$	$\frac{0}{0}$	$\frac{0}{-1}$	$\frac{-1}{1}$	$\frac{-1}{0}$	$\frac{-1}{-1}$
Activation	$p(Y=y X=x)$	$1-p$	$p/2$	$p/2$	$p/2$	$1-p$	$p/2$	$p/2$	$p/2$	$1-p$
Inhibition	$p(Y=y X=x)$	$p/2$	$p/2$	$1-p$	$p/2$	$1-p$	$p/2$	$1-p$	$p/2$	$p/2$

Table 4
Pre-defined regulatory network models for simulation

Network model	No. of genes	No. of source genes	Fan-out factor	Noise level	No. of relations
G1	10	2	3	0.2	12
G2	50	5	4	0.2	83
G3	200	5	4	0.2	392

expression level of a gene (Z) given expression levels of two parent genes (X and Y) is presented in (12). The formula can be easily extended to compute probabilities of expression level of genes given expression levels of multiple parent genes.

$$p(Z = z | X = x, Y = y) = \sum_{z_1 + z_2 = z} p(Z = z_1 | X = x) p(Z = z_2 | Y = y) \quad (12)$$

where $x, y, z = -1, 0$ or 1 , $p(Z = z | X = x)$ and $p(Z = z | Y = y)$, and are defined as in Table 3.

Based on the factors described above, three gene regulatory models were generated using a random procedure. The key factors of the three models are presented in Table 4. The three network models (G1, G2, and G3) contained 10, 50, and 200 nodes, respectively. G1 and G2 represent typical scales of regulatory network analysis reported in the literature, while G3 was used to evaluate the scalability of the proposed algorithms in analyzing large-scale networks. We employed the probabilistic logic sampling method proposed by Henrion [24] For each network model, two simulation datasets with sample sizes of 40 and 80 were generated. These sample sizes were selected to represent typical sample sizes of microarray datasets as those of the real datasets used in the empirical evaluation section.

4.1.2. Simulation-based evaluation results

Gene regulatory network models based on the six simulation datasets were generated using the Bayesian network and association rule techniques described in the previous section. To evaluate the inferred network models, we used an accuracy measure as described in (13).

$$\text{Network accuracy} = \frac{\text{Number of relations within the top } K \text{ relations with the highest strength in the inferred model that match with the relations in the true model}}{K} \quad (13)$$

where K is the number of relations in the true model.

Three versions of the network accuracy measure were computed based on different relation match definitions, including an *exact match*, a *directional match*, and a *non-directional match*. The non-directional match is the basic match definition as it requires only the two relations in comparison to involve the identical genes. A directional match further requires that the directions of the two relations under comparison are identical. An exact match is the strongest relation match. It requires two relations to have identical relation types (activation or inhibition) in addition to satisfying the directional match requirements. The exact match, direction match, and non-directional match network accuracy measures (denoted as NA-EM, NA-DM, and NA-NM) could provide a complete picture of how well an inferred model matches with the true model.

We applied the modified and unmodified Bayesian network learning and association rule mining algorithms to analyze the six simulated datasets based on the three pre-defined gene regulatory network models (G1, G2, and G3). We then used the three types of network accuracy measures introduced previously to evaluate the performances of these four algorithms. We next report the evaluation results of the modified algorithms followed by the results of their unmodified counterparts as comparison benchmarks.

4.1.2.1. Modified algorithm results. The network accuracy measures of the inferred network models using the MBN and MAR algorithms with the six simulation datasets are summarized in Table 5. The key observations of the simulation-based evaluation results presented are summarized as follows.

- Both the MBN and MAR algorithms achieved relatively high network accuracy measures for all three network models. The exact and non-directional match network accuracies of the two algorithms were in the range of 8–32% while the non-directional match network accuracies were in the range of 45–75%. Considering the limited sample sizes and noise introduced during the simulation, these accuracies show the great potential of the two algorithms.
- Both the MBN and MAR algorithms showed roughly invariant performance across different network sizes. We did not observe a significant decrease in network accuracies for G2 and G3 compared to G1. Even for the large network model with 200 nodes (G3), both algorithms achieved about 25% exact match and directional match network accuracies and over 50% non-directional match network accuracies when 80

Table 5
Simulation-based evaluation results of modified algorithms

True model	Sample size	No. of relations	MBN						MAR					
			NA-EM		NA-DM		NA-NM		NA-EM		NA-DM		NA-NM	
G1	40	12	2	16.67%	2	16.67%	7	58.33%	1	8.33%	1	8.33%	6	50.00%
G1	80	12	2	16.67%	2	16.67%	9	75.00%	1	8.33%	1	8.33%	8	66.67%
G2	40	83	19	22.89%	20	24.10%	37	44.58%	24	28.92%	24	28.92%	39	46.99%
G2	80	83	19	22.89%	21	25.30%	46	55.42%	26	31.33%	26	31.33%	47	56.63%
G3	40	392	90	22.96%	94	23.98%	172	43.88%	57	14.54%	57	14.54%	116	29.59%
G3	80	392	98	25.00%	101	25.77%	232	59.18%	102	26.02%	102	26.02%	199	50.77%
Avg				21.18%		22.08%		56.07%		19.58%		19.58%		50.11%

samples were available. These results showed that the two proposed algorithms are especially suitable for large-scale regulatory network analysis from microarray data.

- In general, the MBN algorithm showed slightly better performance than the MAR algorithm, as measured by the average network accuracies of all six datasets. With the G2 dataset the MAR algorithm only slightly outperformed the MBN algorithm.

4.1.2.2. *Unmodified algorithms results as benchmark.* For comparison purposes, we also report the performances of the original information-theory-based Bayesian network algorithm and standard association rule mining algorithms on the six simulation datasets.

We implemented the original information-theory-based Bayesian network algorithm [11] and obtained the network accuracies shown in Table 6 (the BN algorithm results). The average accuracies of exact match, directional match, and non-directional match achieved by the BN algorithm were 4.85%, 10.51%, and 45.86%, respectively. They were consistently lower than those achieved by the MBN algorithm (21.01%, 21.95%, and 55.51%) as presented in Table 5. The poor performance of the original Bayesian network algorithm showed that the dimensionality problem of typical microarray data seriously limited the applicability of the algorithm. The network accuracies of the BN algorithm were especially

low for G2 and G3 because with more genes the dimensionality problem was more severe.

We also implemented the association rule mining algorithm (the AR algorithm) for gene regulatory relation analysis reported in [52]. In their formulation the expression level of gene X was encoded by three Boolean variables, $\langle X\uparrow, X\#, X\downarrow \rangle$. So rules generated by the AR algorithm were in the forms of “ $A\uparrow \rightarrow B\uparrow$ ” and “ $C\uparrow \rightarrow D\downarrow$.” To produce the more intuitive and comparable gene regulatory relations as those generated by the MAR algorithm, we aggregated these relations in the following way: from either “ $A\uparrow \rightarrow B\uparrow$ ” or “ $A\downarrow \rightarrow B\downarrow$ ” we inferred “ $(A \overset{\pm}{\rightarrow} B)$ ” from either “ $A\uparrow \rightarrow B\downarrow$ ” or “ $A\downarrow \rightarrow B\uparrow$ ” we inferred “ $(A \overset{-}{\rightarrow} B)$.” Table 6 shows that the average accuracies of exact match, directional match, and non-directional match achieved by the AR algorithm were 16.81%, 16.81%, and 47.51%, respectively. Accuracy measures were consistently lower than those achieved by the MAR algorithm (19.58%, 19.58%, and 50.11%) as presented in Table 5. Unlike the BN algorithm, the AR algorithm achieved comparable network accuracies, showing that it was less constrained by the dimensionality problem of the microarray data. The relatively poor performance of the AR algorithm may have been partly related to the rule aggregation process and larger search space resulting from the Boolean representation of the gene expression levels.

Table 6
Simulation-based evaluation results of unmodified algorithms

True model	Sample size	No. of relations	BN						AR					
			NA-EM		NA-DM		NA-NM		NA-EM		NA-DM		NA-NM	
G1	40	12	1	8.33%	2	16.67%	6	50.00%	2	16.67%	2	16.67%	7	58.33%
G1	80	12	0	0.00%	1	8.33%	7	58.33%	2	16.67%	2	16.67%	7	58.33%
G2	40	83	4	4.82%	7	8.43%	29	34.94%	11	13.25%	11	13.25%	34	40.96%
G2	80	83	5	6.02%	10	12.05%	43	51.81%	19	22.89%	19	22.89%	41	49.40%
G3	40	392	19	4.85%	33	8.42%	126	32.14%	53	13.52%	53	13.52%	135	34.44%
G3	80	392	20	5.10%	36	9.18%	188	47.96%	70	17.86%	70	17.86%	171	43.62%
Avg				4.85%		10.51%		45.86%		16.81%		16.81%		47.51%

4.1.2.3. Parameter setting and computational issues.

The central parameter for the MBN algorithm is the threshold ε , which is involved in the conditional independency (CI) test that is critical for the decisions to add or remove particular edges in the network model. The network accuracy of the inferred network models and the computational efficiency are both dependent on the value of this threshold. In our experiments, the best network accuracies were achieved when ε was set around 0.3. (The results reported previously were obtained with ε set to be 0.3.) The network accuracies deteriorate as ε deviates from 0.3 to both smaller and larger values. For example, the exact-match network accuracies of the network learned from the 80-sample dataset generated from G3 for different values of ε are the following: 19.64% ($\varepsilon=0.01$), 20.66% ($\varepsilon=0.05$), 21.68% ($\varepsilon=0.1$), 25.00% ($\varepsilon=0.3$), 19.39% ($\varepsilon=0.5$), and 13.27% ($\varepsilon=0.8$). In general the smaller the value of ε , the longer the computation time, because more CI tests need to be performed with a smaller threshold. For example, learning the network from the 80-sample dataset generated from G3 with ε set to be 0.1 roughly doubles the computation time needed with ε set to be 0.3¹. We also observed that the BN algorithm showed similar patterns in network accuracy and computation time with respect to the value of ε . The threshold ε also determines the number of relations to be included in the learned networks. Generally the larger the value of ε , the smaller the number of edges included in the learned network model. The key parameters of the association rule algorithm are the minimum support (*minsup*) and minimum confidence (*minconf*), the values of which determine the number of relations to be included into the learned networks. Because we rely on the relation strength to derive networks of different levels of confidence at the last stage of both algorithms, the total numbers of relations included into the networks are often less relevant than the ranking quality of relation strength within the learned network. This is consistent with the network accuracy measures we employed in this study, which only considers the top K relations as opposed to all relations in the network. Our experiments showed that the network accuracy measures of the MAR

learning results were largely insensitive to both *minsup* and *minconf*, except for the poor accuracy measures for very large values of both parameters (*minsup*>0.5 and *minconf*>0.8) when fewer than the number of relations in the true model can be included into the learned network. In our reported experiments, *minsup* and *minconf* were set to be 0.2 and 0.5, respectively.

In summary, the simulation-based evaluation results showed that acceptable network accuracies were achieved although compromises were made during the algorithm design to accommodate the high dimensionality and small sample sizes of the data available. Based on our simulation experiments, the two modified algorithms outperformed the standard Bayesian network learning and association rule mining algorithms. Genetic pathway researchers could use the proposed algorithms to summarize microarray data of large numbers of genes into approximate regulatory network models. They could expect to see one out of four relations in the learned gene network to have correct relation type and directionality in these approximate models. Without considering relation type and directionality about one out of two learned relations could be correct relations. These promising simulation-based evaluation results, however, need to be interpreted with caution. The simulated network models were largely simplified versions of real genetic regulatory processes. Various sources of noise from the experimental procedures and regulation processes themselves might not be captured by the simulated noise effects.

4.2. Empirical evaluation

Although simulation-based evaluation provides an estimate of the usefulness of the proposed gene regulatory network analysis algorithms, empirical evaluations that compare analysis results based on real experimental data with existing knowledge of gene regulatory relations are still needed to provide direct evidence on the potential usefulness of the approaches. In this section, we present an empirical evaluation study that relied on a domain expert's judgment for assessing the interestingness of the learned regulatory relations. In our study, we employed two real microarray datasets, an *S. cerevisiae* dataset and a *H. sapiens* dataset, to evaluate the two proposed algorithms.

4.2.1. Datasets and data preprocessing

The *S. cerevisiae* yeast cell-cycle dataset of Spellman et al. [47] has been used frequently in previous microarray data analysis studies, which provide good benchmarks for evaluation. This dataset contains 76

¹ We currently implemented Cheng's original algorithm and the modified Bayesian network learning algorithm using MS SQL Server database and stored procedures for fast prototyping. Our current MBN algorithm implementation takes about 40 min to learn a network for the 200-gene 80-sample dataset on a high-end Windows workstation. Based on our previous experiences, substantial reduction in computation time can be achieved by using more efficient programming environments such as C++ and Python.

gene expression measurements of the mRNA levels of 6,177 *S. cerevisiae* ORFs. The experiments measured six time series under different cell cycle synchronization methods [19] The *H. sapiens* dataset was provided by Arizona Cancer Center. It contains 33 samples of 5306 human genes in total.

Based on our gene expression data representation, we applied a ternary discretization to both datasets. For the *S. cerevisiae* dataset, we used the same threshold as in Friedman et al.'s research [19] to perform discretization. With this threshold of 0.5 in logarithmic (base 2) scale, expression levels with ratio to the reference that were lower than $2^{-0.5}$ were considered to be under-expressed; levels higher than $2^{0.5}$ were considered over-expressed; expression levels between $2^{-0.5}$ and $2^{0.5}$ were considered as normal. For the *H. sapiens* dataset we selected a threshold value of 1 in logarithmic (base 2) scale, based on discussions with the domain scientists who conducted the experiments and had substantial experience with the specific dataset. The domain scientists suggested that the network extraction techniques be applied on the 200 genes with greatest expression variations across the 33 samples.

4.2.2. Evaluation results

Regulatory networks learned from real experimental data are typically difficult to evaluate, partly because large portions of the underlying true network remain unknown (which is exactly the reason regulatory network analysis from microarray data is needed). Without a true network as an evaluation benchmark, a less-than-ideal choice is to employ an expert's judgment (supported by detailed literature search) as an approximation of accumulated knowledge about the known portions of the underlying regulatory network.

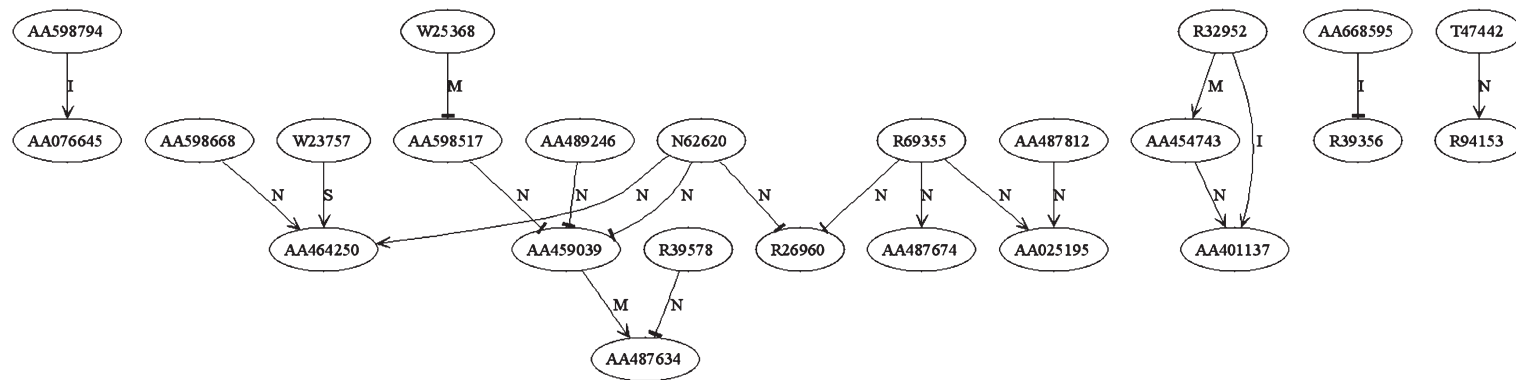
In our study, one domain scientist performed such an evaluation. For each of the four result sets learned from the two datasets using the two proposed learning algorithms, she evaluated the top 20 learned relations with greatest confidence involving known genes/proteins. In this study, a learned relation "*A* activates *B*" was labeled "*interesting*" if (1) *A* and *B* were documented to interact directly or indirectly, or (2) *direct* interactions exist between the homologs of *A* and *B* (homologs of *A* refer to those belonging to the same family of *A*). The relation was labeled "*maybe interesting*" if (1) *indirect* relations exist between the homologs of *A* and *B*, or (2) *A* and *B* are involved in similar functions or pathways. The relation was labeled "*not interesting*" if none of the criteria above was met. Relations involved with unknown genes/proteins were not evaluated in our study. In addition to the three

categories above, if *A* and *B* belong to the same family we labeled the relation as "*spurious*." In such relations, *A* and *B* do not necessarily interact with each other and the inferred relation might be spurious, reflecting the fact that a common regulator of *A* and *B* may exist. The expert's evaluation results were validated by two domain scientists who conduct research with the microarray dataset we used and were quite familiar with the genes involved.

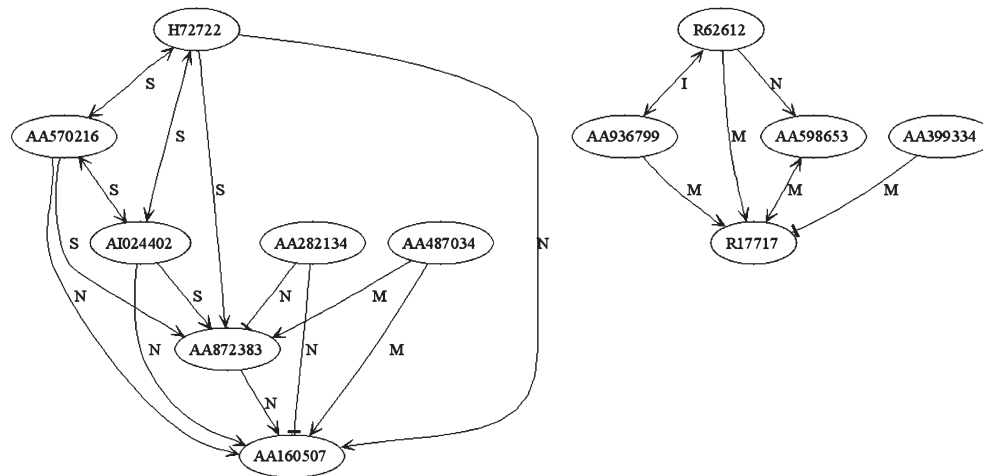
Fig. 2 presents the top 20 relations from the four learned gene networks that were evaluated by the domain expert. With only the top 20 relations from each learned network, we still observe some general topological differences between the MBN and MAR learning results. The networks learned by the MBN algorithm were generally sparser than those learned by the MAR algorithm. This observation is consistent with the fact that the MBN and Bayesian networks algorithm in general explicitly differentiate between direct and indirect relationships while the association rule mining algorithms do not explicitly make this distinction. We also observe that the MAR algorithm performed poorly in determining the direction of these top relations, especially for the yeast dataset.

Table 7 summarizes the expert evaluation results for the four learning result sets. Within the top 20 learned relations having the highest confidence measures using the MBN and MAR algorithms for the human and yeast datasets, the percentages of "*interesting*" and "*maybe interesting*" relations ranged from 20% to 35%. For the human dataset, the MBN algorithm learned more "*interesting*" relations (15%) but fewer "*maybe interesting*" (5%) relations than the MAR algorithm, while the MAR algorithm learned much more "*spurious*" relations (30% as compared to 5% by the MBN algorithm) and fewer "*not interesting*" relations (35% as compared to 65% by the MBN algorithm). For the yeast data set, the MBN and MAR algorithms learned similar numbers of "*interesting*" and "*maybe interesting*" relations. The MBN algorithm learned more "*spurious*" relations (45% as compared to 0% by the MAR algorithm) and fewer "*not interesting*" relations (35% as compared to 75% by the MAR algorithm). The results indicate that neither of the two algorithms outperformed the other for both datasets based on the expert's evaluation of the top 20 learned relations.

Table 8 lists the details of the "*interesting*" relations identified in the top 20 learned relations from the four result sets. One observation is that the interesting relations learned using the MBN and MAR algorithms did not overlap. In this sense, the two algorithms are complementary. By analyzing the learning results of the

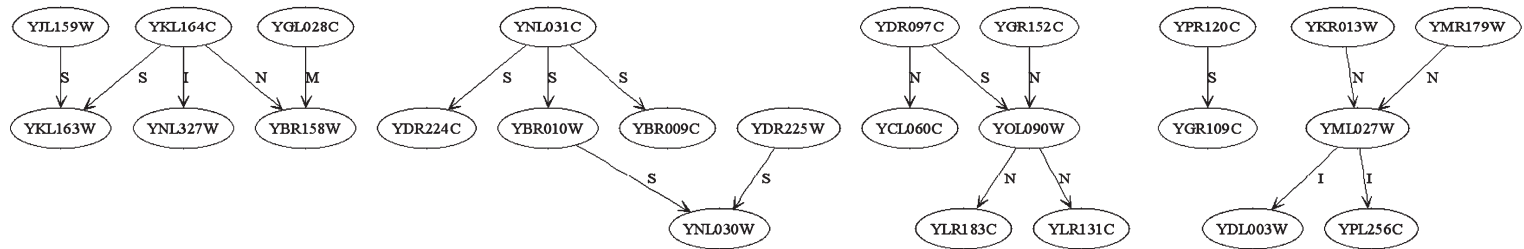


(a) Human dataset - MBN algorithm

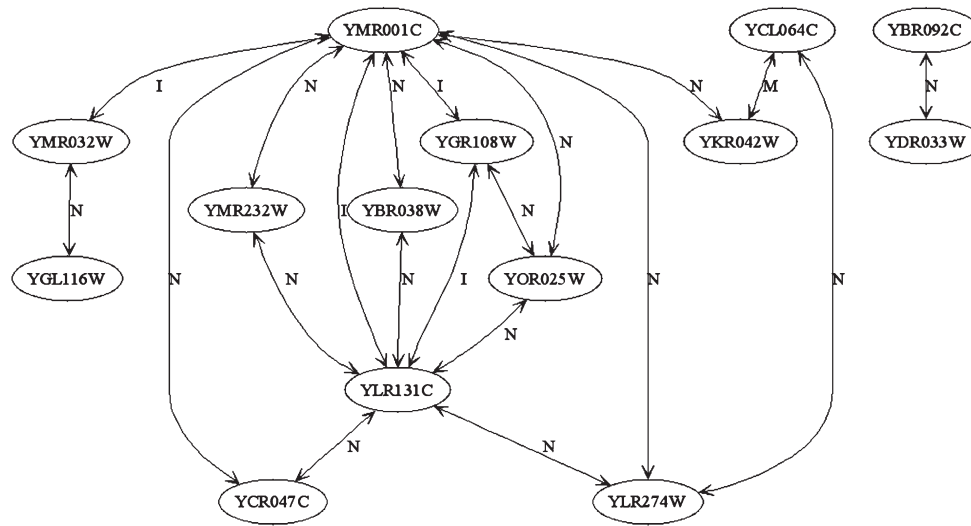


(b) Human dataset - MAR algorithm

Fig. 2. Top 20 relations with the greatest confidence involving known genes/proteins. Genes are labeled with accession Ids; undirected relations are presented as bidirectional arrows; arrowhead shapes denote relation type: open—activation, tee—inhibition; arrow labels denote evaluation results: I—interesting; M—maybe interesting; N—not interesting; S—spurious.



(c) Yeast dataset - MBN algorithm



(d) Yeast dataset - MAR algorithm

Fig. 2 (continued).

Table 7
Expert evaluation results

Category	MBN_human	MAR_human	MBN_yeast	MAR_yeast
Interesting	3 (15.0%)	1 (5.0%)	3 (15.0%)	4 (20.0%)
Maybe interesting	3 (15.0%)	6 (30.0%)	1 (5.0%)	1 (5.0%)
Interesting or maybe interesting	6 (30.0%)	7 (35.0%)	4 (20.0%)	5 (25.0%)
Not interesting	13 (65.0%)	7 (35.0%)	7 (35.0%)	15 (75.0%)
Spurious	1 (5.0%)	6 (30.0%)	9 (45.0%)	0 (0.0%)

two algorithms using the six simulation datasets, we found that significant portions of the matched relations learned by the two algorithms are consistently different. For example, within the two sets of the exact-match relations learned from the 80-sample dataset of 200 genes (98 learned using the MBN algorithm and 101 learned using the MAR algorithm), only 42 relations appear in both sets. Within the two sets of the exact-match relations learned from the 40-sample dataset of 200 genes (90 learned using the MBN algorithm and 57 learned using the MAR algorithm), only 22 relations appear in both sets.

Although the learned relations could potentially help researchers propose new hypotheses that might eventually lead to new discoveries of gene regulatory relations, the relatively low percentages of the interesting relations indicate that the microarray technologies may not be ready to generate high-quality data to facilitate construction of large-scale regulatory networks from experimental data. New experimental technologies need to be developed to directly measure activity levels of the genes and other relevant entities such as proteins and small molecules. New network learning algorithms or further customization on existing algorithms are also needed.

5. Conclusion and future directions

In this study, we examined two algorithms for large-scale regulatory network learning: a modified Bayesian network algorithm (MBN algorithm) and a modified association rule mining algorithm (MAR algorithm). We conducted two types of evaluations to assess the practical value of these two techniques in helping researchers analyze large amounts of gene expression data. The simulation-based evaluation results indicated that the two techniques could infer about 20% of the relations in pre-defined network models. If directionality and relation type were not considered the two algorithms could infer about 50% of the relations in the actual models. Our simulation experiments also showed that both modified algorithms consistently outperformed unmodified Bayesian network learning and association rule mining algorithms across all simulation datasets, confirming that the modifications we had made on both algorithms partly alleviated the dimensionality problem of using microarray data for regulatory network learning. Empirical evaluation results showed that the top relations learned by the two proposed algorithms contained 20–30% “interesting” or “maybe interesting” relations that had potential as experiment hypotheses for

Table 8
Interesting relations from the expert evaluation results

Dataset	Algorithm	Accession Id1	Gene1	Accession Id2	Gene2	Direction ^a	Type
Human	MBN	AA598794	connective tissue growth factor	AA076645	RAB21, member RAS oncogene family	yes	activation
		AA668595	PIG3	R39356	p53	yes	inhibition
		R32952	S100P	AA401137	LCN2	yes	activation
Yeast	MAR	R62612	fibronectin 1	AA936799	MMP2	no	activation
		YML027W	YOX1; Homeodomain protein that binds leu-tRNA gene	YDL003W	Mitotic Chromosome Determinant ^b	yes	activation
	MAR	YML027W	YOX1	YPL256C	role in cell cycle START	yes	activation
		YKL164C	PIR1	YNL327W	EGT2	yes	activation
		YMR001C	CDC5	YLR131C	ACE2	no	activation
		YMR001C	CDC5	YGR108W	CLB1	no	activation
		YGR108W	CLB1	YLR131C	ACE2	no	activation
		YMR001C	CDC5	YMR032W	CYK2	no	activation

^a ‘Yes’ means the direction of the learned relation is from gene1 to gene2, while ‘no’ means the direction of the relation is uncertain.

^b Similar to *S. pombe* RAD21; may function in chromosome morphogenesis from S phase through mitosis.

further investigation. Our general conclusion is that regulatory network analysis on microarray data can capture large portions of underlying regulatory structures. The empirical results also indicate that specialized microarray experiments need to be conducted to rigorously evaluate the effectiveness of regulatory network learning from microarray data.

In our future research, we will perform formal analysis of the effect of the modifications we have introduced to the Bayesian network learning and association rule mining algorithms. We will perform large-scale simulations to further verify the robustness of the network accuracy performance measures we have reported in this paper. We will also assess other network learning techniques to study their performances in learning large-scale genetic regulatory networks from microarray data. Our general objective will be analyzing different algorithms using simulation datasets and real experimental datasets to provide systematic practical guidance for learning large-scale regulatory networks from microarray data. A longer-term future direction of this work is to assess the real value of the large-scale rough genetic regulatory networks learned from small-sample-size gene expression data by systematic long-term user studies that involve researchers using such networks in their daily research activities. Another important practical extension of our research is to integrate the core algorithm development into an interactive gene regulatory network analysis and visualization system, which allows the users to input microarray datasets and manipulate the learning algorithms and results by interactively setting the domain-dependent algorithmic and visualization parameters.

Acknowledgements

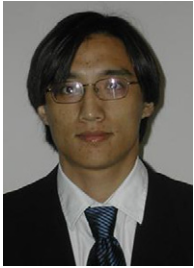
This research is supported by the grant: NIH/NLM, 1 R33 LM07299-01, 2002-2005, “Genescene: a Toolkit for Gene Pathway Analysis.” We would like to thank Dan McDonald for valuable comments and discussions. We would also like to thank Jesse Martinez, Ryan Falsey, and Kerri Kislin from the Arizona Cancer Center for expert evaluations and valuable discussions.

References

- [1] R. Agrawal, T. Imielinski, A. Swami, Mining association rules between sets of items in large databases, *Proceedings of the IACM-SIGMOD International Conference on Management of Data*, 1993, pp. 207–216.
- [2] A.A. Alizadeh, M.B. Eisen, R.E. Davis, C. Ma, E. Al, Distinct types of diffuse large B-cell lymphoma identified by gene expression profiling, *Nature* 403 (2000) 503–511.
- [3] A.P. Arkin, P.-D. Shen, J. Ross, Deduction of a complex reaction mechanism from measured time series: verification of the theory of statistical construction, *Science* 277 (5330) (1997) 1275.
- [4] A. Arnone, B. Davidson, The hardwiring of development: organization and function of genomic regulatory systems, *Development* 124 (1997) 1851–1864.
- [5] W.-H. Au, K.C.C. Chan, Mining fuzzy association rules in a bank-account database, *IEEE Transactions on Fuzzy Systems* 11 (2) (2003) 238–248.
- [6] C. Becquet, S. Blachon, B. Jeudy, J.-F. Boulicaut, O. Gandrillon, Strong-association-rule mining for large-scale gene-expression data analysis: a case study on human SAGE data, *Genome Biology* (2002).
- [7] D.D. Bernardo, T.S. Gardner, J.J. Collins, Robust identification of large genetic networks, *Proceedings of the Pacific Symposium on Biocomputing*, 2004, pp. 486–497.
- [8] D. Berrar, W. Dubitzky, M. Granzow, R. Eils, Analysis of gene expression and drug activity data by knowledge-based association mining, *Proceedings of the Critical Assessment of Microarray Data Analysis Techniques (CAMDA '01)*, 2001, pp. 25–28.
- [9] T. Chen, V. Filkov, S. Skiena, Identifying gene regulatory networks from experimental data, *Proceedings of the RECOMB*, 1999, pp. 94–103.
- [10] R. Chen, K. Sivakumar, H. Kargupta, Collective mining of Bayesian networks from distributed heterogeneous data, *Knowledge and Information Systems* 6 (2) (2004) 164–187.
- [11] J. Cheng, R. Greiner, J. Kelly, D.A. Bell, W. Liu, Learning Bayesian networks from data: an information-theory based approach, *The Artificial Intelligence Journal* 137 (2002) 43–90.
- [12] D.M. Chickering, D. Geiger, D. Heckerman, Learning Bayesian Networks is NP-Hard, Technical Report MSR-TR-94-17, Microsoft Research, Microsoft Corporation, 1994.
- [13] D.M. Chickering, C. Meek, Monotone DAG Faithfulness: A bad Assumption, Microsoft Research, 2003.
- [14] C.K. Chow, C.N. Liu, Approximating discrete probability distributions with dependence trees, *IEEE Transactions on Information Theory* 14 (1968) 462–467.
- [15] L. Chrisman, P. Langley, S. Bay, A. Pohorille, Incorporating biological knowledge into evaluation of causal regulatory hypotheses, *Proceedings of the Pacific Symposium on Biocomputing*, 2003, pp. 128–139.
- [16] H. De Jong, Modeling and simulation of genetic regulatory systems: a literature review, *Journal of Computational Biology* 9 (2002) 67–103.
- [17] P. D’haeseleer, X. Wen, S. Fuhrman, R. Somogyi, Linear modeling of mRNA expression levels during CNS development and injury, *Proceedings of the Pacific Symposium on Biocomputing '99*, 1999, pp. 41–52.
- [18] M.B. Eisen, P.T. Spellman, P.O. Brown, D. Botstein, Cluster analysis and display of genome-wide expression patterns, *Proceedings of the National Academy of Sciences of the United States of America* 95 (1998) 14863–14868.
- [19] N. Friedman, M. Linial, I. Nachman, D. Pe’er, Using Bayesian network to analyze expression data, *Journal of Computational Biology* 7 (2000) 601–620.
- [20] T.R. Golub, D.K. Slonim, P. Tamayo, C. Huard, E. Al, Molecular classification of cancer: class discovery and class prediction by gene expression monitoring, *Science* 286 (1999) 531–537.
- [21] A.J. Hartemink, D.K. Gifford, T.S. Jaakkola, R.A. Young, Combining location and expression data for principled discovery of genetic regulatory network models, *Proceedings of the Pacific Symposium on Biocomputing*, 2002, pp. 437–449.

- [22] D. Heckerman, A tutorial on learning Bayesian networks, 1995, Technical Report MSR-TR-95-06, Microsoft Research, 1995.
- [23] D. Heckerman, M. Wellman, A. Mamdani, Special issue on uncertainty AI, Communications of the ACM 38 (3) (1995) 24–47.
- [24] M. Henrion, Propagating uncertainty in Bayesian networks by probabilistic logic sampling, Uncertainty in Artificial Intelligence 2 (1988) 149–163.
- [25] S. Imoto, T. Higuchi, T. Goto, K. Tashiro, S. Kuhara, S. Miyano, Estimating gene networks by Bayesian networks from microarrays and biological knowledge, Proceedings of the 11th Inter. Conf. on Intelligent Systems for Molecular Biology, 2003.
- [26] T. Ito, T. Chiba, R. Ozawa, M. Yoshida, M. Hattori, Y. Sakaki, A comprehensive two-hybrid analysis to explore the yeast protein interactome, Proceedings of the National Academy of Sciences 98 (8) (2001) 4569–4574.
- [27] A. Jagota, Microarray data analysis and visualization, Bioinformatics, The Bay Press, 2001.
- [28] T.-K. Jenssen, A. Lagreid, J. Komorowski, E. Hovig, A literature network of human genes for high-throughput analysis of gene expression, Nature Genetics 28 (2001) 21–28.
- [29] X.-R. Jiang, L. Gruenwald, Microarray gene expression data association rules mining based on JG-Tree, Proceedings of the 14th International Workshop on Database and Expert Systems Applications, 2003, pp. 27–31.
- [30] S. Kauffman, The Origin of Order: Self-organization and Selection in Evolution, Oxford University Press, 1993.
- [31] M. Khabzaoui, C. Dhaenens, E.-G. Talbi, A multicriteria genetic algorithm to analyze microarray data, Proceedings of the Congress on Evolutionary Computation (CEC 2004), 2004, pp. 1874–1881.
- [32] P. Kotala, A. Perera, J.K. Zhou, S. Mudivarthy, W. Perrizo, E. Deekard, Gene expression profiling of DNA microarray data using Peano count trees (P-Trees), Proceedings of the Online Proceedings of the First Virtual Conference on Genomics and Bioinformatics, 2001.
- [33] I. Lee, S.V. Date, A.T. Adai, E.M. Marcotte, A probabilistic functional network of yeast genes, Science 306 (26) (2004) 1555–1558.
- [34] S. Liang, S. Fuhrman, R. Somogyi, REVEAL, a general reverse engineering algorithm for inference of genetic network architectures, Proceedings of the Pacific Symposium on Biocomputing, 1998, pp. 18–29.
- [35] F. McCormick, Signaling networks that cause cancer, Trends in Cell Biology 9 (1999) M53–M56.
- [36] E. Mjolsness, D.H. Sharp, J. Reinitz, A connectionist model of development, Journal of Theoretical Biology 152 (4) (1991) 429–454.
- [37] E. Mjolsness, T. Mann, R. Castano, B. Wold, From coexpression to coregulation: an approach to inferring transcriptional regulation among gene classes from large-scale expression data, Neural Information Processing Systems 12 (1999) 928–934.
- [38] K. Murphy, S. Mian, Modeling gene expression data using dynamic Bayesian networks, Technical Report, Computer Science Division, University of California, Berkeley, 1999.
- [39] T. Naitou, K. Satou, E. Furuichi, S. Kuhara, T. Takagi, A system for finding association rules from microarray data and public databases, Proceedings of the Genome Informatics, 2000, pp. 356–357.
- [40] D. Pe'er, A. Regev, G. Elidan, N. Friedman, Inferring subnetworks from perturbed expression profiles, Bioinformatics 17 (Suppl 1) (2001) S215–S224.
- [41] J. Pearl, Probabilistic reasoning in intelligent systems: networks of plausible inference, Morgan Kaufmann, 1988.
- [42] M. Pellegrini, E.M. Marcotte, M.J. Thompson, D. Eisenberg, T.O. Yeates, Assigning protein functions by comparative genome analysis: protein phylogenetic profiles, Proceedings of the National Academy of Sciences 96 (8) (1999) 4285–4288.
- [43] J.A. Rushing, H. Ranganath, T.H. Hinke, S.J. Graves, Image segmentation using association rule features, IEEE Transactions on Image Processing 11 (5) (2002) 558–567.
- [44] E. Segal, H. Wang, D. Koller, Discovering molecular pathways from protein interaction and gene expression data, Bioinformatics 19 (Suppl: 1) (2003) i264–i272.
- [45] E.V. Someren, L. Wessels, M. Reinders, Linear modeling of genetic networks from experimental data, Proceedings of the Eighth International Conference on Intelligent Systems for Molecular Biology, 2000, pp. 355–366.
- [46] R. Somogyi, S.A. Sniegowski, Modeling the complexity of genetic networks: Understanding multigenic and pleiotropic regulation, Complexity 1 (6) (1996) 45–63.
- [47] P. Spellman, G. Sherlock, M. Zhang, V. Iyer, K. Anders, M. Eisen, P. Brown, D. Botstein, B. Futcher, Comprehensive identification of cell cycle-regulated genes of the yeast *Saccharomyces cerevisiae* by microarray hybridization, Molecular Biology of the Cell 9 (1998) 3273–3297.
- [48] P. Spirtes, C. Glymour, R. Scheines, An algorithm for fast recovery of sparse causal graphs, Social Science Computer Review 9 (1991) 62–72.
- [49] S. Srinivas, S. Russell, A. Agogino, Automated construction of sparse Bayesian networks from unstructured probabilistic models and domain information, in: M. Henrion, R.D. Shachter, L.N. Kanal, J.F. Lemmer (Eds.), Uncertainty in Artificial Intelligence, North-Holland, Amsterdam, 1990.
- [50] H. Toh, K. Horimoto, Inference of a genetic network by a combined approach of cluster analysis and graphical Gaussian modeling, Bioinformatics 18 (2) (2002) 287–297.
- [51] A.H.Y. Tong, M. Evangelista, A.B. Parsons, H. Xu, G.D. Bader, N. Pagé, M. Robinson, S. Raghibizadeh, C.W.V. Hogue, H. Bussey, B. Andrews, M. Tyers, C. Boone, Systematic genetic analysis with ordered arrays of yeast deletion mutants, Science 294 (5550) (2001) 2364–2368.
- [52] A. Tuzhilin, G. Adomavicius, Handling very large number of association rules in the analysis of microarray data, Proceedings of the SIGKDD '02, 2002.
- [53] P. Uetz, L. Giot, G. Cagney, T.A. Mansfield, R.S. Judson, J.R. Knight, D. Lockshon, V. Narayan, M. Srinivasan, P.P.A. Qureshi-Emili, Y. Li, B. Godwin, D. Conover, T. Kalbfleisch, G. Vijayadmodar, M. Yang, M. Johnston, S. Fields, J.M. Rothberg, A comprehensive analysis of protein–protein interactions in *Saccharomyces cerevisiae*, Nature (2000) 623–627.
- [54] K. Vermeulen, D.R. Van Bockstaele, Z.N. Berneman, The cell cycle: a review of regulation, deregulation and therapeutic targets in cancer, Cell Proliferation 36 (2003) 131–149.
- [55] M. Wahde, J. Hertz, Coarse-grained reverse engineering of genetic regulatory networks, Biosystems 55 (1999) 129–136.
- [56] D. Weaver, C. Workman, G. Stormo, Modeling regulatory networks with weight matrices, Proceedings of the Pacific Symposium on Biocomputing, 1999, pp. 112–123.
- [57] N. Wermuth, S. Lauritzen, Graphical and recursive models for contingency tables, Biometrika 72 (1983) 537–552.
- [58] L. Wessels, E.V. Someren, M. Reinders, A comparison of genetic network models, Proceedings of the Pacific Symposium on Biocomputing, 2001.

- [59] R. Wolff, A. Schuster, Association rule mining in peer-to-peer systems, *IEEE Transactions on Systems, Man and Cybernetics, Part B* 34 (6) (2004) 2426–2438.
- [60] M.L. Wong, S.Y. Lee, K.S. Leung, Data mining of Bayesian networks using cooperative coevolution, *Decision Support Systems* 38 (3) (2004).
- [61] S. Wright, Correlation and causation, *Journal of Agricultural Research* 20 (1921) 557–585.



Zan Huang is an Assistant Professor of the Department of Supply Chain and Information Systems at the Pennsylvania State University. His research interests include recommender systems, data mining and text mining for bioinformatics and financial applications, knowledge management technologies, and experimental economics-related research for electronic markets. His articles have appeared in *ACM Transactions on Information Systems*, *Journal of the American Society for*

Information Science & Technology, *Decision Support Systems*, *Journal of Nanoparticle Research* and other publications. He received the BEng degree in Management Information Systems from Tsinghua University, Beijing, China and the PhD degree in Management Information Systems from the University of Arizona.



Jiexun Li (jiexun@eller.arizona.edu) is a doctoral student in the Department of Management Information Systems and research associate in the Artificial Intelligence Laboratory at the University of Arizona, Tucson, AZ. He earned his MS in MIS from the School of Economics and Management, Tsinghua University, China. His research interests include data mining in biomedical and business applications, knowledge management, and information retrieval.



Hua Su is a Postdoctoral Research Associate in the Artificial Intelligence Laboratory at the University of Arizona. She received her PhD in Plant Sciences and MS in Management Information Systems from the University of Arizona. Her research interests reside in biological and medical data mining and knowledge discovery, and their applications in bioinformatics and genomics. Specifically, she is interested in integration of biological data from various sources, development of

data mining tools for prediction of gene function and interaction from genetic and genomic data, and modeling and reconstruction of genetic pathways and networks from experimental data and biomedical literatures.



George S. Watts received a BS in Chemistry at the University of Delaware in 1993 before relocating to Tucson for a PhD in Pharmacology and Toxicology at the University of Arizona. While earning his PhD studying epigenetic control of gene expression through DNA methylation Dr. Watts began working on microarray technology. Dr. Watts continued developing microarrays at the Arizona Cancer Center as a post-doctoral fellow and became a research assistant professor in 2000.

Since that time Dr. Watts has run the Genomics Shared Service while pursuing his own research in cancer and toxicology. Currently the Dr. Watts is investigating esophageal adenocarcinoma, a cancer with a poor prognosis that arises from a common change in the tissue lining the esophagus referred to as Barrett's esophagus. Dr. Watts has analyzed the changes in gene expression that are associated with progression from Barrett's esophagus to cancer. The genes identified during microarray-based studies are being developed as biomarkers and targets for improving therapy.



Hsinchun Chen is McClelland Professor of Management Information Systems at the University of Arizona and Andersen Consulting Professor of the Year (1999). He received the BS degree from the National Chiao-Tung University in Taiwan, MBA from the SUNY Buffalo, and the PhD degree in Information Systems from New York University. He is author of six books and more than 100 SCI journal articles covering intelligence analysis, data/text/web mining, digital library, knowl-

edge management, medical informatics, and Web computing. He serves on the editorial board of *Journal of the American Society for Information Science and Technology*, *ACM Transactions on Information Systems*, *IEEE Transactions on Systems, Man, and Cybernetics*, *IEEE Transactions on Intelligent Transportation Systems*, and *Decision Support Systems*. Dr. Chen is a Scientific Counselor for National Library of Medicine (NLM), USA, and has served as an advisor for major National Science Foundation (NSF), Department of Justice (DOJ), Department of Homeland Security (DHS), NLM, and other international research programs in digital library, digital government, medical informatics, and national security. Dr. Chen is founding director of the UA Artificial Intelligence Laboratory and Hoffman E-Commerce Laboratory.