IDENTIFYING GENE INTERACTIONS FOR TIME SERIES MICROARRAY DATA USING DYNAMIC BAYESIAN NETWORKS AND EXTERNAL BIOLOGICAL KNOWLEDGE

by

Umut Ağyüz

B.S. Electronics Engineering, Işık University, 2008

Submitted to the Institute of Biomedical Engineering
in partial fulfillment of the requirements
for the degree of
Master of Science
in
Biomedical Engineering

Boğaziçi University
2013

ACKNOWLEDGMENTS

First of all, I would like to thank my thesis advisor, Prof. Dr. Ahmet Ademoğlu, for his academic support and friendly supervision. I am deeply grateful to my co-advisor, Assist. Prof. Dr. Hasan H. Otu, for his guidance, support, and suggestions. I am also grateful to TUBITAK for providing funding for this study (Project Number: 111E042).

I see myself obliged to thank Istanbul Bilgi Univesity Bioengineering, for its comfortable environment during the analysis of this study. I am indebted to Prof. Dr. Yorgo Istefanopulos for his leading me to the field of Biomedical Engineering. In addition, I feel grateful to my friends Claudia Tuncel, Melike Korucuoğlu, Haluk Doğan, Haluk Akalp, Sercan Kaya, Sercan Akdağ, Kadir Akman, Selçuk Ataker and Benjamin Lin for their great support during my master studies. Lastly, I want to thank my father, Yüksel Ağyüz and my uncle, İsmail Erkılıç. They have brought up me to these days with love and confidence.

ABSTRACT

IDENTIFYING GENE INTERACTIONS FOR TIME SERIES MICROARRAY DATA USING DYNAMIC BAYESIAN NETWORKS AND EXTERNAL BIOLOGICAL KNOWLEDGE

DNA hybridization arrays measure the expression levels for thousands of genes. These measurements provide us with a "snapshot" of transcription levels in the cell. A major challenge in computational biology is to identify the gene-protein, gene-gene, and protein-protein interactions using such measurements, as well as some biological features of cellular systems. In our study we aimed at building up our framework on the use of Bayesian networks. A Bayesian network is a graph-based model of joint multivariate probability distributions that captures properties of conditional independence between variables. Such models are deemed attractive for their ability to describe complex stochastic processes. They also provide a clear methodology for learning from observations, even for noisy ones. However, Bayesian Networks work only for stationary data, require prior information in model selection, and applies to acyclic directed graphs. Dynamic Bayesian network (DBN) is an improved model to overcome the cyclicity and stationary limitations.

Keywords: Gene Regulatory Networks, Structure Learning, Bayesian Networks, Dynamic BN, Gene Expression Profiles.

ÖZET

HARİCİ BİYOLOLOJİK BİLGİ VE DİNAMİK BAYES AĞLARI KULLANARAK GENLER ARASI ETKİLEŞİMLERİ TANIMLAMAK

DNA hybridlenme teknolojisi sayesinde DNA'yı ölçülebilir değerlere dönüştürülüp genlerin sinyal değerlerini elde edebilmekteyiz. Bu sinyal değerleri kullanılarak gen-gen, protein-gen ve protein-protein arasındaki etkileşimler çözülmeye çalışılmak, günümüz biyolojisinde önem kazanmaktadır. Bu tezde bu problemi Bayes Ağları (BA) kullanarak çözebilecek bir yöntem takip ettik. BA değişkenler arasındaki şartlı bağımsızlıkları kavrayabilen olasılık dağılımlarının modellenmesidir. BA ve benzeri modeller olasılıksal olayları çok iyi açıklayabildikleri için gözlemsel verilerden öğrenme yapılırken tercih edilirler. Veri gürültüye sahip olsa bile kullanabiliriz. BA modeli aynı zamanda bazı dezavantajlara da sahiptir. BA modeli sadece değişmeyen veriler için kullanılabilir ön bilgi gerektirir ve sadece döngü içermeyen yollar içerdiği için sınırlıdır. Bu gibi kısıtlamaların üstesinden gelebilmek için Dinamik Bayes Ağları geliştirilmiştir.

Anahtar Sözcükler: Gen Düzenleyici Ağlar, Bayes Ağları, Dinamik Bayes Ağları, Mikroerey, Yapı Öğrenme, Zaman Serisi Veriler.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS

D	Data
G	Graph
X_i	Node i
n	Total number of nodes
θ	Graph
$\Gamma(\cdot)$	Gamma function

LIST OF ABBREVIATIONS

BN	Bayesian Networks
DBN	Dynamic Bayesian Networks
mRNA	Messenger Ribonucleic Acid
DNA	Deoxyribonucleic Acid
DAG	Directed Acyclic Graphs
CPD	Conditional Probability Distribution
CPT	Conditional Probability Table
ML	Maximum Likelihood
MLE	Maximum Likelihood Estimations
EM	Expectation-Maximization
BNT	Bayes Net Toolbox
BNP	Bayesian Network Prior
Syntren	Synthetic Transcriptional Regulatory Networks
FC	Fold Change
ROC	Receiver Operation Characteristic
AUC	Area Under Curve
TP	True Positives
FN	False Negatives
FP	False Positives
RV	Random Variables

1. Introduction

1.1 Motivation

Signaling pathways are dynamic events that take place over a given period of time, so expression data over time are required in order to identify them. Dynamic Bayesian network (DBN) is an important approach for predicting the gene regulatory networks as early as possible. However, two fundamental problems greatly reduce the effectiveness of current DBN methods. The first is the lack of a systematic way to determine a biologically relevant transcriptional time lag, which results in relatively low accuracy of predicting gene regulatory networks. The second one is the excessive computational cost of these analyses, which limits the applicability of current DBN analyses to a large-scale microarray data. [1]

1.2 Aim

This project aims for configuring a plan to introduce a Time Variant DBN-based analysis that can predict gene regulatory networks from time course expression data, with significantly increased accuracy and reduced computational time. In doing so, we will utilize external biological knowledge in our structure learning algorithms, a strategy we have successfully applied for static Bayesian Networks [2]. In the proposed framework, the network aimed to be calculated is called the transition network, which defines the dependencies between adjacent time-slices. With the use of external biological knowledge in DBN learning algorithms, this study's target is to impose a search space constraint in identifying the transition network.

1.3 Gene Regulation

Gene Regulation can be briefly described as the process of a gene transcribed into mRNA and then translated into a protein product. Some specific proteins (transcription factors) are responsible for regulating the expression of their target genes, by increasing (activation) or decreasing (suppression) it. Gene networks can be seen as ï¿½projections of the whole biochemical network onto a space where the only observables are gene transcripts (mRNA)'.

Furthermore, a collection of DNA segments in a cell that interact with each other and with other substances in the cell is called a gene regulatory network. It controls the rates at which genes in the network are transcribed into mRNA [3]. The interaction between genes is indirect through their RNA and protein expression products. If microarray data is used for inference of the underlying network, then the regulatory network is called a gene expression network.

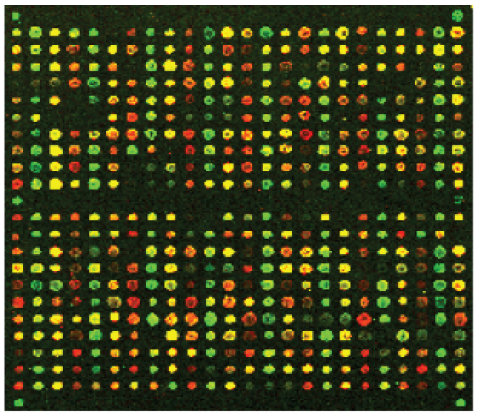
1.4 Microarrays

In the late 1990's microarray data started to be used in genomic studies using oligonucleotide or cDNA probes. With DNA microarrays researchers were able to measure the abundance of thousands of mRNA targets simultaneously. Early microarray experiments examined few samples and mainly focused on differential display across tissues or conditions of interest. DNA microarray experiments were able to measure all the genes of an organism, providing a "genomic" viewpoint on gene expression.

By definition microarrays contain grids of up to tens of thousands of array elements presented in miniaturized format. A way to measure the amount of homologous sequences in the sample is through the intensity of the hybridization over individual spots [4]. Microarrays come in two forms: robotically spotted and oligonucleotides.

The most widely used are oligonucleotide microarrays, and big providers, such as Affymetrix, develop and put them on the market. For example, in the technology utilized by Affymetrix, millions of oligonucleotides of length 25 base pairs each are placed on an array. These oligochips (oligonecleotide chips) are constructed using a photolithographic masking technique similar to the process that is used in microelectronics and integrated circuits fabrication, first described by Stephen Fodor et al. in 1991 [5].

A)



Samples

B)

Genes

Gene Description	1	2	3	4	5	6	7
AFFX-BioB-5_at (endogenous control)	-214	-139	-76	-135	-106	-138	-72
AFFX-BioB-M_at (endogenous control)	-153	-73	-49	-114	-125	-85	-144
AFFX-BioB-3_at (endogenous control)	-58	-1	-307	265	-76	215	238
AFFX-BioC-5_at (endogenous control)	88	283	309	12	168	71	55
AFFX-BioC-3_at (endogenous control)	-295	-264	-376	-419	-230	-272	-399
AFFX-BioDn-5_at (endogenous control)	-558	-400	-650	-585	-284	-558	-551
AFFX-BioDn-3_at (endogenous control)	199	-330	33	158	4	67	131
AFFX-CreX-5_at (endogenous control)	-176	-168	-367	-253	-122	-186	-179
AFFX-CreX-3_at (endogenous control)	252	101	206	49	70	87	126
AFFX-BioB-5_st (endogenous control)	206	74	-215	31	252	193	-20
AFFX-BioB-M_st (endogenous control)	41	19	19	363	155	325	-115
AFFX-BioB-3_st (endogenous control)	-831	-743	-1135	-934	-471	-631	-1003
AFFX-BioC-5_st (endogenous control)	-653	-239	-962	-577	-490	-625	-761
AFFX-BioC-3_st (endogenous control)	-462	-83	-232	-214	-184	-177	-541
AFFX-BioDn-5_st (endogenous control)	75	182	208	142	32	-94	109
AFFX-BioDn-3_st (endogenous control)	381	164	432	271	213	222	435
AFFX-CreX-5_st (endogenous control)	-118	-141	84	-107	1	-1	-129
AFFX-CreX-3_st (endogenous control)	-565	-423	-501	-101	-260	-140	-399
hum_alu_at (miscellaneous control)	15091	11038	16692	15763	18128	34207	30801
AFFX-DapX-5_at (endogenous control)	7	37	183	45	-28	65	43
AFFX-DapX-M_at (endogenous control)	311	134	378	268	118	154	80
AFFX-DapX-3_at (endogenous control)	-231	-161	-221	-27	-153	-49	-87

Figure 1.1 (A) Image of a microarray (B) Microarray signal matrix

In a standard microarray experiment, by using fluorescent (generally Cy3-dNTP or Cy5-dNTP) or radiolabeled deoxynucleotides ([^{33}P]- or [^{32}P]-α-dCTP), the isolated RNA is transcribed reversely into target cDNA. The labeled targets are then purified,

denaturized, and hybridized to the microarrays at a temperature determined by the hybridization buffer used. After hybridization, the arrays are washed in order to clean up nonspecific target binding and are air-dried.

The next step is microarray image processing. In order to obtain a scan of the array for each emission wavelength, such as two 16-bit grayscale TIFF images, differential excitation and emission wavelengths of the atoms are used. An analysis of these images enables the identification of the spots, their associated signal intensities calculation, as well as the assessment of local background noise. Spot-flagging can be managed by using basic filtering tools, usually found in image acquisition software packages; in this manner spots such as extremely low-intensity spots, ghosts spots (where background is higher than spot intensity), or damaged spots (e.g., dust artifacts) can be flagged. After this process an initial ratio of the evaluated channel/reference channel intensity can be calculated for every spot on the chip [3]. These ratios are translated into signal values for each gene and are tabulated as shown in Figure 1.1.

Microarrays have been used in biological and clinical studies to study various phenotypes in health and disease [1]. They have been widely applied to cancer, heart disease, diabetes, stem cells, and neurological disorders, studies analyzing drug response, dosage, and effect. In these studies, microarrays are primarily used to find differentially expressed genes between two different states, such as cancer and normal tissue samples. These genes can further be analyzed to identify their functional roles and be used to develop biological and clinical hypothesis that warrant deeper studies. Microarrays can also be used as a tool for biomarker discovery [2]. In these studies, a signature set of genes predict a certain biological or clinical phenotype, such as predicting which primary tumor samples are most likely to metastasize based on the transcriptional profile of the tumors at the baseline. The PubMed scientific literature database lists 54,780 papers that involve microarrays.

1.5 Bayesian Network

Bayesian Network (BN) is a graphical model for reasoning in a domain where the nodes $X = X_1, ..., X_n$ represent random variables and arcs $X_i \rightarrow X_j$ represent direct connections/dependencies between them. BNs also provide information about the strength of these connections. We score the strength of these dependencies with CPD (conditional probability distribution) values.

In other words, a BN is a compact representation of the joint probability distribution of random variables. BNs do not allow cycles in a graph, indeed must be a directed acyclic graph (DAG), according to the first order Markov assumption. In short, if we start from a node, we would not be able to reach the same node following any path in the graph. In BNs, if there is a link between a pair of nodes, the node where the link starts from is called the parent node and the other is called the daughter. Another condition for a BN is that a node should be conditionally independent of its non-descendants given its parents. Bayesian Networks have been used in a diverse list of fields ranging from finance, economics to physics and biology [6]. These applications mainly deal with understanding the interdependence relation between random variables based on observed data and use of this relation to make predictions about outcomes given new data.

1.6 Dynamic Bayesian Networks

Dynamic Bayesian Networks unlike Bayesian Networks, use time series data for constructing causal relationships among random variables. In addition, as the BN rule says, a node is assumed to be independent of its non-descendants given its parents in the immediate previous time point. Therefore, in order to overcome cycles in a network, a DBN can duplicate the number of nodes per lag applied to the times series. This means that a DBN, assuming inter relations between following time lags, will remove cyclicity duplicating its node elements [7] in the next time lag and giving arcs

only $X_t \rightarrow X_{t+1}$. In the end, the graphical structure of DBNs only represents direct associations between random variables.

Current methods for learning of DBNs can be categorized into two major groups: constraint based methods and score based methods [8, 9, 10]. Constraint based methods are shown to create satisfactory results with sparse networks but are not suitable for large datasets and dense networks [11]. The other, score based methods, consider learning of DBN as an optimization problem. These methods devise a scoring function for a candidate network structure based on the probability of the structure given the input data. They search through the space of possible network structures that minimize the scoring function.

1.7 Learning in Bayesian Networks

1.7.1 Inference

Bayesian inference is basically to obtain probability values of every node using the data and then applying the Bayesian rules. Indeed, Bayesian inference contributes to learn a prior distribution from the data. Bayesian learning can be described in to groups, which are objective (non-informative) and subjective (informative) and might carry Gaussian or Normal distribution. For an objective learning with a normal distribution data, we follow the process of Bayesian learning like following. First, from observations we estimate the joint probabilities of the nodes in the network, which are referred to as: $X_1, X_2, ..., X_n$ .The data is represented by $D = (X_1, X_2, ..., X_n)$. Assume we model the data with normal distribution with all data parameters being independent, so the node i refers to $X_i \sim N(\theta, 1)$ for $\theta = true$ value of the parameter. Most common methods to calculate it are Maximum Likelihood Estimations (MLE) and MAP methods.

Maximum Likelihood Estimations, where n is the number of the nodes in the graph, are computed as:

$$\theta_{MLE} = \frac{1}{n}\sum_{i=1}^{n} X_i \tag{1.1}$$

Until this the calculation is non-informative (objective). Indeed we did not use any prior information. In such case a prior information can be a belief and implemented in to the inference, which refers to $P(\theta)$. We call this type of process "informative (subjective)" learning. This calculation is different from the previous one is with that we also add the prior information as $n + 1^{th}$ node. If we assume they are independent and not weighted, which means all nodes are the same strength, we can compute the posterior probability as:

$$P(\theta|D) = \frac{P(D|\theta)P(\theta)}{P(D)} \tag{1.2}$$

$P(\theta|D)$ is the posterior probability of a graph for given data. Indeed, D is data and θ is graph and $P(\theta)$ is the prior information.

1.7.2 Parameter Learning

In a Bayesian network, the DAG is called the structure and the values in the conditional probability distributions are called parameters. These parameters represent nodes in the graph, and what this study is trying to do is to address the problem of learning these parameters for a given structure. This process is called Parameter Learning, and by running it we aim at assigning the probability distribution to each node. As such, we assume that each node has a space of size 2. This means that each node demonstrates 2 probability (relative frequency) values, one for "presence" and one for "absence".

In order to explain how parameter learning works we can take a look at Beta distribution functions [12, 13]. The function which gives the value of each node is the beta density function, as can be seen in the Equations 1.3, 1.4, and 1.5.

Within the classical Bayesian framework, learning parameters in BNs is based on priors; a prior distribution of the parameter is chosen and a posterior distribution is then derived given the data and priors. An important issue in parameter learning in BNs is that the learning datasets are mostly incomplete and we have to deal with missing observations. Inference with missing data is an old problem in statistics and several solutions have been proposed in the last three decades [14]. For example; the Expectation-Maximization (or EM) algorithm is a routine technique for parameter estimation in statistical models with missing data. Mostly, the Maximum Likelihood (ML) method is used in EM [14].

1.7.3 Beta Functions

We obtain relative frequency values and develop density functions; a set of density functions are put together and thus form a Beta function. Beta functions provide quantifying prior beliefs with relative frequencies and updating these beliefs regarding new evidences. In order to understand Beta Density functions we need to consider Gamma functions, which are the fundamentals of Beta Density functions and refer to $\Gamma(x)$:

$$\Gamma(x) = \int_0^\infty t^{x-1} e^{-t}\, \mathrm{d}t \tag{1.3}$$

Gamma functions are generalization of factorial functions and established on 2 rules:

$$\frac{\Gamma(x+1)}{\Gamma(x)} = x; \tag{1.4}$$

for $a, b, andN$, where $N = a + b$ and $0 \leq f \geq 1$; and

$$p(f) = \frac{\Gamma(N)}{\Gamma(a)\Gamma(b)} f^{a-1}(1-f)^{b-1} \tag{1.5}$$

Any random variable F that has the density function given in the equation above, is said to have a beta distribution.

In short, parameter learning is to learn CPTs for a given or candidate graph, different from inference, because in inference, the network structure is unknown and is expected from data, as explained in the previous subsection. Then posterior probability is calculated with Equation 1.2.

1.7.4 Structure Learning

Learning the structure G of the Bayesian network from data is a very challenging problem. The most common approach to discovering the structure of Bayesian networks from data is to define a graph model to consider, and then practice a scoring function that evaluates how well the model explains the available data. After that, an optimization algorithm is used to find the highest-scoring model.

The scoring function in the logarithm form of the posterior probability of the network structure given the data is in equation 1.6:

$$Score(G;D) = \log P(G|D) = \log P(D|G) + \log P(G) - \log P(D) \tag{1.6}$$

where,

$$P(D|G) = \int_{\theta} P(D|G,\theta)P(\theta|G)\mathrm{d}\theta \tag{1.7}$$

We average over all parameters θ associated with a graph structure G. The likelihood of the data D given parameter vector θ for a known structure G is:

$$L(\theta;D) = P(D|\theta) = \prod_{i=1}^{m} P(X_1(1), ..., X_n(i)|\theta) \tag{1.8}$$

Bayesian methods reduce overfitting by representing and using available knowledge about the parameters in the form of a prior distribution $P(\theta)$. The data D then serves to update the prior $P(\theta)$ to yield the posterior probability distribution $P(\theta|D)$. We know by Bayes Rule, seen in equation 1.2, an equivalent expression. However, the effect of $P(\theta)$ is generally neglected because of the difficulty of calculating this probability. Therefore, $P(\theta)$ is omitted and since the data is given, the $P(D)$ is constant, hence, $P(\theta|D) \cong P(D|\theta) = BayesianScoring_{BDe}(D,\theta)$ could be accepted as equal:

$$BayesianScoring_{BDe}(D,\theta) = \prod_{i=1}^{N}\prod_{j=1}^{q_i}\frac{\Gamma(N_{ij})}{\Gamma(N_{ij}+M_{ij})}\prod_{k=1}^{r_i}\frac{\Gamma(a_{ijk}+s_{ijk})}{\Gamma(a_{ijk})} \tag{1.9}$$

where N = number of nodes, q = number of different states of a node's parents,

r_i = set of values a node can take on, N_{ij} = the sum of corresponding Dirichlet distribution hyper-parameters a_{ijk}, M_{ij} = number of times parents of node i take on configuration j in the dataset, and S_{ink} = the total number of times (of these M_{ij} cases) that node i is observed to have value k when its parents take on configuration j.

2. Graphical Model Applications

The utilization of microarray gene expression data seems promising to create gene interaction networks to uncover biochemical pathways. Several approaches have been proposed for inferring gene networks from experimental data [15]. In particular, Bayesian Network models have gained popularity for the task of learning biological pathways from microarray gene expression data [16]. In gene network modeling studies using BNs, Xi generally represents expression level of a gene and edges represent relationship between genes.

If we observe the random variables (RVs) at different time points, static BNs do not represent how a RV is related to its and other RVs values in successive times. However, DBNs are able to model this phenomenon in the time-series context [12]. Let $X[t]$ denote a column vector of RVs $X_1, ..., X_n$ at time t. In the DBN framework, at each time point t, we have a DAG, G_t, representing the dependency structure between the RVs at time t. G_t is referred to as the intra network for each time point. In addition to the intra networks, if the data set consists of k time points, the DBN structure models $k-1$ inter networks that represent the transition probability distributions between the RVs in successive time points. The inter network, $G_t \leftarrow t+1$, represents the dependency structure between $X[t]$ and $X[t+1]$. Biological events are dynamic in nature and take place over a time period. Therefore, it is often desirable to come up with a time series experimental design that can monitor the biological sample at a series of time points. When gene interaction networks are studied in this setting, use of Dynamic BNs (DBN) is preferred over static BNs as the underlying mechanisms are not static and change over time.

Earlier work applying the DBN framework has mainly focused on non-stationary models with fixed structure. The time varying autoregression (TVAR) model [17], which describes non-stationary linear dynamic systems with continuously changing linear coefficients and noise variances, is one such popular model. The TVAR model

has been further developed with non-Gaussian autoregression models [18] and been applied to gene expression data [13, 19, 20]. Another class of non-stationary models, which has been widely studied, is the switching linear dynamic system (SLDS), which uses latent Markov chains to describe the piecewise changes in the linear systems [12]. Recently, graphical models that change dynamically in both parameter and structure have received more attention. With the assumption that the data sequence is partially stationary in time, non-stationary models are constructed as a cascade of stationary models. Each of these models is learnt from pre-segmented stationary subintervals and a number of methods have been proposed to find these subintervals using Gaussian graphical models [21] or Markov chain Monte Carlo (MCMC) sampling methods [22, 23]. In most such approaches, the network learning phase uses some variant of the Greedy Hill Climbing method [12, 24]. In this thesis, we focus on the REVEAL algorithm [25], which, for a given fully observable discrete time series data, learns an inter-slice adjacency matrix. The parent set for each node in slice (time) t is computed by evaluating all subsets of nodes in slice $t-1$, and picking the largest scoring one.

Most of the BN structures learning techniques employ the likelihood model when finding the best graph that explains the observed data. The likelihood measure is a suboptimal objective function and does not incorporate the probability of the candidate graph in its scoring. In addition to using the optimal model (by use of an appropriate scoring function), incorporation of existing external biological information can also be employed to improve BN learning from experimental data. There are large numbers of biological databases storing vast amounts of data that can be used to infer the interaction probability of two genes. In this thesis, we combine these two areas that could be improved in a novel framework, where external biological knowledge is incorporated in BN learning through calculation of the probability of a candidate graph [26, 27, 28]. This way, the true model is optimized in the structure learning phase and the resulting networks are more reliable as they are partially based on evidence collated from external sources.

3. Use of External Knowledge

3.1 Bayesian Network Prior

Bayesian Network Prior (BNP) predicts if two genes interact based on the expert advised external biological knowledge and given experimental data [29]. In order to establish it, we first gather information from specific external biological sources as described in the next section and BNP is purposed to provide evidence about the interaction of any two given genes. A Gene Interaction (GI) variable was evaluated as "TRUE" or "FALSE" regarding the number of evidence from the external biological knowledge. If there are two or more evidence for the questioned interaction it is marked as TRUE, which means there is an interaction between the pair of genes. This results in an evidence matrix, where rows represent pairs of genes and columns represent evidence types with GI being the last column. A small sample evidence matrix is shown in Table 3.1 with a small subset of external information. BNP was learnt over this data matrix and therefore represented the interrelation between different evidence types and the event of GI. In total, we had 19 evidence types such as "Microarray", "Pathway", "Two-Hybrid", and "Western". There were over 60,000 gene pairs used in building BNP.

As parameters of the nodes in addition to the structure of the network are learnt during the establishment of BNP, it can now be used as a decision support system to predict if a given gene pair interacts. This is done by instantiating BNP with experimental evidence and existing knowledge for a given gene pair and calculating the probability of the GI node being "True". In case of microarray data, the correlation between the expression values of the two genes is used as the experimental data. When this procedure is cycled through all pairs of genes that are part of the input data, we obtain a prior interaction matrix B, where B_{ij} represents the probability that gene i and j interact based on external available evidence data and given observed experimental data.

Table 3.1
A short sample for representative knowledge types and gene interaction information for hypothetical pair wise relations. For each knowledge type we indicate if a "relation" between the genes has been established. 1: False, 2: True, NA: Not Available.

Gene Pair	TF Binding	Co-localization	Affinity Capture	Two-Hybrid	Western	Synthetic Rescue	Gene Interaction
$A \to B$	2	1	1	NA	1	2	2
$A \to C$	1	2	2	2	2	2	2
$C \to D$	2	2	NA	1	2	1	2
$C \to E$	2	1	2	1	2	1	1
$E \to G$	1	1	1	2	NA	2	1

3.2 Sources of External Biological Knowledge

Molecular interactions would occur between molecules belonging to different biochemical families (proteins, DNA, RNA, etc.) and also within a given family. There are many experimental methods that reveal interactions for protein-protein, protein-DNA, protein-RNA and such. In what follows, we list the experimental evidence type used in the proposed BN model.

3.2.1 Physical Interactions

Affinity Capture-Luminescence An interaction is inferred when a bait protein, tagged with luciferase, is enzymatically detected in immunoprecipitates of the prey protein as light emission. The prey protein is affinity captured from cell extracts by either polyclonal antibody or epitope tag.

Affinity Capture-MS An interaction is inferred when a "bait" protein is affinity captured from cell extracts by either polyclonal antibody or epitope tag and the associated interaction partner is identified by mass spectrometric methods.

Affinity Capture-RNA An interaction is inferred when a bait protein is affinity

captured from cell extracts by either polyclonal antibody or epitope tag and associated RNA species identified by Northern blot, RT-PCR, affinity labeling, sequencing, or microarray analysis.

Affinity Capture-Western An interaction is inferred when a bait protein affinity captured from cell extracts by either polyclonal antibody or epitope tag and the associated interaction partner identified by Western blot with a specific polyclonal antibody or second epitope tag. This category is also used if an interacting protein is visualized directly by dye stain or radioactivity. Note that this differs from any co-purification experiment involving affinity capture in that the co-purification experiment involves at least one extra purification step to get rid of potential contaminating proteins.

Biochemical Activity An interaction is inferred from the biochemical effect of one protein upon another, for example, GTP-GDP exchange activity or phosphorylation of a substrate by a kinase. The "bait" protein executes the activity on the substrate "hit" protein.

Co-crystal Structure Interaction directly demonstrated at the atomic level by X-ray crystallography, also used for NMR or Electron Microscopy (EM) structures. If a structure is demonstrated between 3 or more proteins, one is chosen as the bait and binary interactions are recorded between that protein and the others.

Co-fractionation Interaction inferred from the presence of two or more protein subunits in a partially purified protein preparation. If co-fractionation is demonstrated between 3 or more proteins, one is chosen as the bait and binary interactions are recorded between that protein and the others.

Co-localization An interaction is inferred from co-localization of two proteins in the cell, including co-dependent association of proteins with promoter DNA in chromatin immunoprecipitation experiments.

Co-purification An interaction is inferred from the identification of two or more protein subunits in a purified protein complex, as obtained by classical biochemical

fractionation or affinity purification and one or more additional fractionation steps.

Far Western An interaction is detected between a protein immobilized on a membrane and a purified protein probe.

FRET An interaction is inferred when close proximity of interaction partners is detected by fluorescence resonance energy transfer between pairs of fluorophore-labeled molecules, such as occurs between CFP (donor) and YFP (acceptor) fusion proteins.

PCA A protein-protein interaction assay in which a bait protein is expressed as fusion to one of the either N- or C- terminal peptide fragments of a reporter protein and prey protein is expressed as fusion to the complementary N- or C- terminal fragment of the same reporter protein. Interaction of bait and prey proteins bring together complementary fragments, which can then fold into an active reporter.

Protein-peptide An interaction is detected between a protein and a peptide derived from an interaction partner. This includes phage display experiments.

Protein-RNA An interaction is detected between protein and an RNA.

Reconstituted Complex An interaction is detected between purified proteins in vitro.

Two-hybrid / TF Binding Site Localization - Bait protein expressed as a DNA binding domain (DBD) fusion and prey expressed as a transcriptional activation domain (TAD) fusion and interaction measured by reporter gene activation.

3.2.2 Genetic Interactions

Dosage Growth Defect A genetic interaction is inferred when over expression or increased dosage of one gene causes a growth defect in a strain that is mutated or deleted for another gene.

Dosage Lethality A genetic interaction is inferred when over expression or increased dosage of one gene causes lethality in a strain that is mutated or deleted for another gene.

Dosage Rescue A genetic interaction is inferred when over expression or increased dosage of one gene rescues the lethality or growth defect of a strain that is mutated or deleted for another gene.

Phenotypic Enhancement A genetic interaction is inferred when mutation or over-expression of one gene results in enhancement of any phenotype (other than lethality/growth defect) associated with mutation or over expression of another gene.

Phenotypic Suppression A genetic interaction is inferred when mutation or over expression of one gene results in suppression of any phenotype (other than lethality/growth defect) associated with mutation or over expression of another gene.

Synthetic Growth Defect A genetic interaction is inferred when mutations in separate genes, each of which alone causes a minimal phenotype, result in a significant growth defect under a given condition when combined in the same cell.

Synthetic Haploin Sufficiency A genetic interaction is inferred when mutations or deletions in separate genes, at least one of which is hemizygous, cause a minimal phenotype alone but result in lethality when combined in the same cell under a given condition.

Synthetic Lethality A genetic interaction is inferred when mutations or deletions in separate genes, each of which alone causes a minimal phenotype, result in lethality when combined in the same cell under a given condition.

Synthetic Rescue A genetic interaction is inferred when mutations or deletions of one gene rescues the lethality or growth defect of a strain mutated or deleted for another gene.

The interaction evidence from experimental data are gathered from the following databases: KEGG[1] , NCI/Nature Pathway Interaction Database[2] , Reactome[3] , and BioGrid[4] .

3.3 Use of Prior Information

After we calculated a graph matrix using BNP as mentioned before, the candidate graph is scored in the structure learning phase, the probability of this graph, $P(G)$, is obtained using

$$P(G) = C\mathrm{e}^{\beta E(G)} \tag{3.1}$$

Where β is the candidate graph and C is a scaling constant. The choice of C does not affect the relative comparison during scoring of graphs in structure learning. The hyperparameter β can be marginalized using

$$P(G) = C\frac{1}{\beta_H - \beta_L}\int_{\beta_L}^{\beta_H} \mathrm{e}^{\beta E(G)}\mathrm{d}\beta \tag{3.2}$$

In these calculations, $E(G)$ is the energy function and denotes the degree of agreement between the prior interaction matrix, β, and the candidate graph, G. Namely, we define $U_{ij} = 1$ if there does not exist any links between i and j in G and we set $U_{ij} = 1 - B_{ij}$, if there is a link between i and j in G. $E(G)$ is then defined as

[1]Accessible at: http://tinyurl.com/kaalg2h
[2]Accessible at: http://tinyurl.com/63l8cx
[3]Accessible at: http://tinyurl.com/megepcp
[4]Accessible at: http://tinyurl.com/mescm8m

$$E(G) = \sum_{i,j} U(i,j)/N^2 \tag{3.3}$$

where N is the number of nodes in G. In the structure learning process for BNs, the graph that maximizes the probability in equation 1.2 is to be found. Here, D represents the observed data. However, in the standard search algorithms the likelihood $P(D|G)$, instead of the true model $P(G|D)$ is used as the objective function. In the proposed method, we are able to calculate the true model through incorporation of $P(G)$ as described.

Given time-series data, we learn the intra networks using the greedy search algorithm both with incorporation of prior knowledge (proposed method) and with flat/uniform priors (standard methods, likelihood approach). To this end, the data observed at time t only is used to find the intra network at that time. The inter networks are learned along the lines of the REVEAL algorithm as described in the previous section. In applying the basic idea of the REVEAL method in the proposed approach, we use the following modification. We append the data of a node at time (slice) t to the entire data set in time $t-1$ and apply structure learning on this augmented data set to discover the parents of the appended node. The appended node belongs to the t^{th} slice while its parents come from the $(t-1)^{st}$ slice. Going through each node at time t with this procedure reveals the final inter network between times $t-1$ and t. In the inter-network calculation, we compare the results of the proposed approach using prior information with that of the REVEAL method using flat/uniform priors.

4. Results and Discussion

4.1 Synthetic Data

In this chapter we practice synthetic simulations to realize the effect of implementing prior knowledge in structure learning. We generate a synthetic time series interaction network as seen in Figure 4.1. The total network consists of four time points and five nodes at each time point. There are four intra and three inter networks but for purposes of symmetry, we used the first three intra networks in our simulations. For each intra and inter network, data that follow the network's CPTs were generated using the Bayes Net Toolbox (BNT) for Matlab[5] . We generated data sets of length 200, 300, 400, and 500. When the proposed algorithm was used to infer the interaction networks, we obtained the B matrix from the true adjacency matrix of the corresponding DAG, A_T, and introduced a distortion by adding Gaussian noise to each entry. The distortion rate was calculated using $d = Fro(A_T - B)/Fro(A_T)$, where $Fro(A)$ represents the Frobenius norm of the matrix A [4]. The introduced distortion rates were between 0.0 and 0.3 with 0.1 increments yielding similarities to the prior matrix at the levels of 0.7, 0.8, 0.9, and 1.0.

To assess the performance of the proposed method, we used receiver operating characteristic (ROC) curve analysis. The area under the curve (AUC) values for the learnt DAGs were calculated both using posterior probability $P(G|D)$ with informative priors (proposed method) and marginal likelihood $P(D|G)$ scores with uniform flat priors. In total, there were 96 structure learning processes performed when inter and intra networks are learned for four different distortion levels and four different data set sizes. In 100% of the time for the inter and intra networks, in average of 10 run for each, the interaction network learnt using the proposed method revealed a higher AUC value than the standard methods. In Figures 4.2 and 4.3, we represent the mean AUC values of ten runs with standard deviation as errors, for both methods.

[5]Accessible at: http://www.cs.ubc.ca/?murphyk/Software/BNT/bnt.html

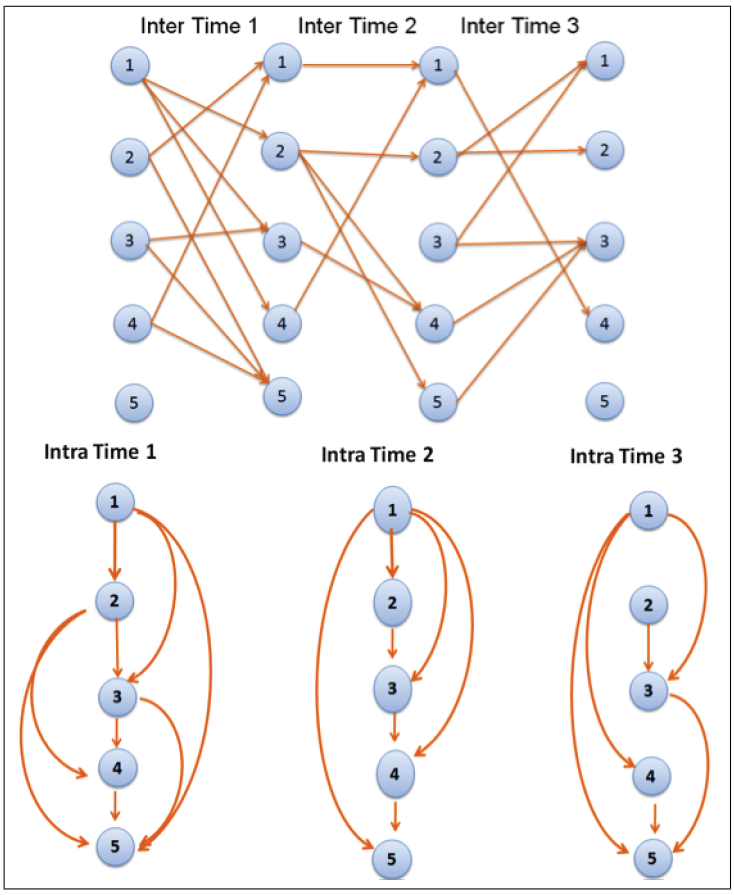


Figure 4.1 Topologies of the inter- and intra- networks used for synthetic data.

Our results suggest that the proposed method greatly outperforms standard methods and the proposed method is more successful in identifying the intra networks than in identifying the inter networks. This is expected as the intra networks exhibit the static structure inherent in the data and the challenging task in the time-series data is in identifying the temporal, inter networks. For the inter networks, the average AUC values for the proposed method when all distortion and data set sizes were considered was about 70-90%. This value increased to 90-100% for the intra networks. On the other hand, standard methods performed at around the 60-70% AUC level both for the inter and intra networks.

The proposed system's performance increased with data set size mainly because the structure of the fitting DAG is more reliably found as more data are observed. For the intra networks, when all distortion levels are combined, the average AUC values were 92%, 88%, 93%, and 96%, respectively with data set sizes increasing from 200 to

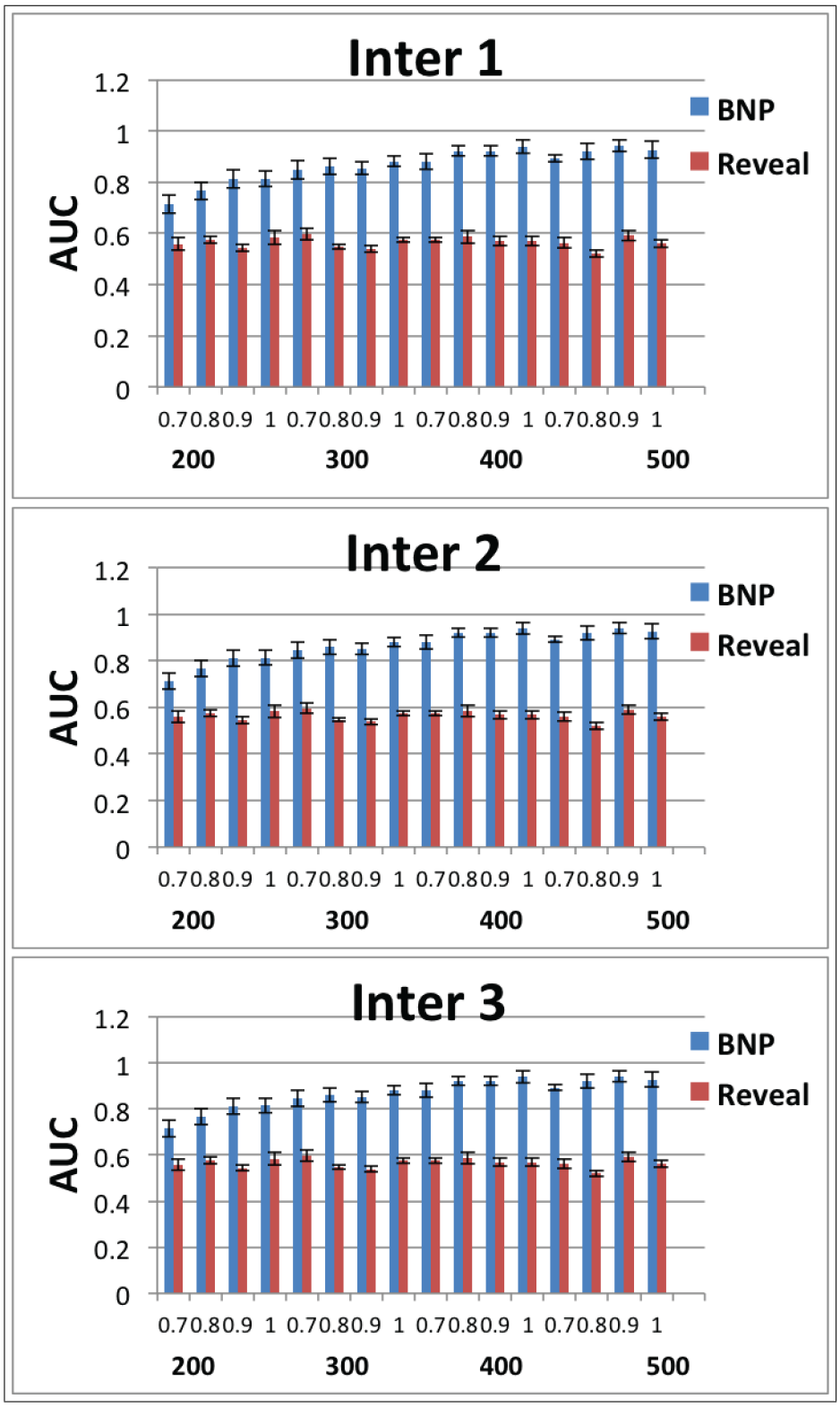


Figure 4.2 AUC values for the proposed (blue) and standard (red) DBN based algorithms for the inter 1-2-3 networks for varying data sizes. The x-axis denote the similarity between the employed prior matrix and the true network.

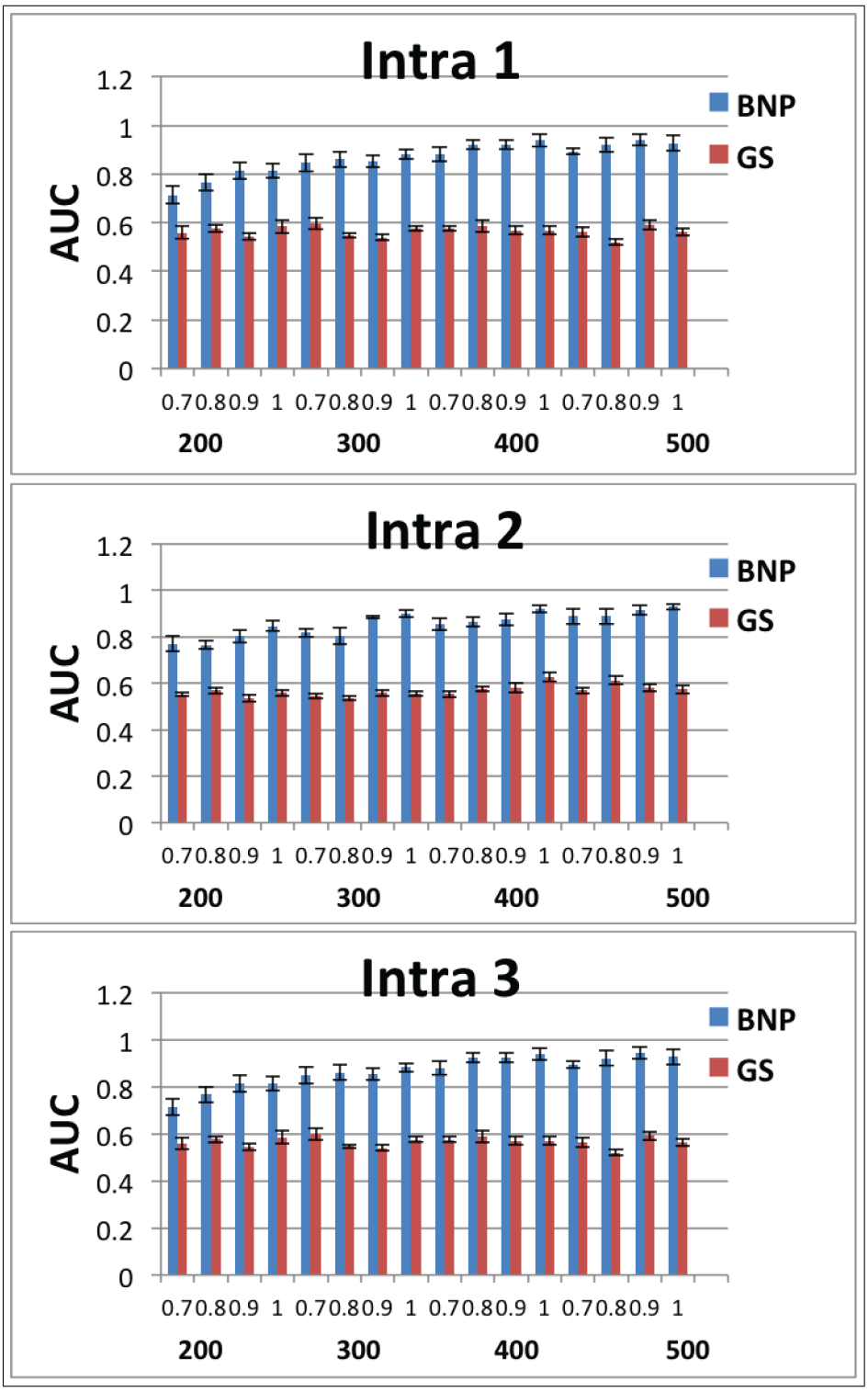


Figure 4.3 AUC values for the BNP (blue) and GS (red) DBN based algorithms for the intra 1-2-3 networks for varying data sizes. The x-axis denote the similarity between the employed prior matrix and the true network.

500. Similar results were observed for the inter networks where the AUC values were increasing from 78% to 98%, respectively with increasing data set sizes. These results

suggest that factors other than the data set size are limiting in learning inter networks as the performance increase seems more dependent on the data set size for the case of intra networks.

Although not very apparent, the proposed method resulted in higher AUC values as the prior matrix, B, utilized in calculation of $P(G)$, approached the true interaction network. For example, for the case of the data set size of 500, the average AUC values for the inter networks were 69%, 73%, 74%, and 76%, respectively as the similarity between the prior matrix and the true DAG increased from 0.7 to 1.0. We believe that lack of a structured dependence of the proposed methodï¿½s performance to the distortion introduced in the prior matrix exhibits the robustness of the proposed method to errors in the prior informative structure, B.

In Figure 4.4, we show an example instance of the identified inter networks using the proposed method where data set size was 500 and the similarity rate was 0.7. We omit the intra networks in this figure for the sake of visual simplicity. In this depiction, green edges represent true positives (TP), i.e., the edges that were in the true inter networks and were correctly found by the proposed method. Red edges represent false positives (FP), i.e., the edges inserted by the proposed method that did not exist in the true network. Yellow dashed edges represent false negatives (FN), i.e., the edges that existed in the true network but were not found by the proposed method.

4.2 Simulated Gene Expression Data

4.2.1 Network Generator: Syntren

The use of simulated data is gaining an increased attention because of the limitations of real experimental data for structure learning researches. The term network generator is used to explain a system that generates synthetic networks and simulates gene expressions derived from these networks. A synthetic network consists of a topology. Different approaches have been used to create a network topology. In the

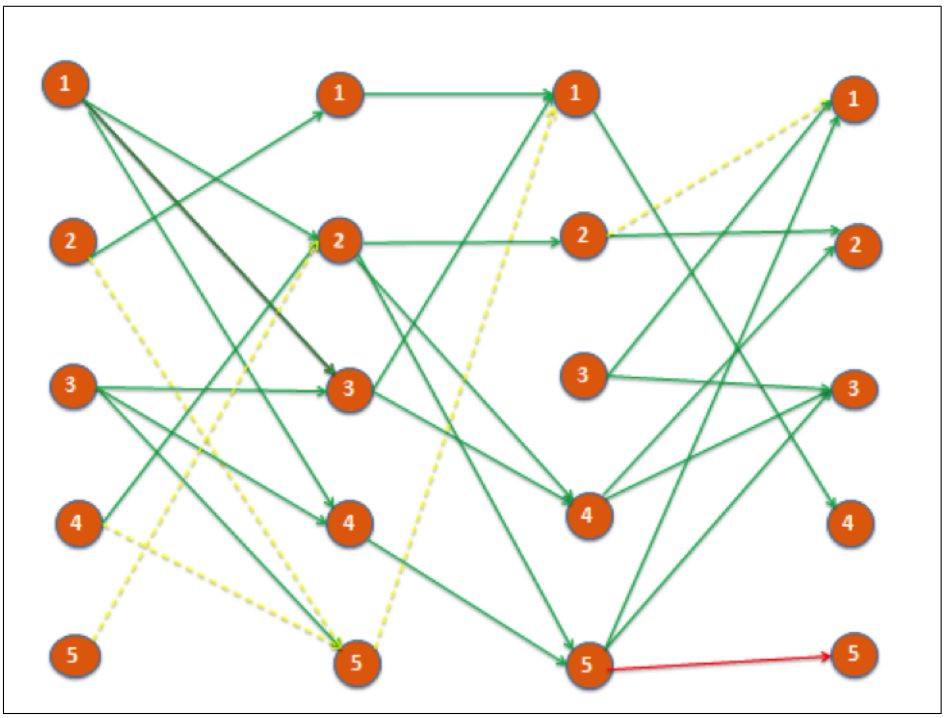


Figure 4.4 The inter networks identified by the proposed method using a data set size of 400 and a similarity of 0.9 between the prior matrix and the true network. Green: TP; Red: FP, Yellow: FN.

generation of small networks, handcrafted topologies might be used, but for producing topologies of large networks, including big number of nodes, random graph models are preferred. Using known regulatory network topologies increases the biological reality. For simulation of the regulatory networks, the interactions between genes need to be quantitatively modeled. Some models have been proposed for this purpose, including Boolean [30], continuous [31] and probabilistic [32] approaches. What we use in our synthetic data simulations is Syntren. SynTReN (Synthetic Transcriptional Regulatory Networks), which is implemented in Java, generates topologies instead of using random graph models, and is based on previously described source networks, providing better approximation of the statistical properties of biological networks. The computational performance of Syntren simulations is directly proportional with the number of genes and significantly successful in simulations of large networks[6] .

In our second half biological data simulations, we used the biological KEGG pathway of "Phenylalanine metabolism" to generate our second Syntren data with the

[6]Accessible at http://homes.esat.kuleuven.be/ kmarchal/SynTReN/

same number of samples and distortion rate. This network includes 17 genes and 93 edges.

4.2.2 Data Processing and Discretization

Most experimental designs consist of two groups of samples (e.g. cancer versus normal). Fold change (FC) is a number describing how much a quantity changes going from an initial to a final value and calculated by dividing the cancer type by the normal type value. For a given biological pathway, we can obtain observed fold changes for genes in this pathway by pairwise comparisons of samples in each group.

This approach provides a distribution of FC values and a reasonable dataset size [31]. This approach provides a distribution of FC values and a reasonable dataset size [40]. Briefly, for an example data set that consists of N normal and C cancer samples, we obtain an observation matrix that is composed of NxC rows. Each column represents a gene (a node on the BN) and the values in a given column represent the FC for that gene due to each of the NxC pairwise Normal vs. Cancer comparison.

For the simulated gene expression data set, we used the Phenylalanine metabolism (hsa00360) KEGG pathway [33], which consists of 17 genes and 93 edges. In Figure 4.5, we show the DAG obtained from this pathway. When modeling the pathway as BN, we used our previously established framework [34]. Briefly, repeating entries in the pathway were merged as a single node in the DAG while conserving edge relations. Cyclic paths were eliminated using Spirtes' method [35] such that the d-separations in the collapsed graph entails the same independency relations defined by the pathway.

We constructed a four time point temporal scenario where at each time point, a randomly chosen 20% of the genes in the network were deactivated. The DAG obtained from the pathway minus the deactivated genes' edges represented the true intra network at each time point. For the inter networks, we expect the DBN based learning algorithms to identify the links that exist in the pathway except for the ones

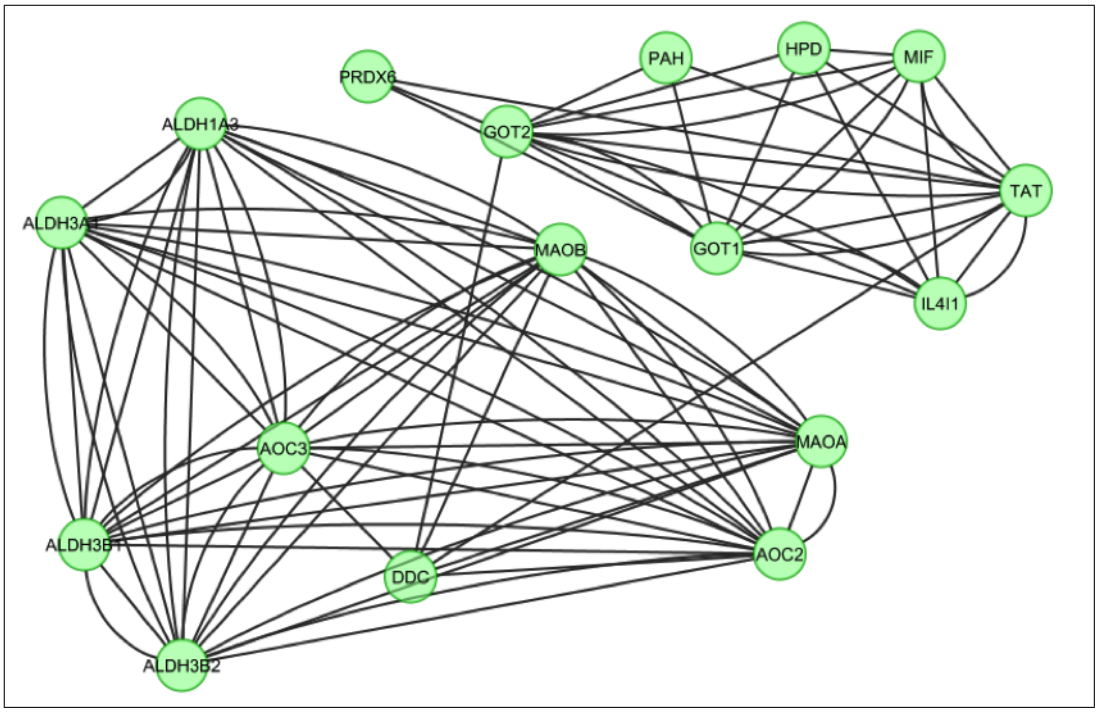


Figure 4.5 The DAG obtained from the Phenylalanine metabolism (hsa00360) KEGG pathway.

that involved at least one deactivated gene in the two successive time points. The gene expression data for each time point was generated using SynTReN v1.12 with 20 control and 20 test samples and 10% background noise. Observed data matrices to be used in the structure learning phase were obtained as previously described [30]. The input matrix consisted of 400 observations (20 control x 20 test) and reflected the distribution of fold change values between the two classes of samples. This matrix was discretized into 3-levels using k-means clustering [32].

For each inter network, the inferred DAGs using prior knowledge (proposed method) and uniform prior knowledge (flat prior, standard methods) were compared to the original pathway structures using AUC values. In Figure 4.6, we show the AUC values obtained by both methods. The proposed method outperforms the standard approaches where the average AUC values obtained by the proposed method was 70% and the average AUC values obtained by the standard approaches was 52%.

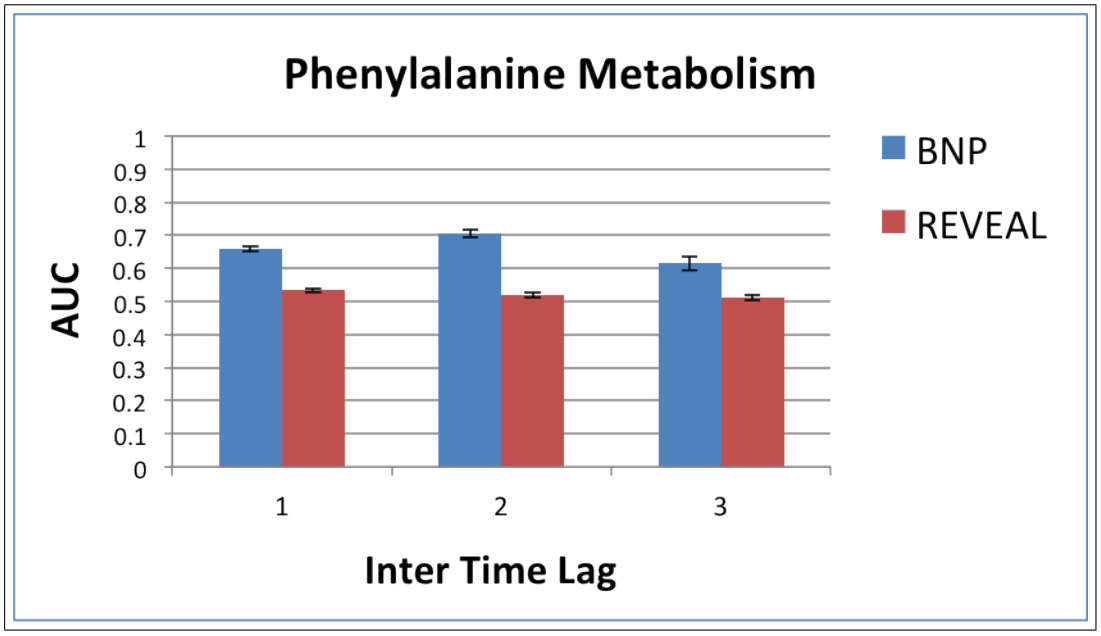


Figure 4.6 AUC performance of the proposed (blue) and standard (red) approaches on the inter networks for the Phenylalanine metabolism (hsa00360) KEGG pathway.

5. Conclusion

In this thesis, we provide a framework that learns inter and intra networks for a given time-series microarray experiment. Time-series experiments are frequently employed in various biological and clinical states. Examples of such applications include growth of certain tissue or cell types over time or response of certain cell lines to drugs. By applying microarrays, expression levels of tens of thousands of genes are measured for the utilized biological samples under various conditions. This expression data that represents the transcriptomics of the underlying samples and can be used to find differentially expressed genes at different states. However, one of the major goals in Bioinformatics, Computational Biology, and Systems Biology is to learn the topology that represents the interaction network of genes and/or gene products. Bayesian Networks provide a suitable framework for this task. Alas, BNs are stationary in nature and do not allow for cycles in its representation. These obstacles provide challenges in applying the BN framework to the task of gene interaction network learning.

Dynamic Bayesian Networks can help us overcome the cyclicity and stationarity problems, which are naturally observed in biological systems. The proposed system in this thesis uses DBNs to learn gene interaction networks for given time series microarray data. In the DBN system, the inter network that defines the links between genes in successive time points enables us to link a gene to itself. At each time, we search, for a given gene, its parents in the previous time point. In doing so, we utilize the external biological knowledge in a framework called BNP. To this end, we constrain our search space using our belief in certain links based on external knowledge and given experimental data. BNP system is also used to calculate the probability of a candidate graph, G, in the network learning process. This way, we not only utilize external knowledge but also optimize the true parameter $P(G|D)$ instead of the likelihood parameter, $P(D|G)$ in our structure learning phase.

To demonstrate the utility of the proposed approach, we have applied it to synthetic and simulated data sets and compared our results to the state-of-the-art DBN workflow used to learn interaction networks from time series microarray data. In the synthetic data sets, we generated data that follows a network structure that spans three time slices. The four intra and the three inter networks were chosen to mimic typical biological network topologies and the proposed method outperformed the existing methods in all simulations with varying data size and distortion levels. In the simulated microarray data set, we generated data that follows a known biological pathway. To define inter networks, we deactivated a randomly chosen 20% of the genes in the pathway at each of the three time points. We expected the algorithms to discover links that did not involve the deactivated genes in the original pathway. Similar to the synthetic data set results, the proposed workflow identified gene interaction networks that are closer to the expected topology when compared with the existing methods.

We believe that the framework proposed in this thesis successfully identifies a dynamic interaction network that spans over time for given microarray data. However, in real life applications, the number of genes measured on the microarray platforms are much larger than the number of nodes used in our simulations. Therefore, we expect that the proposed approach should be augmented with a module that first identifies the genes for which the dynamic network should be built. Nevertheless, the proposed approach would still be applied to real life applications as in biological applications there often lies a targeted set of genes. For example, if we were to assess the effect of a drug on a certain pathway over time, we could design a time series experiment and apply the proposed approach on the genes that are on our target pathway of interest. Biological phenomena are dynamic and cyclic in nature. Therefore, though applicable on a modest number of genes, the proposed algorithm is suitable to the biology under investigation as the algorithm allows for cycles and changes the topology over time. Our approach renders itself as a novel methodology as it incorporates external knowledge and identifies the parents of the nodes at each time slice in newly described ways.

REFERENCES

1. Stern, C., "Predicting chaotic time series with wavelet networks," *Daedalus*, Vol. 99, pp. 882–908, Fall 1970.

2. Liu, W., K. Chang, J. Zheng, J. Divya, J. J. Kim, and J. C. Rajapakse, "Gene regulatory networks from gene ontology," in *Bioinformatics Research and Applications*, ISBRA 2013, (Charlotte, NC, USA), pp. 87–98, 9th International Symposium,, Springer Berlin Heidelberg, 20-22 May 2013.

3. Macgregor, P. F., and J. A. Squire, "Application of microarrays to the analysis of gene expression in cancer," *Clinical Chemistry*, Vol. 48, pp. 1170–1177, August 2002.

4. Ghahramani, Z., and G. E. Hinton, "Variational learning for switching state-space models," *Neural Computation*, Vol. 12, pp. 831–864, April 2000.

5. Friedman, N., M. Goldszmidt, and A. Wyner, "Data analysis with bayesian networks: a bootstrap approach," in *Proceedings of the Fifteenth conference on Uncertainty in artificial intelligence* (Kaufmann, M., ed.), UAI'99, (San Francisco, CA, USA), pp. 196–205, Morgan Kaufmann Publishers Inc., 1999.

6. Arcellana-Panlilio, M., *Principles of Application of DNA Microarrays*, pp. 239–260. Wiley-VCH Verlag GmbH, 2008.

7. Kohane, I. S., A. J. Butte, and A. Kho, *Microarrays for an Integrative Genomics*, Cambridge, MA, USA: MIT Press, 2002.

8. Tseng, G. C., D. Ghosh, and E. Feingold, "Comprehensive literature review and statistical considerations for microarray meta-analysis," *Nucleic Acids Research*, 2012.

9. Quackenbush, J., "Microarray analysis and tumor classification," *New England Journal of Medicine*, Vol. 354, no. 23, pp. 2463–2472, 2006. PMID: 16760446.

10. Baldi, P., and G. W. Hatfield, *DNA Microarrays and Gene Expression*, Cambridge, United Kingdom: Cambridge University Press, 2002. Available: http:/www.atm.com.

11. Stroncek, D. F., J. P. Wang, E. J. Ren, M. Sabatino, and F. M. Marincola, "Global transcriptional analysis for biomarker discovery and validation in cellular therapies," *Molecular Diagnosis & Therapy*, Vol. 13, no. 3, pp. 181–193, 2009.

12. Neapolitan, R. E., *Learning Bayesian Networks*, Prentice Hall, 2003.

13. Friedman, N., M. Linial, and I. Nachman, "Using bayesian networks to analyze expression data," *Journal of Computational Biology*, Vol. 7, pp. 601–620, 2000.

14. DÕhaeseleer, P., S. Liang, and R. Somogyi, "Genetic network inference: from co-expression clustering to reverse engineering," *Bioinformatics*, Vol. 16, no. 8, pp. 707–726, 2000.

15. Koller, D., and N. Friedman, *Probabilistic Graphical Models Principles and Techniques*, Cambridge, Massachusetts: The MIT Press, 2009.

16. Ramirez, N. C., *Building Bayesian Networks from Data: a Constraint-based Approach*. PhD thesis, The University of Sheffield, 2001.

17. Dempster, A. P., N. M. Laird, and D. B. Rubin, "Maximum Likelihood from Incomplete Data via the EM Algorithm," *Journal of the Royal Statistical Society. Series B (Methodological)*, Vol. 39, no. 1, pp. 1–38, 1977.

18. De Jong, H., "Modeling and simulation of genetic regulatory systems: A literature review," *Journal of Computational Biology*, Vol. 9, pp. 67–103, 2002.

19. Imoto, S., T. Goto, and S. Miyano, "Estimation of genetic networks and functional structures between genes by using Bayesian networks and nonparametric regression.," *Pac Symp Biocomput*, pp. 175–186, 2002.

20. Pe'er, D., A. Regev, G. Elidan, and N. Friedman, "Inferring subnetworks from perturbed expression profiles," *Bioinformatics*, Vol. 17, no. 1, pp. 215–224, 2001.

21. Prado, R., G. Huerta, and M. West, "Bayesian time-varying autoregressions: Theory, methods and applications," *Resenhas*, Vol. 4, pp. 405–422, 2001.

22. Gençağa, D., E. E. Kuruoğlu, and A. Ertüzün, "Modeling non-gaussian time-varying vector autoregressive processes by particle filtering," *Multidimensional Syst. Signal Process.*, Vol. 21, pp. 73–85, Mar. 2010.

23. Rao, A., A. O. Hero, III, D. J. States, and J. D. Engel, "Inferring time-varying network topologies from gene expression data," *EURASIP J. Bioinformatics Syst. Biol.*, Vol. 2007, pp. 7–7, Jan. 2007.

24. da Piedade, I., M.-H. Tang, and O. Elemento, "Dispare: Discriminative pattern refinement for position weight matrices," *BMC Bioinformatics*, Vol. 10, no. 1, pp. 1–16, 2009.

25. Liang, S., S. Fuhrman, and R. Somogyi, "Reveal, a general reverse engineering algorithm for inference of genetic network architectures," pp. 18–29, Pacific Symposium on Biocomputing, 1998.

26. Fearnhead, P., "Exact and efficient bayesian inference for multiple changepoint problems," *Statistics and Computing*, Vol. 16, pp. 203–213, June 2006.

27. Robinson, J. W., and A. J. Hartemink, "Learning non-stationary dynamic bayesian networks," *J. Mach. Learn. Res.*, Vol. 9999, pp. 3647–3680, December 2010.

28. Grzegorczyk, M., and D. Husmeier, "Non-stationary continuous dynamic bayesian networks," in *Advances in Neural Information Processing Systems* (Bengio, Y., D. Schuurmans, J. Laftery, C. Williams, and A. Culotta, eds.), no. 22 in Advances in neural information processing systems, pp. 682–690, La Jolla, CA, USA: Curran Associates, 2009.

29. Isci, S., U. Agyuz, C. Ozturk, and H. H. Otu, "Detecting gene interactions within a bayesian network framework using external knowledge," (Nevsehir,Turkey), pp. 82–87, Health Informatics and Bioinformatics (HIBIT), 2012 7th International Symposium, IEEE, May 2012.

30. Anthony, M., G. Brightwell, and J. Shawe-Taylor, "On specifying boolean functions by labelled examples," *Discrete Applied Mathematics*, Vol. 61, no. 1, pp. 1 – 25, 1995.

31. Kaderali, L., E. Dazert, U. Zeuge, B. Frese, and R. Bartenschlager, "Reconstructing signaling pathways from rnai data using probabilistic boolean threshold networks," *BMC Bioinformatics*, Vol. 25, no. 17, 2009.

32. Ghahramani, Z., "Probabilistic modelling, machine learning, and the information revolution," 2012. Available: http://mlg.eng.cam.ac.uk/zoubin/talks/mit12csail.pdf.

33. Kanehisa, M., S. Goto, Y. Sato, M. Furumichi, and M. Tanabe, "Kegg for integration and interpretation of large-scale molecular data sets," *Nucleic Acids Research*, Vol. 40, pp. D109–14, January 2012.

34. Isci, S., C. Ozturk, J. Jones, and H. H. Otu, "Pathway analysis of high-throughput biological data within a bayesian network framework," *Bioinformatics*, Vol. 27, pp. 1667–1674, June 2011.

35. Spirtes, P., "Directed cyclic graphical representations of feedback models," in *Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence UAI-95*, pp. 491–549, 1995.