Inferring gene networks from time series microarray data using dynamic Bayesian networks

Sun Yong Kim, Seiya Imoto and Satoru Miyano

Date received (in revised form): 23rd June 2003

Abstract

Dynamic Bayesian networks (DBNs) are considered as a promising model for inferring gene networks from time series microarray data. DBNs have overtaken Bayesian networks (BNs) as DBNs can construct cyclic regulations using time delay information. In this paper, a general framework for DBN modelling is outlined. Both discrete and continuous DBN models are constructed systematically and criteria for learning network structures are introduced from a Bayesian statistical viewpoint. This paper reviews the applications of DBNs over the past years. Real data applications for Saccharomyces cerevisiae time series gene expression data are also shown.

INTRODUCTION

The development of microarray technology produces a huge amount of gene expression data and provides an innovative perspective for whole genome analyses. The estimation of a gene network from cDNA microarray gene expression data is one of the most important computational topics. Several methods have been proposed for modelling gene networks including: Boolean networks,1 2 Bayesian networks (BNs)3–5 and differential equations.6,7 In particular, researchers have paid great attention to BNs, which model causal relationships between variables based on probabilistic measure. Since microarray data are usually very noisy, the use of statistical methods is expected to be effective for extracting useful information from such noisy data. Friedman et al.3 proposed both a discrete BN model and a continuous BN model based on a linear regression for modelling gene networks. Friedman et al.4,5 succeeded in employing a non-parametric regression for capturing even non-linear relationships between genes.

Although the above methods are effective to some degree, BNs have a limitation that no cycles are allowed. This can be a serious problem since real gene networks have cyclic regulatory pathways including feedback loops. When we have time series microarray data, the use of dynamic Bayesian networks (DBNs) is a promising alternative, since DBNs can treat time delay information and can construct cyclic networks. DBNs have been used in the field of signal processing and were recently introduced into the analysis of time series microarray data. Friedman et al.8 first applied DBNs to the analysis of gene networks. They constructed a discrete DBN model and used the BDe9 metric for learning networks. Smith et al.10,11 and Ong et al.12 also used discrete models. An interesting point of Ong et al. is that they imported biological knowledge into the modelling of network structures. Their target organism, Escherichia coli, is already known to have sets of genes, called operons, which are transcribed together into mRNA. Reflecting this information, they added some nodes representing operons to the network and restricted edge directions. Although discrete models have
Joint probability

some advantages such as robustness, simplicity of learning and non-linearity, discretisation often tends to be a problem for the following two reasons. First, discretisation might cause information loss. Secondly, the threshold value for discretisation must be chosen very carefully since resulting networks will be affected by this value. To avoid discretisation, Kim et al.\(^{13}\) defined a continuous DBN and non-parametric regression model to capture more than linear dependencies.

This paper reviews the methodology of estimating gene networks from time series microarray data using DBN models. A general theory of DBN models is introduced first, and discrete and continuous models are then elicited. Information criteria for learning unknown network structures from a Bayesian statistical viewpoint are derived. The methods in Friedman et al.,\(^{8}\) Smith et al.,\(^{10,11}\) Ong et al.\(^{12}\) and Kim et al.\(^{13}\) will be presented in this framework. The effectiveness of DBN models through the analysis of \textit{S. cerevisiae} microarray data will be shown.\(^ {14}\)

Gene network

Time series microarray data

DBN MODEL

DBNs can be viewed as an extension of BNs. In contrast to BNs that are based on static data, DBNs use time series data for constructing causal relationships among random variables. In this section, we describe a DBN model under a general framework.

Suppose that we have \(n\) microarrays and each microarray measures expression levels of \(p\) genes. The microarray data, then, can be summarised as an \(n \times p\) matrix \(\mathbf{X} = (x_1, \ldots, x_p)^T\) whose \(i\)th row vector \(x_i = (x_{i1}, \ldots, x_{ip})^T\) corresponds to a gene expression level vector measured at time \(t\). Note that \(x_{ij}\) is considered as an observation from a random variable \(X_{ij}\).

In DBN modelling, the process of model construction can be divided into two steps. First, the DBN models assume a time dependency. Note that, in general, edges in a time slice can be allowed, but in this paper, models are assumed within which the state vector of time \(i\) depends only on that of time \(i-1\). Figure 1 shows this relationship as a directed acyclic graph. Therefore, the joint probability can be decomposed as:

\[
P(X_{11}, \ldots, X_{np}) = P(X_1)P(X_2|X_1) \times \ldots \times P(X_p|X_{p-1})
\]

where \(X_i = (x_{i1}, \ldots, x_{ip})^T\) is a \(p\)-dimensional random variable vector.

Next, we consider the gene regulations described in the right side of Figure 1. The gene regulations can be modelled through the construction of \(P(X_i|X_{i-1})\) for \(i = 2, \ldots, n\). We assume that gene \(j\) has \(q_j\) genes as its parents. As is shown in Figure 1, the network structure is assumed to be stable through all time points. Furthermore, according to the time dependency, only forward edges, ie edges from time \(i-1\) to \(i\), are allowed in these networks. Hence DBNs can model cycles, as is shown in Figure 2. Under these conditions, the conditional probability \(P(X_i|X_{i-1})\) can also be decomposed into the product of conditional probabilities of each gene given its parent genes:

\[
P(X_i|X_{i-1}) = P(X_{i1}|P_{i-1,1}) \times \ldots \times P(X_{ip}|P_{i-1,p})
\]

where \(P_{i-1,j} = (P_{i-1,1}, \ldots, P_{i-1,q_j})^T\) is a random variable vector of parent genes of \(j\)th gene at time \(i-1\).

Conditional probability

Dynamic Bayesian network

DBNs can be modelled through the construction of \(P(X_i|X_{i-1})\) for \(i = 2, \ldots, n\). We assume that gene \(j\) has \(q_j\) genes as its parents. As is shown in Figure 1, the network structure is assumed to be stable through all time points. Furthermore, according to the time dependency, only forward edges, ie edges from time \(i-1\) to \(i\), are allowed in these networks. Hence DBNs can model cycles, as is shown in Figure 2. Under these conditions, the conditional probability \(P(X_i|X_{i-1})\) can also be decomposed into the product of conditional probabilities of each gene given its parent genes:

\[
P(X_i|X_{i-1}) = P(X_{i1}|P_{i-1,1}) \times \ldots \times P(X_{ip}|P_{i-1,p})
\]

where \(P_{i-1,j} = (P_{i-1,1}, \ldots, P_{i-1,q_j})^T\) is a random variable vector of parent genes of \(j\)th gene at time \(i-1\).

Equations (1) and (2) hold when we use density or probability functions instead of probabilistic measure. We then obtain a DBN model in the form:

\[
f(x_{i1}, \ldots, x_{ip}) = \prod_{j=1}^p \prod_{f=1}^{q_j} g_j(x_{ij}|P_{i-1,j})
\]

where \(P_{i-1,j} = \emptyset\).

In statistics, we parameterise \(f\) by a parameter vector \(\theta\) and transfer the construction of \(f\) into the estimation of \(\theta\).
Discrete model

Although microarray data are measured as continuous data, discretisation is sometimes applied in order to remove noise. Then a discrete DBN is used for estimating gene networks. Let $U = \{u_1, \ldots, u_m\}$ be a finite set of discrete values and $I_1, \ldots, I_m$ be regions satisfying $\bigcap_{i=1}^{m} I_i = \emptyset$ and $I_i \cap I_j = \emptyset$ ($i \neq j$). Here $\mathcal{R}$ is the set of real values.

An expression value $x_{ij}$ is then transformed to $u_i$ when $x_{ij} \in I_i$. The values $g_j(x_{ij} | p_{i-1,j})$ themselves are considered as parameters, that is $\theta_{ijkl} = P(X_{ij} = u_l | P_{i-1,j} = u_{jk})$, where $u_{jk}$ is the $k$th entry of the state table of parents of the $j$th gene. For example, suppose that we discretise the expression values into two classes and that the $j$th gene has two parents. That is, the expression value $x_{ij}$ is transformed into 0 or 1 and the state table will have four entries, $u_{j1} = (0, 0)$, $u_{j2} = (0, 1)$, $u_{j3} = (1, 0)$, $u_{j4} = (1, 1)$. Then $f(x_{11}, \ldots, x_{np}; \theta)$ can be modelled as a multinomial distribution function:

**Multinomial distribution**

**Figure 1:** Graphical view of a dynamic Bayesian network model

**Figure 2:** Example of a network containing a cyclic regulation. The network (left) contains a cycle $X_1 \rightarrow X_2 \rightarrow X_4 \rightarrow X_5 \rightarrow X_1$. A Bayesian network model cannot treat such a network. On the other hand, the dynamic Bayesian network can construct a cyclic regulation by dividing states of a gene by time points (right)
Inferring gene networks from time series microarray data

Recall that microarray data need to be discretised while using discrete models. In general, discretisation is performed as follows: Let \( t_0, \ldots, t_m \) be thresholds for discretisation satisfying \( t_0 = \min x_{ij} < t_1 < \ldots < t_m = \max x_{ij} \) and \( x_{ij} \) will be classified to \( u_i \) if 
\[
t_{t-1} < x_{ij} < t_t. \quad \text{Note that several methods, such as the k-means algorithm, have been investigated for discretising microarray data (see, for example, Friedman and Goldszmidt and Pe’er et al.)}
\]
discretised thresholds as data into four classes and set the system of birds. They discretised their from a simulator of a communication system of birds. They discretised their data into four classes and set the thresholds as \( t_1 = r(x_{ij})/4, \ t_2 = r(x_{ij})/2 \) and \( t_3 = 3r(x_{ij})/4 \), where \( r(x_{ij}) = \max x_{ij} - \min x_{ij} \). This discretisation seems to be suitable for their data. Further discussion is needed when we apply this discretisation method to real microarray data in practice, however. Ong et al. used E. coli microarray data discretised into two classes. An expression value \( x_{ij} \) is transformed to \( u_i \) if \( x_{ij} > x_{i-1,j} \) or \( u_2 \) otherwise. This discretisation could be sensitive to noise when the expression level changes in a narrow range.

Continuous model
When we treat microarray data as continuous values, \( g_j(x_{ij}|p_{i-1,j}; \theta_j) \) can be modelled as a normal density function:

\[
g_j(x_{ij}|p_{i-1,j}; \theta_j) = \frac{1}{\sqrt{2\pi \sigma_j^2}} \exp \left\{ -\frac{(x_{ij} - m(p_{i-1,j}))^2}{2\sigma_j^2} \right\}
\]
where \( \theta_j \) is a parameter vector in \( g_j() \) and \( m(p_{i-1,j}) \) is a regression function from \( \mathcal{X}^{i-1} \) to \( \mathcal{X} \). For example, if we define
\[
m(p_{i-1,j}) = \beta_1^{(j)} p_{i-1,1} + \ldots + \beta_{M_j}^{(j)} p_{i-1,M_j}
\]
we obtain a linear DBN model, where \( \beta_1, \ldots, \beta_{M_j} \) are parameters.

There is no guarantee that the linear models can approximate the relationships between genes, however. For capturing even non-linear relationships between genes, Kim et al. used a non-parametric regression model based on B-splines:

\[
m(p_{i-1,j}) = \sum_{m=1}^{M_j} \gamma_{m1}^{(j)} b_m^{(j)} (p_{i-1,1}) + \ldots + \sum_{m=1}^{M_j} \gamma_{mk}^{(j)} b_m^{(j)} (p_{i-1,k})
\]
where \( \gamma_{11}^{(j)}, \ldots, \gamma_{M_k}^{(j)} \) are coefficient parameters and \( \{ b_1^{(j)}(), \ldots, b_{M_k}^{(j)}() \} \) is a prescribed set of B-splines.

CRITERION FOR LEARNING NETWORKS
By using DBN models, we can model a gene network from time series microarray data, when we know the true relationships among genes completely. Many parts of the true gene network are...
Posterior probability of the network

Still unknown and need to be estimated from microarray data, however. Hence, construction of a criterion for evaluating the goodness of the specified model is an essential point of gene network modelling. Under a Bayesian statistics framework, we can choose an optimal network by maximising the posterior probability of the network. The posterior probability of the network \( G \) is given by:

\[
P(G|X) = \frac{P(G, X)}{P(X)}
\]

where

\[
P(G, X) = \int P(G, X, \theta)d\theta
\]

\[
= P(G) \int P(X|G, \theta)P(\theta|G)d\theta
\]

\[
P(X) = \sum_{G \in \Omega} P(G, X)
\]

Here \( \Omega \) is the set of possible networks, \( P(G) \) and \( P(\theta|G) \) are prior probabilities of the network \( G \) and the parameter \( \theta \), respectively. By using the density or probability functions, the posterior probability can be expressed as

\[
\pi(G|X) \propto P(G) \prod_{j=1}^{n} f(x_{i1}, \ldots, x_{ip}; \theta_c)
\]

\[
\pi(\theta_c)d\theta_c \tag{4}
\]

Note that since the form of \( \theta \) is equivalent to the network structure, we write \( \theta_c \) as a parameter vector given network \( G \). The problem now is how to compute the high-dimensional integration in equation (4). Usually this integration can be solved analytically by using the conjugate prior as \( \pi(\theta_c) \).

BDe metric

Conjugate prior

Laplace approximation

For the discrete model defined by equation (3), the parameter vector \( \theta_c = (\theta_1^T, \ldots, \theta_{Q_1}^T)^T \) can be rewritten as \( \theta_j = (\theta_{j1}, \ldots, \theta_{jQ_j})^T \), where \( \theta_{jkl} \) corresponds to \( P(X_{ij}|n_j|P_{i<j} = n_{jk}) \). In this case, the Dirichlet distribution is often used as the prior distribution on the parameter \( \theta_{jkl} \):

\[
D(\theta_j|\alpha_j) = \frac{\Gamma\left(\sum_{k} \sum_{l} \alpha_{jkl}\right)}{\prod_{k} \prod_{l} \Gamma(\alpha_{jkl})} \prod_{k} \prod_{l} \theta_{jkl}^{\alpha_{jkl}-1}
\]

where \( \Gamma(\cdot) \) is the gamma function and \( \alpha_j = (\alpha_{j11}, \ldots, \alpha_{jQ_j})^T \) is a hyperparameter vector in the Dirichlet distribution.

Then the integration in the marginal likelihood can be solved in a closed form:

\[
\int f(x_{i1}, \ldots, x_{ip}; \theta_c)\pi(\theta_c|\alpha) d\theta_c = \prod_{j=1}^{n} \frac{\Gamma\left(\sum_{k} \alpha_{jkl}\right)}{\Gamma(\sum_{k} \alpha_{jkl} + N_{jkl})} \prod_{k} \prod_{l} \Gamma(\alpha_{jkl})
\]

where \( f(x_{i1}, \ldots, x_{ip}; \theta_c) \) is defined by equation (3) and \( \pi(\theta_c|\alpha) = \Pi_j D(\theta_j|\alpha_j) \) with \( \alpha = (\alpha_1^T, \ldots, \alpha_{Q_1}^T)^T \).

In Heckerman et al.,\(^8\) when \( \sum_{j} \sum_{k} \alpha_{jkl} \) is assumed to be constant, the posterior probability of the network results in the BDe metric. Note that Friedman et al.\(^8\) and Smith et al.\(^10\) used the BDe metric as a criterion for learning networks.

Continuous model

For computing high-dimensional integration in the marginal likelihood, Kim et al.\(^13\) used Laplace approximation.\(^17,18\) An advantage of using the Laplace approximation is that it is not necessary to consider the use of the conjugate prior distribution. Let \( \pi(\theta_c|\lambda) \) be a prior distribution on \( \theta_c \) with a hyperparameter vector \( \lambda \), satisfying \( \log \pi(\theta_c|\lambda) = O(n) \). By using Laplace approximation, the integration can be computed as:
Inferring gene networks from time series microarray data

\[ f(x_{11}, \ldots, x_{ni}; \theta_c) \pi(\theta_c | \lambda) d\theta_c \]
\[ = \exp \left\{ nl_x(\theta_c | X) \right\} d\theta_c \]
\[ = \frac{(2\pi/n)^{l/2}}{|J_\lambda(\theta_c)|^{1/2}} \exp \left\{ nl_x(\theta_c | X) \right\} \]
\[ \{ 1 + O_p(n^{-1}) \} \]

where \( r \) is the dimension of \( q_c \).

Then Kim et al. defined a criterion, called BNRC\(_{\text{dynamic}}\), of the form:

\[
BNRC_{\text{dynamic}}(G) = -2 \log \left\{ \pi(G) \left\{ f(x_{11}, \ldots, x_{ni}; \theta_c) \pi(\theta_c | \lambda) d\theta_c \right\} \right\} \\
\quad \quad \approx -2 \log \pi(G) - r \log(2\pi/n) \\
\quad \quad + \log |J_\lambda(\theta_c)| - 2 nl_x(\theta_c | X)
\]

The optimal network is chosen such that the criterion \( BNRC_{\text{dynamic}} \) is minimal.

For computing the score of criterion (4), we need to consider a prior probability of a network denoted by \( \pi(G) \). Friedman and Goldszmidt used a prior based on the MDL encoding of network \( G \). Kim et al. set \( \pi(G) \) based on the number of parent genes.

On the other hand, we can embed biological knowledge in a prior probability. Imoto et al. constructed a prior probability of a network based on biological knowledge such as binding site information, DNA–protein interaction and so on.

**CONCLUSION**

A general framework for the DBN models for constructing gene networks from time series microarray data is summarised. Both discrete and continuous models are shown in detail and criteria were introduced for evaluating these models. Three discrete models and one continuous model were focused on and the strengths and weaknesses of these methods were discussed.

We need to find the optimal network...
that gives the best score. The number of possible DAGs however is huge, even if we estimate a network containing a somewhat smaller number of genes. For example, when we have 20 genes, the number of DAGs is over $10^{72}$. Therefore the use of heuristic search methods is required and several methods such as greedy hill-climbing,\textsuperscript{8,13} simulated annealing\textsuperscript{10,23} and junction tree algorithm\textsuperscript{12} have been used to find a solution. Development of effective methods for learning networks is needed to find a better solution.

Although microarray data gives us valuable information, it is difficult to know whole gene networks by using only microarray data. Like Ong et al.\textsuperscript{12} and Imoto et al.,\textsuperscript{20} many researchers are now interested in combining microarray data with another technique, such as protein–protein interactions and binding site information,\textsuperscript{24} for extracting more information.

Acknowledgment

We would like to thank the anonymous referees for valuable and helpful comments.

References

Inferring gene networks from time series microarray data


21. URL: http://www.genome.ad.jp/kegg/

