Residual Bootstrapping and Median Filtering for Robust Estimation of Gene Networks from Microarray Data

Seiya Imoto[1,*], Tomoyuki Higuchi[2,*], SunYong Kim[1], Euna Jeong[1], and Satoru Miyano[1]

[1] Human Genome Center, Institute of Medical Science, University of Tokyo, 4-6-1 Shirokanedai, Minato-ku, Tokyo, 108-8639, Japan
{imoto, sunk, eajeong, miyano}@ims.u-tokyo.ac.jp
[2] Institute of Statistical Mathematics, 4-6-7, Minami-Azabu, Minato-ku, Tokyo, 106-8569, Japan
higuchi@ism.ac.jp

Abstract. We propose a robust estimation method of gene networks based on microarray gene expression data. It is well-known that microarray data contain a large amount of noise and some outliers that interrupt the estimation of accurate gene networks. In addition, some relationships between genes are nonlinear, and linear models thus are not enough for capturing such a complex structure. In this paper, we utilize the moving boxcel median filter and the residual bootstrap for constructing a Bayesian network in order to attain robust estimation of gene networks. We conduct Monte Carlo simulations to examine the properties of the proposed method. We also analyze *Saccharomyces cerevisiae* cell cycle data as a real data example.

1 Introduction

In recent years, estimation of gene networks based on microarray gene expression data has received considerable attention and the use of various computational methods, such as Boolean networks [1], differential equations [2, 3] and Bayesian networks [5, 6, 9, 10, 16], have been proposed in bioinformatics. In the estimation of gene networks based on microarray data, we need to consider two issues: One is how to capture the nonlinear relationships between genes. Due to the nonlinearity, the methods based on linear transformation of the data cannot be guaranteed to give sufficient results. The other problem arises from outliers included in microarray data. The outliers sometimes inhibit correct relationships or lead to spurious correlations between genes.

To estimate gene networks from microarray data, Bayesian networks [12] provide a probabilistic framework that is suitable for extracting effective information from high-dimensional noisy data. Unlike Bayesian network models based

*These authors contributed equally to this work.

V. Danos and V. Schachter (Eds.): CMSB 2004, LNBI 3082, pp. 149–160, 2005.
© Springer-Verlag Berlin Heidelberg 2005

on linear regression, the discrete Bayesian networks [5, 6, 16] can capture nonlinear relationships between genes. However, since microarray data take continuous variables, the discretization possibly leads to information loss. Furthermore, the threshold values and the number of categories for discretization are parameters that should be optimized. To avoid the discretization and capture the nonlinearity, Imoto *et al.* [9, 10] proposed a Bayesian network and nonparametric regression model for estimating gene networks. However, a problem that still remains to be solved is how we treat the effect of outliers. Since nonparametric regression model employed in a Bayesian network is based on the Gaussian distribution, the outliers in microarray data sometimes affect the resulting networks. Therefore, development of statistical methods that can handle outliers and nonlinearity appropriately is considered as an important problem.

In this paper, we propose the use of moving boxcel median filter and residual bootstrap [4] for constructing Bayesian networks aimed at robust estimation of gene networks. By using the moving boxcel median filter, we can reduce the effect of outliers and can estimate nonlinear relationships between genes suitably. Residual bootstrap virtually realizes a model for measurement noise included in microarray data and gives a stable estimation of gene networks. In Section 2.1 we give the explanation of Bayesian networks. Since microarray data contain measurement noise, we introduce a "virtual sample method" to realize a measurement noise model in Section 2.2. The moving boxcel median filter and the residual bootstrap are introduced in Section 2.3 and 2.4, respectively. A greedy hill-climbing algorithm for choosing the optimal graph from candidates is introduced in Section 2.5. We conduct Monte Carlo simulations to show the effectiveness of the proposed method in Section 3.1. In Section 3.2, we analyze *Saccharomyces cerevisiae* cell cycle data collected by Spellman *et al.* [19] as a real data example.

2 Proposed Method

2.1 Bayesian Networks

In the context of Bayesian networks, we consider the directed acyclic graph (DAG) encoding the Markov assumption between nodes, i.e. a graph contains no cyclic regulations and a node depends only on its direct parents. In the Bayesian network models, a gene is regarded as a random variable and shown as a node. Under above assumptions, we can decompose the joint probability of all genes into the product of the conditional probabilities as

$$P(X_1, ..., X_p) = P(X_1|parent(X_1)) \times \cdots \times P(X_p|parent(X_p)), \qquad (1)$$

where X_j $(j = 1, ..., p)$ is a random variable and corresponds to the jth gene, denoted by gene_j, and $parent(X_j)$ is a random variable vector of the direct parents of gene_j. For example, if gene_2 and gene_3 are the direct parents of gene_1 in the DAG, G, we then have $parent(X_1) = (X_2, X_3)^T$. Therefore, an essential problem for constructing a Bayesian network is the computation of each conditional probability $P(X_j|parent(X_j))$.

The computation of $P(X_j|parent(X_j))$ is essentially the same as the regression problem. In general, a regression model can capture the relationship between X_j and $parent(X_j)$ as

$$X_j = h_j(parent(X_j)) + \varepsilon_j, \tag{2}$$

where ε_j is noise satisfying $E[\varepsilon_j] = 0$ and $V(\varepsilon_j) = \sigma_j^2$, and $h_j(parent(X_j))$ is a function that describes the structure between X_j and $parent(X_j)$. Imoto *et al.* [9, 10] gave $h_j(parent(X_j))$ as the additive form

$$h_j(parent(X_j)) = h_{j1}(parent(X_j)_1) + \cdots + h_{jq_j}(parent(X_j)_{q_j}), \tag{3}$$

where $parent(X_j)_k$ $(k = 1, ..., q_j)$ is the kth parent of gene$_j$, and $h_{jk}(x)$ is a smooth function from R to R. For the noise, Imoto *et al.* [10] assumed the heterogeneous error variances for reducing the effect of outliers in microarray data. In the next section, we introduce the moving boxcel median filter to achieve more robustness in the estimation of the relationships between genes against outliers.

2.2 Virtual Samples

Suppose that we have p genes' expression data observed by n microarrays. That is, $\boldsymbol{x}_i = (x_{i1}, ..., x_{ij}, ..., x_{ip})^T$ is a p-dimensional gene expression vector from ith microarray and $\boldsymbol{X} = (\boldsymbol{x}_1, ..., \boldsymbol{x}_n)^T$ is an $n \times p$ gene expression matrix whose (i, j)th element, x_{ij}, is an expression value of gene$_j$ of ith microarray. Since microarray data contain various noise including measurement noise, we can decompose

$$\boldsymbol{x}_i = \boldsymbol{x}_i^{internal} + \boldsymbol{\eta}_i, \tag{4}$$

where $\boldsymbol{\eta}_i$ is a p-dimensional noise vector. By using $\boldsymbol{x}_i^{internal}$ and density functions instead of probability measure in (1), the purpose is to express a Bayesian network model as

$$f(\boldsymbol{x}_i^{internal}) = \prod_{j=1}^{p} f_j(x_{ij}^{internal}|\boldsymbol{p}_{ij}^{internal}),$$

where $\boldsymbol{p}_{ij}$ is a parent gene vector of gene$_j$ of ith microarray, i.e. if $parent(X_1) = (X_2, X_3)^T$, we have $\boldsymbol{p}_{i1} = (x_{i2}, x_{i3})^T$, and $\boldsymbol{p}_{ij}^{internal}$ is defined by the same as (4). In previous works, Bayesian network models are expressed as $f(\boldsymbol{x}_i) = \prod_{j=1}^{p} f_j(x_{ij}|\boldsymbol{p}_{ij})$. Therefore, one of the differences between the proposed method and previous works [9, 10] is in (4). However, gene expressions are observed by a few (typically one) microarrays for an experimental condition. Therefore, it is difficult to separate the true signal, $\boldsymbol{x}_i^{internal}$, from the noise like (4). As an alternative approach to realize the model (4), we make virtual observations for each gene as follows: We generate M virtual samples, $\{x_{ij}^{*1}, ..., x_{ij}^{*M}\}$, for each observation, x_{ij}, from the following system

$$x_{ij}^{*m} = x_{ij} + \varepsilon_{ij}^{*m}, \quad m = 1, ..., M, \tag{5}$$

where ε_{ij}^{*m} depends independently and normally on mean 0 and variance $\sigma_{j_0}^2$. For the setting of $\sigma_{j_0}^2$, we set $\sigma_{j_0}^2 = \alpha \sum_i (x_{ij} - \sum_{i'} x_{i'j}/n)^2/n$ with $\alpha = 0.2$ empirically. However, it is often the case that the setting $\sigma_{j_0}^2$ is inappropriate. We then update x_{ij}^{*m}'s by using the residual bootstrap described in Section 2.4. By using the system (5), we can model the measurement noise included in x_{ij} virtually. In standard regression methods, the measurement noise is usually ignored. On the other hand, the proposed method allows the measurement noise and estimate the relationship between variables suitably.

2.3 Moving Boxcel Median Filter

For constructing $h_j(\boldsymbol{x})$ in (2), we apply the moving boxcel median filter to $((\boldsymbol{p}_{ij}^{*m})^T, x_{ij})$ for $i = 1, ..., n;\ m = 0, ..., M$, where $\boldsymbol{p}_{ij}^{*0} = \boldsymbol{p}_{ij}$. To explain the moving boxcel median filter, we consider a simple example that gene$_1$ has one parent gene, gene$_2$. The moving boxcel median filter estimate ($X_1 = \hat{h}_1(X_2)$) is obtained as follows: First, we compute

$$\Delta x_j = \frac{\max_{i=1,\ldots,n}(x_{ij}) - \min_{i=1,\ldots,n}(x_{ij})}{2\lambda},$$

for $j = 2$. Here λ is a constant and we set 20 as an appropriate value in the later section. The estimated value at a, i.e. $\hat{h}_1(a)$, is the median of the observations of x_{i1} whose parent observations x_{i2}^* are included in the interval $I_{\Delta x_2}(a) = [a - \Delta x_2, a + \Delta x_2]$, where $a \in Val(X_2) = [\min_i(x_{i2}), \max_i(x_{i2})]$. The moving boxcel median filter curve is thus obtained by moving the interval $[a - \Delta x_2, a + \Delta x_2]$ from $a = \min_i(x_{i2})$ to $a = \max_i(x_{i2})$. Figure 1 shows an example of the moving boxcel median filter estimate. Figure 1 (a) is the scatterplot of the expression data of gene$_2$ and gene$_1$, which are generated numerically. Figure 1 (b) is the scatterplot of $\{(x_{i2}^{*m}, x_{i1}^*) | i = 1, ..., n; m = 0, ..., M\}$ with $x_{i2}^{*0} = x_{i2}$. We set $\alpha = 0.2$, $\lambda = 20$ and $M = 10$, and generate virtual samples. In Figure 1 (b), the moving boxcel median filter estimate at $X_2 = a$ is given as the median of the data in the shadowed area. In Figure 1 (c), the true relationship between gene$_1$ and gene$_2$ is shown as the dotted curve and the moving boxcel median filter estimate is the solid curve. It is clear that the moving boxcel median filter can reduce the effect of the outliers (in right-bottom) and construct a suitable relationship, which is close to the true one.

The moving boxcel median filter can be easily extended for more than two parent genes cases. If gene$_1$ has three parents, gene$_2$, gene$_3$ and gene$_4$, the interval defined above is extended as

$$I_{\Delta x_2}(a) \otimes I_{\Delta x_3}(b) \otimes I_{\Delta x_4}(c)$$

with $a \in Val(X_2)$, $b \in Val(X_3)$ and $c \in Val(X_4)$. Note that, by using the moving boxcel median filter, we do not need to assume the additive form (3) as a regressor and can model the relationship between genes by using more general form given in (2).

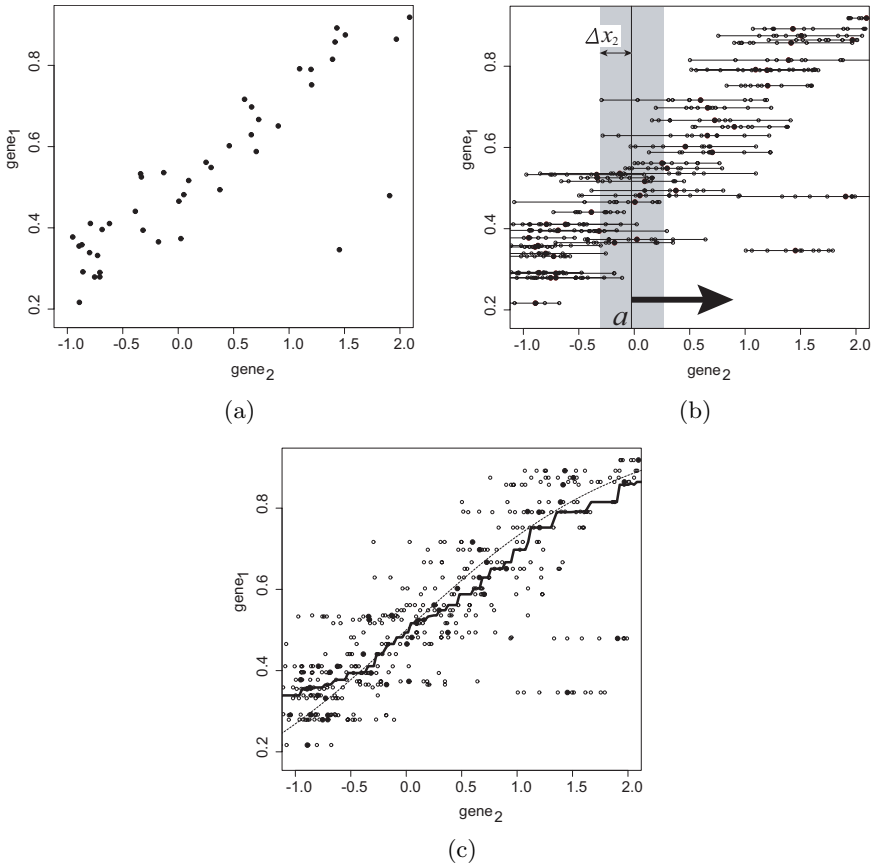


Fig. 1. Example of the moving boxcel median filter. (a) Scatterplot of $\{(x_{i2}, x_{i1})|i = 1, ..., n\}$. (b) Scatterplot of $\{(x_{i2}^{*m}, x_{i1}^{*m})|i = 1, ..., n; m = 0, ..., M\}$. (c) Solid curve: moving boxcel median filter estimate, Dotted curve: true curve

2.4 Residual Bootstrap

Since microarray data contain various noise including measurement noise, we model this fact by the model defined by (4) and (5). However, in the system (5), we determined the volume of the error variance $\sigma_{j_0}^2$ empirically. Therefore, it is often the case that $\sigma_{j_0}^2$ is not appropriate. Although the value of $\sigma_{j_0}^2$ can be set as a somewhat appropriate value by using some information about the true relationships or resulting networks, it is clear that this approach has a limitation in practice. To solve this problem, we use the residual bootstrap method and recreate M virtual samples for each observation.

Suppose that parent genes of each gene are temporarily obtained and the moving boxcel median filter estimate for each relationship is computed, the procedure of the residual bootstrap can be expressed as follows:

Homogeneous Error Variance Model
We compute the residuals for each gene as

$$\varepsilon^*_{ij} = x_{ij} - \hat{h}_j(\boldsymbol{p}_{ij}).$$

For each observation, we recreate M virtual samples $\{x^{*1}_{ij}, ..., x^{*M}_{ij}\}$ by

$$x^{*m}_{ij} = \hat{h}_j(\boldsymbol{p}_{ij}) + \varepsilon^{*m}_{ij}, \quad m = 1, ..., M,$$

where ε^{*m}_{ij} is a bootstrap sample obtained by resampling from $\{\varepsilon^*_{1j}, ..., \varepsilon^*_{nj}\}$ with replacement. Hence, we obtain new virtual samples x^{*m}_{ij} $(m = 1, ..., M)$ for x_{ij}.

Heterogeneous Error Variance Model
In heterogeneous error variance model, we assume that the distribution of ε_{ij} depends not only on the index i (gene), but also the index j (microarray). This situation is more natural than the case of the homogeneous error variance, e.g. Imoto *et al.* [10]. First, we define the neighborhood of $\boldsymbol{p}_{ij}$, denoted by $Neighbor(\boldsymbol{p}_{ij})$, by using Δx_j and the virtual samples $\boldsymbol{p}^{*m}_{ij}$. For example, if $\boldsymbol{p}_{ij} = (x_{i2}, x_{i3})^T$, then we have

$$Neighbor(\boldsymbol{p}_{ij}) = \{(x^{*m'}_{i'2}, x^{*m'}_{i'3}) | x^{*m'}_{i'2} \in I_{\Delta x_2}(x_{i2}), x^{*m'}_{i'3} \in I_{\Delta x_3}(x_{i3})\}.$$

By using the virtual samples included in $Neighbor(\boldsymbol{p}_{ij})$, the residuals are obtained as

$$\varepsilon^{*m'}_{i'j} = x_{i'j} - \hat{h}_j(\boldsymbol{p}^{*m'}_{i'j})$$

with $\boldsymbol{p}^{*m'}_{i'j} \in Neighbor(\boldsymbol{p}_{ij})$. We then make M virtual samples for x_{ij} as

$$x^{*m}_{ij} = \hat{h}_j(\boldsymbol{p}_{ij}) + \varepsilon^{*m}_{ij}, \quad m = 1, ..., M,$$

where ε^{*m}_{ij} is a bootstrap sample from $\{\varepsilon^{*m'}_{i'j}\}_{i',m'}$.

After updating the virtual samples, we will fit the moving boxcel median filter described in the previous section to the microarray data and the updated virtual samples. We repeat this iteration until the stable estimate is obtained. Note that the moving boxcel median filter and the residual bootstrap can be applied when we set the parents of each gene. In the next section, we describe the selection of the graph structure and show an algorithm for estimating gene networks by using the moving boxcel median filter and the residual bootstrap method.

2.5 Graph Selection

In the estimation of gene networks from gene expression data, an essential problem is the choice of the optimal graph structure that gives the best approximation of the system underlying the data. From a statistical view point, this problem

can be considered as a statistical model selection problem. For the graph selection problem, we use the residual sum of squares as a criterion for choosing the optimal graph structure

$$\sigma_G^2 = \frac{1}{p}\sum_{j=1}^{p}\hat{\sigma}_j^2, \tag{6}$$

where $\hat{\sigma}_j^2$ is defined by

$$\hat{\sigma}_j^2 = \begin{cases} \sigma_{j0}^2/\alpha & \text{for top genes,} \\ \dfrac{1}{n}\displaystyle\sum_{i=1}^{n}\{x_{ij} - \hat{h}_j(\boldsymbol{p}_{ij})\}^2 & \text{otherwise.} \end{cases}$$

The optimal graph $\hat{G}$ is obtained as the minimizer of σ_G^2. Note that the criterion (6) can evaluate graphs that are obtained by the same λ, M and α.

When we focus on a small gene networks, the optimal graph structure can be obtained by using a suitable learning algorithm, e.g. Ott *et al.* [15]. However, for large gene networks, we use a greedy hill-climbing algorithm for learning graph structures. Our greedy hill-climbing algorithm can be written as follows:

Initial Step

Step1. For all genes $\boldsymbol{x}_{(j)} = (x_{1j}, ..., x_{nj})^T$, create M virtual samples

$$\boldsymbol{x}_{(j)}^* = (x_{1j}^{*1}, ..., x_{nj}^{*1}, ..., x_{1j}^{*m}, ..., x_{nj}^{*m}, ..., x_{1j}^{*M}, ..., x_{nj}^{*M})^T,$$

where $x_{ij}^{*m} \sim N(x_{ij}, \sigma_{j0}^2)$.

Step2. For each pair $\{(x_{ik}^{*m}, x_{ij}) | i = 1, ..., n; m = 0, ..., M\}$, apply the moving boxcel median filter and take 10 best genes in terms of $\hat{\sigma}_j^2$ as candidate parents of gene$_j$. We denote 10 candidate parents of gene$_j$ as "pa$_{jk}$" for $k = 1, ..., 10$.

Learning Step

Step3. For each gene$_j$ $(j = 1, ..., p)$:

Step3-1. For each candidate parent pa$_{jk}$ $(k = 1, ..., 10)$:

Step3-1-(a). Test one of the following operations, apply moving median filter, and calculate $\hat{\sigma}_{j,test}^2$.

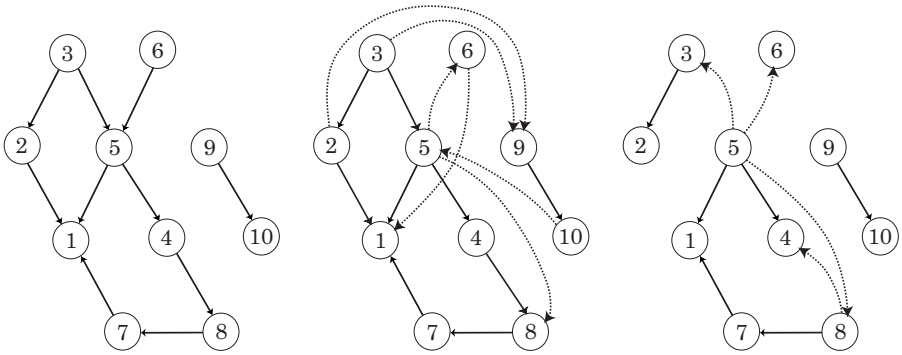
- If pa$_{jk} \rightarrow$ gene$_j$ does not exist, add this edge.
- If pa$_{jk} \rightarrow$ gene$_j$ exists, remove this edge.
- If pa$_{jk} \leftarrow$ gene$_j$ exists, reverse this edge.

Step3-1-(b). If $\hat{\sigma}_{j,test}^2 < \hat{\sigma}_j^2$, apply the operation in Step3-1(a) and set $\hat{\sigma}_j^2 = \hat{\sigma}_{j,test}^2$. Otherwise, no operation is conducted.

Step3-2. Update virtual samples for gene$_j$.

Step4. Repeat Step3 until σ_G^2 converges.

In the learning step of the above algorithm, the resulting network depends on the learning order of genes. Therefore, we permute the learning order and take the best network out of 10 networks as the optimal one.



(a) (b) (c)

$$X_1 = X_2^2 + 2\sin(X_5) - 2X_7 + \varepsilon_1, \quad X_2 = \{1 + \exp(-4X_3)\}^{-1} + \varepsilon_2,$$
$$X_3 = \varepsilon_3, \quad X_4 = X_5^2/3 + \varepsilon_4, \; X_5 = X_3 - X_6^2 + \varepsilon_5, \quad X_6 = \varepsilon_6,$$
$$X_7 = \begin{cases} -1 + \varepsilon_7, \; X_8 \leq -0.5 \\ X_8 + \varepsilon_7, \; -0.5 < X_8 \leq 0.5 \\ 1 + \varepsilon_7, \quad 0.5 < X_8 \end{cases}$$
$$X_8 = \exp(-X_4 - 1)/2 + \varepsilon_8, \quad X_9 = \varepsilon_9, \quad X_{10} = \cos(X_9) + \varepsilon_{10}.$$

(d)

Fig. 2. True model and estimated networks of Monte Carlo simulations: (a) True network. (b) Estimated network by the previous method [9]. (c) Estimated network by the proposed method ($n = 100$, $\alpha = 0.2$, $\lambda = 20$ and $M = 100$). (d) Functions between nodes

3 Computational Experiments

3.1 Monte Carlo Simulations

We conduct Monte Carlo simulations to examine the properties of the proposed method by comparing with the previous method [9]. The simulated microarray data were generated from the artificial network of Figure 2 (a) with the functional structures between nodes shown in Figure 2 (d). The observations of the child variable are generated after transforming the observations of the parent variables to mean 0 and variance 1. After generating the data from the true system and making a matrix $\boldsymbol{X}' = (\boldsymbol{x}'_1, ..., \boldsymbol{x}'_n)^T$, we then add the noise corresponding to the measurement noise by

$$\boldsymbol{x}_i = \boldsymbol{x}'_i + \boldsymbol{\eta}_i, \quad i = 1, ..., n,$$

where $\boldsymbol{\eta}_i = (\eta_{i1}, ..., \eta_{ip})^T$ is a p-dimensional noise vector and η_{ij} depends on the mixture normal distribution of the form

$$\eta_{ij} \sim (1 - \kappa)N(0, \{Rng(x_{ij})/20\}^2) + \kappa N(0, \{Rng(x_{ij})/10\}^2).$$

Here we set $\kappa = 0.05$ and $Rng(x_{ij}) = \max_i(x_{ij}) - \min_i(x_{ij})$. Hence a microarray data matrix we used is defined by $\boldsymbol{X} = (\boldsymbol{x}_1, ..., \boldsymbol{x}_n)^T$. A network was rebuilt from simulated data consisting of $n = 50$ or $n = 100$ observations, which corresponds to 50 or 100 microarrays.

Figure 2 (b) and (c) are typical examples of the estimated networks for $n = 100$. We tried various settings of α, M and λ and set $\alpha = 0.2$, $M = 100$ and $\lambda = 20$ as appropriate values. In Figure 2 (b) and (c), the solid edges are correctly estimated edges and the dotted edges are the falsely estimated edges by the previous method [9] and the proposed method. It is clear that by adding measurement noise and some outliers, the previous method estimated some spurious relations that are false positives. On the other hand, by comparing with the previous method, the proposed method can reduce the number of false positives. We observe that a shortcoming of the proposed method from this simulation that the proposed method sometimes estimates edges that are inverse direction. However, if we consider the estimated network as an undirected graph, the sensitivity (the number of correctly estimated edges divided by the number of the estimated edges) of the proposed method is much higher than that of the previous method.

In the result of Monte Carlo simulations, it is shown that our method is robust to the noise which is independent of fluctuations of the gene network. Various instrumental and observation noises can be properly removed together with constructing Bayesian networks for estimating gene networks, resulting in giving robust and reliable estimates.

3.2 Real Data Example

In the real data example, we use *Saccharomyces cerevisiae* cell cycle data collected by Spellman *et al.* [19] and focus on 52 genes. These genes made a subnetwork estimated by Imoto *et al.* [9]. Figure 3 is the resulting network obtained by the proposed method with the same α, M and λ in the Monte Carlo simulations in the previous section.

The results of the real data example can be summarized as follows: In budding yeast, *Saccharomyces cerevisiae*, the homeodomain protein, *YOX1*, is a repressor that restricts early cell cycle boxes (ECB)-mediated transcription to the M/G1 phase of the cell cycle [17]. As a transcription factor, *YOX1* binds nearly 30 genes, including *CLN2* [7], that are important for DNA synthesis and repair. An ECB element (*CLN3*) activates SBF and MBF, two late G1-specific transcription complexes. Both SBF and MBF cause a burst of transcription of the late G1 cyclins, *CLN1* and *CLN2*, and other genes required for S phase (B cyclins), *CLB5* and *CLB6*, respectively. The two late G1 cyclins, *CLN1* and *CLN2*, have important effects for progression into S phase, i.e. an increase in the level of *CLB5,6-CDC* kinase activity sufficient to permit initiation of DNA replication [8]. In the resulting network shown in Figure 3, *YOX1* connects directly to *CLN2* and *CLN2* connects to *CLN1* and *CLB6*, while the resulting network of Imoto *et al.* [9] contains connections from *CLN2* to *CLN1* through *YOX1*. In comparison with Imoto *et al.* [9], our network can reflect functional correlations

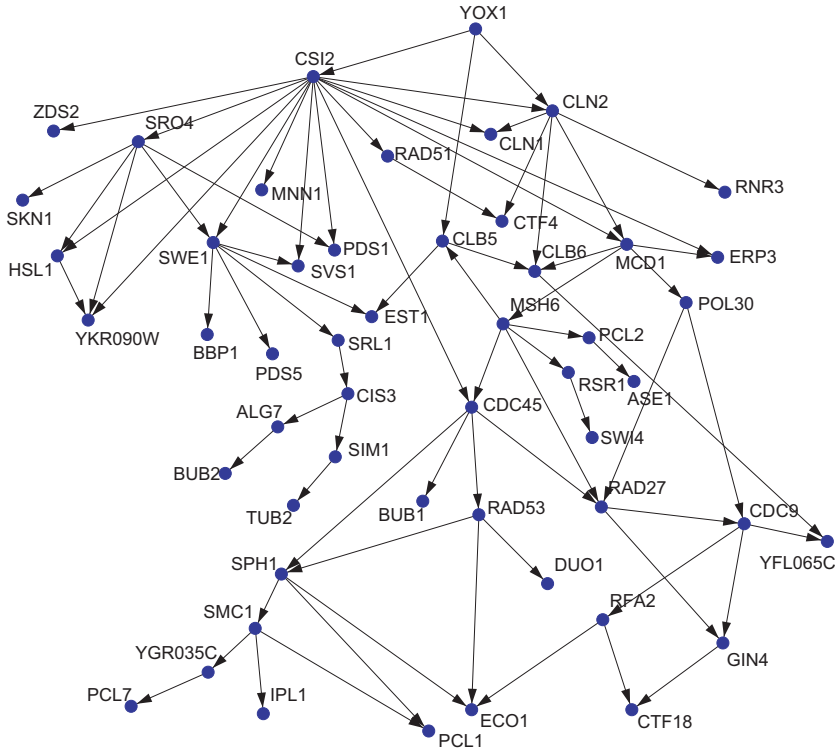


Fig. 3. Resulting cell cycle gene network obtained by the proposed method

between *YOX1* and the late G1-cyclins and B-cyclins more precisely. An interesting relationship is found that the cell wall biogenesis gene *CSI2* triggers many genes in our resulting network, while only three genes are regulated by *CSI2* in the network of Imoto *et al.* [9]. Although the function of *CSI2* as a structural component of the chitin synthase 3 complex is still unclear, Friedman *et al.* [5] showed that *CLN2*, *RNR3*, *SVS1*, *SRO4* and *RAD51* are highly correlated each other and *CSI2-SRO4* pair works together as cell wall regulation at the plasma membrane. In addition, the yeast protein function assignment [21] by Marcotte *et al.* [13] shows that *CSI2* is functionally linked with vanadate sensitive suppressor *SVS1*. Based on the resulting network in Figure 3, we may suggest that *CSI2* interacts many genes and may play an important role in cell cycle. Based on those observations, our proposed methods are capable of extracting causal relationships between genes effectively from gene expression data.

4 Discussion

We proposed the use of the moving boxcel median filter and the residual bootstrap for constructing Bayesian networks aimed at robust estimation of gene

networks from microarray data. Main difficulties of the estimation of gene networks based on microarray data are caused by the outliers and the nonlinearity of the relationships between genes. We solved these problems by the proposed method and attained an accurate gene network rather than the previous method.

We consider the following problems as our future topics: We set the parameters α, M and λ empirically. These parameters, however, could affect the resulting networks and we need to develop a suitable criterion for choosing them from a statistical point of view. Recently, researches have been focused on using multiple types of genomic data such as binding site information, protein-protein interaction and so on, together with microarray data for extracting more reliable information [11, 14, 18, 20]. We would like to extend our method to handle such genomic data for estimating more accurate gene networks.

References

1. Akutsu, T., Miyano, S., Kuhara, S.: Identification of genetic networks from a small number of gene expression patterns under the Boolean network model. Pac. Symp. Biocomput. **4** (1999) 17–28.
2. Chen, T., He, H.L., Church, G.M.: Modeling gene expression with differential equations. Pac. Symp. Biocomput. **4** (1999) 29–40.
3. De Hoon, M.J.L., Imoto, S., Kobayashi, K., Ogasawara, N., Miyano, S.: Inferring gene regulatory networks from time-ordered gene expression data of *Bacillus subtilis* using differential equations. Pac. Symp. Biocomput. **8** (2003) 17–28.
4. Efron, B., Tibshirani, R.J.: An Introduction to the Bootstrap. Chapman & Hall/CRC. (1993).
5. Friedman, N., Linial, M., Nachman, I., Pe'er, D.: Using Bayesian network to analyze expression data. J. Comp. Biol. **7** (2000) 601–620.
6. Hartemink, A.J., Gifford, D.K., Jaakkola, T.S., Young, R.A.: Combining location and expression data for principled discovery of genetic regulatory network models. Pac. Symp. Biocomput. **7** (2002) 437–449.
7. Horak, C.E., Luscombe, N.M., Qian, J., Bertone, P., Piccirrillo, S., Gerstein, M., Snyder, M.:Complex transcriptional circuitry at the G1/S transition in *Saccharomyces cerevisiae*. Genes & Development **16** (2002) 3017–3033.
8. Huberman, J.A.: Cell cycle control of S phase: a comparison of two yeasts. Chromosoma **105** (1996) 197–203.
9. Imoto, S., Goto, T., Miyano, S.: Estimation of genetic networks and functional structures between genes by using Bayesian network and nonparametric regression. Pac. Symp. Biocomput. **7** (2002) 175–186.
10. Imoto, S., Kim, S., Goto, T., Aburatani, S., Tashiro, K., Kuhara, S., Miyano, S.: Bayesian network and nonparametric heteroscedastic regression for nonlinear modeling of genetic network. J. Bioinform. Comp. Biol. **1**(2) (2003) 231–252.
11. Imoto, S., Higuchi, T., Goto, T., Tashiro, K., Kuhara, S., Miyano, S.: Combining microarrays and biological knowledge for estimating gene networks via Bayesian networks. J. Bioinform. Comp. Biol. **2** (2004) in press. (Preliminary version has appeared in Proc. 2nd IEEE Computational Systems Bioinformatics (2003) 104–113).
12. Jensen, F.V.: An Introduction to Bayesian Networks. University College London Press, (1996).

13. Marcotte, E.M., Pellegrini, M. Thompson, M.J., Yeates, T.O., Eisenberg, D.: A combined algorithm for genome-wide prediction of protein function. Nature **402** (1999) 83–86.
14. Nariai, N., Kim, S., Imoto, S., Miyano, S.: Using protein-protein interactions for refining gene networks estimated from microarray data by Bayesian networks. Pac. Symp. Biocomput. **9** (2004) 336–347.
15. Ott, S., Imoto, S., Miyano, S.: Finding optimal models for small gene networks. Pac. Symp. Biocomput. **9** (2004) 557–567.
16. Pe'er, D., Regev, A., Elidan, G., Friedman, N.: Inferring subnetworks from perturbed expression profiles. Bioinformatics **17** (ISBM2001) S215–S224.
17. Pramila, T., Shawna, M., GuhaThakurta, D., Jemiolo, D., Breeden, L.L.: Conserved homeodomain proteins interact with MADS box protein Mcm1 to restrict ECB-dependent transcription to the M/G1 phase of the cell cycle. Genes & Development **16** (2002) 3034–3045
18. Segal, E., Wang, H., Koller, D.: Discovering molecular pathways from protein interaction and gene expression data. Bioinformatics **19** (ISMB2003) i264–i272.
19. Spellman, P.T., Sherlock, G., Zhang, M.Q., Iyer, V.R., Anders, K., Eisen, M.B., Brown, P.O., Botstein, D., Futcher, B.: Comprehensive identification of cell cycle-regulated genes of the yeast *Saccharomyces cerevisiae* by microarray hybridization. Mol. Biol. Cell **9** (1998) 3273–3297.
20. Tamada, Y., Kim, S., Bannai, H., Imoto, S., Tashiro, K., Kuhara, S., Miyano, S.: Estimating gene networks from gene expression data by combining Bayesian network model with promoter element detection. Bioinformatics **19** (ECCB2003) ii227–ii236.
21. `http://www.doe-mbi.ucla.edu/Services/GPofYPF/yeastlist.html`