Machine Learning, 52, 67–89, 2003

Self-Organizing Latent Lattice Models for Temporal Gene Expression Profiling

BYOUNG-TAK ZHANG btzhang@bi.snu.ac.kr
Biointelligence Laboratory, School of Computer Science and Engineering, Seoul National University, Seoul 151-742, Korea; Center for Bioinformation Technology (CBIT), Seoul National University, Seoul 151-742, Korea

JINSAN YANG jsyang@bi.snu.ac.kr
Biointelligence Laboratory, School of Computer Science and Engineering, Seoul National University, Seoul 151-742, Korea

SUNG WOOK CHI swchi@bi.snu.ac.kr
Center for Bioinformation Technology (CBIT), Seoul National University, Seoul 151-742, Korea

Editors: Paola Sebastiani, Isaac S. Kohane and Marco F. Ramoni

Abstract. DNA microarrays are a high-throughput technology useful for functional genomics and gene expression analysis. While many microarray data are generated in sequence, most expression analysis tools are not utilizing the temporal information. Temporal expression profiling is important in many applications, including developmental studies, pathway analysis, and disease prognosis. In this paper, we develop a learning method designed for temporal gene expression profiling from massive DNA-microarray data. It attempts to learn probabilistic lattice maps of the gene expressions, which are then used for profiling the trajectories of temporal expressions of co-regulated genes. This self-organizing latent lattice (SOLL) model combines the topographic mapping capability of self-organizing maps and the generative property of probabilistic latent-variable models. We empirically evaluate the SOLL model on a set of cell-cycle regulation data, demonstrating its effectiveness in discovering the temporal patterns of correlated genes and its usefulness as a tool for generating and visualizing interesting hypotheses.

Keywords: DNA-microarray data, correlated genes, temporal expression profiling, learning latent-variable models, visualization

1. Introduction

Recently, advanced microarray technologies make it possible to study biological processes at the level of gene expression (Lockhart et al., 1996; Young, 2000). An important aspect of these techniques is the monitoring of the expression profiles of thousands of genes during biological processes that develop over time under selected time points (De Risi, Iyer, & Brown, 1997). In the cell cycle process, for example, the activation of genes is controlled through a complex biological process. In many cases, the expression levels of one gene regulate those of another gene and the identification of these relations is important to the understanding of the transcriptional regulation of the cell cycle process (Brian, 1997; Cho et al., 1998).

Several methods for gene expression profiling have been proposed. Most of the studies are focused on the analysis of single gene's expressions by clustering similar expression patterns over the expression time (Eisen et al., 1998). Since the transcriptional regulation occurs between two sets of genes with mutual interaction, the expression profiling based on multiple genes is more appropriate than that on the single genes. For the functional analysis of the mutually co-regulated genes, it is usually assumed that the various measurements of co-regulations are coming from a small number of underlying sources. Thus feature-reduction methods are used widely for the analysis of single gene expressions. Raychaudhuri, Stuart and Altman (2000) have used principal component analysis to select a meaningful set of time intervals and clustering genes for yeast data. Tamayo et al. (1999) implemented a self-organizing map (SOM) algorithm to the clustering of yeast genes.

In this paper, we present a gene expression profiling method that analyzes the temporal patterns of multiple correlated genes. It attempts to learn probabilistic lattice maps of the gene expressions, which are then used for profiling the trajectories of temporal expressions of multiple correlated genes. This self-organizing latent lattice (SOLL) model combines the topographic mapping capability of self-organizing maps and the generative property of probabilistic latent-variable models. Latent-variable models are widely used when there are inherent sources for the observed features. The use of latent-variables reduces the number of features and the lattice shape is useful for data visualization as demonstrated in generative topographic mapping or GTM (Bishop, Svensen, & Williams, 1998). Unlike SOM and similar to GTM, SOLL selects the grid points in the latent lattice probabilistically. As in SOM, the locations of the grid points of SOLL are modified during learning. This distinguishes SOLL from GTM where the locations of grid points are fixed. The construction of elastic latent-lattice structures in SOLL avoids the careful positioning of the initial grid points.

SOLL tries to find out the temporal co-regulation patterns of pairs of genes. The rationale behind the pairwise correlation profiling is biological: to find out the multiple correlated genes out of given gene groups, the combined information about mutual expressions should be used rather than separate expression levels. In our approach, we select all the possible combinations of genes and their features to find out meaningful pairs of correlated genes. When using the learned latent lattice model, a pair of gene expression values is given and from this SOLL generates a trajectory on the 2D lattice. The trajectory is assigned a cluster of trajectories acquired and labelled during the learning phase. The generation and use of trajectory patterns (rather than a single point) on the lattice are defining features of SOLL as a temporal expression profiler.

The main contribution of this paper is twofold. From the machine learning point of view, we present a novel learning model for the analysis of correlated variables through dimension reduction and visualization. From the bioinformatics point of view, we present a novel tool for temporal expression analysis of multiple correlated genes. Because of their complex nature, correlated genes have not been widely studied although their importance in biology is significant (Price, Nasmyth, & Schuster, 1991). By the nature of DNA expression data and gene mechanism, the analysis of the temporal trajectory patterns is the most natural way of profiling these mutually co-regulated genes (Sasik et al., 2002). By finding characteristic patterns of trajectories on the latent lattice, biologically meaningful genes can be discovered.

Our application of the self-organizing latent lattice model to the yeast cell-cycle regulation data makes some interesting biological discoveries. These include confirmation of the phenomenon of the activation of the genes containing the MCB or SCB element in the late G1 phase (Koch & Nasmyth, 1994). In the SOLL approach, the profiling of correlated genomic expressions can be analyzed through temporal trajectories of time intervals. The high dimensional features can be represented in the latent space and their trajectories are visualized. In the analysis of a subset of yeast genes, patterns and trajectories of biologically meaningful co-regulated genes (i.e., genes in late G1 phase regulated by CLN3) are obtained.

The paper is organized as follows. In Section 2, temporal gene expression profiling is described. In Section 3, the algorithm and the underlying idea of SOLL are explained. In Section 4, the application of SOLL to the temporal gene expression profiling is presented. In Section 5, the experimental results for the yeast data are reported and compared with other methods. Finally, possible limitations and further developments of the method are discussed in Section 6.

2. Temporal profiling of gene expression data

We used Spellman's (1998) microarray data sets consisting of amplified genomic DNA of *S. cerevisiae* ORFs for cell cycle analysis. In these data sets, the time series of relative levels of mRNA were measured in cell cultures synchronized by three different methods. In the first set, the alpha mating factor pheromone was used to arrest cells in G1 in every 7 minute for 140 minutes. In the second set, centrifugal elutriation was used to collect small G1 cells in every 30 minutes during 6.5 hours. In the third set, temperature sensitive cdc15 mutant was used to arrest cells in late mitosis in every 10 minutes during 300 minutes. Figure 1 shows the change of expression levels for 104 known yeast genes in the alpha factor arrested data set. The analysis of these data sets enables researchers to identify distinct cycles or waves of expressions that are meaningful in biological processes.

To ensure proper progression of the cell cycle, the proteins such as cyclins must be temporarily expressed at appropriate times. These proteins regulate the expression of other genes which mediate the essential events in cell cycle such as DNA replication and cytokinesis (Futcher, 2000). These temporarily regulated genes activate the transcription of the genes that will be necessary in the following cell cycle and also down-regulate the previously expressed genes (figure 2). So temporal profiling of correlated genes dependent on the cell cycle is important to investigate the mechanism of gene expression in cell cycle regulation.

In the analysis of the gene expression data, most approaches are based on the expression levels of a single gene as a time series or part of the experimental conditions (see e.g. Lukashin & Fuchs, 2001; Aach & Church, 2001). Clustering has been the most popular method for this approach. For example, the self-organizing map (SOM) is an unsupervised learning method that provides robust and accurate clustering of large amounts of noisy data (Kohonen, 1990). SOM provides a two-dimensional grid structure that facilitates interpretations of the results. SOM is a topology-preserving neural network where the number of nodes is fixed from the beginning. This makes it difficult to recover the natural cluster structure of the data set (Herrero, Valencia, & Dopazo, 2001).

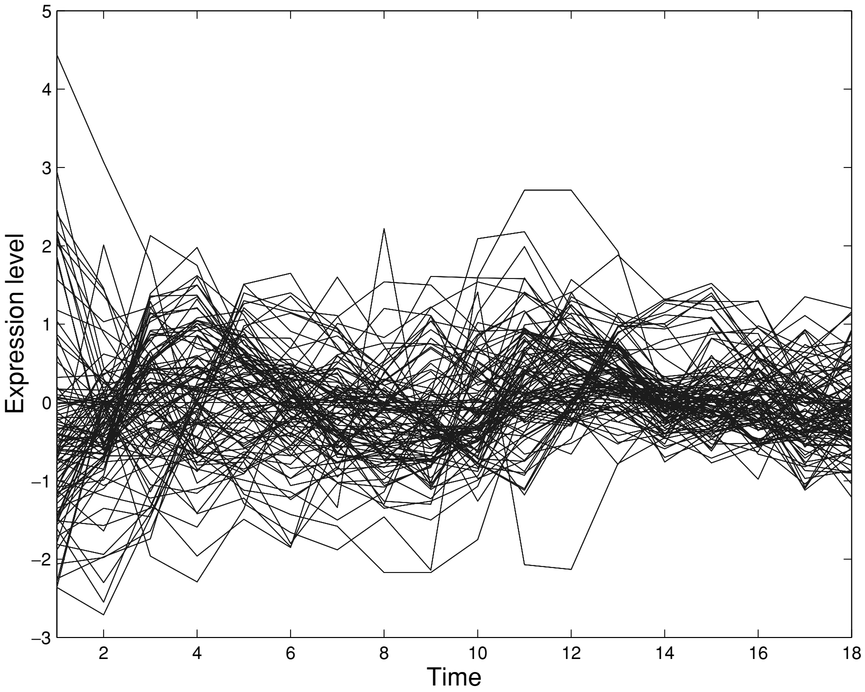

Figure 1. The change of expression levels for 104 known yeast genes in the alpha factor arrested data set. For mutually co-regulated genes, the expression levels of one gene can activate or deactivate those of another gene through biological regulation mechanisms. At each point of the time series, various characteristic features of co-regulation, like the difference of expression levels and the change of slopes before the current measurement, can be measured.

Although clustering is the most widely used method for the analysis of gene expression patterns, there are several drawbacks in this approach. In the case of hierarchical clustering (Alon et al., 1999; Eisen et al., 1998), for example, it lacks robustness and the solutions may not be unique and dependent on the order of data. Also clustering has the weakness when dealing with the data containing a large amount of noise (Tamayo et al., 1999). When analyzing microarray data for the cell cycle, clustering has an advantage in grouping genes for overall cell cycle regulation but not in the analysis of the relationship between cell cycle regulated genes. In this case, profiling of temporal expressions might be an efficient approach to identifying correlated genes during the cell cycle.

Recently, several methods for profiling the temporal gene expressions have been proposed. One is the time warping algorithm (Aach & Church, 2001). Here, the two time series are compared with respect to the similarity of their patterns regardless of expression timings. The motivation of the time warping algorithm is based on the fact that the same gene may have a different reaction time under different experimental conditions. But in the cases of cell cycle regulation, most genes are under the same conditions and the timing of their expressions is also important besides their patterns. In contrast to the time warping algorithm (Ramoni, Sebastiani, & Kohane, 2002), has recently proposed a method that directly

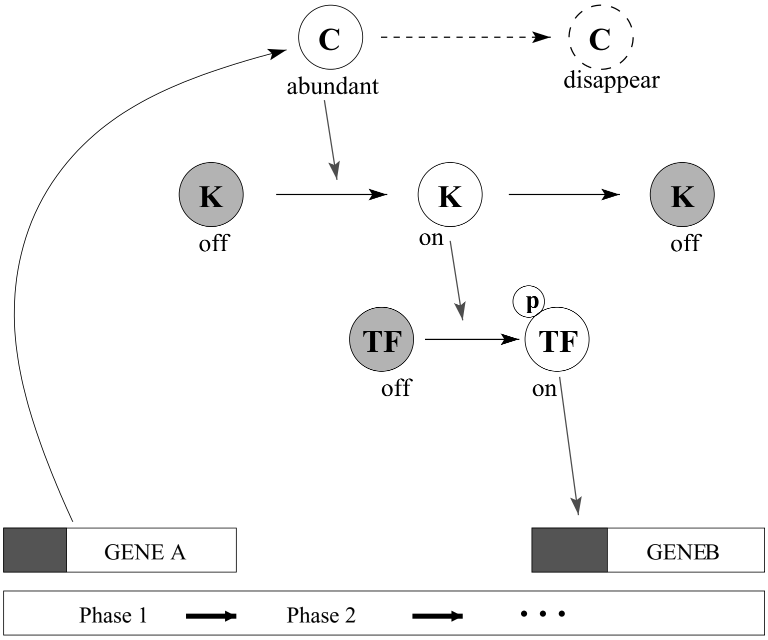

Figure 2. The roles of cyclin (C), kinase (K) and transcription factor (TF) in the cell cycle regulation between gene A and gene B. As the cell cycle regulating protein cyclin (C) from gene A is being abundant, the kinase (K) induces the phosphorylation of the transcription factor (TF) to express the target gene B. As a result of these mechanisms, gene A regulates the expression levels of gene B.

takes into account the dynamics of the gene expression time series. This method represents gene expression dynamics as autoregressive models and uses an agglomerative procedure to search for the most probable clusters. They also provide a principled way to identify the number of distinct clusters. Another related work is the bi-clustering method (Cheng & Church, 2000). Here, the common expression patterns in a group of genes are found by clustering both genes and conditions. The bi-clustering method can provide automatic discovery of similarity for both conditions and genes and allows overlapped grouping of genes to provide a better interpretation for genes with multiple functions or factors based on the similarity scores of a subset of genes.

In this paper, we propose a more flexible method based on the latent lattice model. In this model, each observed data item is represented by a point in the latent space and the temporal profile of a sequence of data points can be obtained by plotting them through the experimental time intervals. But the cell cycle regulation is a complex process where two or more genes are involved to activate or inactivate each other. In order to find out the multiple correlated genes out of given gene groups, the combined information about mutual expressions should be used rather than separate expression levels. In our approach, we select all the possible combinations of genes and appropriate features to find out meaningful pairs of correlated genes. The gene pairs with similar co-regulations are visualized in specific patterns on the 2D lattice. For example, figure 3 shows the time series of 5 genes (3a) and the

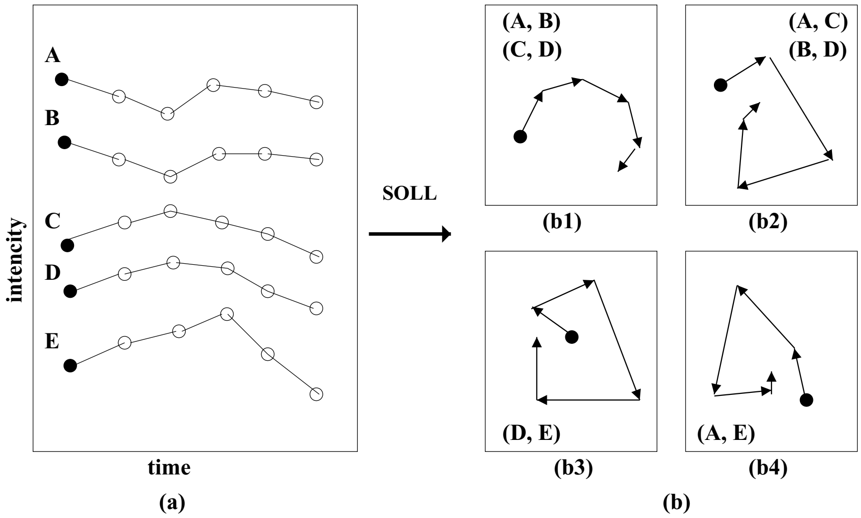

Figure 3. Schematic of generating co-regulation trajectory patterns in SOLL. (a) Expression sequences of five different genes. (b) Various trajectory patterns on SOLL for pairs of genes. The gene pairs with similar co-regulation are visualized in specific patterns. For example, gene pairs A-B and C-D are expressed differently in the time series domain (a) but have the same co-regulation pattern on the lattice as shown in (b1).

pairwise co-regulation patterns visualized on the 2D lattice. Here gene pairs A-B and C-D have expression levels activated and inhibited together and this can be represented by the common co-regulation pattern in (b1) on the 2D lattice. On the other hand, gene pairs A-C and B-D have a similar co-regulation pattern which can be visualized by another common trajectory on the lattice.

The trajectory patterns on the lattice can be used in several ways. Generally, similar trajectory patterns on the lattice imply the co-regulation patterns of the two genes are similar. By generating and observing various trajectory shapes for different combinations of genes, we can find the groups of gene pairs having similar correlation patterns. On the other hand, differently co-regulated genes show different patterns, helping to identify some unexpected characteristics of co-regulations. Loops in the trajectory mean the co-regulation patterns repeat. Thus, the loopy trajectories can be used to detect the cycles in gene expressions. The size of the trajectory indicates the magnitude of the phase shift in gene interactions. Thus, the interpretation of trajectory patterns should take into account both size and shape of the patterns.

SOLL is a visualization tool based on the latent-variable model. Figure 4 illustrates the characteristics of SOLL in learning as compared to SOM and GTM. SOM learns the data structure directly using a set of prototype vectors without assuming a specific model. The topographic nature of data is learned by the prototype vectors retaining similar topographic relations with the node arrays. The GTM model assumes a latent-variable model and selects each node with some probability for each input to estimate the model. The latent variables are set to be a set of fixed grid points regardless of the distribution of data. In SOLL, the topographic characteristics of input data are learned by using the prototype vectors and the

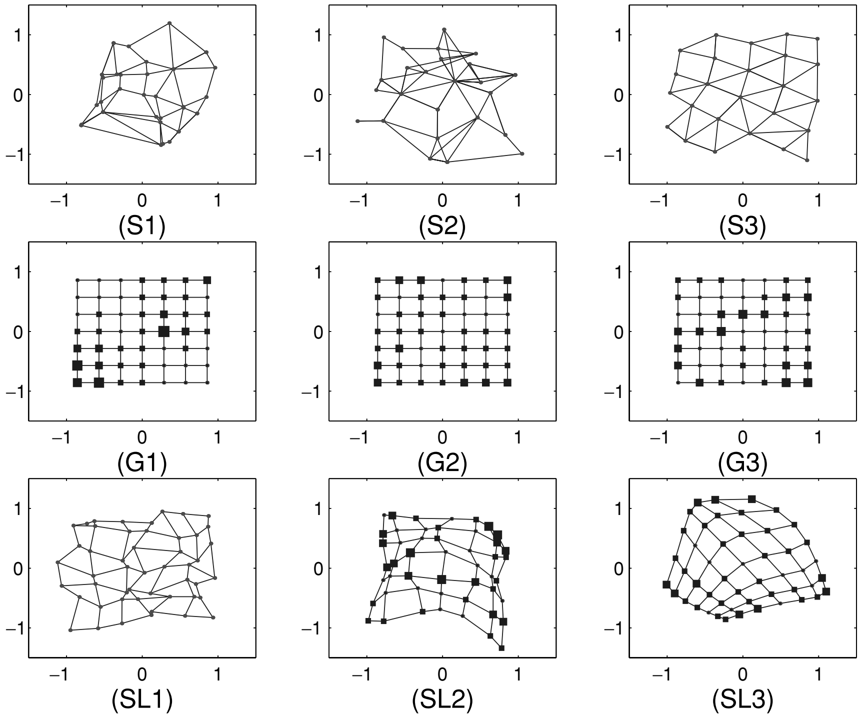

Figure 4. Comparison of the learning behaviors in SOM (S1–S3), GTM (G1–G3) and SOLL (SL1–SL3). In SOM, the structure of the input data is reflected by the prototype vectors topographically in a deterministic way. In GTM, grid points are fixed and the corresponding winning probabilities (marked by squares) are learned. In SOLL, the positions are elastic and the posterior distribution of the grid points are learned. The winning probabilities are learned for the input data with respect to this grid.

corresponding nodes are selected probabilistically. Here the grid can take a more flexible form and the posterior density of the latent point is used in the estimation of the hidden variables. The self-organizing latent lattice model will be discussed in more detail in the next section.

3. Self-organizing latent lattice models

SOLL assumes a probabilistic latent-variable model to generate the input data from an arbitrary node of latent lattice. The latent-variable model assumes a set of hidden variables that are responsible for generating high dimensional complex data. Here, complex measurements of co-regulations between two genes can be expressed in a more compact form by latent features and can provide useful insights into understanding the data. Especially when the latent features are imbedded in a 2-dimensional latent space, the structure of the given co-regulation expressions can be explored visually.

Learning in SOLL consists of two steps (figure 5): E-step for the visualization of high dimensional features with time dependent trajectory and M-step for the self-supervised

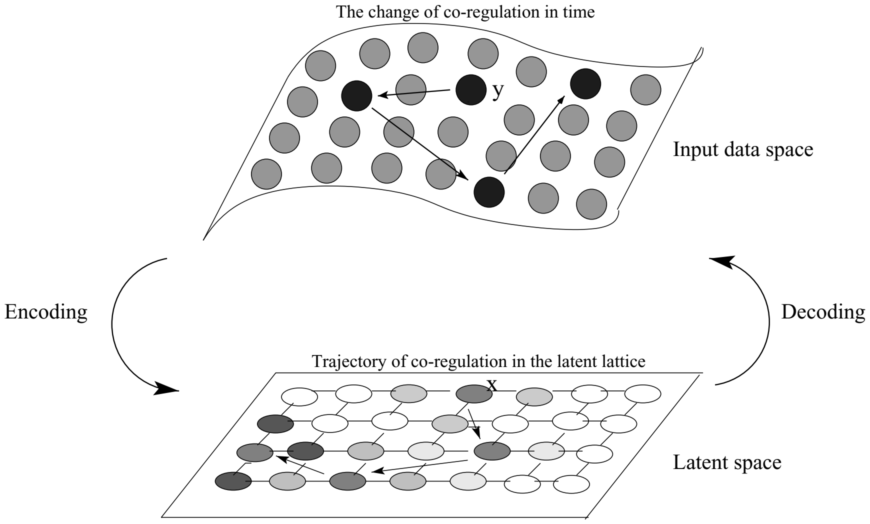
The change of co-regulation in time
y
Input data space
Encoding
Decoding
Trajectory of co-regulation in the latent lattice
x
Latent space

Figure 5. The learning process of a self-organizing latent lattice model consists of encoding and decoding processes. A trajectory in the high dimensional data space through time can be mapped into the latent lattice embedded in the latent space and visualized as a pattern.

learning of the latent-variable model which generates the co-regulation expression features from a given latent lattice. Table 1 summarizes the whole process for learning SOLL. The basis function set and the size of lattice are fixed at the outset. In the E-step, the lattice point corresponding to the closest prototype vector (a mapping of the lattice point to the data space through the latent-variable model) for each input data is selected as the winner and the prototype vectors are updated (Eq. (1) in Section 3.2). Based on these prototype vectors, the posterior density of each lattice point is updated. In the M-step, the model parameters are updated using posterior density of the lattice and projection of the data by error minimization (Eq. (11) in Section 3.3).

Note the similarity between SOLL and SOM in the first part of E-step and the similarity between SOLL and GTM in the second part of E-step and in the M-step. In effect, SOLL combines the self-organizing property of SOM by winner-take-all and the generative property of GTM by latent-variable models.

3.1. Co-regulation measure and feature extraction

In the gene co-regulation, one gene is activating or inhibiting another gene with or without a certain amount of time delay. In this paper, gene co-regulation is defined as a temporal relationship between two time series of gene expression levels. The difference of responses before and after the present time t (i.e., differences of slope) as well as the differences between the expression levels are among the most natural candidates in measuring such biological characteristics of inter-activities between genes. We define the difference of expression levels at present time t as a measure of co-regulation when there is no time delay between the two genes. The change of slope at time $t-1$, t and $t+1$ is used for measuring

Table 1. The EM algorithm for learning SOLL.

```
(Initialization)
• Set the penalty parameter $\lambda$.
• Set the learning rate $\eta$.
• Set the $M$ basis functions.
• Initialize the parameter matrix $\boldsymbol{W}$ randomly or by PCA.
• Initialize the error variance $\beta\boldsymbol{I}$.
• Initialize the latent lattice $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_K$.
• Compute the matrix of basis set $\boldsymbol{\Phi}$.
(E-Step: Lattice update and projection)
• Set the weight $z_k$ $(k = 1, \ldots, K)$ from $\boldsymbol{W}$ and $\boldsymbol{\Phi}$.
• For each input $\boldsymbol{y}_j$ $(j = 1, \ldots, N)$, update the weight:
      $z_{k+1} \leftarrow z_k + \eta\mathcal{N}(k, j^*)(\boldsymbol{y}_j - z_k)$
• Update latent lattice:
      $p(\boldsymbol{x}_k) \leftarrow p(\boldsymbol{x} \mid z_k)$
      $\boldsymbol{x}_k \leftarrow \sum_{j=1}^{K} \boldsymbol{x}_j p(\boldsymbol{x}_j \mid z_k)$
(M-Step: Parameter update)
• Compute $\boldsymbol{W} \leftarrow (\Phi^T \boldsymbol{R}\boldsymbol{R}^T \Phi + \lambda\boldsymbol{I})^{-1} \Phi^T \boldsymbol{R}\boldsymbol{Y}$ from $\Phi$ for a fixed $\lambda$.
• Compute $\Delta = (\Delta_{kn})$ with $\Delta_{kn} \leftarrow \|\boldsymbol{y}_n - \phi_k \mathbf{W}\|^2$
• Update $\beta$ from $\Delta$.
(Repeat)
Repeat E-step and M-step until the convergence criterion is fulfilled.
```

how fast or slow the level of one gene is activating that of another. Figure 6 shows how the four features of co-regulation $\boldsymbol{y}(t) = (y_1, y_2, y_3, y_4)$ in data space are mapped into the latent space at experimental time t. When a_t and b_t indicate the expression levels of gene A and gene B at time t respectively, we define each component as

$$
\begin{aligned}
y_1 &= |a_t - b_t|, \\
y_2 &= |(a_t - a_{t-1}) - (b_t - b_{t-1})|, \\
y_3 &= |(a_{t+1} - a_t) - (b_{t+1} - b_t)|, \\
y_4 &= |(a_{t-1} - a_{t-2}) - (b_{t-1} - b_{t-2})|.
\end{aligned}
$$

Here, y_1 measures the difference of present expression levels, y_2, y_3 and y_4 measure the differences of slopes at present, past and future respectively. Note that our choice of four features is robust against fluctuations since important biological features like activation and inhibition occurring in the gene pairs through time is measured completely by slope changes (rather than the absolute values) through the time window.

After encoding into the latent space, the trajectories of correlated genes in time can be expressed and visualized. Thus, multi-dimensional correlation features between two genes

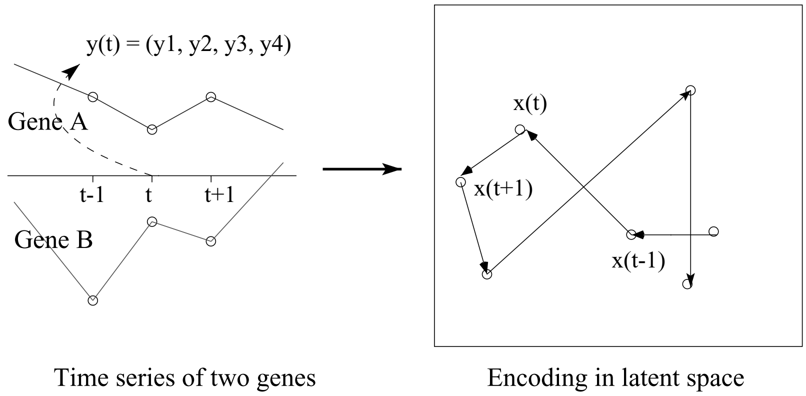

Figure 6. Encoding of co-regulation data into the latent space. At each experimental time, features of multiple correlated genes are encoded in the latent lattice making a trajectory pattern.

are reduced to hypothetical latent features and the characteristics of correlation in time can be visualized as a specific pattern in the latent space.

3.2. Lattice update

In E-step of SOLL learning, each node of the latent lattice is connected to the node of input data. For each input data $\boldsymbol{y}_j$, a winning node $\boldsymbol{x}_{j^*}$ in the latent lattice is the one which generates the closest input data z_{j^*} to $\boldsymbol{y}_j$ based on the latent-variable model. Thus, the weight z_k connecting the node $\boldsymbol{x}_k$ to the input data is updated according to the following rule:

$$z_{k+1} = z_k + \eta\mathcal{N}(k, j^*)(\boldsymbol{y}_j - z_k), \tag{1}$$

where $\mathcal{N}(k, j^*)$ is a decreasing neighborhood function of $\|\boldsymbol{x}_k - \boldsymbol{x}_{j^*}\|$ and η is a learning rate. In the updating process of the weights in (1), the updated prototypes are compared with the mapping of the grid points by the generative property of the latent-variable model. Unlike SOM, the winner node is not selected deterministically but with some winning probability. This process corresponds to the projection of prototype vectors in the latent-variable space. In GTM, each data point is projected to a noninformative fixed set of grid points (E-step) and used in estimating the model parameter (M-step). Here, the prototype vectors are reflecting topographical properties of data and projected to the latent space using the latent-variable model. The projection of data is based on these posterior distribution of the latent variables. The remaining process follows similar to GTM. A data point $\boldsymbol{y}_i$ can also be encoded probabilistically by computing Bayesian posterior probability of each point in the latent lattice by

$$p(\boldsymbol{x} \mid \boldsymbol{y}_i) = \frac{p(\boldsymbol{y}_i \mid \boldsymbol{x})p(\boldsymbol{x})}{\int p(\boldsymbol{y}_i \mid \boldsymbol{x})p(\boldsymbol{x})\, d\boldsymbol{x}}. \tag{2}$$

The encoded values can be estimated by taking the expectation of the corresponding conditional probability for each input data $\boldsymbol{y}_i$ by

$$\hat{\boldsymbol{x}} = \int \boldsymbol{x} p(\boldsymbol{x} \mid \boldsymbol{y}_i)\, d\boldsymbol{x} \tag{3}$$

Though the probabilistic encoding gives a clear formulation between the latent features and the features of co-regulation, there are some drawbacks. The calculation of posterior probability is computationally infeasible except under simple forms of priors over the latent variables in (2) or (3) (Bishop, Svensen, & Williams, 1998). For computational and algorithmic reasons, the latent space is further simplified by putting some form of lattice as

$$p(\boldsymbol{x}) = \frac{1}{K} \sum_{k=1}^{M} \delta(\boldsymbol{x} - \boldsymbol{x}_k), \tag{4}$$

where K is the number of lattice points and δ is the delta function. The set of node indices in SOM or the discrete latent points in (4) are examples of lattices applied to the latent space. In assuming (4) for the prior distribution of the latent variable, the complicated integration in (2) or (3) is replaced by a simple summation.

3.3. Parameter update

Let D represent the number of features measuring the co-regulation for a given pair of genes and L be the number of corresponding latent features. Then a D-dimensional vector $\boldsymbol{z}$ where each component represents a feature of co-regulation is a realization of a latent variable $\boldsymbol{x}$ under a mapping $\boldsymbol{\Gamma}$ defined on the latent space $\Re^L$.

$$\boldsymbol{\Gamma} : \boldsymbol{x} \in \Re^L \longmapsto \boldsymbol{z} \in \Re^D. \tag{5}$$

When $\boldsymbol{\Gamma}$ is a simple linear transformation and $\boldsymbol{x}$ has an isotropic Gaussian distribution, the mapping (5) becomes a factor analysis model after convolving $\boldsymbol{z}$ with an isotropic Gaussian distribution. If the features of co-regulation are related nonlinearly with the hypothetical latent features by a set of M fixed basis functions and K latent points, the corresponding model becomes

$$\boldsymbol{z} = \boldsymbol{\phi}(\boldsymbol{x})\mathbf{W} + \boldsymbol{e}, \tag{6}$$

where $\boldsymbol{\phi}(\boldsymbol{x})$ is a mapping of latent value $\boldsymbol{x}$ with respect to a set of M fixed basis functions, $\boldsymbol{W}$ is an $M \times D$ matrix of coefficients and $\boldsymbol{e}$ is a noise term for convolving $\boldsymbol{z}$. Examples of basis functions include 1 for a bias term, $\{x_1, x_1^2, x_1 x_2, \ldots\}$ for a polynomial basis set and $\{\exp(-\frac{(x_i-\mu_i)^2}{\sigma_i})\}_{i=1}^M$ for an exponential basis set with corresponding terms of means and variances. For a Gaussian noise, the variance of $\boldsymbol{e}$ is set to be $\beta \boldsymbol{I}$ and used in the computation of the responsibilities of grid points (Eqs. (9) and (10)).

The latent-variable model (6) can be learned by the least squares estimation method (see e.g. Anderson, 1984). Suppose $\boldsymbol{Z}$ is a $K \times D$ matrix with each row representing a data vector generated from each of K latent points, $\mathbf{\Phi}$ is a $K \times M$ basis matrix of M given basis functions including a bias term taking values at each latent point $\boldsymbol{x}$. Then the form of the matrix of basis function becomes

$$\mathbf{\Phi} = (\phi_{km}) = (\phi_m(\boldsymbol{x}_k)), \tag{7}$$

where $(\phi_m(\boldsymbol{x}_k))$ represents the value of the m-th basis function evaluated at the latent point $\boldsymbol{x}_k$. In a matrix form, the latent-variable model (6) becomes:

$$\boldsymbol{Z} = \mathbf{\Phi}(\boldsymbol{x})\boldsymbol{W} + \boldsymbol{E}. \tag{8}$$

$\boldsymbol{Z}$ is updated by estimating the corresponding model parameter $\boldsymbol{W}$ in (8). Note that in SOM, each row vector in $\boldsymbol{Z}$ corresponds to a weight vector and is updated directly from a winner-take-all neural network.

Suppose $\boldsymbol{Y}$ is an $N \times D$ matrix of the data set, $\boldsymbol{R}$ is a $K \times N$ matrix that assigns each data vector $\boldsymbol{y}$ to a vector z in (6). Then the computation of the matrix $\boldsymbol{R}$ depends on the different learning methods. In GTM, each element r_{ij} of the matrix $\boldsymbol{R}$ is formulated as *responsibility* of a grid point $\boldsymbol{x}_i$ for the input $\boldsymbol{y}_j$ as:

$$r_{ij} = p(\boldsymbol{x}_i \mid \boldsymbol{y}_j) = \frac{p(\boldsymbol{y}_j \mid \boldsymbol{x}_i)p(\boldsymbol{x}_i)}{\int p(\boldsymbol{y}_j \mid \boldsymbol{x}_i)p(\boldsymbol{x}_i)\,d\boldsymbol{x}}. \tag{9}$$

In SOLL, the density of the grid point $\boldsymbol{x}_i$ is conditional on the input $\boldsymbol{y}_j$ and the prototype vectors $\boldsymbol{Z} = \{z_1, \ldots, z_K\}$. The *responsibility* of the grid point $\boldsymbol{x_i}$ with respect to the input $\boldsymbol{y}_j$ and the prototype $\boldsymbol{Z}$ becomes:

$$r_{ij} = p(\boldsymbol{x}_i \mid \boldsymbol{y}_j, \boldsymbol{Z}) = \frac{p(\boldsymbol{y}_j, \boldsymbol{Z} \mid \boldsymbol{x}_i)p(\boldsymbol{x}_i)}{p(\boldsymbol{y}_j, \boldsymbol{Z})} = \frac{p(\boldsymbol{y}_j \mid \boldsymbol{x}_i)p(\boldsymbol{Z} \mid \boldsymbol{x}_i)p(\boldsymbol{x}_i)}{\int p(\boldsymbol{y}_j \mid \boldsymbol{x})p(\boldsymbol{Z} \mid \boldsymbol{x})p(\boldsymbol{x})\,d\boldsymbol{x}}. \tag{10}$$

The selection of the winning node is based on the estimation of the probability density $p(\boldsymbol{y}_j \mid \boldsymbol{x}_i)$ obtained from the latent model (6) and the posterior distribution $p(\boldsymbol{Z} \mid \boldsymbol{x}_i)$ of the grid points. When no prior knowledge is assumed on the distribution of the grid set, the winning node in (1) is selected based on the distance measure between the reference vector z and the input data $\boldsymbol{y}$. The flexibility of grid points becomes important for high dimensional latent space. In GTM the grid points are a set of rigid points and are not allowed to move freely. As the dimension of the latent space are growing higher, the size of grid points increases exponentially causing a heavy computational burden. In SOLL, these problems are avoided by the flexibility of grid points.

The parameter matrix $\boldsymbol{W}$ estimated by the least squared estimation method becomes:

$$[\mathbf{\Phi}^{\boldsymbol{t}}(\boldsymbol{x})\boldsymbol{R}\boldsymbol{R}^{\boldsymbol{t}}\mathbf{\Phi}(\boldsymbol{x})]\hat{\boldsymbol{W}} = \mathbf{\Phi}^{\boldsymbol{t}}(\boldsymbol{x})\boldsymbol{R}\boldsymbol{Y}. \tag{11}$$

When $\mathbf{\Phi}^{\boldsymbol{t}}\boldsymbol{R}\boldsymbol{R}^{\boldsymbol{t}}\mathbf{\Phi}$ is not a full rank matrix, an additional term $\lambda\mathbf{I}$ with a penalty parameter λ needs to be added to avoid the singularity and the addition corresponds to supposing an isotropic Gaussian prior over $\boldsymbol{W}$ (Bishop, 1999):

$$\boldsymbol{\Phi}^t \boldsymbol{R}\boldsymbol{R}^t \boldsymbol{\Phi} + \lambda \mathbf{I}. \tag{12}$$

Some remarks on the computational complexity of SOLL is in order. With respect to the number of parameters, SOLL requires a $K \times M$ matrix for M basis functions, an $M \times D$ matrix for $\boldsymbol{W}$, and a $K \times N$ matrix for $\boldsymbol{R}$. When $K \approx M$, the overall computational complexity for SOLL is $O(KND) + O(K^3)$ which is the time $O(KND)$ for SOM plus the time $O(K^3)$ for GTM. The dimensionality of the trajectories is proportional to the number K of grid points. In choosing K, there should be some tradeoff between the resolution of trajectories and the computing time. Large K is preferred if fine-grained trajectory patterns are required while small K is better for getting general patterns (Zhang, Ohm, & Muehlenbein, 1997).

4. Temporal gene expression profiling by SOLL

4.1. Application of SOLL

The expressions of genes are regulated through complicated biological pathways during the cell cycle and the profiling of their time trends can provide important guidelines and information to the analysis of the cell cycle co-regulation process. The temporal profiling of gene expression consists of three parts: generating features from the expression levels of gene pairs, modelling temporal profiles based on these features, and finding patterns from these temporal profiles. In feature generation, the characteristic features are defined and generated from the combinations of possible pairs of gene expression levels. The SOLL algorithm is used in the modelling of temporal profiles. In pattern finding, we trace temporal expression profiles of genes in the latent lattice, cluster them according to the temporal patterns, and compare specific patterns and trajectories of different clusters. After initializing lattice points as weights of the neural network by grid points and model parameter $\boldsymbol{W}$ by principal component analysis (PCA), the training of SOLL is performed in two steps:

(E-step) Given a lattice, fit the corresponding prototype vectors to the data topographically. Compute the responsibility of each input.

(M-step) Given encoded input values from E-step, estimate parameters of the latent-variable model by proper optimization criteria.

Typical parameter values used for the experiments are as follows: penalty parameter $\lambda = 0.1$, learning rate $\eta = 0.1$, the number of basis functions $M = 100$, $\boldsymbol{W}$ was initialized by PCA, grid size $K = 100$.

4.2. Demonstrations on artificial data

Knowing the order of co-regulation timing as well as specific patterns is important in studying the co-regulated genes (Futcher, 2000). We extracted 4 features at each time point as the input data and extracted two latent features using the latent model (figure 6). The resulting latent features are used as temporal patterns by making them in a single vector form (see Section 3.1 for more details).

To test the SOLL algorithm, a set of simulated data is created. Since the genomic expression is activated periodically during the cell cycle, the sine function is used to create the data. In figure 7, the collection of simulated data is plotted where four different sine curves with different peak times are used with some noises in the amplitudes and peak times. There are 6 different co-regulation patterns out of 4 distinct sine curves. We applied the SOLL method to identify these specific co-regulation patterns and compared them with other temporal pattern profiling methods. As a baseline method, we used a single time series made by joining the two current time series to express the temporal pattern of the two genes. We also compare the performances of SOLL with those of SOM. Here, the data set is mapped into a 10 by 10 grid of SOM and the resulting patterns in the node set (the patterns of SOM in figure 8) are taken as temporal patterns. In each of the SOLL and SOM methods, the two-dimensional temporal pattern is entered as a single time series whose components consist of the components of the mapping point in the latent space. After that, a set of temporal time series generated from each method is clustered by a k-means algorithm where the total number of different patterns is set to be 6 (figure 7). The k-means algorithm (Duda & Hart, 1973) is a clustering method using the Euclidean distance as a dissimilarity measure:

$$d(\boldsymbol{p}_i, \boldsymbol{p}_j) = \sum_{t=1}^{T}(p_{it} - p_{jt})^2 = \|\boldsymbol{p}_i - \boldsymbol{p}_j\|. \tag{13}$$

In Eq. (13), each component of $\boldsymbol{p}_i$ is the element of mapping point in the latent space where $T = 36$ for the baseline method and $T = 30$ for other methods by ignoring the first and last 3 intervals as noise.

In figure 7, pattern 1 represents a pair of sine curves which is almost identical through time, while patterns 2, 3, 4, 5 represent shifted ones with different peak times respectively and pattern 6 represents a pair of sine curves opposite each other. In gene co-regulation analysis, pattern 1 is the characteristic of genes in the same cluster, while patterns 2, 3, 4, 5, 6 are characteristic of co-regulation patterns where the expression level of one gene is regulating or is being regulated by that of another gene through time. When the two genes show similar expression levels, they might be in the same group of genes which are regulated under the influence of other gene. In the study of co-regulation, this group of genes are not of much interest. In the profiling method by SOLL, they are represented by concentrated points (SL1 in figure 8).

To evaluate the profiling method we calculate, for each and every trajectory in cluster c, the Euclidean distance to the six meaningful patterns and identify the majority pattern p in the cluster. We then compute the identification rate r_c^p by

$$r_c^p = \frac{N_c^p}{N_c}, \tag{14}$$

where N_c is the total number of trajectories in cluster c and N_c^p is the number of trajectories of the majority pattern p in the cluster c. In Table 2, the identification rate of the baseline, SOM, GTM and SOLL methods are compared. Patterns of co-regulation (patterns 2 to 6)

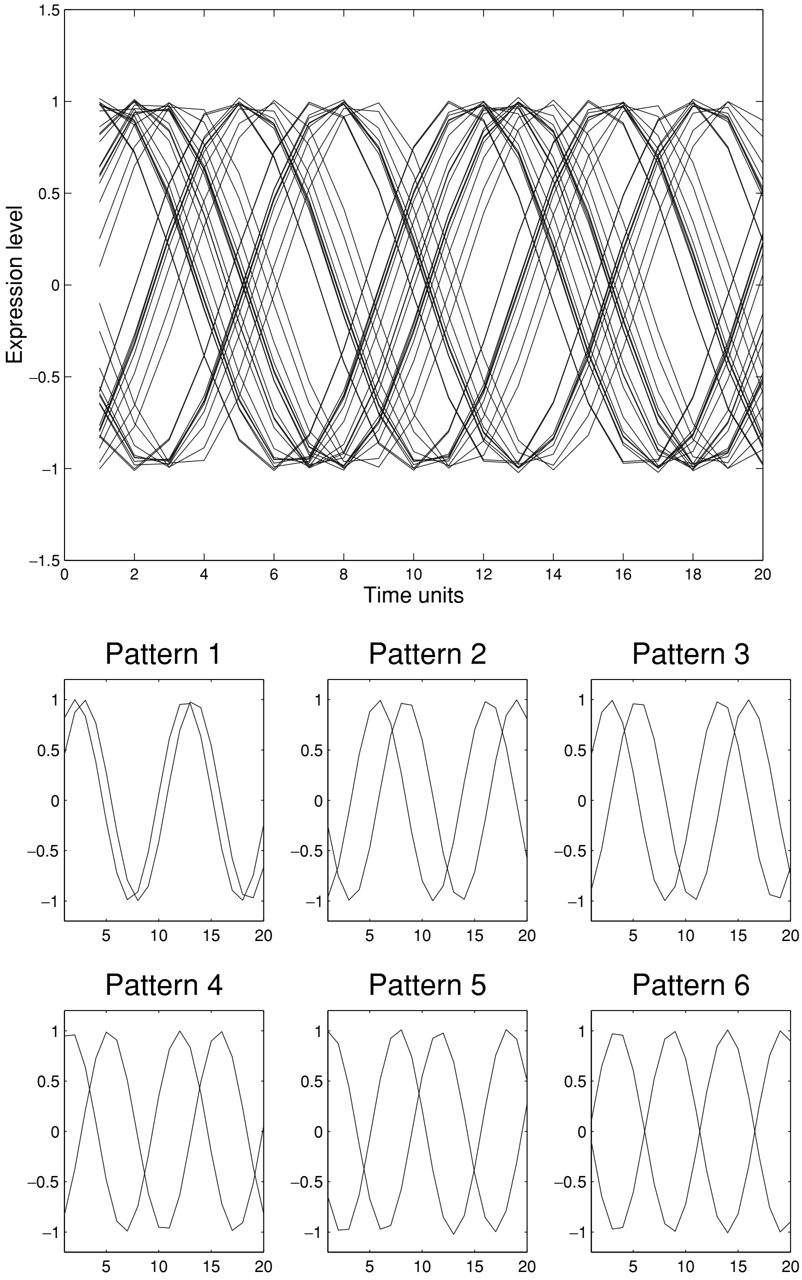

Figure 7. Above: The temporal patterns of artificial data expressed as time series. Below: The 6 different expression patterns of artificial data.

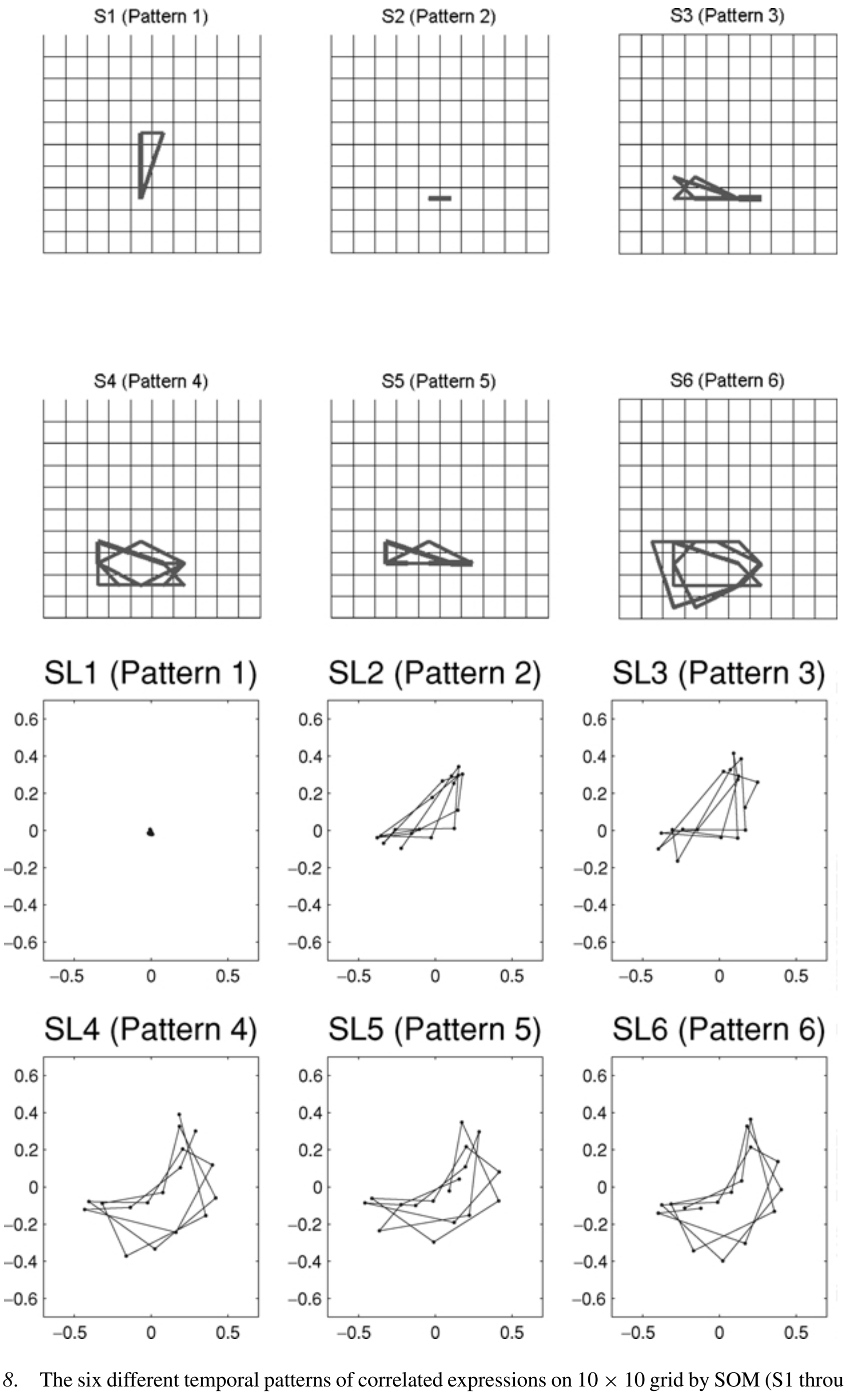

Figure 8. The six different temporal patterns of correlated expressions on 10 × 10 grid by SOM (S1 through S6) and on the latent space by SOLL (SL1 through SL6).

Table 2. Comparison of SOLL and other temporal profiling methods in identifying 6 possible patterns in the artificial data set. Each entry in the table represents the identification rate $r_c^p = N_c^p / N_c$ of cluster c for the majority pattern p in c (see text for definition).

Pattern	Baseline	SOM	GTM	SOLL
Pattern 1	0.600	0.859	0.974	0.983
Pattern 2	0.880	0.717	0.829	0.846
Pattern 3	0.195	0.594	0.634	0.736
Pattern 4	0.175	0.509	0.683	0.930
Pattern 5	0.890	0.780	0.638	0.838
Pattern 6	0.530	0.455	0.804	0.842
Average	0.545	0.652	0.760	0.862

are identified with high rate in SOLL and GTM methods while patterns 3 and 4 are not identified effectively in the baseline method.

5. Empirical results

5.1. Experimental setup

We have tested the SOLL approach on a DNA microarray data for possible gene combinations from a subset of known genes of yeast. The data set we used is based on the alpha factor arrested synchronization set among the three data sets (Spellman et al., 1998) as described in Section 2.

The data set of alpha factor arrested expressions have 18 measurements for every 7 minute interval. In building the co-regulation features, we do not use the first 3 measurements due to the noises in the beginning of the microarray data. In the baseline method the raw data is used without taking differences of expressions and $T = 36$ of the whole time interval is used. Since different cluster sizes give similar results, the various temporal patterns are clustered by k-means clustering procedure with $k = 100$ using Euclidean dissimilarity measure (13). For the encoding of the co-regulation features onto the latent lattice, we assume a 10 by 10 grid and learn the winning grid points by computing the posterior probabilities for each input data.

5.2. Temporal profiling of multiple correlated genes

We have tested 5356 different combinations of genes out of 104 genes of yeast. Analysis of all possible pairs of the yeast genes is possible, but we did not do that for two reasons. The first is because of the enormous computation time for performing this study. We still lack biological data for the confirmation of the results. For the analysis of all yeast genes (more than 6000), the number of possible pairs of genes becomes too large for matrix computation. A possible remedy for this would be clustering the genes and using prototype genes in each cluster for the generation of pairs. The second, more important, reason is that, at the current

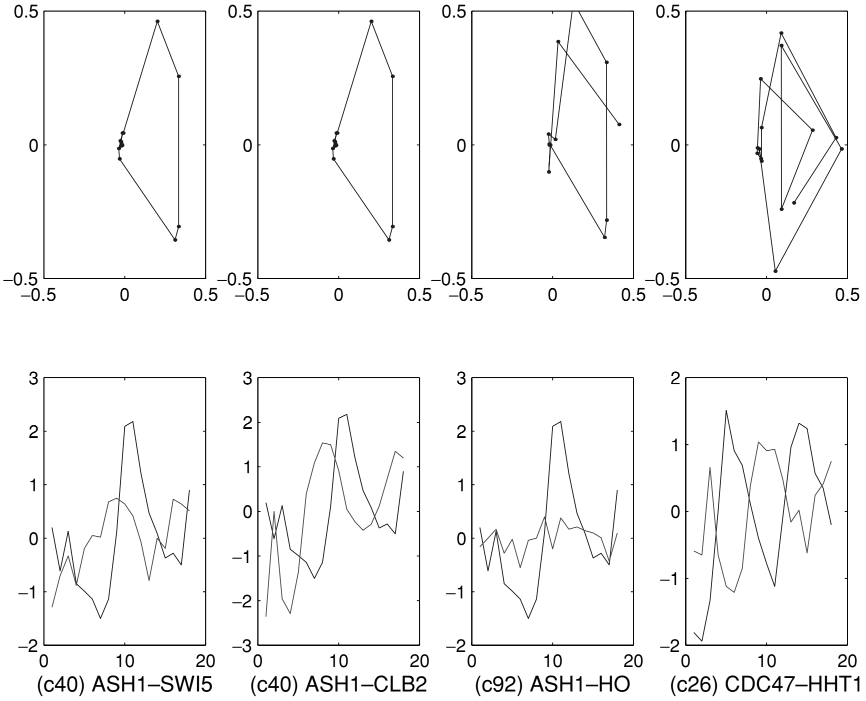

Figure 9. Above: The various trajectories obtained by SOLL for the correlated genes. Below: The corresponding time series for each pattern. Both cluster 40 and cluster 92 contain ASH1 gene but show different relation between two correlated genes.

time of biology, such a study does not help the biological research very much. The use of all the genes without having a large amount of data would only increase the noise of data.

At each experimental time, 4 co-regulation features are expressed in a 2 dimensional latent feature space and their trajectories are visualized. A total of 100 different patterns are obtained by clustering the corresponding set of time series. Figure 9 shows part of the trajectories obtained by SOLL by plotting the mean encoding values. Note that patterns are distinct from each other and some of the clusters contain a set of biologically important genes with respect to the co-regulation mechanism in the cell cycle.

Especially, cluster 18 contains most of the correlated gene pairs which are known to regulate each other in the corresponding biological mechanisms (Table 3). In the cell cycle process, Cln3 is the G1 specific cyclin whose expression is highly peaked at about G1/S transition (Koch, 1994). It is well known to positively regulate the expressions of G1 specific genes that contain MCB or SCB elements in their promoters (figure 10). As can be seen in Table 3, analysis of the set of CLN3-containing pairs in cluster18 shows that most of SCB and MCB genes are included in these pairs. These results are in accordance with the expectation from biological knowledge. But other genes in M/G1, S, S/G2 and G2/M phase also contain the CLN3 gene in cluster 18. We used the Cln3 values to examine the direct effects of induced Cln3 from spellman's experiments (Spellman et al., 1998). The genes that

Table 3. Average cln3 values of gene pairs of pattern 18 by SOLL. The CLN3-containing pairs have higher values than the rest of genes in M/G1, S, S/G2 and G2/m phase. This means that genes other than MCB or SCB genes can be also activated by CLN3 with certain possibility.

	SCB	MCB	S	S/G2	G2/M	M/G1
CLN3-containing pairs	0.55	0.93	1.28	0.23	−0.36	−0.35
Other pairs	1.66	0.64	N/A	0.00	−1.50	−0.38

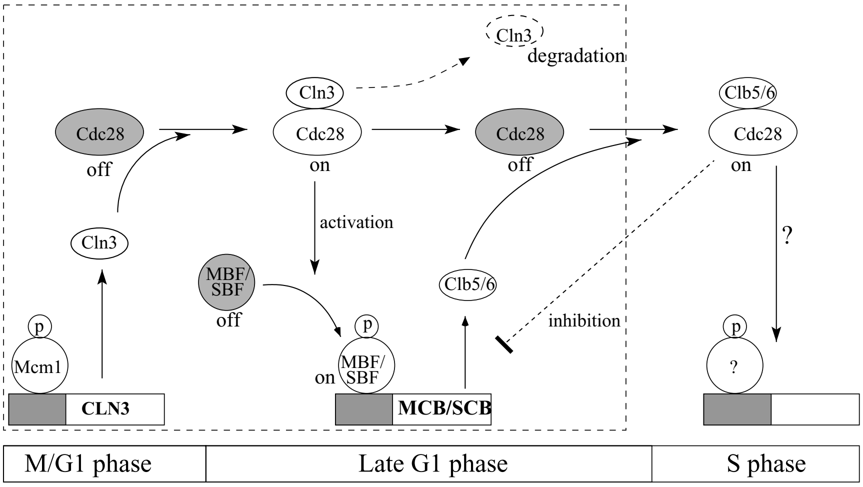

Figure 10. The roles of CLN3, a late G1 specific cyclin , in the yeast cell cycle regulated transcription. Transcription of CLN3 is started at M/G1 phase and then Cln3-associated kinase (Cdc28) activates late G1 specific transcription factors such as SBF (SCB binding factor) and MBF (MCB binding factor) by phosphorylation. Activated SBF and MBF initiate the expression of CLB5 and CLB6 as well as late G1 specific genes, leading to budding and S phase entry. Cln3 proteolysis at the begining of S phase and expression of CLB genes (CLB5 and CLB6) change the activity of cdc28. Clb5,6/Cdc28 activate transcription of S and G2-specific genes and represses SBF-mediated transcription.

have greater Cln3 values are more likely to be activated by CLN3 than others. According to the Table 3, average Cln3 values of CLN3-containing pairs are greater than the rest of genes in M/G1, S, S/G2 and G2/m phase. This means that genes other than MCB or SCB genes are also activated by CLN3 with certain possibility. As a result of analyzing the pair of genes for pattern 18, it is confirmed that the activation of MCB or SCB genes by CLN3 and find novel CLN3-corresponding genes.

Cluster 26 contains both CDC47 and all S phase histone genes except HHF2. During S-phase when the DNA is replicated, replicated DNA is packed into nucleosome through binding with histone proteins (Stein, Stein, & Lian, 1992). The prior elevation of histone protein is necessary to support DNA replication and histone-dependent formation of nucleosome (Osley, 1991). CDC47 is a minichromosome maintenance protein (MCM) which is essential for DNA replication and interacts with histone proteins (Ishimi et al., 1996). Cluster 26 shows the relationship between the prior expression of histone proteins which is

Table 4. Identification rate of CLN3-containing pairs in each method. SOLL contains the largest number of CLN3-containing pairs (83.0%) while the baseline method does not identify them effectively (39.6%). SOM and GTM contain moderate portions of them (43.3% and 56.6%).

	Cluster size	No. of CLN3-containing pairs	Portion within the cluster	Portion in all CLN3 pairs
Baseline	121	21	0.173	0.396
SOM	27	23	0.851	0.433
GTM	105	30	0.285	0.566
SOLL	99	44	0.444	0.830

required for sequential DNA replication process and sequential expression of CDC47 that execute the DNA replication.

In addition, both cluster 40 and cluster 92 contain ASH1 gene but show different relation between two correlated genes (figure 9). ASH1 is transcriptional repressor protein whose expression is peaked at M/G1 phase. The well known function of it is that it translocates to the daughter nucleus and repress HO gene for lineage-specific transcription (Sil & Herskowitz, 1996). Cluster 92 presents this negative relationship between ASH1 and its target repressed genes including HO. When SWI5, a transcription factor, is phosphorylated by CLB2/CDC28 complex, SWI5 is concentrated in the cytoplasm (Tebb et al., 1993). At anaphase, clb2 is degraded, and migrates to the nucleus and induces the expression of ASH1 and other genes in early G1 (Nasmyth et al., 1990; Moll et al., 1991). Cluster 40 shows the positive relation between ASH1 and its transcriptional activators because it contains not only SWI5 but also CLB2, the key start regulator in cell cycle. Comparative analysis of raw graphs and trajectory paths in cluster 40 and cluster 92 confirmed these positive and negative relations (figure 9).

Although regulation of abundance of many gene products through cell cycle regulator is the major regulation mechanism, post transcription events like proteolysis, phosphorylation and localization are likely to mediate to control the basic timing of cell cycle (Futcher, 2000). So, there might be a lot of complex patterns of correlation among genes. Further analysis of other clusters will help biologists to investigate the regulation mechanism and novel correlated genes in cell cycle.

We compared in Table 4 the result of SOLL with those of other profiling methods with respect to the identification rate of CLN3-containing pairs. SOLL contains the largest number of CLN3-containing pairs (83.0%) while the baseline method does not identify them effectively (39.6%). SOM and GTM contain moderate portions of them (43.3% and 56.6%). For the baseline method, the reason might be the lack of enough features for the temporal nature of the expression levels. In figure 11, part of the pairs of genes in the cluster 18 found by SOLL is represented. Note that the shapes of time series are not necessarily the same while the pattern of co-regulation is similar.

6. Concluding remarks

We proposed a self-organizing latent lattice (SOLL) model for analyzing temporal patterns with multi-dimensional features at each cell cycle time. The temporal trajectories of the

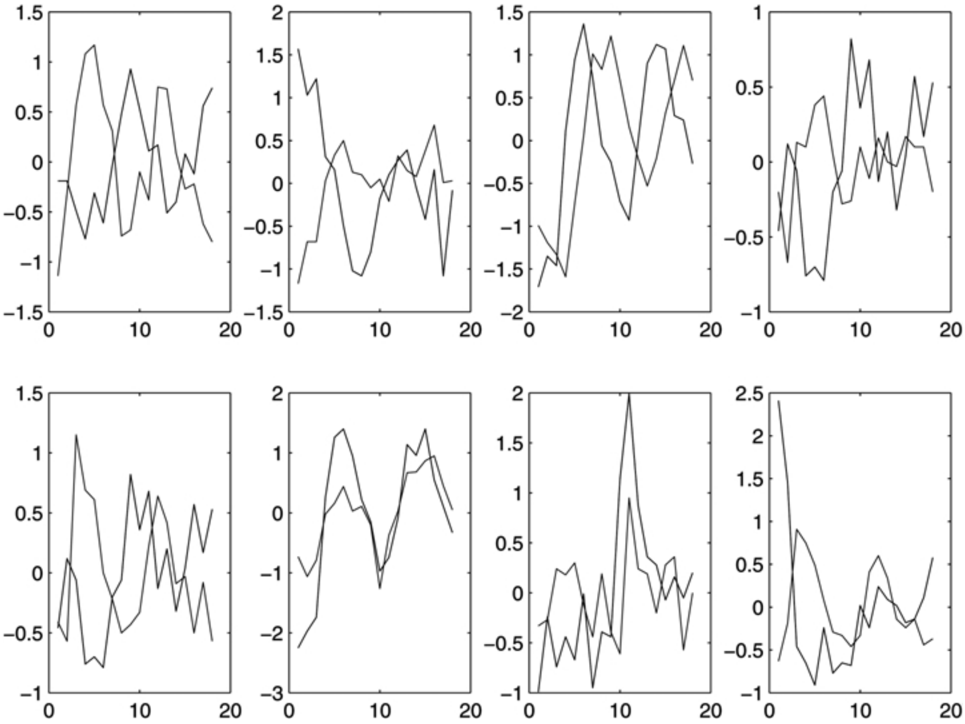

Figure 11. The various gene-expression time series corresponding to pattern 18.

correlated gene expressions allow us to analyze complex interactions of more than two genes during the cell cycle period. The usual practice of the analysis in microarray expression data is focused on the expression levels of single genes, and the search for patterns or clusters is based solely on each single gene's expression levels, neglecting more complex and realistic gene-to-gene interactions. Unlike simple expression levels of a single gene, multiple features in correlated genes are no longer possible to visualize or analyze by current gene-based methods. SOLL is expressing these multi-dimensional features in terms of hypothetical latent features which are simple enough to visualize and analyze.

For the simulated experiments of 6 different correlation patterns of periodic time series (Section 4.2), the performance of SOLL is evaluated and compared with those of other methods. The experimental evidence supports that SOLL outperforms the other methods in detecting several different peaks and patterns. The baseline method which neglects gene-to-gene interactions showed poor performance in detecting different peaks while concentrating individual gene shapes. On the real-life cell-cycle regulation data, SOLL was able to identify most of the CLN3 activated regulation gene pairs with several unknown pairs of genes while the baseline methods showed less effective results (Table 4). The CLN3 is biologically known to affect a group of genes during the cell cycle process.

Since SOLL is based on the generative latent-variable model, it can be used effectively in estimating the missing data for the data set with missing values. Another merit of SOLL is the visualization ability of the complex co-regulation process in the latent space. This feature can be applied to discovering novel gene-to-gene relations in intuitive and visual way.

Some remarks are in order with respect to the current approach to temporal expression profiling of correlated genes. In this paper, we restricted ourselves to the case of finding correlations between two genes. Though this proved very useful in our analysis, we may also want to analyze the correlation patterns of three or more genes. In this case, the selection of features to be presented to the SOLL's input layer should be extended appropriately. However, the SOLL model itself can be used without modification since it basically learns to map the temporal sequences of input vectors to the trajectory patterns on the latent space.

Acknowledgments

This work was supported by the Korean Government through BK21-IT, BrainTech, IMT2000 Bioinformatics and NRL Programs.

References

Aach, J., & Church, G. M. (2001). Aligning gene expression time series with time warping algorithms. *Bioinformatics, 17:6*, 495–508.

Alon, U., Barkai, N., Notterman, D. A., Gish, K., Ybarra, S., Mack, D., & Levine, A. J. (1999). Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays. *Proceedings of the National Academy of Sciences, 96*, 6745–6750.

Anderson, T. W. (1984). *An Introduction to Multivariate Statistical Analysis*, 2nd ed. (pp. 285–290). New York: John Wiley.

Bishop, C. M., Svensen, M., & Williams, C. K. I. (1998). GTM: The generative topographic mapping. *Neural Computation, 10:1*, 215–234.

Bishop, C. M. (1999). Latent variable models. *Learning in Graphical Models* (pp. 371–404). The MIT Press.

Brian, D. D. (1997). Regulation of transcription by proteins that control the cell cycle. *Nature, 389*, 149–152.

Buntine, W. L. (1994). Operations for learning with graphical models. *Journal of Artificial Intelligence Research, 2*, 159–225.

Butte, A. J., Tamayo, P., Slonim, D., Golub, T. R., & Kohane, I. S. (2000). Discovering functional relationships between RNA expression and chemotherapeutic susceptibility using relevance networks. *Proceedings of the National Academy of Sciences, 97*, 12182–12186.

Calinski, T., & Harabasz, J. (1974). A dendrite method for cluster analysis. *Communications in Statistics, 3*, 1–27.

Cheng, Y., & Church, G. M. (2000). Biclustering of expression data. In *Proceedings of the Eighth International Conference on Intelligent Systems for Molecular Biology*, ISMB2000 (pp. 93–103). AAAI Press.

Chiang, D. Y., Brown, P. O., & Eisen, M. B. (2001). Visualizing associations between genome sequences and gene expression data using genome-mean expression profiles. *Bioinformatics, 17*, s49–s55.

Cho, R. J., Campbell, M., Winzeler, E., Steinmetz, L., Conway, A., Wodicka, L., Wolfsberg, T., Gabrielian, A., Landsman, D., Lockhart, D., & Davis, R. (1998). A genome-wide transcriptional analysis of the mitotic cell cycle. *Molecular Cell, 2*, 65–73.

Dash, M., & Liu, H. (1997). Feature selection for classification. *Intelligent Data Analysis, 3:1*, 131–156.

De Risi, J. L., Iyer, V. R., & Brown, P. O. (1997). Exploring the metabolic and genetic control of gene expression on a genomic scale. *Science, 278*, 680–686.

Dempster, A. P., Laird, N. M., & Rubin, D. B. (1977). Maximum likelihood from incomplete data via the EM algorithm (with discussion). *Journal of the Royal Statistical Society, B39*, 1–38.

Duda, R. O., & Hart, P. E. (1973). *Pattern Classification and Scene Analysis*. New York: John Wiley & Sons, Inc.

Eisen, M. B., Spellman, P. T., Brown, P. O., & Botstein, D. (1998). Cluster analysis and display of genome-wide expression patterns. *Proceedings of the National Academy of Sciences, 95*, 14863–14868.

Futcher, B. (2000). Microarray and cell cycle transcription in yeast. *Current Opinion in Cell Biology, 12*, 710–715.

Herrero, J., Valencia, A., & Dopazo, J. (2001). A hierarchical unsupervised growing neural network for clustering gene expression patterns. *Bioinformatics, 17*, 126–136.

Ishimi, Y., Ichinose, S., Omori, A., Sato, K., & Kimura, H. (1996). Binding of human minichromosome maintenance proteins with histone H3. *Journal of Biological Chemistry, 271*, 24115–24122.

Jain, A., & Dubes, R. (1988). *Algorithms for Clustering Data*. Prentice-Hall, Eaglewood Clifts, NJ.

Koch, C., & Nasmyth, K. (1994). Cell cycle regulated transcription in yeast. *Current Opinion in Cell Biology, 6*, 451–459.

Kohonen, T. (1990). The self-organizing map. *Proceedings of the IEEE, 78:9*, 1464–1480.

Lockhart, D. J., Dong, H., Byrne, M. C., Follettie, M. T., Gallo, M. V., Chee, M. S., Mittmann, M., Wang, C., Kobayashi, M., Horton, H., & Brown, E. L. (1996). DNA expression monitoring by hybridization to high-density oligonucleotide arrays. *Nature Biotechnology, 14*, 1675–1680.

Lukashin, A. V., & Fuchs, R. (2001). Analysis of temporal gene expression profiles: Clustering by simulated annealing and determining the optimal number of clusters. *Bioinformatics 17:5*, 405–414.

Moll, T., Tebb, G., Surana, U., Robitsch, H., & Nasmyth, K. (1991). The role of phosphorylation and the CDC28 protein kinase in cell cycle-regulated nuclear import of S. cerevisiae transcription factor SWI5. *Cell 66*, 743–758.

Nasmyth, K., Adolf, G., Lydall, D., & Seddon, A. (1990). The identification of a second cell cycle control on the HO promoter in yeast; cell cycle regulation of SWI5 neclear entry. *Cell, 62*, 631–647.

Osley, M. A. (1991). The regulation of histone synthesis in the cell cycle. *Annual Review of Biochemistry, 60*, 827–861.

Price, C., Nasmyth, K., & Schuster, T. (1991). A general approach to the isolation of cell cycle regulated genes in the budding yeast, Saccharomyces cerevisiae. *Journal of Molecular Biology, 218*, 543–556.

Ramoni, M. F., Sebastiani, P., & Kohane, I. S. (2002). Cluster analysis of gene expression dynamics. *Proceedings of the National Academy of Sciences, 99*, 9121–9126.

Raychaudhuri, S., Stuart, J. M., & Altman, R. B. (2000). Principal component analysis to summarize microarray experiments: Application to sporulation time series. In *Pacific Symposium on Biocomputing*, 455–466.

Sasik, R., Iranfar, N., Hwa, T., & Loomis, W. F. (2002). Extracting transcriptional events from temporal gene expression patterns during Dictyostelium development. *Bioinformatics, 18:1*, 61–66.

Sharan, R., & Shamir, R. (2000). CLICK: A clustering algorithm with applications to gene expression analysis. In *Proceedings of the Eighth International Conference on Intelligent Systems for Molecular Biology, ISMB2000*, 307–316.

Sil, A., & Herskowitz, I. (1996). Identification of asymetrically localized determinant, Ash1p, required for lineage-specific trascription of the yeast HO gene. *Cell, 84*, 711–722.

Spellman, P. T., Sherlock, G., Zhang, M. Q., Iyer, V. R., Anders, K., Eisen, M. B., Brown, P. O., Botstein, D., & Futcher, B. (1998). Comprehensive identification of cell cycle-regulated genes of the yeast saccharomyces cerevisiae. *Molecular Biology of the Cell, 9*, 3273–3297.

Spevak, C., & Polfreman, R. (2000). Analyzing auditory representations for sound classification with self-organizing neural networks. In *Preceedings of COST G-6 on Digital Audio Effects*, Verona, Italy, Dec. 2000.

Stein, G. S., Stein, J. L., & Lian, J. B. (1992). Regulation of histone gene expression. *Current Opinion in Cell Biology, 4*, 166–173.

Tamayo, P., Slonim, D., Mesirov, J., Zhu, Q., Kitareewan, S., Dmitrovsky, E., Lander, E. S., & Golub, T. R. (1999). Interpreting patterns of gene expression with self-organizing maps: Methods and application to hematopoietic differentiation. *Proceedings of the National Academy of Sciences, 96:6*, 2907–2912.

Tebb, G., Moll, T., Dowzer, C., & Nasmyth, K. (1993). SWI5 instablilty may be neccesarry but is not sufficient for asymmetric HO expression in yeast. *Genes & Development, 7*, 517–528.

Young, R. A. (2000). Biomedical discovery with DNA arrays. *Cell, 102*, 9–15.

Zhang, B.-T., Ohm, P., & Muehlenbein, H. (1997). Evolutionary induction of sparse neural trees. *Evolutionary Computation, 5(2)*, 213–236.

Received July 2, 2002
Revised December 23, 2002
Accepted December 23, 2002
Final manuscript December 31, 2002