Tree-Dependent Components of Gene Expression Data for Clustering

Jong Kyoung Kim and Seungjin Choi

Department of Computer Science
Pohang University of Science and Technology
San 31 Hyoja-dong, Nam-gu
Pohang 790-784, Korea
{blkimjk,seungjin}@postech.ac.kr

Abstract. Tree-dependent component analysis (TCA) is a generalization of independent component analysis (ICA), the goal of which is to model the multivariate data by a linear transformation of latent variables, while latent variables fit by a tree-structured graphical model. In contrast to ICA, TCA allows dependent structure of latent variables and also consider non-spanning trees (forests). In this paper, we present a TCA-based method of clustering gene expression data. Empirical study with yeast cell cycle-related data, yeast metabolic shift data, and yeast sporulation data, shows that TCA is more suitable for gene clustering, compared to principal component analysis (PCA) as well as ICA.

1 Introduction

Clustering genes from expression data into biologically relevant groups, is a valuable tool for finding characteristic expression patterns of a cell and for inferring functions of unknown genes. Clustering is also widely used in modelling transcriptional regulatory networks, since it reduces the data complexity [1]. On one hand, classical clustering methods such as k-means, hierarchical clustering and self-organizing map (SOM), have widely been used in bioinformatics. On the other hand, linear latent variables models were recently used in the task of gene clustering. This includes principal component analysis (PCA) [2], factor analysis [3], independent component analysis (ICA) [4–6], independent subspace analysis (ISA) [7, 8], and topographic ICA [9].

The underlying assumption in linear latent variable models, is that gene expression profiles (measured by microarray experiments) are generated by a linear combination of linear modes (corresponding to prototype biological processes) with weights (encoding variables or factors) determined by latent variables. In such a case, latent variables indicates the portion of contributions of each linear mode to a specific gene profile. Clustering gene profiles can be carried out by investigating the significance of latent variables and representative biological functions directly come from linear modes of latent variable models. It was shown that clustering by latent variable models outperforms classical clustering algorithms (e.g., k-means) [5].

Tree-dependent component analysis (TCA) is a generalization of ICA, the goal of which is to seek a linear transform with latent variables well-fitting by a tree-structured graphical model, in contrast to ICA which restricts latent variable to be statistically independent [10]. TCA allows the dependent structure of latent variables and also incorporates with non-spanning trees (forests). In this paper, we present a method of gene clustering based on TCA. We compare the performance of TCA to PCA and ICA, for three yeast data sets, evaluating the enrichment of clusters through the statistical significance of *Gene Ontology* (GO) annotations [11].

2 Linear Latent Variable Models

Gene expression patterns measured in microarray experiments, result from unknown generative processes contributed by diverse biological processes such as the binding of transcription factors and environmental change outside a cell [4]. Genome-wide gene expression involves a very complex biological system and the characteristics of biological processes is hidden to us. A promising way to model such a generative process, is to consider a linear latent variable model such as PCA and ICA.

The linear generative model assumes that a gene profile $\boldsymbol{x}_t \in \mathbb{R}^m$ (the elements of $\boldsymbol{x}_t$ represent the expression levels of gene t at m samples or m time points) is assumed to be generated by

$$\boldsymbol{x}_t = \boldsymbol{A}\boldsymbol{s}_t + \boldsymbol{\epsilon}_t, \quad t = 1, \ldots, N, \tag{1}$$

where $\boldsymbol{A} = [\boldsymbol{a}_1 \cdots \boldsymbol{a}_n] \in \mathbb{R}^{m \times n}$ contains linear modes in its columns and $\boldsymbol{s}_t \in \mathbb{R}^n$ is a latent variable vector with each element s_{it} associated with the contribution of the linear mode $\boldsymbol{a}_i$ to the gene profile $\boldsymbol{x}_t$. The noise vector $\boldsymbol{\epsilon}_t \in \mathbb{R}^m$ takes the uncertainty in the model into account and it is assumed to be statistically independent of $\boldsymbol{s}_t$. For the sake of simplicity, we neglect the noise vector $\boldsymbol{\epsilon}_t$. Then the linear generative model (1) can be written in a compact form:

$$\boldsymbol{X} = \boldsymbol{A}\boldsymbol{S}, \tag{2}$$

where $\boldsymbol{X} = [X_{it}] \in \mathbb{R}^{m \times N}$ is the data matrix with each element X_{it} associated with the expression level of gene t at sample i (or time i). The latent variable matrix $\boldsymbol{S} \in \mathbb{R}^{n \times N}$ contains $\boldsymbol{s}_t$ for $t = 1, \ldots, N$.

Given a data matrix $\boldsymbol{X}$, latent variables $\boldsymbol{S}$ are determined by $\boldsymbol{S} = \boldsymbol{W}\boldsymbol{X}$, where the linear transformation $\boldsymbol{W}$ is estimated by a certain optimization method. Depending on restrictions or assumptions on $\boldsymbol{A}$ and $\boldsymbol{S}$, various methods including PCA, ICA, and TCA have been developed. A brief overview of those methods is given below.

2.1 PCA

PCA is a widely-used linear dimensionality reduction technique which decomposes high-dimensional data into low-dimensional subspace components. PCA is

illustrated as a linear orthogonal transformation which captures maximal variations in data. Various algorithms for PCA have been developed [12–14]. Singular value decomposition (SVD) is an easy way to determine principal components.

The SVD of the data matrix $\boldsymbol{X} \in \mathbb{R}^{m \times N}$ is given by

$$\boldsymbol{X} \approx \boldsymbol{U}\boldsymbol{D}\boldsymbol{V}^{\top}, \tag{3}$$

where $\boldsymbol{U} \in \mathbb{R}^{m \times n}$ $(n \leq m)$ contains n principal left singular vectors (eigenvectors) in its columns, $\boldsymbol{D} \in \mathbb{R}^{n \times n}$ is a diagonal matrix with eigenvalues on diagonal entries, and $\boldsymbol{V} \in \mathbb{R}^{N \times n}$ contains n right singular vectors in its columns.

In the framework of gene expression data analysis, the n column vectors of $\boldsymbol{U}$ correspond to *eigengenes* and the n column vectors of $\boldsymbol{V}$ are associated with *eigenarrays*. Exemplary applications of SVD or PCA to gene expression data, can be found in [15, 2].

2.2 ICA

ICA is a statistical method which model the observed data $\{\boldsymbol{x}_t\}$ by a linear model $\{\boldsymbol{A}\boldsymbol{s}_t\}$ with restricting non-Gaussian latent variables $\boldsymbol{s}_t$ to have statistically independent components. In contrast to PCA where the multivariate data is modelled by an orthogonal transformation of independent (or uncorrelated) Gaussian latent variables, ICA seeks a non-orthogonal transformation that makes non-Gaussian components to be as independent as possible. Refer to [16–18] for details and recent review of ICA.

The non-Gaussianity constraint for independent components, is very useful in the gene expression data analysis. Hidden biological processes affect only a few relevant genes and a large portion of genes remains unaffected. Gaussian distribution does not model this encoding process correctly. In fact, heavy-tailed distributions are more suitable for encoding variables $\{\boldsymbol{s}_t\}$ in gene expression data [5, 4]. The independence assumption on hidden variables $\{\boldsymbol{s}_t\}$ was shown to be an effective hypothesis for separating linearly-mixed biological signals in gene expression data. Despite of this effectiveness of the independence assumption, it is not realistic since biological systems are known to be highly inter-connected networks.

2.3 TCA

For the sake of simplicity, we omit the index t in both $\boldsymbol{x}_t$ and $\boldsymbol{s}_t$, unless it is necessary. As in ICA, we also assume that the data is pre-processed by PCA such that its dimension is reduced down to n. TCA is a generalization of ICA, where instead of seeking a linear transformation $\boldsymbol{W}$ that makes components $\{s_i\}$ independent (s_i is the ith-element of $\boldsymbol{s} = \boldsymbol{W}\boldsymbol{x}$), it searches for a linear transform $\boldsymbol{W}$ such that components (latent variables) $\{s_i\}$ well-fit by a tree-structured graphical model [10]. In TCA, s_i are referred to as *tree-dependent components*. In contrast to ICA, TCA allows the components s_i to be dependent and its dependency is captured by a tree-structured graphical model. Thus, it is

expected that TCA will be more suitable for gene clustering than ICA, since it is more realistic in seeking hidden biological processes. A brief overview of TCA is given below, and see [10] for more details.

Let us denote by $T(\mathcal{V}, \mathcal{E})$ an undirected tree, where $\mathcal{V}$ and $\mathcal{E}$ represent a set of nodes and a set of edges, respectively. The objective function considered in TCA model, involves the *T-mutual information* $I_T(\boldsymbol{s})$:

$$\begin{aligned} \mathcal{J}(\boldsymbol{x}, \boldsymbol{W}, T) &= I_T(\boldsymbol{s}) \\ &= I(s_1, \ldots, s_n) - \sum_{(i,j)\in\mathcal{E}} I(s_i, s_j), \end{aligned} \tag{4}$$

where $I(\cdot)$ is the mutual information. Note that in the case of ICA, only the mutual information $I(s_1, \ldots, s_n)$ serves as the objective function. The objective function (4) results from the minimal KL-divergence between the empirical distribution $p(\boldsymbol{x})$ and the model distribution $q(\boldsymbol{x})$ where the linear model $\boldsymbol{x} = \boldsymbol{A}\boldsymbol{s}$ is considered and $\boldsymbol{s}$ is assumed to factorize in a tree T.

In terms of entropies (denoted by $H(\cdot)$), the objective function (4) can be written as

$$\begin{aligned} \mathcal{J}(\boldsymbol{x}, \boldsymbol{W}, T) = \sum_j H(s_j) - \sum_{(i,j)\in\mathcal{E}} [H(s_i) + H(s_j) - H(s_i, s_j)] \\ - \log|\det \boldsymbol{W}|, \end{aligned} \tag{5}$$

where $H(\boldsymbol{x})$ is omitted since it is constant. The objective function (5) involves the calculation of entropy, which requires the probability distribution of $\boldsymbol{s}$ that is not available in advance. Several empirical contrast functions were considered in [10]. These include: (1) kernel density estimation (KDE); (2) Gram-Charlier expansion; (3) kernel generalized variance; (4) multivariate Gaussian stationary process-based entropy rate. In the case of ICA, Gaussian latent variables are not interesting. In such a case, the transformation $\boldsymbol{W}$ is defined up to an orthogonal matrix. On the other hand, TCA imposes a tree-structured dependency on latent variables, hence, this indeterminacy disappears and the transformation $\boldsymbol{W}$ can be estimated with a fixed tree T.

Incorporating with a non-spanning tree in TCA allows us to model inter-cluster independence, while providing a rich but tractable model for intra-cluster dependence. This is desirable for clustering since an exact graphical model for clusters of variables would have no edges between nodes that belong to different clusters and would be fully connected within a cluster. In order for non-spanning trees to be allowed, the following prior term (penalty term), $\zeta(T) = \log p(T)$, was considered in [10]:

$$\zeta(T) = \log p(T) = \sum_{(i,j)\in\mathcal{E}} \zeta_{ij}^0 + f(\#(T)), \tag{6}$$

where ζ_{ij}^0 is a fixed weight of (i, j), f is a concave function, and $\#(T)$ is the number of edges in T.

Model parameters $\boldsymbol{W}$ and non-spanning trees T in TCA are determined by alternatively minimizing the objective function $\widetilde{\mathcal{J}} = \mathcal{J}(\boldsymbol{x}, \boldsymbol{W}, T) - \zeta(T)$ [1]. Minimization of the objective function with respect to the discrete variable T, is solved by a greedy algorithm involving the maximum weight forest problem. The second minimization with respect to $\boldsymbol{W}$, is done by the gradient descent method. More details on TCA are found in [10].

3 Proposed Method for Clustering

ICA has been successfully applied to clustering genes from expression data in a non-mutually exclusive manner [5, 6]. Each independent component is assumed to be a numerical realization of a biological process relevant to gene expression. The genes having extremely large or small values of the independent component can be regarded as significantly up-regulated or down-regulated genes. However, the assumption that the hidden variables are mutually independent is too strong to model the real biological processes of gene expression properly. This limitation of ICA-based method of clustering can be solved by using TCA. The tree-structured graphical model of TCA is enough rich to model the real biological processes. The procedures of TCA-based clustering are summarized below.

Algorithm Outline: TCA-Based Clustering

Step 1 [**Preprocessing**] The gene expression data matrix $\boldsymbol{X}$ is preprocessed such that each element is associated with $X_{it} = \log_2 R_{it} - \log_2 G_{it}$ where R_{it} and G_{it} represent the red and green intensity of cDNA microarray, respectively. Genes whose profiles have missing values more than 10% are discarded. Missing values in $\boldsymbol{X}$ are filled in by applying the *KNNimpute*, a method based on k-nearest neighbors [19]. The data matrix is centered such that each row vector has zero mean. In the case of high-dimensional data, PCA could be applied to reduce the dimension, but it is not always necessary.

Step 2 [**Decomposition**] We apply the TCA algorithm to the preprocessed data matrix to estimate the demixing matrix $\boldsymbol{W}$ and the encoding variable matrix $\boldsymbol{S}$.

Step 3 [**Gene clustering**] In the case of ICA, the row vectors of $\boldsymbol{S}$ are statistically independent. Thus clustering is carried out for each row vector (associated with each linear mode that is the column vector of $\boldsymbol{A}$). In other words, for each row vector of $\boldsymbol{S}$, genes with strong positive and negative values of associated independent components, are grouped into two clusters,

[1] This objective function is the case where whitening constraints are imposed. In such a case, the minimization is carried out subject to $\boldsymbol{W}\boldsymbol{\Sigma}\boldsymbol{W}^\top = \boldsymbol{I}$ where $\boldsymbol{\Sigma}$ is the covariance matrix of $\boldsymbol{x}$.

each of which is related to induced and repressed genes, respectively. On the other hand, TCA reveals a dependency structure in the row vectors of $\boldsymbol{S}$. Hence, the row vectors of $\boldsymbol{S}$ associated with a spanning tree undergo a weighted sum. These resulting row vectors (the number of these row vectors is equal to the number of spanning trees in the forest) are used for grouping genes into up-regulated and down-regulated genes. Denote by $\mathcal{C}_i$ the cluster associated with an isolated spanning tree determined by TCA. The up-regulated ($\mathcal{C}_i^u$) and down-regulated ($\mathcal{C}_i^d$) genes are grouped by the following rule:

$$\mathcal{C}_i^u = \left\{ \text{gene } j \;\middle|\; \sum_{k \in \mathcal{C}_i} \|\boldsymbol{a}_k\|_2^2 \, \text{sign}(\overline{\boldsymbol{a}_k}) \, S_{kj} \geq c\,\sigma \right\},$$
$$\mathcal{C}_i^d = \left\{ \text{gene } j \;\middle|\; \sum_{k \in \mathcal{C}_i} \|\boldsymbol{a}_k\|_2^2 \, \text{sign}(\overline{\boldsymbol{a}_k}) \, S_{kj} \leq -c\,\sigma \right\}, \tag{7}$$

where σ denotes the standard deviation of $\sum_{k \in \mathcal{C}_i} \|\boldsymbol{a}_k\|_2^2 \, \text{sign}(\overline{\boldsymbol{a}_k}) \, S_{k,:}$, where $\overline{\boldsymbol{a}_k}$ is the average of $\boldsymbol{a}_k$ and $S_{k,:}$ is the kth row vector of $\boldsymbol{S}$. In our experiment, we chose $c = 1.5$.

4 Numerical Experiments

4.1 Datasets

We used three publicly available gene expression time series data sets, including yeast sporulation, metabolic shift, and cell cycle-related data. The details on these data sets are described in Table 1.

Table 1. The three data sets are summarized. The number of open reading frames (ORF) represents the total number of genes which are not discarded in the preprocessing step. The number of time points is equal to the dimension of the observation vector $\mathbf{x}$. We chose the number of clusters of hidden variables by using the TCA algorithm.

No.	Dataset	# of ORFs	# of time points	# of clusters	Reference
D1	sporulation	6118	7	2	[20]
D2	metabolic	6314	7	3	[21]
D3	cdc28	5574	17	9	[22]

4.2 Performance Evaluation

Evaluating statistical significance of clustering is one of the most important and difficult steps in clustering gene expression data [1]. For biologists, the contents of

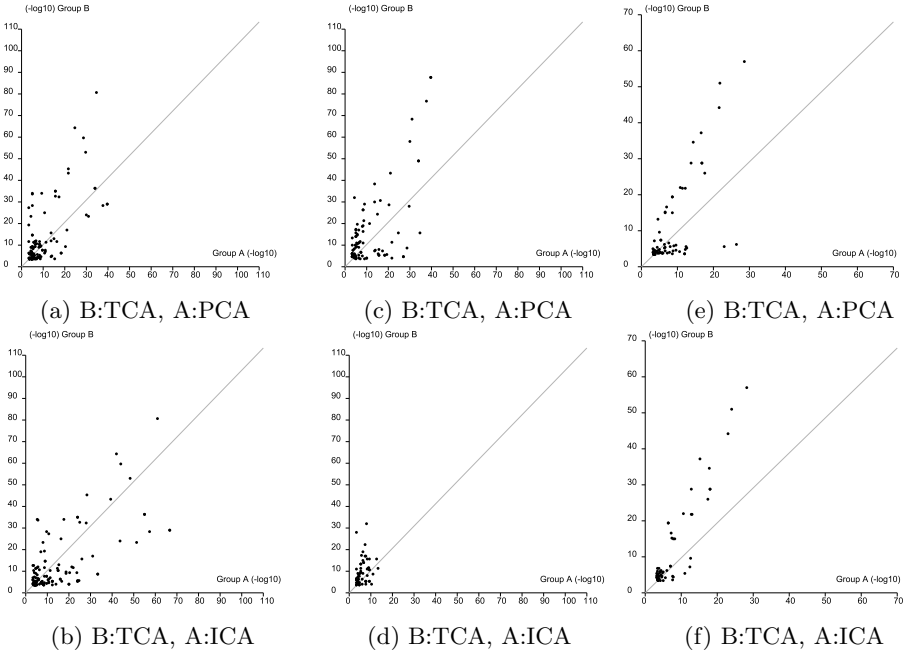
a cluster should be correctly interpreted in order to extract biologically valuable information from the results of clustering. The correct interpretation is guided by the analysis of statistical significance of clustering. In statistics, statistical significance is usually determined in the framework of hypothesis testing considering the null and alternative hypotheses. To apply the hypothesis testing framework to this work, we use the *Gene Ontology* (GO) database annotating gene products of many well-known genomes in terms of their associated biological processes, cellular components, and molecular functions [11]. From the gene list of a cluster, we obtain several annotation categories in which some genes of the cluster are contained. If the genes contained in a certain annotation category are observed within the cluster by chance, the number of genes follows the hypergeometric distribution. This is the null hypothesis H_0 and the opposite one is called the alternative hypothesis H_1. Under the null hypothesis H_0, the p-value of the probability to observe the number of genes as large or larger than k from an annotation category within a cluster of size n is given by

$$p = 1 - \sum_{i=0}^{k-1} \frac{\binom{f}{i}\binom{g-f}{n-i}}{\binom{g}{n}} \tag{8}$$

where f is the total number of genes within an annotation category of the GO database and g is the total number of genes within the genome. If the p-value is smaller than a fixed significance level α, we reject the null hypothesis H_0 and conclude that the genes contained in the annotation category are statistically significant [1]. To compare the statistical significance of two clustering results, we collect the minimum p-value smaller than α for each annotation category observed in both clustering results. A scatter plot of the negative logarithm of the collected p-values are finally drawn for visual comparison [5]. In the experiments, we set $\alpha = 0.005$ for the significance level. We have developed a software called *GOComparator* which calculates p values of GO annotations and compares the two clustering results visually by plotting the minimum p-values shared in both. It is freely available at *http://home.postech.ac.kr/∼blkimjk/software.html*.

4.3 Results

We compared the performance of TCA-based clustering with PCA and ICA by using the three yeast datasets. The method of clustering with the two algorithms is very similar to TCA except that decomposition is performed by PCA and ICA, respectively. In addition, the weighted summation of tree-dependent components in the gene clustering step is not done as there are no clusters of hidden variables in the two algorithms. We compared three different ICA algorithms to choose one showing the best clustering performance in ICA-based clustering. The used ICA algorithms are Self Adaptive Natural Gradient algorithm with nonholonomic constraints (SANG), Joint Approximate Diagonalization of Eigenmatrices (JADE), and Fixed-Point ICA (FPICA) [23]. Among the three



(a) B:TCA, A:PCA (c) B:TCA, A:PCA (e) B:TCA, A:PCA

(b) B:TCA, A:ICA (d) B:TCA, A:ICA (f) B:TCA, A:ICA

Fig. 1. Comparison of TCA based clustering to PCA and ICA on three yeast datasets. For each dataset, TCA has more points above the diagonal, which indicates that TCA has more significant GO annotations. (a), (b): D1, (c), (d): D2, (e), (f): D3

ICA algorithms, SANG shows the best performance in terms of statistical significance of GO annotations for each dataset. We also compared TCA algorithms with different empirical contrast functions: CUM, KGV, KDE, and STAT. The TCA algorithm based on Gaussian stationary process (STAT) outperforms the others for each dataset. The performance of TCA with a non-spanning tree was better than that of a spanning tree. The comparison results of three datasets are shown in Fig. 1. It confirms that TCA-based clustering outperforms PCA- and ICA-based clustering. The number of clusters of tree-dependent components chosen by TCA is given in Table 1. By applying PCA, we reduced the number of hidden variables in PCA- and ICA-based clustering to the chosen number of clusters of TCA-based clustering. Because of the computational cost of TCA, we reduced the dimension of the data vector to 10 by applying PCA for the dataset D3. For each dataset, the edge prior,ζ_{ij}^0, in (6) was chosen to $\frac{8\log(N)}{N}$, where N is the total number of genes.

The clustering based on the linear latent variable models can reveal hidden biological processes determining gene expression patterns. In the case of TCA-based clustering, each non-spanning tree corresponds to an unknown biological process. The characteristics of the unknown biological processes can be revealed by referring to the most significant GO annotations. The most significant GO annotations of the dataset D2 selected by TCA are given in Table 2. The dataset

Table 2. The most significant GO annotations of the dataset D2 selected by TCA. The results of cluster 2 are not shown since it did not contain any significant GO annotations.

Cluster	Induced functions	Repressed functions
1	sporulation, spore wall assembly	structural molecule activity, macromolucule biosynthesis
3	aerobic respiration, cellular respiration, carbohydrate metabolism	ribosome biogenesis and assembly, cytoplasm organization and biogenesis

D2 shows the diauxic shift which is a switch from anaerobic growth to aerobic respiration upon depletion of glucose [21]. The selected significant GO annotations of the cluster 3 represent the unknown biological processing related with the diauxic shift of yeast.

5 Conclusions

In this paper, we have presented a method of TCA-based clustering for gene expression data. Empirical comparison to PCA and ICA, with three different yeast data sets, has shown that the TCA-based clustering is more useful for grouping genes into biologically relevant clusters and for finding underlying biological processes. The success of TCA-based clustering has confirmed that a tree-structured graph (a forest consisting of Chow-Liu trees) for latent variables is a more realistic and richer model for modelling hidden biological processes.

Acknowledgments: This work was supported by National Core Research Center for Systems Bio-Dynamics and POSTECH Basic Research Fund.

References

1. Tavazoie, S., Hughes, J.D., Campbell, M.J., Cho, R.J., Church, G.M.: Systematic determination of genetic network architecture. Nature Genetics **22** (1999) 281–285
2. Raychaudhuri, S., Stuart, J.M., Altman, R.B.: Principal components analysis to summarize microarray experiments: Application to sporulation time series. In: Proc. Pacific Symp. Biocomputing. (2000) 452–463
3. Girolami, M., Breitling, R.: Biologically valid linear factor models of gene expression. Bioinformatics **20** (2004) 3021–3033
4. Liebermeister, W.: Linear modes of gene expression determined by independent component analysis. Bioinformatics **18** (2002) 51–60
5. Lee, S., Batzoglou, S.: ICA-based clustering of genes from microarray expression data. In: Advances in Neural Information Processing Systems. Volume 16., MIT Press (2004)
6. Kim, S., Choi, S.: Independent arrays or independent time courese for gene expression data. In: Proc. IEEE Int'l Symp. Circuits and Systems, Kobe, Japan (2005)

7. Kim, H., Choi, S., Bang, S.Y.: Membership scoring via independent feature subspace analysis for grouping co-expressed genes. In: Proc. Int'l Joint Conf. Neural Networks, Portland, Oregon (2003)
8. Kim, H., Choi, S.: Independent subspaces of gene expression data. In: Proc. IASTED Int'l Conf. Artificial Intelligence and Applications, Innsbruck, Austria (2005)
9. Kim, S., Choi, S.: Topographic independent component analysis of gene expression time series data. In: Proc. Int'l Conf. Independent Component Analysis and Blind Signal Separation, Charleston, South Carolina (2006) 462–469
10. Bach, F.R., Jordan, M.I.: Beyond independent components: Trees and clusters. Journal of Machine Learning Research **4** (2003) 1205–1233
11. Ashburner, M., Ball, C.A., *et al.*: Gene ontology: Tool for the unification of biology. Nature Genetics **25** (2000) 25–29
12. Diamantaras, K.I., Kung, S.Y.: Principal Component Neural Networks: Theory and Applications. John Wiely & Sons, INC (1996)
13. Jolliffe, I.T.: Principal Component Analysis, 2nd Edition. Spinger (2002)
14. Choi, S.: On variations of power iteration. In: Proc. Int'l Conf. Artificial Neural Networks. Volume 2., Warsaw, Poland (2005) 145–150
15. Alter, O., Brown, P.O., Botstein, D.: Singular value decomposition for genome-wide expression data processing and modeling. Proceedings of the National Academy of Sciences, USA **97** (2000) 10101–10106
16. Hyvärinen, A., Karhunen, J., Oja, E.: Independent Component Analysis. John Wiley & Sons, Inc. (2001)
17. Cichocki, A., Amari, S.: Adaptive Blind Signal and Image Processing: Learning Algorithms and Applications. John Wiley & Sons, Inc. (2002)
18. Choi, S., Cichocki, A., Park, H.M., Lee, S.Y.: Blind source separation and independent component analysis: A review. Neural Information Processing - Letters and Review **6** (2005) 1–57
19. Troyanskaya, O., Cantor, M., Sherlock, G., Brown, P., Hastie, T., Tibshirani, R., Botstein, D., Altman, R.B.: Missing value estimation methods for DNA microarrays. Bioinformatics **17** (2001) 520–525
20. Chu, S., DeRisi, J., Eisen, M., Mulholland, J., Botstein, D., Brown, P.O., Herskowitz, I.: The transcriptional program of sporulation in budding yeast. Science **282** (1998) 699–705
21. DeRisi, J.L., Iyer, V.R., Brown, P.O.: Exploring the metabolic and genetic control of gene expression on a genomic scale. Science **278** (1997) 680–686
22. 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. Molecular Biology of the Cell **9** (1998) 3273–3297
23. Cichocki, A., Amari, S., Siwek, K., Tanaka, T.: ICALAB Toolboxes (2002)