Time lagged recurrent neural network for temporal gene expression classification

Yulan Liang

Department of Family and Community Health,
University of Maryland,
655 W. Lombard. St., Rm 404K,
Baltimore, MD, 21201, USA
Fax: 410-706-0730
E-mail: ylian001@umaryland.edu

Arpad Kelemen*

Department of Organisational Systems and Adult Health,
University of Maryland,
655 W. Lombard. St., Rm 475A,
Baltimore, MD, 21201, USA
Fax: 410-706-3289
E-mail: akele001@umaryland.edu
*Corresponding author

Abstract: Heterogeneous gene expressions provide insight into the biological role of gene interaction with the environment, disease development and drug effect at the molecular level. We propose Time Lagged Recurrent Neural Network with trajectory learning for identifying and classifying gene functional patterns from the heterogeneous nonlinear time series microarray experiments. The proposed procedures identify gene functional patterns from the dynamics of a state-trajectory learned in the heterogeneous time series and the gradient information over time. Trajectory learning with Back-propagation through time algorithm can recognise gene expression patterns vary over time. This reveals more information about the regulatory network underlying gene expressions.

Keywords: gene expression; heterogeneous; time lagged neural network; trajectory learning; backpropagation through time.

Reference to this paper should be made as follows: Liang, Y. and Kelemen, A. (2009) 'Time lagged recurrent neural network for temporal gene expression classification', *Int. J. Computational Intelligence in Bioinformatics and Systems Biology*, Vol. 1, No. 1, pp.86–99.

Biographical notes: Yulan Liang is an Associate Professor in the Department of Family and Community Health and Senior Biostatistician at the Research Office of the University of Maryland, Baltimore. She was an Assistant Professor at the Department of Biostatistics at the University at Buffalo from 2002 to 2008. She received her PhD and MS Degrees in Applied Statistics from the University of Memphis. Her research interests include Bayesian inference, categorical data modelling, multivariate statistics, time series, data mining, neural networks, statistical learning, statistical pattern recognition. Her application areas include statistical genomics and bioinformatics, epidemiology, neuroimaging, medical imaging and clinical trials.

Arpad Kelemen is an Associate Professor in the Department of Organisational Systems and Adult Health at the University of Maryland, Baltimore. He received his PhD from the University of Memphis and his MS Degree from the University of Szeged, Hungary, both in Computer Science. His research interests include computational intelligence, artificial intelligence, intelligent agents, machine learning, neural networks, bioinformatics, neuroimaging, medical imaging, and computational biology. He published two books and more than 50 papers in international journals and conferences. He received several awards and honors.

1 Introduction

Understanding the function of each gene in the human/animal genome is not a trivial task. Learning the gene interactions with the changing environment, with the development of a disease or under different treatment is an even greater challenge and critical to improve human life. As an important step, extracting the knowledge from heterogeneous types of gene expressions may provide a better insight into the biological role of gene interactions with disease development and drug effect at the molecular level. Heterogeneous types of gene expressions contain different experimental conditions. The experimental conditions may correspond to different time points under different dosages of a drug, measures from different individuals, different organs or different diseases. DNA microarrays allow the measurement of expression levels for thousands of genes, perhaps all genes of a cell or an organism, within a number of different experimental conditions (Gasch et al., 2000). The dynamic patterns of genes expressed under different conditions can be useful indicators about gene state-trajectories and may reveal possible states and trajectories of disease and treatment effects (Holter et al., 2001; Ramoni et al., 2002; Yeung and Ruzzo, 2001; Yeung et al., 2003; Hastiel et al., 2000; Romualdi et al., 2003; Neal et al., 2000). Also, the analysis of the gene state patterns can help identifying important and reliable predictors of diseases, such as cancer, in order to develop therapies and new drugs (Brown et al., 2000). Biologists, computer scientists and statisticians have had a decade of research on the use of microarrays to model gene expressions (D'haeseleer et al., 1999; Herrero et al., 2001; Holter et al., 2001; Raychaudhuri et al., 2000 etc.). Most of the studies are interested in the genes that co-express in similar conditions with fewer results on the heterogeneous types of gene expressions. Moreover, most of these studies focus on the mean profiles of the gene expression time course, which can make the cluster or classification of gene expressions largely simplified but ignores the important time updated (varied) information.

One feature of gene expression data in time series microarray experiment is that it includes a large number of attributes with high correlation and with high level noise. Because of its massive parallelism, potential for fault and noise tolerance, an Artificial Neural Network (ANN) based information processing is capable of taking the task to deal with this feature. ANNs can adapt their structure in response to the change of the gene expressions under different conditions in order to extract knowledge, which contributes to a deep understanding of gene interactions and identifies certain causal relationships among the genes with diseases and drugs (Bishop, 1995; Haykin, 1999).

The study of the heterogeneous gene expressions under different experimental conditions in a multivariate nonlinear time series may involve the study of dynamic changing of the statistical variations of non-stationary processes of gene expressions. There are several types of artificial neural networks for temporal processing, which can be used to model the natural characteristics of the gene changing under different conditions and update the information in the training data over time. Recurrent Neural Networks (RNNs) has the ability of dealing with time varying input and output and it can define neurons as states of the network (Pearlmutter, 1995). The output of the hidden layer is fed back to the input layer via time delay. An internal state of the network encodes a representation of some characteristics or a biological mechanism of gene interactions, based on the transition function of the state from a recursive neural network, eventually to control the production of the internal information. State space model can be viewed as a special case of RNN, which combines a stochastic process with observation data model uniformly based on the recursive neural network. Hidden Markov processes can also be used to model the gene activity systems in which the gene states are unobservable, but can be represented by a state transition structure determined by the state parameters and the state transition matrix while processing the patterns over time. Time Lagged Recurrent Neural Networks (TLRNNs) are extensions of conventional RNNs and outperform them in the terms of network size. A TLRNN use short memory structure instead of static topology networks to develop advanced classification systems and use a complex learning algorithm: Back-Propagation Through Time (BPTT) to learn the temporal pattern (Stornetta et al., 1988; Werbos, 1993). This dynamic learning process is well adapted to the heterogeneous time series gene expression domain. TLRNNs can be used in nonlinear time series prediction, system identification and temporal pattern classification.

The goal of this paper is to investigate the performance of heterogeneous types of multivariate time series data using time lagged recurrent neural networks with dynamic trajectory learning. The question we are interested in is whether the dynamic heterogeneous gene activity patterns can be well identified or classified through the trajectory learning with a time lagged recurrent neural network.

The rest of the paper is divided as follows: in Section 2 we describe how the data was acquired and preprocessed. In Section 3 TLRNNs, statistical criteria for searching for the optimal model and related learning algorithms are presented. Experimental results are given in Section 4. We survey related work in Section 5 and finally we provide some concluding remarks in Section 6.

2 Data acquisition and preprocessing

2.1 Data acquisition

We used the widely studied set of yeast expression measurements data produced by Eisen et al. (Chu et al., 1998; Eisen et al., 1998; Spellman et al., 1998). The total data contained 2465 genes. Each data point represented the base two logarithm of the ratio of expression levels of a particular gene under two different conditions (CY5 and CY3 with red and green fluorescence intensity). The gene matrix contained 79 experiments over a variety of experimental conditions. The data were generated from spotted arrays using samples collected at various time points during diauxic shift, the mitotic cell division cycle, sporulation, the temperature, reduing shocks, and was taken from the Stanford genome research website (http://www-genome.stanford.edu). The training labels were extracted from Saccharomyces cerevisiae functional catalogue databases (Jansen and Gerstein, 2000). The study divided the data into three cases:

- *Identify two classes of gene functional patterns*: 121 ribosomal protein sample genes and 2346 non-ribosomal protein genes. We used two third of the data as training to construct the best model and the rest as testing the network performance.
- *Identify three different classes of functional gene expressions*: three classes and the sizes are listed as can be seen in Table 1.
- Identify multiple gene functional patterns from four to ten classes and see the network performance trends.

Ten subclasses are also extracted from the original functional catalogue, the last class as negative control and the other nine as learnable classes. The selected data and corresponding classes are given in Table 2.

Table 1 Three gene functional classes and size

Class	*Size*
Ribosomal	121
Transcription	159
Secretion	96

Table 2 Gene functional classes with heterogeneous data

Class	*Size*
Cell	168
Chromatin	48
Cytoskeleton	72
DNA	103
mRNA	103
Nuclear	43
Protein	477
Secretion	116
Transcription	136
Transport	129

2.2 Data preprocessing

Figure 1 gives scatter plots of data under different experimental conditions. Figure 2 provides time series plots of gene expressions under different experimental conditions.

Figure 1 Scatter plots of three different classes of functional gene expressions (ribosomal, transcription and secretion) under heterogeneous conditions (see online version for colours)

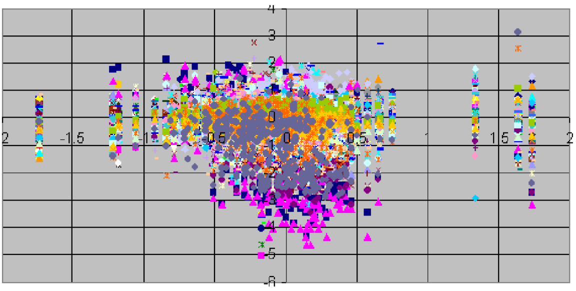
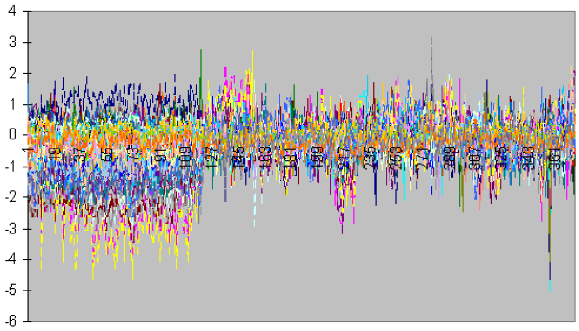


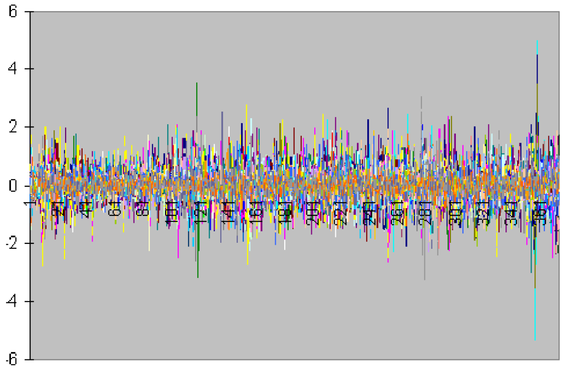
Figure 2 Time series plot of three classes functional gene expressions under heterogeneous conditions (see online version for colours)



2.3 *Smoothing the data*

As we have seen in the time series plots in Figure 2, the data contains high-frequency components and are non-stationary, which may make modelling difficult. To remove these factors, we processed the data by differencing the input series. After difference transformation, the time series were plotted in Figure 3. The figure shows that transformation can make the data more stationary.

Figure 3 Time series plots after difference transformation for three classes functional gene expressions under heterogeneous conditions (see online version for colours)



2.4 Input selection

Seventy nine inputs may be too large for a recurrent neural network, which is difficult to train (particularly if the data is noisy) and may result in overfitting problems, which do not provide good general solution. In order to select the neural network inputs, a statistical analysis has been carried out to determine the correlation between the inputs (experimental conditions) and the outputs (the class or pattern of genes). We compute the Pearson correlation coefficients of inputs and outputs, then based on the p-values we setup the acceptance threshold: if the *p*-value of the correlation coefficient is less than 0.0001, then we consider correlation and accept it as input, otherwise we drop it. The selected inputs and computed correlation coefficients are given in Table 3. The number of inputs was reduced from 79 conditions to 47. Several input permutation runs were also employed in order to find the combination, which produce the lowest error in the testing set. After we filtered out the low correlation inputs, the data were fed into the time lagged recurrent neural network. One advantage of the TLRNNs is that they can use the memory layer confined to the input, which can be used as further input preprocessors to reduce the redundant information.

Table 3 Input selection: Pearson correlation coefficients of inputs and outputs (class), Prob > |*r*| under H0: Rho = 0

Inputs	*Correlation coefficients*
alpha0	–0.29655
alpha56	–0.26218
alpha63	–0.21622
alpha70	–0.20749
alpha84	–0.34190
alpha91	–0.26009
alpha98	–0.44897
alpha105	–0.2258
alpha112	–0.39606
Elu0	0.39482
Elu60	–0.43226
Elu90	–0.57070
Elu120	–0.63138
Elu150	–0.59320
Elu180	–0.52208
Elu210	–0.48014
Elu240	–0.37021
Elu270	0.34820
Elu300	–0.25142
cdc1570	–0.26273
cdc1590	–0.32061
cdc15110	–0.40302

Table 3 Input selection: Pearson correlation coefficients of inputs and outputs (class), Prob > |*r*| under H0: Rho = 0 (continued)

Inputs	*Correlation coefficients*
cdc15130	–0.32735
cdc15150	–0.23325
cdc15210	–0.33316
spo0	–0.22003
spo2	0.63531
spo5	0.61170
spo7	0.58318
spo9	0.41863
spo511	–0.67660
spoearly	0.67581
spomid	0.67611
heat10	0.57206
heat20	0.77191
heat40	0.60434
heat80	0.57094
heat160	0.47246
dtt15	–0.25992
dtt60	0.51186
dtt120	0.75011
cold20	0.32741
cold40	0.38936
cold160	0.59074
diaua	0.33820
diauf	0.65262
diaug	0.67092

3 Design of time lagged recurrent neural network

3.1 Architecture

The heterogeneous gene expression contains information in its time structure, i.e., how the gene expression changes with time. Time Lagged Recurrent Neural Networks are useful to identify the gene functional patterns with time information. TLRNNs are extensions of Multi-Layer Perceptrons (MLPs) with short-term memory structures. In our study we choose the network architecture with three layers, the feedback connection from the hidden layer back to the input layer. The input layer use the inputs delayed by L samples before presented in the network. The trainings of the TLRNN are processed using back-propagation through time with trajectory learning and the parameters can be learned by examples (Werbos, 1993). The advantage of the BPTT with trajectory learning is that the learning system is trained to recognise or capture a set of patterns, which may vary with time since the number of inputs can be varied with time. Therefore it perfectly meets the requirements of the knowledge discovery purpose for biological systems and living systems like gene expression microarray data. In this case the static topology neural networks must be extended to dynamic networks, i.e., have short-term memory structures in which they capture the gene information of temporal patterns.

There are three memory structures at the input layer we can choose from: one sample delay, Gamma memory, and Laguerre memory. In our study we employed three cases to search for the best structure for the selected data. Based on Akaike and Bayesian information statistical criteria and classification accuracy the Gamma structure worked better than the other two.

3.2 Learning algorithms

One way to achieve the best network structure without information loss and accuracy loss is constructing the learning algorithm based on its capability of decreasing the effective number of weights during training to reduce the danger of overfitting. There are two ways to achieve this, one is using back-propagation through structural learning with forgetting in which the small weights are dropped off through regularisation parameters or decayed values (Miller and Zurada, 1998). The other way is using Back-Propagation Through Time (BPTT) in which the number of parameters in the network can be confined with the memory structure (Werbos, 1993). The BPTT algorithm uses the derivatives from state updated functions and cost functions, then trains over a trajectory of the input space, enabling it to capture the temporal dynamics of the time series information. Trajectory learning is based on gradient information over time. BPTT can adapt the depth of the memory using gradient descent, instead of changing the number of inputs. Initial memory depth in our study is setup to ten, which later can be adapted by the network according to the type of the memory structure. The learning rule for each layer applied back-propagation with momentum, where the momentum was setup to 0.7. Tangent sigmoid transfer function works well for the given data.

3.3 Statistical criteria for model selection

The goal of model selection is to find the best network architecture that can achieve the balance between data fitting and model complexity in order to avoid over-fitting and to optimise generalisation performance. In a time lagged recurrent neural network, the model complexity level varies through trajectories with distinct types of common states. There are several dynamic parameters such as the number of hidden neurons, the memory taps (depth in the samples) and the number of trajectories in the search space that need to be optimised in order to achieve optimal models. The memory taps (the depth in samples) parameter can be adapted through BPTT. We designed two-way factorial arrays to search for the best values of the trajectory and the number of hidden neurons. In our study the number of trajectories is ranged from 2 to 20 and the number of hidden nodes from 2 to 20.

To avoid a sequence of hypothesis testing for the model selection, we use statistical criteria such as the Akaike Information Criterion (AIC), or the Bayesian Information Criterion (BIC) to determine the optimal vales for optimal network size and structure

(Akaike, 1974). All these statistical criteria share the common form of composition, which is the model complexity criterion (equals to log-likelihood) plus a complexity penalty term. The advantage of using statistical criteria like AIC/BIC other than using sequential tests for model selection as proved by Stone (1974) is that there is an asymptotic equivalence of the choice by cross-validation and AIC. Therefore we can avoid the computational drawback of cross-validation and do not need to separate the data into cross-validation set when selecting the model. Note that the difference between AIC and BIC is that BIC includes the size of the input examples, and it may be more informative. The model with lowest AIC/BIC is considered to be the preferred one and if the model complexity is changed with the size of the sample, BIC is preferred, otherwise we prefer AIC (Li et al., 2000). The best neural network is the one with the highest classification accuracy and the lowest AIC/BIC. The best model was chosen for the rest of the gene classification and future predictions.

4 Experimental results

4.1 Two classes: ribosome and non-ribosome protein genes

We used TLRNN with one hidden layer and Gamma memory function to identify two classes. The learning rule used back-propagation with momentum with step size 1, the learning rate was 0.7, and tangent sigmoid function was used as transfer function. After 1000 epochs of training the MSE dropped below 0.000059. Classification accuracy for testing set provided the mean with standard deviation value 99.427% ± 0.366% with ten independent runs. This result is even better than the reported results by prediction algorithm: CLEAVER, with a classification accuracy of 99.229834% for the same data (Raychaudhuri et al., 2000).

We also employed Nearest Neighbor with Mahalanobis-Distances and Self-Organised Map (SOM) methods for comparison study, which gave correct classification rates of 97.39% and 98.53%, respectively. Most of the methods are available with Neurosolutions 4.1 software.

4.2 Three classes: ribosomal, transcription and secretion gene functional classes

Table 4 provides computed statistical criteria for model selection: values of AIC/BIC with average of five independent runs based on the number of trajectories and the number of hidden nodes for the case of three classes. Table 5 reports generalisation rates for the same runs. As it can be seen, the AIC/BIC values increase nearly linearly with the number of hidden nodes, while their values are essentially independent of the number of trajectories. Yet it is arguable if the low AIC/BIC values reported at trajectories 2, 5, and 18 are significant. Regarding Table 5 most of the low error rates are reported around the upper left corner, which corresponds to low number of hidden nodes with low number of trajectories, but the overall optimum can be found at four hidden nodes with 18 trajectories. Since the number of classes to be recognised is only three, it is not surprising that small number of hidden nodes and number of trajectories can provide good performance. Table 5 also shows that the learning capability of the model varies with the complexity of the gene patterns, which mainly depends on the number of patterns in the data and on the complexity of the trajectories. Our preliminary results show that by increasing the number of patterns (classes) to be recognised, the number of trajectories and the number of hidden nodes needs to be increased in order to get optimal performance.

Table 6 provides comparison results with some other popular machine learning approaches for gene expression classification: Nearest Neighbor with Mahalanobis Distances (NNMD), Self-Organised Map and Support Vector Machine (SVM). The values given in the table provided us the confidence with the mean and standard deviation of the correct classification rate for five independent runs. SVM in our case did not provide higher performance than TLRNN as opposed to most gene expression studies. The reason may come from the heterogeneous expression data, TLRNN particularly performs well for this kind of time series data. We also computed the results with another recurrent neural network: the Jordan/Elman Recurrent Neural Network (JERNN). According to the results for the heterogeneous time series gene expression patterns with three classes the TLRNN works best.

Table 4 Factorial array for model selection with dynamic trajectory learning: value of AIC/BIC based on the number of trajectories (T) and the number of hidden neurons (H)

T\H	*2*	*4*	*8*	*12*	*15*	*20*
2	1861/2143	3300/4289	7661/8698	11398/12970	14273/16234	19021/21629
5	1846/2127	3828/4367	7638/8694	11404/12978	14316/16263	19092/21699
8	1953/2234	3788/4327	7636/8692	11444/13018	14293/16254	19091/21700
10	1904/2184	3861/4400	7631/8687	11410/12985	14154/16345	19046/21653
12	1773/2250	3722/4582	7688/8744	11460/13033	14195/17325	19128/21735
15	1945/2225	3815/4354	7679/8735	11505/13078	14288/16250	19111/21719
18	1907/2187	3807/4346	7624/8680	11420/12992	14282/16243	19059/21666
20	1936/2217	3834/4373	7665/8721	11445/13018	14378/16339	19139/21747

Table 5 The generalisation error percentage with TLRNN based on the number of learned trajectories (T) and the number of hidden neurons (H)

T\H	*2*	*4*	*8*	*12*	*15*	*20*
2	4.42	3.96	5.72	4.70	5.30	5.01
5	3.74	4.97	6.46	5.61	5.09	7.97
8	5.73	4.36	5.29	7.99	6.16	7.72
10	4.53	7.43	5.61	4.63	5.90	8.66
12	5.43	6.53	9.15	9.31	7.36	11.60
15	5.96	4.50	9.71	10.41	5.51	9.41
18	4.39	3.48	5.75	6.23	5.03	5.90
20	6.03	5.26	8.39	6.36	10.58	12.66

4.3 More broad functional classes: identify four to ten classes using TLRNN

The data distribution for more broad gene functional classes is given in Table 2. The correct classification rates with TLRNN are given in Table 7, which are based on the optimal structure given by the AIC/BIC. As it can be seen in the table the correct classification rate decreases with the number of classes, which is not surprising. Also, as we discussed earlier both the number of hidden nodes and the number of trajectories increases as the number of classes increases for achieving better performance.

Table 6 Correct classification rates of different methods for three classes

Methods	*Correct classification rate* (%) ± *STD*
NNMD	73.28 ± 0.012
SVM	74.65 ± 0.002
SOM	80.44 ± 0.053
TLRNN	95.61 ± 0.018
JERNN	94.04 ± 0.015

Table 7 The correct classification rates of TLRNN with BPTT with dynamic trajectory learning corresponding to the number of patterns to be discovered

Number of patterns (*classes*)	*Correct classification rate* (%)
3	96.52
4	87.14
5	85.06
6	76.15
10	62.14

5 Related works

A large number of approaches have been proposed, implemented and tested by computer scientists and statisticians in order to discover or identify the gene functional patterns with microarray experiments (Golub et al., 1999; Alter et al., 2000). For example, a genetic network approach was discussed and developed by Thieffry and Thomas (1998) and D'haeseleer et al. (1999). Time series was studied by Socci and Mitra (1999) and so on. Self-organised hierarchical neural network was done by Herrero et al. (2001). Unsupervised neural network and associated memory neural network was done by Azuaje (2001) and Bicciato et al. (2001), classification and diagnostic prediction of cancers using gene expression profiling and artificial neural networks was investigated by Khan et al. (2001). Comparison of discrimination methods for the classification of tumors using gene expression data was done by Dudoit et al. (2002). We reported a Bayesian neural network approach earlier (Liang et al., 2002). Previous study showed that traditional statistical models can provide some insight into gene expressions and has precise results, but the weaknesses of statistical models are that they can not capture the dynamic changing of gene expressions from time to time well and are sensitive to noise and assumptions. Neural networks are more efficient and flexible for studying gene expressions. We, as an addition to our efforts reported in this paper currently explore other kinds of neural network models for discovering correlation in gene patterns, and refine the Jordan/Elman neural network approach to study the heterogeneous time series gene expression patterns.

6 Conclusion

In this paper, we proposed and explored the use of TLRNNs with dynamic trajectory learning for investigating the gene functional patterns with heterogeneous microarray experiments. Results show that the TLRNN works better than the nearest neighbour with Mahalanobis Distances, SVM and SOM. For SVM, this is a little surprise since most well known results using SVM provided the highest performance, and it has properties of dealing with high level noise and large number of attributes, which both exist in the gene expression data. The possible reasons may be found in the heterogeneous time series gene expression data, since it involved different conditions with time information. Another reason is that TLRNN can iteratively construct the network, train the weights and update the time information. Results show that the best generalisation capability largely depends on the complexity of the patterns in which TLRNN can be monitored by the complexity of the trajectory with distinct types of states. With the increase in the number of gene functional patterns, the generalisation performance decreased. However, with changing the number of trajectories and the number of hidden nodes, the performance of the model can be improved based on the statistical criteria for model selection, in which two or three way factorial design can be employed in order to search for the best network architecture for prediction and medical diagnosis.

References

Akaike, H. (1974) 'A new look at the statistical model identification', *IEEE Transactions on Automatic Control*, Vol. 19, pp.716–723.

Alter, O., Brown, P.O. and Botstein, D. (2000) 'Singular value decomposition for genome-wide expression data processing and modeling', *PNAS*, Vol. 97, pp.10101–10106.

Azuaje, F. (2001) 'Unsupervised neural network for discovery of gene expression patterns in B-cell lymphoma', *Online Journal of Bioinformatics*, Vol. 1, pp.26–41.

Bicciato, S., Pandin, M., Didone, G. and Bello, C. (2001) 'Analysis of an associative memory neural network for pattern identification in gene expression data BIOKDD01', *Workshop on Data Mining in Bioinformatics*, pp.22–30.

Bishop, C.M. (1995) *Neural Networks for Pattern Recognition*, Oxford University Press, Oxford, UK.

Brown, M.P., Grundy, W.N., Lin, D., Cristianini, N., Sugnet, C.W., Furey, T.S., Junior, M.A. and Haussler, D. (2000) 'Knowledge-based analysis of microarray gene expression data by using supported vector machines', *PNAS, USA*, Vol. 97, No. 1, pp.262–267.

Chu, S., DeRisi, J., Eisen, M., Mulholland, J., Botstein D. and Brown, P.O. (1998) 'The transcriptional program of sporulation in budding yeast', *Science*, Vol. 282, pp.699–705.

D'haeseleer, P., Liang, S. and Somogyi, R. (1999) 'Gene expression analysis and genetic networks modeling', *Pacific Symposium on Biocomputing: Tutorial Session on Gene Expression and Genetic Networks*, pp.1–34.

Dudoit, S., Fridlyand, J. and Speed, T. (2002) ‘Comparison of discrimination methods for the classification of tumors using gene expression data’, *JASA*, Vol. 97, No. 457, pp.77–87.

Eisen, M.B., Spellman, P.T., Brown, P.O. and Botstein, D. (1998) ‘Cluster analysis and display of genome-wide expression patterns’, *PNAS*, Vol. 95, No. 25, pp.14863–14868.

Gasch, A.P., Spellman, P.T., Kao, C.M., Carmel-Harel, O., Eisen, M.B., Storz, G., Botstein, D. and Brown, P.O. (2000) ‘Genomic expression programs in the response of yeast cells to environmental changes’, *Mol. Biol. Cell*, Vol. 11, pp.4241–4257.

Golub, T.R., Slonim, D.K., Tamayo, P., Huard, C., Gaasenbeek, M., Mesirov, J.P., Coller, H., Loh, M.L., Downing, J.R., Caliguiri, M.A., Bloomfield, C.D. and Lander, E.S. (1999) ‘Molecular classification of cancer: class discovery and class prediction by gene expression monitoring’, *Science*, Vol. 286, pp.531–537.

Hastiel, T., Tibshirani, R., Eisen, M.B., Alizadeh, A., Levy, R, Staudt, L., Chan, W.C., Botstein, D. and Brown, P.O. (2000) ‘Gene shaving as a method for identifying distinct sets of genes with similar expression patterns’, *Genome Biology*, Vol. 1, No. 2, research0003.1–0003.21.

Haykin, S. (1999) *Neural Networks*, Prentice-Hall, Upper Saddle River, New Jersey.

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

Holter, N., Maritan, A., Cieplak, M., Fedoroff, N. and Banavar, J. (2001) ‘Dynamic modeling of gene expression data’, *PNAS*, *USA*, Vol. 98, p.1693–1698.

Jansen, R. and Gerstein, M. (2000) ‘Analysis of the yeast transcriptome with structural and functional categories: characterizing highly expressed proteins’, *Nucleic Acids Research*, Vol. 28, No. 6, pp.1481–1488.

Khan, J., Wei, J.S., Ringner, M., Saal, L.H., Ladanyi, M., Westermann, F., Berthold, F., Schwab, M., Antonescu, C.R., Peterson, C. and Meltzer, P.S. (2001) ‘Classification and diagnostic prediction of cancers using gene expression profiling and artificial neural networks’, *Nature Med.*, Vol. 7, No. 6, pp 673–679.

Li, W., Sherriff, A. and Liu, X. (2000) ‘Assessing risk factors of human complex diseases by Akaike and Bayesian information criteria (abstract)’, *Am. J. Hum. Genet.*, Vol. 67, Suppl.p. S222.

Liang, Y., George, E.O. and Kelemen, A. (2002) ‘Bayesian neural network for microarray data’, *Proceeding of the IEEE International Joint Conference on Neural Network*, pp.193–198.

Miller D.A. and Zurada, J.M. (1998) ‘A dynamical system perspective of structural learning with forgetting’, *IEEE Transactions on Neural Networks*, Vol. 9, No. 3, pp.508–515.

Neal, S.H., Madhusmita, M., Holter, N.S., Mitra, M., Maritan, A., Cieplak, M., Banavar, J.R. and Fedoroff, V.F. (2000) ‘Fundamental patterns underlying gene expression profiles: simplicity from complexicity’, *PNAS*, Vol. 97, pp.8409–8414.

Pearlmutter, B.A. (1995) ‘Gradient calculation for dynamic recurrent neural networks’, *IEEE Transactions on Neural Networks*, Vol. 6, No. 5, pp.1212–1228.

Ramoni, M., Sebastiani, P. and Kohane, I. (2002) ‘Cluster analysis of gene expression dynamics’, *PNAS*, Vol. 99, pp.9121–9126.

Raychaudhuri, S., Sutphin, P.D., Stuart, J.M. and Altman, R.B. (2000) *CLEAVER: Analyzing Microarray Data using known Biological Categories*, Stanford Medical Informatics Technical Report 2000-0839.

Romualdi, C., Campanaro, S., Campagna, D., Celegato, B., Cannata, N., Toppo, S., Valle, G. and Lanfranchi, G. (2003) ‘Pattern recognition in gene expression profiling using DNA array: a comparative study of different statistical methods applied to cancer classification’, *Human Molecular Genetics*, Vol. 12, No. 8, pp.823–836.

Socci, N.D. and Mitra, P. (1999) ‘Time series analysis of yeast S. Cerevisiae cell cycle expression data’, in Bronberg-Bauer, E., De Beucklaer, A., Kummer, U. and Rost, U. (Eds.): *Proceedings of Workshop on computation of Biochemical Pathways and Genetic Networks*, Heidelberg.

Spellman, P.T., Sherlock, G., Zhang, M.W., Iyer, V.R., Anders, K., Eisen, M.B., Brown P.O. and Futcher, B. (1998) 'Comprehensive identification of cell cycle regulated genes of the yeast Saccharomyces cerevisiae by microarray hybridization', *Mol. Biol. Cell*, Vol. 9, pp.3273–3297.

Stone, M. (1974) 'Cross-validatory choice and assessment of statistical predictions (with discussion)', *Journal of the Royal Statistical Society*, Vol. 36, Series B, pp.111–147.

Stornetta, W.S., Hogg, T. and Huberman, B. (1988) 'A dynamic approach to temporal pattern processing', in Anderson, D.Z. (Ed.): *Neural Information Processing Systems*, Springer Verlag, pp.750–759.

Thieffry, D. and Thomas, R. (1998) 'Qualitative analysis of gene networks', *Pacific Symp. Biocomp.*, Vol. 3, pp.77–88.

Werbos, P.J. (1993) 'Backropagation through time: What it does and how to do it', *Proc. ICNN*, San Francisco, CA.

Yeung, K.Y. and Ruzzo, W.L. (2001) 'Principal component analysis for clustering gene expression data', *Bioinformatics*, Vol. 17, pp.763–774.

Yeung, K.Y., Medvedovic, M. and Bumgarner, R.E. (2003) 'Clustering gene-expression data with repeated measurements', *Genome Biology*, Vol. 4, pp.R34.1–R34.17.