Classification of Time-course Gene Expression Data Using a Hybrid Neural-based Model

Saeede Eftekhari[1], Mohammad Reza Hassan Zadeh[2], Mehdi Khashei[*3]

Department of Industrial Engineering, Isfahan University of Technology
Isfahan, Iran

[1]s.eftekhari@in.iut.ac.ir; [2]mr.hasanzadeh@usc.ac.ir; [*3]khashei@in.iut.ac.ir

Abstract

The emergence of DNA microarray technologies leads to significant advances in molecular biology. Time-course gene expression data are often measured in order to study dynamic biological systems and gene regulatory networks. It is believed that genes demonstrating similar expression profiles over time might give an informative insight into how underlying biological mechanisms work. This importance leads that in microarray gene research, statistical and intelligent models have received considerable attention to analyze complex time-course gene expression data. Recently, various classification models have been applied in order to find genes which show similar periodic pattern expression. In this paper, we consider the classification of gene expression in temporal expression patterns; then propose an Intelligent hybrid classification model, in which an artificial neural network is combined with a multiple linear regression model to classify genes data. Empirical results show that the proposed model can yield more accurate results than other well-known statistical and intelligent classification models. Therefore, it can be applied as an appropriate approach for classification of gene expression data.

Keywords

Medical Diagnosis; Time-course Gene Expression; Classification Models; Artificial Intelligence (AI); Multilayer Perceptrons (MLPs)

Introduction

Microarray technology allows the monitoring of the expression profiles for tens of thousands of genes in parallel process which produce huge amounts of data. More and more studies have shown that microarray technology is a powerful and revolutionary tool for biological and medical researches by allowing this simultaneous monitoring of the expression levels of genes. Gene expression can be generally examined from two points of view, static and dynamic (Song et al., 2007). The gene expression in static microarray experiments is a snapshot at a single time, whereas, in time-course experiments the expression profiles of genes are repeatedly measured over a time. Since many biological systems are dynamic systems, temporal profiles of gene expression levels during a given biological process can often provide more insight about how gene expression levels evolve in time and how genes are dependent during a given biological process (Luan & Li, 22003). In particular, time-course microarray experiments are effective not only in studying gene expression profile levels over a period of time but also in exploring functions of genes and the interactions with their products. It is the main reason that there has been increasing application of statistical and intelligent approaches for analyzing time-course gene expression data. These approaches have been generally applied for clustering and classification purposes.

Spellman et al. (1998) used DNA microarrays and samples from yeast cultures to create a comprehensive catalog of yeast genes, whose transcript levels vary periodically within the cell cycle. They identified 800 genes that meet an objective minimum criterion for cell cycle regulation, thus providing an important and complete set of data for testing various statistical methodologies by other investigators who engaged themselves in time-course gene expression research subsequently. Luan and Li (2003) proposed a model-based approach to cluster time-course gene expression data using B-splines in the framework of a mixture model. They have compared the proposed model with other existing model-based clustering methods (Fraley & Raftery, 2002). Their results indicated that model works well in clustering noisy curves into respective clusters which are different in terms of either curve shapes or times to expression peaks. Song et al. (2007) proposed a unified approach for gene clustering and dimension reduction based on functional data analysis (FNDA) to group observed curves with respect to their shapes or patterns by using the sample information in time-course microarray experiments. They applied this method to a time-course microarray data on the yeast cell cycle, and demonstrated that the proposed method is able to identify tight clusters of genes with expression profiles focused on particular phases of the cell cycle.

Although clustering approaches can find profile groupings, they are not designed in order to reliably reproduce groupings that are known from independent sources of information. Therefore, there is sometimes an analytical challenge in understanding how the novel groupings relate to known groupings. Classification approaches take known groupings and create rules for reliably assigning genes or conditions into these groups (Raychaudhuri et al., 2001) Combining classification with microarray technology plays an important role in diagnosing and predicting disease such as cancer. Gui and Li (2003) proposed mixture functional discriminant analysis, namely MFDA, to classify genes based on time-course gene expression data, which accounts for time dependency of the measurements and the noisy nature of the microarray data. B-spline transformations and Gaussian mixtures are respectively used for dimension reduction and classification in MFDA. They have illustrated the proposed method with predicting functional classes of uncharacterized yeast ORFs.

Liang and Kelemen (2005) proposed a regularised neural network for classification of multiple gene temporal patterns and compared their propose method to other classification techniques. Leng and Muller (2006) proposed a model of classifying collections of temporal gene expression curves in which individual expression profiles are modeled as independent realizations of a stochastic process. The method uses a recently developed functional logistic regression tool based on functional principal components, aimed at classifying gene expression curves into known gene groups. They applied the methodology to two real data sets, yeast cell cycle gene expression profiles (Spellman et al., 1998) and Dictyostelium cell-type specific gene expression profiles (Iranfar et al., 2001). Park et al. (2008) applied the functional support vector machine for classification of gene functions. Their method also provides valuable functional information about interactions between genes and allows the assignment of new functions to genes with unknown functions.

Using hybrid classification models or combining several models has become a common practice in order to overcome the limitations of each component model (Khashei et al., 2009). The basic idea of these multi- model approaches is the use of each component model's unique capability to better capture different patterns in the data. In recent years, several hybrid classification models have been proposed using multilayer perceptrons and successfully applied to the classification problems. In this paper, the hybrid intelligent model is proposed to classification gene expression data based on time course pattern. In our proposed model, multiple linear regression models and multilayer perceptrons which are one of the most accurate and widely used linear and nonlinear classification techniques are combined together. The rest of the paper is organized as follows. In the next section, the formulation of the hybrid proposed model to classification tasks is reviewed. Then, the data set is introduced. In the next section, the proposed model is applied to gene expression data classification and the performance of the proposed model is compared to those of other well-known classifiers presented in the literature. Finally, the conclusions are discussed.

Formulation the Hybrid Proposed Model

Despite the numerous classification models available, the accuracy is fundamental to many decision processes, and hence, never research into ways of improving the effectiveness of the classification models has been given up. Many researchers have combined the predictions of multiple classifiers to produce a better classifier and yield accurate results (Chen et al., 2011). The effectiveness of a hybrid relies on the extent to which its classifiers make different errors, or are error independent. Errors come from four aspects, that is, different data sampling methods, different parameter settings, different classifiers, and different combination strategies (Amanda, 1999). By means of the combined predictions of several classifiers, a better performance than that of any of the individual classifiers is sought. Breiman (1999) refered to multiple experts of classifiers that have demonstrated the potential to reduce the generalization error of a classifier model from 5% to 70%. In the other hand, multiple classifiers may provide more accurate classification results than single classifier.

In this paper, a hybrid classification model of multilayer perceptrons (MLP) is proposed in order to yield more accurate results using the multiple linear regression (MLR) models. The main aim of the proposed model is to use the unique advantages of the MLR models in linear modeling in order to overcome the linear limitation of the traditional multilayer perceptrons. Therefore, in the first phase of the proposed model, a multiple linear regression model is used in order to magnify the linear components in the attributes for better modeling by MLP in the second phase. Then the magnified linear components are summarized in a new attribute as L ($n+1^{th}$ attribute). The main goal of using the multiple linear regression models is to evaluate the relationship between attributes as independent or predictor variables and class value as dependent variable. This is done by fitting a straight line to a number of observations. Specifically, a line is produced so that the squared deviations of the observed points from that line are minimized. Thus, this procedure is generally referred to as least squares estimation (sahoo et al., 2009). Mathematically, if the class value is linearity dependent on the values of their attributes, then a multiple regression model is as follows:

$$L = \alpha_0 + \alpha_1 x_1 + \alpha_2 x_2 + ... + \alpha_n x_n = \sum_{i=0}^{n} \alpha_i x_i , \tag{1}$$

Where x_i $(i = 0,1,2,...,n)$ are attributes and α_i $(i = 0,1,2,...,n)$ are unknown coefficients that are estimated by the least squares method. Then, in the second phase, a multilayer perceptron is used in order to jointly model both linear and nonlinear structures and classify using original attributes and a generated linear attribute by multiple linear regression as follows:

$$\begin{aligned} y_t &= w_0 + \sum_{j=1}^{q} w_j \cdot g(w_{0,j} + \sum_{i=1}^{n} w_{i,j} \cdot x_{t,i} + w_{n+1,j} \cdot x_{t,n+1}) + \varepsilon_t \\ &= \sum_{j=0}^{q} w_j \cdot g(w_{0,j} + \sum_{i=1}^{n+1} w_{i,j} \cdot x_{t,i}) + \varepsilon_t , \end{aligned} \tag{2}$$

where, $g(w_{0,0} + \sum_{i=1}^{n+1} w_{i,0} \cdot x_{t,i}) = 1$, w_j $(j = 0,1,2,...,q)$ and $w_{i,j}$ $(i = 0,1,2,...,n+1, \quad j = 0,1,2,...,q)$ are connection weights, n+1 is the number of the all attributes (input nodes), and q is the number of hidden nodes. The architecture of the proposed hybrid model is shown in Fig. 1.

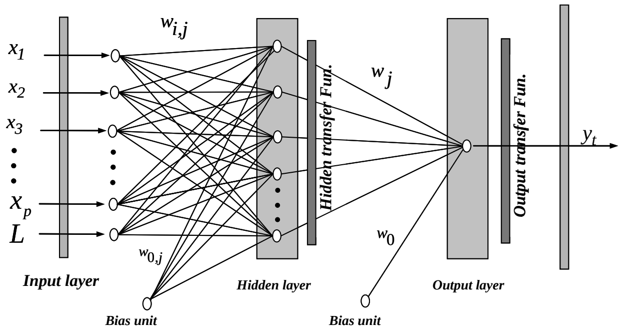

FIG. 1 ARTICHITECTURE OF THE HYBRID PROPOSED MODEL

Hierarchical Proposed Model for Multiple Class Classification

The outputs of the proposed model are continuous, whereas outputs of the classification models are discrete. However, classification can also be viewed as the process of drawing a partition between classes (Khashei et al., 2012). The proposed model can be used to approximate a function that identifies this partition. Our proposed model does not assume the shape of the partition, unlike the linear and quadratic discriminant analysis. Additionally, in contrast to the K-nearest neighbor method, the proposed model does not require storage of training data. Once the model has been trained, it performs much faster than K-nearest neighbor does, because it does not need to iterate through individual training samples. The proposed model does not require experimentation and also final selection of a kernel function and a penalty parameter as is required by the support vector machines. Our proposed model solely relies on a training process in order to identify the final classifier model.

Finally, the proposed model dose not has the mixed results in linear problems and also, in problems that consist both linear and nonlinear correlation structures as traditional multilayer perceptrons. Therefore, in order to apply the proposed model to classification, certain modifications to the model needed to be made. Similar to other models, continuous output of the proposed model is converted to a discrete class by assigning a sample to the class to which the output was closest. Each class is assigned a numeric value. The difference between the output and each numeric value is then calculated, and the sample is put in the class with which its output has the smallest difference. Then, the hierarchical schema of the proposed model is introduced for multiple class classification.

The additional complexity inherent in multiple class classification problems presents a challenge to many classification models. An approach that has been commonly used in order to improve multiple class performance of classifiers is hierarchical models. Generally, reasons for using hierarchical classifiers focus on reducing complexity yield more accurate results. Porter and Liu (1996) described hierarchical classifiers as a subset of modular classifiers. They suggest that modular classifiers often arise when a combination of factors include a large number of classes, classes have difficult shapes (are not compact, convex, or connected), classes do not have distinct boundaries, boundaries are highly nonlinear, and misclassification of some points carries a high penalty. Lee and Ersoy (2007) described hierarchical classification as a way in which data is detected which are more difficult to classify in order to distinguish these data.

In this paper, three different approaches, namely "one versus one"; "one versus rest"; and "one versus all" are examined in order to develop a hierarchical version of the proposed model following the aforementioned reasoning. In all of these approaches it is postulated that if a class is removed from the data set, the remaining classes may become easier to classify without the influence of the removed class. On the other hand, in order to simplify the training and improve the performance, multiple class classification problems can be broken down into several two-class data sets. The modeling cost of the "one versus one" approach in order to develop of the hierarchical proposed model is too high. For a case of k classes, the "one versus one" approach needs $\binom{k}{2} = \frac{k!}{(k-2)!\ 2!}$ $= \frac{k(k-1)}{2}$ two-class classifiers, while the "one versus rest", and the "one versus all"; approaches only need $k-1$ and k two-class classifiers, respectively. Therefore using the "one versus one" approach for developing the hierarchical proposed model is not reasonable.

In the "one versus rest" approach, for a case of k classes, a class from these k classes is first considered as a category, and the rest $k-1$ classes as another category, and a two-class classifier is constructed. Next, this class is excluded, and then the described process is repeated for a case of $k-1$ classes. On the other hand, a class from remaining $k-1$ classes is considered as a category, and the rest $k-2=(k-1)-1$ classes as another category, and a second two-class classifier is constructed, and so on and so forth till the last two-class classifier is constructed. In this way, $k-1$ two-class classifier must be constructed in all for a case of k classes. The "one versus all" approach is similar to the "one versus rest" approach with a bit difference. In the "one versus all" approach, for a case of k classes, a class from these k classes is also considered as a category, and the rest $k-1$ classes as another category, and a two-class classifier is constructed; however, this class is not excluded. In this way, k two-class classifier must be constructed in all for a case of k classes.

Gene Expression Microarray Data Set

In this section, the gene expression microarray data is applied in order to examine the performance of the proposed model in incomplete data conditions. This data set is one of the most well-known data with the "high dimension small sample" characteristic in the field of classification problems. In addition, this data set has been used in multiple published studies to assess classification performance of various classification models.

Structure of Microarray Data

A microarray is a small chip onto which a large number of DNA molecules (probes) are attached in fixed grids. The chip is made of chemically coated glass, nylon, membrane or silicon. Each grid cell of a microarray chip corresponds to a DNA sequence. For cDNA microarray experiment, the first step is to extract RNA from a tissue sample and amplification of RNA. Thereafter two mRNA samples are reverse-transcribed into cDNA (targets) labeled using different fluorescent dyes (red-fluorescent dye Cy5 and green fluorescent dye Cy3) (Saha et al., 2011). Due to the complementary nature of the base-pairs, the cDNA binds to the specific oligonucleotides on the array. In the subsequent stage, the dye is excited by a laser so that the amount of cDNA can be quantified by measuring the fluorescence intensities. The log ratio of two intensities of each dye is used as the gene expression profiles:

$$Gene\ \exp ression\ level = \log_2 \frac{Intensity(Cy5)}{Intensity(Cy3)} \tag{3}$$

A microarray experiment typically measures the expression levels of large number of genes across different experimental conditions or time points. A microarray gene expression data consisting of n genes and m conditions can expressed as a real valued $n \times m$ matrix $M=[g_{i,j}], i=1,2,...,n; j=1,2,...,m$. Here each element $g_{i,j}$ represents the expression level of the ith gene at the jth experimental condition or time point (Fig. 2) (Maulik & Mukhopadhyay, 2010).

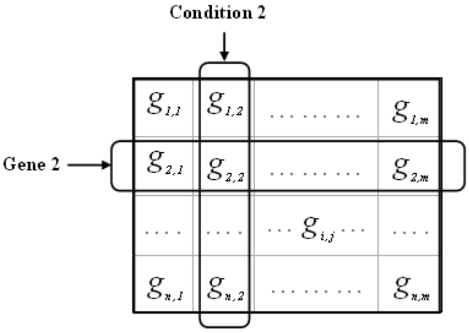

FIG. 2 GENE EXPRESSION MATRIX

Data Set and Pre-processing

In this section, the gene expression data that used in order to show the appropriateness and effectiveness of the proposed model for classification problems with scant data is introduced. The full data set include 6118 genes; however, only 477 of these are classified into seven temporal expression patterns (Liang & Kelman, 2005). The raw data included measurements of green signal, green background, red signal, and red background at seven time points: 0, 0.5, 3, 6, 7, 9, and 11.5 h. Chu et al. (1998) divided the genes into seven temporal classes of induced transcription, reflecting the sequential activation of genes during the program of the experiment, and for each class a model expression profile is derived from a representative set of genes. They call these patterns Metabolic, Early I, Early II, Early-Mid, Middle, Mid-Late and Mid-Late. The patterns are shown in Fig. 3.

Looking at the behavior of these model profiles, we can identify two qualitative shapes, corresponding to the [Increasing Decreasing] and [Increasing] clusters (Mualik et al., 2011). The first one groups the Metabolic, Early I, and Early II genes, and the other the Mid-Late and Late genes. The genes identified as Early-Mid and Middle present a qualitative profile that may be considered as potentially belonging to both clusters.

The particular distribution of the genes is shown in Table 1. Notice that the [Increasing Decreasing] cluster groups up to 93% of Early and Metabolic genes, while the [Increasing] cluster groups up to 80% of Late genes.

Liang and Keleman (2005) examined the use of a regularized neural network for classification of genes into expression patterns based on a series of microarray measurements at seven time points. These expression patterns are useful in studying the function of individual genes, as genes expressed at similar times are often involved in the same or related functions.

The neural networks tested by Liang and Keleman (2005) are feed-forward, back-propagation neural networks with a single hidden layer with five to twenty neurons.

The regularization is carried out in order to smooth the data and involved modification of the cost function is derived with a penalty term for complexity. Liang and Keleman (2005) used a gene expression data set that is derived experimentally by Chu et al. (1998). As described by Liang and Keleman (2005), in this paper, the data for each time point is transformed as follows:

$$X_t = Log\left[\frac{\mathrm{Re}\,d\ signal - \mathrm{Re}\,d\ background}{Green\ signal - Green\,background}\right] \tag{4}$$

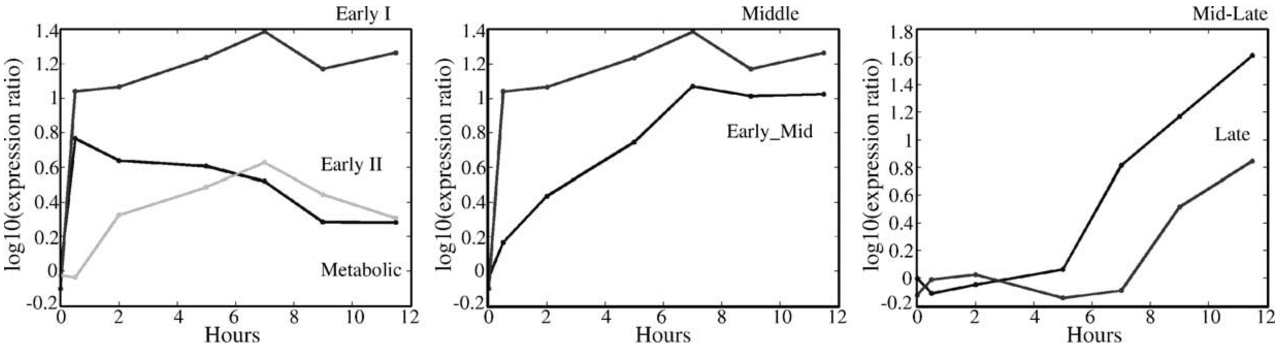

FIG. 3 THE PATTERNS OF GENE EXPRESSION DATA

TABLE 1 DISTRIBUTION OF GENE EXPRESSION DATA IN TWO CLASSES [INCREASING DECREASING] AND [INCREASING]

	Metabolic	Early I	Early II	Early-Mid	Middle	Mid-Late	Mid-Late
Number of gene in each class	52	61	45	95	158	61	5
Increasing–Decreasing	75.5%	60.7%	93.3%	51.6%	44.3%	14.8%	0.0%
Increasing	0.0%	1.6%	0.0%	37.9%	34.8%	70.5%	80.0%

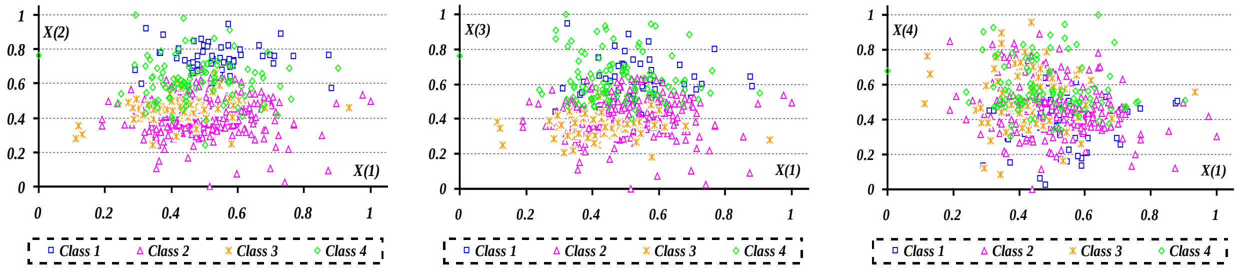

FIG. 4 THE TWO DIMENSIONAL DISTRIBUTION OF GENE EXPRESSION DATA CLASSES

This data set is randomly divided into training and test data sets. As Liang and Keleman (2005) used two third of the samples for training and one third for testing, we do the same. In the process of splitting the data, it is found out that one of the classes, late class, has only five samples. As this is too few samples for the classification models to work properly, the classes are combined together based on the most similarities in their attribute values in four classes. The resulting four classes a Early (Early I and Early II), Middle (Early-Mid and Middle), Late (Mid-Late and Late), and Metabolic. The two-dimensional distribution of these four classes against the (X_1, X_2), (X_1, X_3), and (X_1, X_4), as example, is shown in Fig. 4.

Comparative Assessment of the Gene Expression Data Classification

This data set is divided into a training set and a test set, and each model is applied accordingly. The classification performance of the proposed model is compared with traditional multilayer perceptrons (MLPs) and also some other well-known statistical and intelligent classification models such as linear discriminant analysis (LDA), quadratic discriminant analysis (QDA), K-nearest neighbor (KNN), and support vector machines (SVMs).

Application of the Proposed Model to Gene Expression Data Classification

According to the process of the proposed model, in the first stage, a multiple linear regression model is modeled and the magnified linear component of MLR, is then summarized in eighth attribute (L) of the model. In the second stage, a multilayer perceptron is designed in order to jointly model both linear and nonlinear structures existing in the original attributes and a generated linear attribute by MLR model. In order to obtain the optimum network architecture of the proposed model, based on the concepts of multilayer perceptrons design (Khashei at el., 2011) and using pruning algorithms in MATLAB 7 package software, different network architectures are evaluated to compare the MLPs performance. The best fitted network which is selected, and therefore, the architecture which presents the best accuracy with the test data, is composed of eight inputs, six hidden and one output neurons (in abbreviated form, N(8-6-1)). The architecture of the proposed model is shown in Fig. 5. The misclassification percentages of the each model and improvement percentages of the proposed model in comparison with those of other classification models in both training and test data sets are summarized in Table 2 and Table 3, respectively. The hierarchical proposed model is constructed according to the "one versus all" approach because of better results in comparison with "one versus rest" approach.

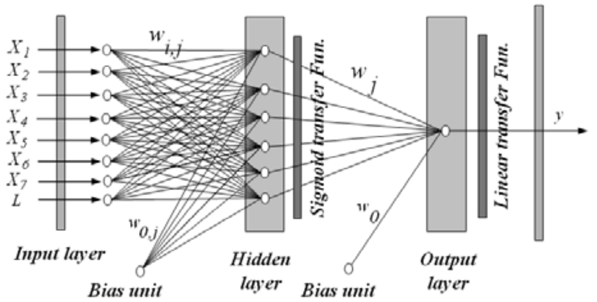

FIG. 5 STRUCTURE OF THE BEST FITTED PROPOSED MODE N(8-6-1)

TABLE 2 GENE EXPRESSION DATA SET CLASSIFICATION RESULTS

Model	Classification error (%)	
	Training Data	Test Data
Linear Discriminant Analysis (LDA) [c=0]	14.3	17.2
Quadratic Discriminant Analysis (QDA) [c=0]	9.7	13.4
K-Nearest Neighbor (KNN) [K=6]	7.8	10.2
Support Vector Machines (SVM) [C=3220]	4.1	5.7
Multi-Layer Perceptrons (MLP) [N(7-13-9-1)]	5.3	18.0
Hybrid proposed model (non-hierarchical) [N(8-6-1)]	5.3	6.3
Hybrid proposed model (hierarchical) [N(8-6-1)]	1.9	3.8

TABLE 3 PERCENTAGE IMPROVEMENT OF THE HIERARCHICAL PROPOSED MODEL IN CMPARISON WITH THOSE OF OTHER CLASSIFICATION MPDELS

Model	Improvement (%)	
	Training Data	Test Data
Linear Discriminant Analysis (LDA)	86.71	77.91
Quadratic Discriminant Analysis (QDA)	80.41	71.64
K-Nearest Neighbor (KNN)	75.64	62.75
Support Vector Machines (SVM)	53.66	33.33
Multi-Layer Perceptrons (MLP)	64.15	78.89

Comparison with other Classification Models

According to the obtained results (Tables 2 & 3), our hierarchical proposed model has the lowest error on the training and test portions of the gene expression data in comparison to those used classification models, with a misclassification rate of 1.9% and 3.8% in the training and test samples, respectively. Several different architectures of multilayer perceptron are designed and examined. The best performing architecture for a traditional multilayer perceptron (N(7-13-9-1)) produces a 5.3% and 18.0% error rate in training and test samples, in which the hierarchical proposed model improves by 64.15% and 78.89%, respectively. Support vector machine (SVM) with C=3220 performs second best with an error rate of 5.7% on the test portion of the data, 33.33% higher than the hierarchical proposed model error rate. K-nearest neighbor (KNN) with k=6 has an error rate of 10.2%, 62.75% higher than that for the hierarchical proposed model. Quadratic discriminant analysis (QDA) with c=0 and linear discriminant analysis (LDA) also with c=0, have error rates of 13.4% and 17.2%, which is 71.64% and 77.91% higher than that of the hierarchical proposed model, respectively.

Conclusions

Monitoring the behavior of gene expression over certain time plays an important role in exploring and investigating regulation of gene expression. Analysis of time-course gene expression data can potentially provide more insights into the dynamic biological systems. Various classification models have been used for gene expression data, which is collected over time in order to identify groups. In biological researches, it is important to find an optimal classification of genes or samples by assessing and comparing various methods. In this paper, the multiple linear regression models and multilayer perceptrons which are one of the most accurate and widely used linear and nonlinear classification techniques; respectively, are combined together in order to construct a new hybrid model. The main aim of the proposed model is to overcome the linear deficiency of MLP models and yield a more accurate classification model than traditional multilayer perceptrons. Linear discriminant analysis (LDA), quadratic discriminant analysis (QDA), K-Nearest neighbor (KNN), support vector machines (SVM) and traditional multilayer perceptrons (MLP), as well-known statistical and intelligent classification models, have been used to compare the classification performances. Empirical results indicate that the proposed model has the lowest error on the training and test portions of the gene expression data in comparison to other those classification models. Therefore, it can be applied as an appropriate approach for classification of gene expression data, especially when higher classification accuracy is needed.

ACKNOWLEDGEMENTS

The authors wished to express their gratitude to Dr. G. A. Raissi, industrial engineering department, Isfahan University of Technology who greatly helped us.

REFERENCES

Amanda, J., C. Combining artificial neural nets: Ensemble and modular multinet systems. London: Springer, 1999.

Breiman, L., "Prediction games and arcing algorithm. "Neural Computation 11 (1999): 1493–1517.

Chen, S., Lin, S., & Chou, S. "Enhancing the classification accuracy by scatter search based ensemble approach." Applied Soft Computing 11 (2011): 1021- 1028.

Chu, S. "The transcriptional program of sporulation in budding yeast." Science 282 (1998): 699– 705.

Fraley, C., Raftery, A.E., "Model-based clustering, discriminant analysis and density estimation." J. Am. Stat. Assoc 97 (2002): 611–631.

Gui, J., Li, H. "Mixture functional discriminant analysis for gene function classification based on time course gene expression data." Paper presented at Proc. Joint Stat. Meeting (Biometric Section), 2003.

Iranfar, N., Fuller, D., Sasik, R., Hwa, T., Laub, M., Loomis, W.F. "Expression patterns of cell-type specific genes in Dictyostelium." Mol. Biol. Cell 12 (2001): 2590–2600.

Khashei, M. Bijari, M. "A novel hybridization of artificial neural networks and ARIMA models for time series forecasting" Applied Soft Computing 11(2011): 2664– 26.

Khashei, M., Bijari, M., & Raissi, GH A. "Improvement of auto-regressive integrated moving average models using fuzzy logic and artificial neural networks (ANNs)." Neurocomputing 72 (2009): 956–967.

Khashei, M., Hamedani, A., Bijari, M. "A novel hybrid classification model of artificial neural networks and multiple linear regression models." Expert Systems with Applications 39 (2012): 2606–2620.

Lee, J., & Ersoy, O. "Consensual and hierarchical classification of remotely sensed multispectral images." IEEE Transactions on Geoscience and Remote Sensing 45 (2007): 2953–2963.

Leng, X., Müller, H.-G. "Classification using functional data analysis for temporal gene expression data." Bioinformatics 22 (2006): 68–76.

Liang, Y., Kelemen, A. "Temporal gene expression classification with regularized neural network." Int. J. Bioinformatics Res. Appl. 1 (2005): 399–413.

Luan, Y., Li, H. "Clustering of time-course gene expression data using amixedeffects model with B-splines." Bioinformatics 19 (2005): 474–482.

Maulik,U. "Analysis of gene microarray data in a soft computing framework." Applied Soft Computing 11 (2011): 4152– 4160.

Maulik, U. A. Mukhopadhyay "Simulated annealing based automatic fuzzy clustering combined with ANN classification for analyzing microarray data." Comput. Oper. Res. 37 (2010): 1369–1380.

Park, L., Koo, J., Kim, S., Sohn, I.,Won Lee, J. "Classification of gene functions using support vector machine for time-course gene expression data." Computational Statistics & Data Analysis 52 (2008): 2578 – 2587.

Porter, W., & Liu, W. "Fuzzy HMC classifiers." Information Sciences 94 (1996): 151–178.

Raychaudhuri, S., Sutphin P. D., Chang, J. T., Altman, R. B. "Basic microarray analysis: grouping and feature reduction." TRENDS in Biotechnology 19 (2001): 189-193.

Saha, I. U. Maulik, S. Bandyopadhyay, D. Plewczynski. "Improvement of new nautomatic differential fuzzy clustering using SVM classifier for microarray analysis." Expert Syst. Appl 38 (2011): 15122–15133.

Sahoo, G., Schladow, S., & Reuter, J. "Forecasting stream water temperature using regression analysis, artificial neural network, and chaotic non-linear dynamic models." Journal of Hydrology 378 (2009): 325–342.

Spellman, P.T., Sherlock, G., Zhang, M.Q., Iyer, V.R., Anders, K., Eisen, M.B., Brown, P.O., Botstein, D., Futcher, B. "Comprehensive identification of cell cycle-regulated genes of the yeast Saccharomyces cerevisiae by microarray hybridization." Mol. Biol. Cell 9 (1998): 3273–3797.

Song, J. J., Lee, H.-J., Morris, J.S., Kang, S. "Clustering of time-course gene expression data using functional data analysis." Comput. Biol. Chem. 31 (2007): 265– 274.