International Journal on Recent and Innovation Trends in Computing and Communication ISSN: 2321-8169
Volume: 4 Issue: 2 221 - 224

Classification of Acute Leukemia using Fuzzy Neural Networks

Dr. B. B. M. Krishna Kanth
Principal, Hindu College of Engineering& technology
Guntur, India
bbkkanth@yahoo.com

Dr. B. G. V. Giridhar
Department of Endocrynology,Andhra Medical College
Visakhapatnam,India
kanthkrishna978@gmail.com

***Abstract*—** Accurate classification of cancers based on microarray gene expressions is very important for doctors to choose a proper treatment. In this paper, we compared ten filter based gene selection methods in order to differentiate acute lymphoblastic leukemia (ALL) and acute myeloid leukemia (AML) in leukemia dataset. Dimensionality reduction methods, such as Spearman Correlation Coefficient and Wilcoxon Rank Sum Statistics are used for gene selection. The experimental results showed that the proposed gene selection methods are efficient, effective, and robust in identifying differentially expressed genes. Adopting the existing SVM-based and KNN-based classifiers, the selected genes by filter based methods in general give more accurate classification results, typically when the sample class sizes in the training dataset are unbalanced.

Keywords- *microarrays; gene expression data; gene selection; classification;*

*****

I. INTRODUCTION

Microarrays [1], also known as gene chips or DNA chips, provide a convenient way of obtaining gene expression levels for a large number of genes simultaneously. Each spot on a microarray chip contains the clone of a gene from a tissue sample. Some mRNA samples are labeled with two different kinds of dyes, for example, Cy5 (red) and Cy3 (blue). After mRNA interact with the genes, i.e., hybridization, the color of each spot on the chip will change. The resulted image reflects the characteristics of the tissue at the molecular level. Microarrays can thus be used to help classify and predict different types of cancers. Traditional methods for diagnosis of cancers are mainly based on the morphological appearances of the cancers; however, sometimes it is extremely difficult to find clear distinctions between some types of cancers according to their appearances. Hence the microarray technology stands to provide a more quantitative means for cancer diagnosis. For example, gene expression data have been used to obtain good results in the classifications of lymphoma, leukemia [2], breast cancer, and liver cancer. It is challenging to use gene expression data for cancer classification because of the following two special aspects of gene expression data. First, gene expression data are usually very high dimensional. The dimensionality ranges from several thousands to over ten thousands. Second, gene expression data sets usually contain relatively small numbers of samples, e.g., a few tens. If we treat this pattern recognition problem with supervised machine learning approaches, we need to deal with the shortage of training samples and high dimensional input features.

Recent approaches to solve this problem include unsupervised methods, such as Clustering [3], and Self-Organizing Maps (SOM)[4] and supervised methods, such as Support Vector Machines (SVM)[5], Multi-Layer Perceptrons (MLP)[6 7], K-Nearest Neighbor(KNN) method[8 9], and Decision Trees (DT)[7]. But most of the current methods in microarray analysis can not completely bring out the hidden information in the data. Meanwhile, they are generally lacking robustness with respect to noisy and missing data. Some studies have shown that a small collection of genes [10] selected correctly can lead to good classification results [11]. Therefore gene selection is crucial in molecular classification of cancer. Although most of the algorithms mentioned above can reach high prediction rate, any misclassification of the disease is still intolerable in acute leukemia's treatment. Therefore the demand of a reliable classifier which gives 100% accuracy in predicting the type of cancer therewith becomes urgent.

We first experiment the classifiers with 38 leukemia samples and test the classifier with another 34 samples to obtain the accuracy rate. Meanwhile, this study reveals that the classification result is greatly affected by the correlativity with the class distinction in the data set. The remainder of the paper is organized as follows. The Gene selection methods for choosing effective predictive genes in our work are introduced in Section 2. Then Sections 3 gives a brief introduction for the architecture of the MFSHNN, followed by its learning algorithm in section 4. Section 5 examines the experimental results of the classifiers operated on leukemia data set. Conclusions are made in Section 6.

II. GENE SELECTION METHODS

Among the large number of genes, only a small part may benefit the correct classification of cancers. The rest of the genes have little impact on the classification. Even worse, some genes may act as noise and undermine the classification accuracy. Hence, to obtain good classification accuracy, we need to pick out the genes that benefit the classification most.

A. Spearman Correlation Coefficient Gene Selecton Method

In order to score the similarity of each gene, an ideal feature vector [13] is defined. It is a vector consisting of 0 in one class and 1 in other class. It is defined as follows:

$$ideal_i = (0,0,0,0,0,0,1,1,1,1,1,1)$$

The ideal feature vector is highly correlated to a class. If the genes are similar with the ideal vector (the distance from the ideal vector and the gene is small.), we consider that the genes

are informative for classification. The similarity of g_i and g_{ideal} using similarity measures such as the Spearman coefficient (SC)

$$SC = 1 - \frac{6\sum_{i=1}^{n}\left(ideal_i - g_i\right)^2}{n\times\left(n^2-1\right)} \quad (1)$$

Where n is the number of samples; μ_g is the mean of the gene and μ_{ideal} is the mean of ideal feature vector, g_i is the i_{th} real value of the gene vector and $ideal_i$ is the corresponding i_{th} binary value of the ideal feature vector.

B. Wilcoxon Rank Sum Test Gene Selection Method

The Wilcoxon rank-sum test [14, 15] is a big category of non-parametric tests. The general idea is that, instead of using the original observed data, we can list the data in the value ascending order, and assign each data item a rank, which is the place of the item in the sorted list. Then, the ranks are used in the analysis. Using the ranks instead of the original observed data makes the rank sum test much less sensitive to outliers and noises than the classical (parametric) tests. The WRST organizes the observed data in value ascending order. Each data item is assigned a rank corresponding to its place in the sorted list. These ranks, rather than the original observed values, are then used in the subsequent analysis. The major steps in applying the Wilcoxon rank-sum test are as follows:

(i) Merge all observations from the two classes and rank them in value ascending order.

(ii) Calculate the Wilcoxon statistics by adding all the ranks associated with the observations from the class with a smaller number of observations.

III. MODIFIED FUZZY HYPERSPHERE NEURAL NETWORK CLASSIFIER

The MFHSNN consists of four layers as shown in Figure 1(a). The first, second, third and fourth layer is denoted as F_R, F_M, F_N and F_O respectively. The F_R layer accepts an input pattern and consists of n processing elements, one for each dimension of the pattern. The F_M layer consists of q processing nodes that are constructed during training and each node represents hypersphere fuzzy set characterized by hypersphere membership function. The processing performed by each node of F_M layer is shown in Figure 1(b). The weights between F_R and F_M layer represent centre points of the hyperspheres. As shown in Figure 1(b), $C_j = \left(c_{j1}, c_{j2}, c_{j3}.....c_{jn}\right)$ represents center point of the hypersphere m_j. In addition to this each hypersphere takes one more input denoted as threshold T, which is set to one and the weight assigned to this link is ζ_j.

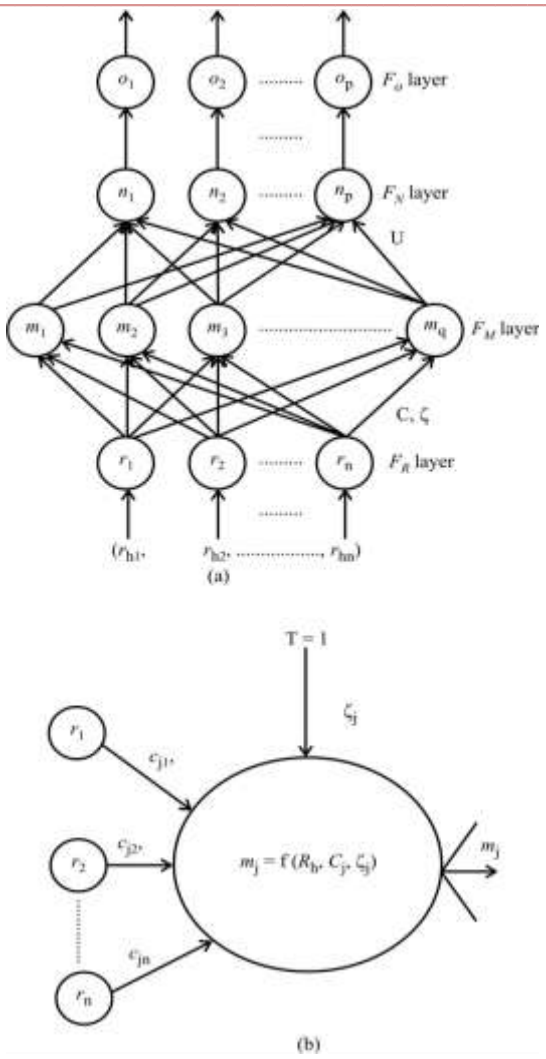

Figure 1(a) Modified Fuzzy hypersphere neural network,

Figure 1(b) Implementation of fuzzy hypersphere

The ζ_j represents radius of the hypersphere m_j, which is updated during training. The center points and radii of the hyperspheres are stored in matrix C and vector ξ respectively. The maximum size of hypersphere is bounded by a user defined value λ, where $0 \le \lambda \le 1$. The λ is called as growth parameter that is used for controlling maximum size of the hypersphere and it puts maximum limit on the radius of the hypersphere. Assuming the training set defined as $R \in \{R_h | h = 1,2,.....P\}$, where $R_h = \left(r_{h1}, r_{h2}, r_{h3}.....r_{hn}\right) \in I^n$ is the h_{th} pattern, the membership function of the hypersphere node m_j is

$$m_j\left(R_h, C_j, \zeta_j\right) = 1 - f\left(l, \zeta_j, \gamma\right) \quad (8)$$

where $f(\,)$ is three-parameter ramp threshold function defined as

$$f\left(l,\zeta_j,\lambda\right)=\begin{bmatrix}0,\ if\ \ (0\le l\le\zeta_j)\\(l-\zeta_j)\gamma,\ if\ \ (\zeta_j\le l\le 1)\\1,\ if\ \ (l\ge 1)\end{bmatrix} \quad (9)$$

and the argument l is defined as,

$$l=\left(\sum_{i=1}^{n}\left(c_{ji}-r_{hi}\right)^2\right)^{1/2} \quad (10)$$

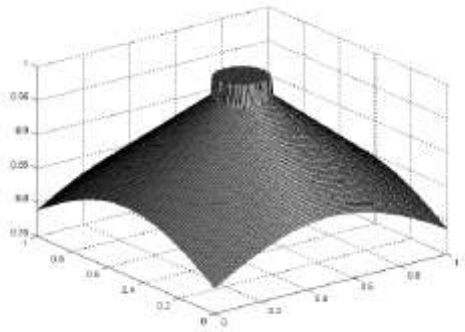

Figure2 (a) Plot of fuzzy hypersphere membership function for γ = 1

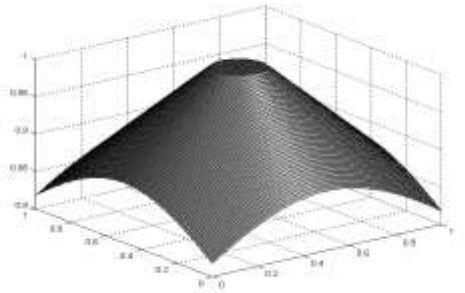

Figure2 (b) Plot of modified fuzzy hypersphere membership function for γ = 1

The membership function returns m_j =1, if the input pattern R_h is contained by the hypersphere. The parameter γ, $0\le\gamma\le 1$, is a sensitivity parameter, which governs how fast the membership value decreases R_h outside the hypersphere when the distance between R_h and C_j increases. The sample plot of membership function for FHSNN with centre point [0.5 0.5] and radius equal to 0.3 is shown in Figure 2(a) and the sample plot of membership function for MFHSNN with the same centre point and radius is shown in figure 2(b). The membership function of MFHSNN is improved. It can be observed that the membership values decrease steadily with increasing distance from the hypersphere.

Each node of F_N and F_O layer represents a class. The F_N layer gives fuzzy decision and output of k_{th} F_N node represents the degree to which the input pattern belongs to the class n_k. The weights assigned to the connections between F_M and F_N layers are binary values that are stored in matrix U and updated during learning as

$$u_{jk}=\begin{cases}1 \text{ if } m_j \text{ is a hypershere of class } n_k\\0 \text{ otherwise}\end{cases}$$

(11)

For k = 1, 2. . . p and j = 1, 2. . . q

where m_j is the j_{th} F_M node and n_k is the k_{th} F_N node. Each F_N node performs the union of fuzzy values returned by the fuzzy set hyperspheres of same class, which is described by equation (12)

$$n_k=\max_{j=1}^{q} m_j u_{jk} \text{ for } k=1,2,.....p$$

Each F_O node delivers non-fuzzy output, which is described by equation (13).

$$o_k=\begin{cases}0 \text{ if } n_k\le T\\1 \text{ if } n_k=T\end{cases} \quad \text{for } k = 1,2,3,....p$$

Where $T=\max\left(n_k\right)$, for k=1, 2, 3...p.

IV. EXPERIMENT AL RESULTS

Dataset that we have used is a collection of expression measurements reported by Golub et al [2]. Gene expression profiles have been constructed from 72 people who have either acute lymphoblastic leukemia (ALL) or acute myeloid leukemia (AML). Each person has submitted one sample of DNA microarray, so that the database consists of 72 samples. Each sample is composed of 7129 gene expressions, and finally the whole database is a 7129 X 72 matrix. The number of training samples in ALL/AML dataset is 38 which of them contain 27 samples of ALL class and 11 samples of AML class; here we randomly applied the training samples to the MFSHNN classifier. The number of testing samples is 34 where 20 samples belong to ALL and remaining 14 samples belongs to AML class respectively. This well-known dataset often serves as bench mark for microarray analysis methods. Before the classification, we need to find out informative genes (features) that are related to predict the cancer class out of 7129.

In this data set, we first ranked the importance of all the genes with respect to SC and WRST gene selection methods. We picked out top 10 genes with the largest values to do the classification. We input these genes one by one to the classifiers according to their ranks. That is, we first input the gene ranked No.1 then; we trained the classifiers with the training data and tested the classifiers with the testing data. After that, we repeated the whole process with top 2 genes, and then top 3 genes, and so on. All source codes were

implemented with MATLAB 7.1 and experiments were conducted on a Pentium IV IBM Laptop with 1.5 GHz Centrino processor and 1GB RAM.main memory.

It should also be noted that this high classification accuracy has been obtained using only two genes with Gene id's 4847 and 1882 which are selected by using Spearman correlation and Wilcoxon rank sum test gene selection methods. But traditional classifiers such as Support vector machine and K-nearest neighbor produced the best accuracy of 97.1% only using all the top 10 genes. As shown from Table 1 the average training time and testing time of MFHSNN classifier is in the range of 0.25 -0.39 seconds which is very fast compared to any other classifier published so far. Meanwhile the average training and testing time of SVM and KNN classifiers is around 2.60-3.5 seconds respectively which is very slow comparative to MFHSNN classifier.

TABLE 1**. Comparison of training and testing time for the three classifiers**

Classifier	**Average Training time (seconds)**	**Average Testing time (seconds)**
MFHSNN	0.25	0.35
KNN	2.60	2.65
SVM	3.20	3.50

The average classification accuracy of the three classifiers for all the gene selection methods used in this paper is shown in Table 2. The highest average classification accuracy achieved by MFHSNN is 97.94% which clearly dominates other classifiers published so far.

TABLE 2**. Average classification accuracy.**

Gene selection\Classifier	**MFHSNN**	**KNN**	**SVM**
Wilcoxon Rank Sum Test	97.64	87.63	81.17
Spearman Coefficient	97.94	87.04	76.47

V. CONCLUSIONS

In order to predict the class of cancer, we have demonstrated the effectiveness of the MFHSNN classifier on Leukemia data set using an informative genes extracted by methods based on their correlation with the class distinction, and statistical analysis. Experimental results show that the MFHSNN classifier is the most effective in classifying the type of leukemia cancer using only two of the most informative genes. MFHSNN yields 100% recognition accuracy and is well suited for the ALL/AML classification in cancer treatment. By comparing the performance with previous publications that used the same dataset, we confirmed that the proposed method provided the competitive, state-of-the-art results. Under the same context, it not only leads to better classification accuracies, but also has higher stability and speed. The training and testing time of MFSHNN is less than 0.4 seconds which will further drastically reduce if the proposed classifier is implemented in hardware.

VI. REFERENCES

[1] M. Schena, D. Shalon, R. W. Davis and P. O. Brown, Quantitative monitoring of gene expression patterns with a complementary DNA microarray, *Science* 267 (1995) 467–470.

[2] T. R. Golub, D. K. Slonim, P. Tamayo, C. Huard, M. Gaasenbeek, J. P. Mesirov, H. Coller, M. L. Loh, J. R. Downing, M. A. Caligiuri, C. D. Bloomfield and E. S. Lander, "Molecular Classification of Cancer: Class Discovery and Class Prediction by Gene Expression Monitoring," *Science*, vol. 286, pp. 531–537, 1999.

[3] R. Baumgartner, C. Windischberger, and E. Moser, "Quantification in functional magnetic resonance imaging: fuzzy clustering vs. correlation analysis." *Magn Reson Imaging*, vol. 16, no. 2, pp. 115–125, 1998.

[4] T. Kohonen, Ed., *Self-organizing maps*. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 1997.

[5] T. S. Furey, N. Cristianini, N. Duffy, D. W. Bednarski, M. Schummer, and D. Haussler, "Support vector machine classification and validation of cancer tissue samples using microarray expression data," *Bioinformatics*, vol. 16, pp. 906–914, 2000.

[6] J. Khan, J. S. Wei, M. Ringner, L. H. Saal, M. Ladanyi, F. Westermann, F. Berthold, M. Schwab, C. R. Antonescu, C. Peterson, and P. S. Meltzer, "Classification and diagnostic prediction of cancers using gene expression profiling and artificial neural networks," *Nature Medicine*, vol. 7, pp. 673–679, 2001.

[7] C. Shi and L. Chen, "Feature dimension reduction for microarray data analysis using locally linear embedding," in *APBC*, 2005, pp. 211–217.

[8] L. Li, C. R. Weinberg, T. A. Darden, and L. G. Pedersen, "Gene selection for sample classification based on gene expression data: study of sensitivity to choice of parameters of the ga/knn method," *Bioinformatics*, vol. 17, pp. 1131–1142, 2001.

[9] T. Jirapech-Umpai and S. Aitken, "Feature selection and classification for microarray data analysis: Evolutionary methods for identifying predictive genes," *Bioinformatics*, vol. 6, pp. 168–174, 2005.

[10] Saeys Y, Inza I, Larranaga P: A review of feature selection techniques in bioinformatics. Bioinformatics 2007, 23(19):2507-2517.

[11] Wang X, Gotoh O. Microarray-Based Cancer Prediction Using Soft Computing Approach. *Cancer Informatics*. 2009; 7: 123–39.