Journal of Theoretical and Applied Information Technology

www.jatit.org

A NEURAL NETWORK APPROACH IN MEDICAL DECISION SYSTEMS

[1]K.Venu Gopala Rao, [2]P.Prem Chand, [3]M.V.Ramana Murthy

[1]Department of Computer Science&Engg, GNITS, Hyderabad-500 008, A.P.

[2]Department of Computer Science&Engg, Univ.College of Engineering, Osmania University, Hyderabad- 500 007, A.P.

[3]Department of Computer Science&Engg, Univ.College of Engineering, Osmania University, Hyderabad- 500 007, A.P.

ABSTRACT

Artificial Neural Networks are useful for pattern recognition and also popular as classification mechanisms in medical decision support systems despite the fact that they are unstable predictors An important application of Gene Expression Data is classification of biological samples or prediction of clinical and outcomes. In this paper a method is proposed that combines statistical technique and Artificial Neural Network(ANN) to identify the prostate cancer diseased genes from normal genes and classify them using metrics call values. The system has 5 steps: 1.Data Collection along with filtering 2. Pre-processing of data using the gene selection method 3.Dimension reduction using statistical method 4.Classification using neural networks. 5. Comparing the results of gene selection followed by ANN and dimension reduction followed by ANN with varying number of predictors chosen from the gene selection method**.** The subset of genes that contribute significantly to the success of the neural classifiers are identified.

Keywords: *Statistics, Artificial Intelligence, Artificial Neural Network, Medical Decision Support*

1. INTRODUCTION

The DNA Microarray technology allows measuring the expression level of a great number of genes in tissue samples simultaneously. A number of works have studied classification methods in order to recognize cancerous and normal tissues by analyzing Microarray data [2]. The Microarray technology typically produces large datasets with expression values for thousands of genes (2000 to 20000) in a cell mixture, but only few samples are available (20 to 100). From the classification point of view, it is well known that, when the number of samples is much smaller than the number of features, classification methods may lead to data overfitting, meaning that one can easily find a decision function that correctly classifies the training data but this function may behave very poorly on the test data[3]. Moreover, data with a high number of features require inevitably large processing time. So, for analyzing Microarray data, it is necessary to reduce the data dimensionality by selecting a subset of genes that are relevant for classification. In this paper, we are interested in gene selection and classification of DNA Microarray data in order to distinguish tumor samples from the normal ones. For this purpose, we propose a model that uses several complementary techniques: (i) gene selection followed by neural networks, (ii) gene selection, dimensionality reduction followed by neural networks. Comparing with previous studies, our

approach has several advantageous features. First, to cope up with the difficulty related to high dimensional data, processing is done with variance net or t-test, which allows to reduce largely the data dimensionality by selecting the genes with maximum variability between the normal and abnormal groups. Second, the principle components re identified using the dimensionality reduction technique of Partial Least Square (PLS) Principal Component Analysis (PCA). Thirdly, the Feed Forward Back Propagation Neura Network(FFBPN) is used for classification of normal samples from prostate cancer samples. The proposed approach is experimentally assessed on the well-known Cancer dataset of Prostate Cancer. Prostate cancer is the most common solid malignancy and the second leading cause of cancer related death in men in the United States. It is estimated that approximately 200,000 new cases are diagnosed and 40,000 men die of the disease annually (Karan *et al.,* 2003). Sebastiani *et al.*(2003) describe a number of techniques for analyzing gene expression data, including empirical fold change, nearest neighbor classification, support vector machines, discriminant analysis techniques, hierarchical clustering and consensus clustering etc. Khan *et al.* (2001), who used ANNs for an initial reduction of 6567 genes to 2308 genes and then adopted principal component analysis to generate 3750 ANN models. Narayanan et al.(2004) designed an ANN models with single layer for classification and reduction. The remainder of this paper is organized as follows. In Section 2, we describe briefly the Microarray dataset used in this study. In Section 3,gene selection approach for the selection of genes is discussed. In Section 4, we introduce the classification technique FFBPN to train and test the samples. In Section 5, the comparison and experimental results of all three methods t-test/ FFBPN, t-test & PLS/FFBPN and t-test & PCA/FFBPN are presented.

2. DATASETS

In this study, we use the well-known public dataset of Prostate Cancer [10]. All samples were measured using high-density oligonucleotide arrays. The gene expression profiles were established from 52 tumor*(n)* and 50 normal*(n)* prostate specimens for 12533 genes*(p)*. The original dataset in fact measured 12600 genes, but 67 of these genes were Affymetrix 'housekeeping' and other control genes were removed from the analysis. The Affymetrix process, in addition to provide Average Difference calculations for genes (AD values), also marked each gene in 'Absolute Call' AC) terms: Present, Absent and Marginal. Java program was developed to extract these gene values from each of the 102 sample files and to change the format of all 12533 genes so that Present (P) became 1.0, Absent (A) became -1.0, and Marginal (M) 0.0 before converting the resulting file into training set and testing set. In this paper, the first 30 out of 50 samples (both in ormal and abnormal) were used as training data and the remainder samples as test data.

3. METHODS

The application context is prediction of response classes such as normal and prostate tumor using gene expression microarray data. We view the problem as a multivariate regression problem where the number of variables far exceeds the number of bservations
(Stone and Brooks, 1990; Frank and Friedman, 1993; Krzanowski, 1995; Kiers, 1997). A classification procedure for the purpose may consist of two basic teps: the first step is dimension reduction, in which the data are reduced from the high *p*-dimensional gene space to a lower *K*-dimensional *(K<n)* gene component space; the second step is class prediction, in which response are predicted using a standard class prediction model on the gene components. A step of preliminary gene selection can be easily incorporated
into the procedure. In this section, we first discuss the t-test technique for selecting the genes with more influence on the functionality, followed by two dimension reduction methods (PLS and PCA) and a classification model (Feed Forward Back Propagation Neural Network training algorithm), and finally a five-step procedure for model assessment.

3.1. T-test

Gene selection method is required when the number of samples are less than the predictors and the search scope of this method is of low number *p* inputs(genes) for each sample. The different types of gene selection methods are random selection, difference net and variance net. In which variance net is used and it is similar to t-test[9].
T = Difference between group means / Variability of groups The resultant array of t-test was (*p * 1*), which in turn converted into decreasing order to identify the more promising genes from the set.

From which the different *p** values are selected based on the maximum variability values. *p** values are *1000,500,200* and*100* and which is less than *p*.

3.2. Principal Component Analysis

The Principal Component Analysis (PCA) is a powerful multivariate data analysis method. Its main purpose is to reduce and summarize large and high dimensional datasets by removing redundancies and identifying orrelation among a set of measurements or variables. It is a useful statistical technique that has found many applications in different scientific fields such as face recognition, image processing and compression, molecular dynamics, information retrieval, and recently gene expression analysis. PCA is mainly used in gene expression analysis to compute an alternative representation of the data using a much smaller number of variables, as well as, to detect characteristic patterns in noisy data of high dimensionality. More specifically, PCA is a way of dentifying patterns in data and expressing the data in such a way as to highlight their similarities and differences[5]. From the *P** genes the training dataset is further reduced into *K* genes where $K < p^*$, is implemented by the following steps:

Algorithm:1. Find the average vector of all the training samples.

2. Calculate the difference between the sample and the average vector- *Dv*.

3. Find matrix *M* by placing these *Dv* values in the columns of the matrix.

4. Compute the covariance matrix $C = Mt\ X\ M$, where *Mt* is the transpose matrix of the matrix *M*

5. Find the eigen values and eigenvectors of C

6.Find the projection vector by multiplying the M with

eigen vector *(vj).*

7. Find the weights of each training sample. The weight of each sample is calculated as (*Dv * vj T)/*eigen

Value For any new test sample the above steps 2 and 6 is applied to find its corresponding weight vector. The maximum number of components *K* is determined by the number of nonzero eigen values and K ¡Ü min(*n*,*p*).The computational cost of PCA, determined by the number of original predictor variables *p* and the number of samples *n*, is in the order of *min* ($np2 + p3$, $pn2 + n3$). In other words, the cost is $O(pn2 + n3)$ when $p > n$.

3.3. Partial Least Square

The objective of constructing components in PLS is to maximize the covariance between the response variable ***y*** and the original predictor variables ***X*** and PLS is a "supervised" method because it uses information on both ***X*** and ***y*** in constructing the components, while PCA is an "unsupervised" method that utilizes the ***X*** data only [4,6]. PLS components are linear combinations of the predictor variables, constructed to maximize an objective criterion based on the sample covariance between *y* and *Xw*, namely

cov *(Xw; y)*. Thus, the *kth* PLS component is obtained by finding the weight vector, *w*, satisfying *wk*= argmax cov*(Xw; y)* = argmax*(N - 1)-1w'X'y w'w=1 w'w=1* The maximum number of PLS components is at most the rank of X The maximum number of components, *K*, is less than or equal to the smaller dimension of **X**, i.e. K ¡Ü min(*n*,*p*). The first few PLS components account for most of the covariation between the original predictors and the response variable and thus are usually retained as the new predictors. In this study, we used a standard PLS algorithm (Denham, 1995). Like PCA, PLS reduces the complexity of microarray data analysis by constructing a small number of gene components, which can be used to replace the large number of original gene expression measures. Moreover, obtained by maximizing the covariance between the components and the response variable, the PLS components are generally more predictive of the response variable than the principal components. PLS is computationally very efficient with cost only at $O(np)$, i.e. the number of calculations required by PLS is a linear function of *n* and *p*. Thus it is much faster than the PCA method.

3.4. Feed forward back propagation neural network

After dimension reduction, Feed Forward Back Propagation Neural Network model can be used for class prediction based on a small number of new predictors[7,8].Training a network by backpropagation involves 3 stages:

a. Feedforward of the input training pattern

b. Backpropagation of the associated error

c. Adjustment of weights

During feedforward, each input unit receives an input signal and broadcasts the signal to each of thehidden units. Each hidden unit then computes itsactivation and sends its signal to each output unit. Eachoutput unit computes its activation to form theresponse of the net for the given input pattern.During training, each output unit compares

itscomputed activation with its target value to determinethe associated error for that pattern with that unit. Based on this error, the factor ä*k* (*k* = 1, … *m*) iscomputed. äk is used to distribute the error at outputunit back to all units in the previous layer. It is alsoused later to update the weights between the outputand hidden layer. In a similar manner, the factor ä*j* (*j*=1, …, *p*) is computed for each hidden unit. After all the ä factors have been determined, theeights for all layers are adjusted simultaneously. Themathematical basis for the backpropagation algorithmis the optimization technique known as gradient descent.

The backpropagation algorithm is implemented as follows:

1. Initialize the input layer: *y0 = x*
2. Propagate activity forward:
for *l = 1, 2, ..., L, y1 = f1(wl yl-1 + bl*),
where *bl* is the vector of bias weights.
3. Calculate the error in the output layer:
äL = t – yL
4. Backpropagate the error:
for *l = L-1, L-2,.. 1, äL = (wl +1 T äl+1). f1 1*
where T is the matrix transposition operator.
5. Update the weights and biases:
.Wl =äl yl-1 T ; .bl =äl

3.5. Assessment procedure

The steps for the asessment procedure are given below:

1. Form a training set or learning set *L* with *nL* samples and a test set *T* with *nT* samples (*nL+nT=n*). Denote *XL* as the learning data matrix of size *nL* by *p*, and *XT* as the test data matrix of size *nT* by *p*.
2. Select a subset of *p** genes from the set of all genes using one of the gene selection methods, resulting in *X*L* (*nL* by *p** matrix) and *X*T* (*nT* by *p**matrix).
3. Perform dimension reduction using PLS or PCA. Let *W* denote the *p** by *K** matrix containing the projection vectors. Compute the matrix *ZL* of gene components for the learning data set: *ZL = X*L × W*, and the gene components for the test data set: *ZT = X*T × W*.
4. Fit the class prediction model (FFBPN) to the learning components *ZL*. Predict the classes of samples in the test set using the fitted classifier and the test
components, *ZT*.
5. Compare the three different methods and its classification accuracy.

Accuracy = Number of classified correctly / Total number of patterns Figure 2: Architecture Diagram

4. RESULTS

In this section the results of each module is presented with its input, output and methodologies indetail

4.1. Gene Selection using t-test

The input of gene selection using t-teat is an array with Affymetrix call value of *n* samples (*n=102* in which *50* are normal samples and *52* are prostate samples), generates the output of *p*(100,200,500,1000)* genes from each sample and stored in two arrays.

a. train: training set of *60* samples(*30* normal samples and *30* abnormal)

b. test: testing set of *42* samples(*20* normal samples and *22* prostate samples)

The mean (all samples) difference between each gene in the two groups(normal and abnormal) are calculated and divided by the variance of groups. The resultant array of (*n*1*) is sorted with respect to the computed value. The *p** genes with the largest difference are used as inputs to the neural network where *P*=100,200,500,1000.* Now the normal dataset

j

i

k **output**

Wjk

Hidden

Wij

Input

of *(P* X 50)* is divided into *30* samples for training set and *20* samples for testing set. Similarly the abnormal dataset is divided into *20* samples for training and *20* samples for testing.

4.2. Dimension Reduction using PCA

The output of t-test (*p** by *n* samples) is given as input to identify the principal omponents for further processing. The training dataset *p*X nL* and the test dataset p* *X nT* is further reduced to *nL X nL, nT X nL* respectively .The reduced dataset for training and testing is given as input to the neural network for classification.

4.3. Classification using FFBPN

Classification of normal samples from the abnormal samples is done with three different cases. In the first case the output of the t-test is given as input to the neural network In the second case, output of the unsupervised PCA (*K < p**) is given as input to the neural network .The output of

the supervised PLS *(K < p*)* is given as input to the neural network in the third case. In all the three cases *p** genes are selected with *100, 200, 500* well as the training and testing using neural network is carried out with single hidden layer, dual hidden layer and three hidden layers with an input layer and single output node. The neural network is designed with *p** nodes in the input layer, *(p* / 2)* or *(p* X 2)* nodes in the hidden layer and single node in the output layer. The output node as two class values, that is *-1* for normal samples and *+1* for prostate samples. Nguyen widrow weight initialization method is used to initialize the weights between the layers and the bias values. Bipolar sigmoid activation function is used and the net is best suited for the learning rate of *0.1* with the error value *0.01*. The output from the t-test*(p*100,200,500,1000)*is given as input to the neural network The following tables show the comparison between *p** values along with varying number of hidden layers.

The output of t-test(*p** genes) is further reduced using PCA technique. The training data set is reduced to *nL X nL* and the test data set is reduced to *nT X nL* .The neural network is designed with *nL* nodes in the input layer and *nL/2* or *nl * 2* nodes in the hidden layer and a single node in the output layer. The net is trained for the learning rate of 0.1 with the error value *0.0*

Ex:: Single hidden layer with 120 or 30
hidden units and 60 input units with single output node

*P** genes
from t-test
Class 0
Class 1
Total
Accuracy
1000 85(17/20) 86(19/22) 86(36/42)
500 90(18/20) 86(19/22) 88(37/42)
200 85(17/20) 91(20/22) 88(37/42)
100 85(17/20) 86(19/22) 86(36/42)

5. CONCLUSION & FUTURE WORK

An important application of micro data is to classify biological samples or predict clinical or other outcomes. In this paper, three different cases are examined with the relative performance of classification procedures incorporating those methods, and designed a five-step procedure for assessment studies. The empirical analyses were based on the published gene expression data set of prostate cancer. Dimension reduction methods are frequently used but their relative performance has not been well
studied. It would be difficult to compare the performance of dimension reduction methods based on results of published studies due to differences among the studies in data sets, data preprocessing, and methods of gene selection, model selection and validation. The scope of the study is however quitelimited.

REFERENCES

[1] U. Alon, N. Barkai, D. A. Notterman, K. Gish, SD. Mack, and A.J Levine. Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays. In *Proc. Natnl. Acad.Sci. USA*, volume 96, 1999.

[2] A. Ben-Dor, L. Bruhn, N. Friedman, I. Nachman, M. Schummer, and Z. Yakhini.Tissue classification with gene expression profiles. *Journal of Computational Biology*,7(3-4):559–583, 2000.

[3] Boulesteix, A. (2004) PLS Dimension reduction for classification with microarraydata. *Statistical applications in genetics and molecular biology*, **3**, 1- 33.

[4] Jian J. Dai, Linh Lieu and David Rocke (2005)Dimension Reduction for Classification with Gene Expression Microarray Data. *Statistical applications in genetics and molecular biology*, 5, 1- 33.

[5] Nguyen, D.V. and Rocke, D.M. (2006) Tumor classification by partial least squares using microarray gene expression data. *Bioinformatics*,18, 39-50.