Gene Expression Data Analysis Using Support Vector Machines

Feng Chu and Lipo Wang

School of Electrical and Electronic Engineering
Nanyang Technological University
Block S1, Nanyang Avenue, Singapore 639798
E-mail: elpwang@ntu.edu.sg

***Abstract*—Cancer classification is an important problem both for clinical treatment and for biomedical research. Considering the good performance of support vector machines (SVMs) on solving pattern recognition problems, we use a C-SVM to process the B-cell lymphoma data. Principal Components Analysis (PCA) is used for gene selection. A voting scheme is used to do multi-group classification by k(k-1) binary SVMs. The classification results show that SVMs are effective tools for this problem.**

***Index Terms*—Cancer Classification, Gene Expression Data, Microarry, Principal Component Analysis, Support Vector Machine**

I. Introduction

Microarrays are also called DNA chips. Through this newly appeared technology, researchers are able to analyze expression information of thousands of genes simultaneously. One of the important applications of microarrays is cancer classification. For example, lymphoma, a kind of cancer, has several subtypes. The clinical treatment to different subtypes should also be different. Unfortunately, traditional methods are not able to give a reliable classification of these subtypes. Therefore, microarry has been used in this field in recent years. [1]-[2]

Support Vector Machines (SVMs) pioneered by Vapnik and his colleagues [3]-[5], try to find optimal hyperplane for separable patterns. Compared with other kinds of supervised learning techniques, SVMs pay much more attention to the points (vectors) with shortest distance to the optimal separation hyperplane, i.e., the support vectors. Among the whole dataset, only very small parts are support vectors. That means only a small set of crucial vectors play key roles in classification. This feature makes SVM a powerful tool in pattern recognition. Actually, SVMs have already been used in the fields such as handwritten character recognition, human face recognition, radar target identification, and gene expression data analysis as well. [11]-[14]

In this paper, SVMs will be used to classify the lymphnoma microarray dataset from Alizadeh and et al. [6]. In this set of data, samples belong to three classes, i.e., diffuse large B-cell lymphoma (DLBCL), follicular lymphoma (FL), and chronic lymphocytic lymphoma (CLL). The objective of the work is to classify these three kinds of lymphomas by using SVM. All the data is available on the web site (http://llmpp.nih.gov/lymphoma). This paper is organized as follows. Section II describes the formation and pre-processing of the Microarray dataset. The method fulfilling gene feature selection is elaborated in the subsequent section. In section IV, foundations of SVMs are provided. Experimental results and discussions are given in sections V and VI. Finally, conclusions are made and some future works are suggested.

II. Microarry Dataset

A. Dataset

One microarry experiment (one sample) usually conducts several thousands hybridizations. One hybridization process means one specific gene takes part in the experiment. To get meaningful results, in one microarray dataset, there are usually several tens to over one hundred experiments. One experiment can be seen as an input vector. The number of genes will determine the dimension of the input vector. In the dataset we use, there are 4026 genes. Therefore, the input vector's dimension is 4026. The whole dataset contains 62 experiments (samples). Among these samples, we randomly chose 31 to train the SVM classifier; we use the rest 31 samples to test the classification result. In the samples for testing, 21 belong to DLCL, 4 belong to FL, and 6 belong to CLL.

Because not all the genes take part in all the experiments and because not all the hybridization process are successful, it is very common for microarray to have some missing data in the input vectors. We put zero to the places where data are missing.

B. Normalization

To limit the influence of different distributions of the input vectors on classification, we normalize all the input vectors with the below methods:

$$\overline{X(j)} = \frac{X(j)}{Max(X_j) - Min(X_j)}$$

Where $\overline{X(j)}$ is normalized j-th attribute of vector X, $Max(X_j)$ and $Min(X_j)$ are the maximum and minimum of the j-th attribute in the dataset. X(j) is the original value.

III. Gene (Feature) selection

According to Cover's theorem on the separability of patterns [7], vectors in a higher dimensional space are more likely to be separated than vectors in lower dimensional space. However, using too high dimensional input vectors will

require much more computing resources. Therefore, casting the input vectors to a space with reasonable number of dimension is an important preprocessing before classification.

In our approach, principal components analysis (PCA) [8] is used for feature selection.

PCA is a classical dimension reduction method. It transforms the data set into a new space described by principal components (PC's). All the PC's are orthogonal and they are ordered according to the absolute value of their eigenvalues. The k-th PC is the vector with the k-th biggest eigenvalue. In fact, the PC's indicate the directions with largest variations of input vectors. Because PCA choose vectors with biggest eigenvalues, it can cover most of directions in which big variations happen in the input dataset. PCA also rejects some directions, because vector variations in these directions are very small, therefore, such variations can be looked as "noise" Furthermore, by calculating the sum of all the absolute values of all the PC's eigenvalues, we can estimate the percentage of the newly obtained dataset compared with the original dataset.

In microarray data analysis, PCA can be used both for experiments and for genes. In lymphoma dataset, because the genes greatly outnumber the experiments, we use PCA to select genes. After this preprocessing, 62 genes with greatest eigenvalues are chosen. The eigenvalues and their tendency of change are showed in Fig 1.

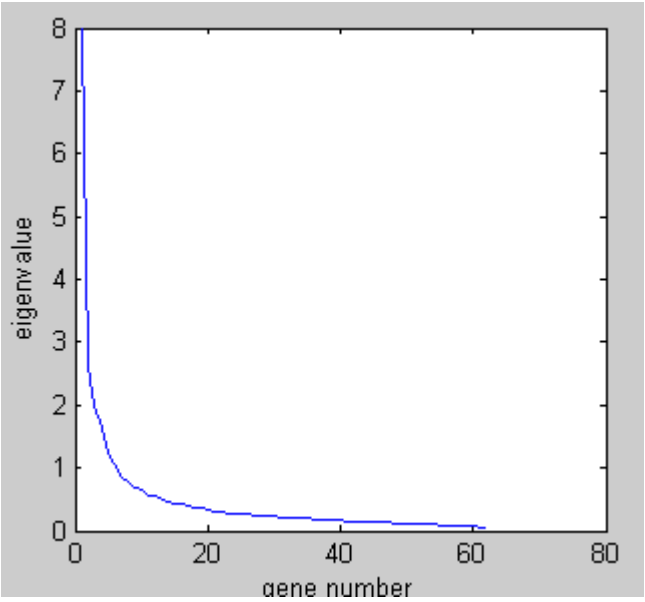

Fig 1. The variation of PCs' eigenvalues.

IV. Support Vector Machines

A. Binary SVM classifier

Support Vector Machines are comparatively new learning method. Just like multilayer perceptrons (MLP) and radial basis function (RBF) networks, SVMs are universal approximators. Their good performance on pattern recognition classification attracts researchers to work on them.

In our approach, we use a C-SVM. The basic idea of this C-SVM can be described as below. [15]

Given training vectors $x_i \in R^n, i = 1,...,l,$ in two classes, vector $y \in R$ and $y_i \in \{1, -1\}$, C-SVM can solve the following primal problem:

Find the optimum values of weight vector w and bias b such that they satisfy the constraint

$$y_i(w^T \phi(x_i) + b) \geq 1 - \xi_i,$$
$$\xi_i \geq 0, i = 1,...,l.$$

Where $\phi(x_i)$ is a function mapping i-th pattern vector to a potentially much higher dimensional feature space. Also, the weight vector w and the slack variables ξ_i should minimize the cost function:

$$\psi(w,\xi) = \frac{1}{2} w^T w + C\sum_{i=1}^{l} \xi_i$$

Where C is a positive constant term.

This primal problem also has a dual:

Find the Lagrange multipliers α_i, $i = 1,...,l$, that minimize the objective function:

$$\theta(\alpha) = \frac{1}{2}\alpha^T Q\alpha - e^T\alpha$$

Subject to the constraints:

$$0 \leq \alpha_i \leq C,$$
$$y^T\alpha = 0,$$

Where e is the vector of all ones, C (>0) is the upper bound, Q is a l x l positive semi-definite matrix, $Q_{ij} \equiv y_i y_j K(x_i, x_j)$, and $K(x_i, x_j) \equiv \phi(x_i)^T \phi(x_j)$ is the kernel. Here training vectors x_i are mapped into a higher dimensional space by the function ϕ.

The decision function that discriminates different pattern classes can be expressed as:

$$sign(\sum_{i=1}^{l} y_i \alpha_i K(x_i, x) + b)$$

B. Multi SVM classifiers

In practical applications, it is very common that there are more than two classes in the dataset. Therefore, binary SVM classifiers are usually not enough to solve a whole problem.

In these cases, a group of binary SVM classifiers are used. Each classifier is responsible for classifying two classes.

For any two classes, there must be one (and only one) classifier taking charge of the classification. Therefore, for a dataset with k classes, k(k-1)/2 binary classifiers are used. To get the ultimate result, a voting scheme is used. [9] For every input vector, all the classifiers give their votes so there will be k(k-1)/2 votes, when all the classification (voting) finished, the vector is designated to the class gets highest number of

votes. If one vector gets highest votes for more than one class, it is randomly designated to one of them.

V. Classification results

We randomly divide the whole 62 samples into two parts, 31 for training and 31 for testing. To find the classification results using different gene groups, first, we feed the data with only one gene to the SVM. After the SVM finish the classification, we add one gene to the input data, and then do classification again. We do this "classification then add gene" again and again until all the genes are fed to the SVM. We feed the genes in the order given out by PCA. The first gene is the one whose eigenvalue has biggest absolute eigenvalue.

The classification results are shown in Table 1. In this table, Gen No. is the number of genes fed into the classifier. The numbers showed under DLCL, FL, CLL are the numbers of the samples correctly classified in these classes. As mention in the former part, among the testing samples, there are 21 DLCL, 4 FL, and 6 CLL. The rate in the chart gives the classification accuracy (Fig 2).

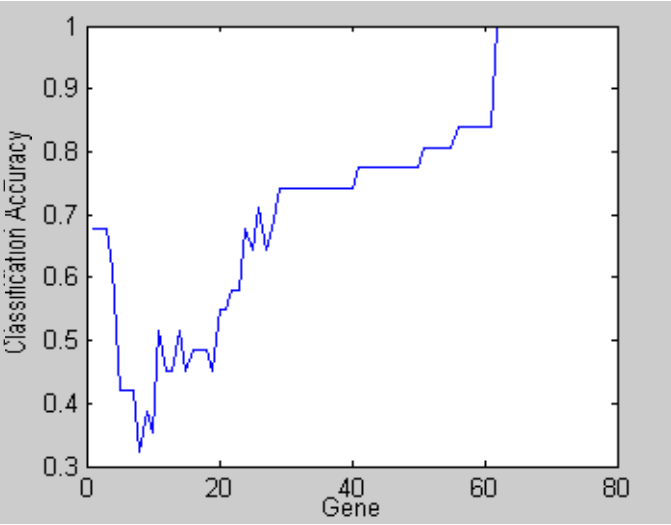

Fig 2. Classification Results Vs. the Number of Genes Included.

From Fig 2, it is can be seen that although there is drop when the number of genes fed in less than 10, the classification accuracy (the accuracy of the number of correctly classification to the number of testing samples, 31) increases gradually when more genes are fed in. At last, when all the 62 genes are fed in, the classification accuracy reaches 100%.

VI. Discussion

Compared with the popular acute myeloid leukemia (AML) and acute lymphoblastic (ALL) data, the B-cell lymphoma data is a relatively new dataset. Robert Tibshirani et al [10] have used "nearest shrunken centroids" method on this dataset.

Table 1. Classification Accuracy Vs. the Number of Genes Included.

Gen No.	DLCL	FL	CLL	Accuracy
1	21	0	0	0.677419
2	21	0	0	0.677419
3	21	0	0	0.677419
4	19	0	0	0.612903
5	13	0	0	0.419355
6	13	0	0	0.419355
7	13	0	0	0.419355
8	9	0	1	0.322581
9	11	0	1	0.387097
10	10	0	1	0.354839
11	14	1	1	0.516129
12	12	1	1	0.451613
13	12	1	1	0.451613
14	13	1	2	0.516129
15	12	0	2	0.451613
16	13	0	2	0.483871
17	13	0	2	0.483871
18	13	0	2	0.483871
19	13	0	1	0.451613
20	15	1	1	0.548387
21	15	1	1	0.548387
22	16	1	1	0.580645
23	16	1	1	0.580645
24	19	1	1	0.677419
25	18	1	1	0.645161
26	18	1	3	0.709677
27	18	1	1	0.645161
28	19	1	1	0.677419
33	21	1	1	0.741935
37	21	1	1	0.741935
40	21	1	1	0.741935
41	21	1	2	0.741935
44	21	1	2	0.741935
47	21	1	2	0.741935
53	21	1	2	0.741935
54	21	2	2	0.806452
55	21	2	2	0.806452
56	21	3	2	0.83871
58	21	3	2	0.83871
60	21	3	2	0.83871
61	21	3	2	0.83871
62	21	4	6	1

They got 100% classification accuracy with a dataset of 48 genes. From this point of view, their method is successful. Our approach is an much different approach. Compared with statistical classification methods, SVMs are more convenient to use. It is because after a SVM classifier has been trained, the user do not need to train it again when use it for classification. However, for statistical classification methods, all the needed factors must be calculated for every

classification. Considering the similar classification accuracy and similar gene numbers to get the optimal result, it can be concluded that using SVMs is more convenient than statistical approach is.

VII. Conclusion and Future Work

Cancer classification based on gene expression data is an important and relatively new pattern recognition problem. From our application of SVM classifiers to B-cell lymphoma gene expression data analysis, it is found that SVMs are a promising tool for this problem. In addition, principal component analysis (PCA) is proved to be an effective way for feature selection in this problem.

In our classification approach, there are still some improvement can possibly be achieved in some aspects. First of all, we can find from the classification results that the classification accuracies show great difference between DLCL, FL and CLL. The unbalanced numbers of DLCL, FL and CLL in training dataset may have caused this. We will consider the unbalance in our future classifiers. In addition, we now replace the missing data with 0, in the future; we will try to find some better methods to handle missing data. Last, in the voting scheme, when more than one highest votes appear, we now randomly designate the sample. We will try to find more appropriate ways for this problem.

We will also use RBF neural networks for processing gene expression data.

REFERENCES

[1] T.R.Golub, D.K.Simon, P.Tamayo, et al, "Molecular Classification of Cancer: Class Discovery and Class Prediction by Gene Expression Monitoring" Science, vol. 286, pp. 531-537, Oct. 1999

[2] Bittner,M et al, "Molecular classification of cutaneous malignant melanoma by gene expression profiling" Nature, vol. 406, pp 536-540, Aug. 2000

[3] Boser. B, I.Guyon, V.N.Vapnik, "A training algorithm for optimal margin classifiers" Fifth Annual Workshop on Computational Learning Theory, pp. 144-152. San Mateo, CA, 1992

[4] Cortes.C, V.Vapnik, "Support vector networks", Machine Learning, vol. 20, pp. 273-297, 1995

[5] V.N.Vapnic, "Statistical learning theory", New York, Wiley, 1998

[6] Ash A. Alizadeh, et al, "Distinct types of diffuse large B-cell lymphoma identified by gene expression profiling" Nature, vol. 403, pp 503-511, Feb. 2002

[7] Cover. T.M., "Geometrical and statistical properties of systems of linear inequalities with applications in pattern recognition", IEEE Transactions on Electronic Computers, vol. EC-14, pp. 326-324, 1965

[8] Simon. Haykin, "Neual networks, a comprehensive foundation" 2nd Editon, New Jersey, Prentice Hall, 1999

[9] B. Scholkopf, C.J.C. Burges, A.J.Smala. "Advances in kernel methods-support vector learning", Cambridge, MA, MIT Press

[10] Robert Tibshirani, Trevor Hestie, et al, "Class prediction by nearest shrunken centroids, with application to DNA microarrays" http://www-stat.stanford.edu/~tibs/research.html

[11] Li Zeyu, Tang Shiwei, Wang Hao, "Fast recognition of handwritten digits using pariwise coupling support vector machine", Neural Networks, 2002. IJCNN'02. Proceedings of the 2002 International Joint Conference on, Volume:1, pp. 878-883, 2002.

[12] Ng, J., Shaogang Gong, "Multi-view face detection and pose estimation using a composite support vector machine across the view sphere", Recognition, Analysis, and Tracking of Faces and Gestures in Real-Time Systems, Proceedings, pp 14-21. 1999.

[13] Zhao, Q., Principe, J. "Support vector machine for SAR automatic target recognition." IEEE Transactions on Aerospace and Electronics, 37(2). Apr. 2001.

[14] Michael P. S. Brown, William Noble Grundy, David Lin, Nello Cristianini, Charles Sugnet, Terrence S. Furey, Manuel Ares, Jr., David Haussler. "Knowledge-based Analysis of Microarray Gene Expression Data Using Support Vector Machines", Proceedings of the National Academy of Sciences. pp262-267. 1997.

[15] Vorgetleg von, "Support vector learning". Oldenbourg Verlage, Munich, 1997.