Gene Expression Data Classification Using SVM-KNN Classifier

Xiaoqiao Shen, Yaping Lin
School of Computer and Communication, Hunan University, China

Email: tracyqiao@163.com , yplin@hnu.cn

Abstract

In this paper we propose a new classifier that combines Support Vector Machine (SVM) with K Nearest Neighbor (KNN) for gene expression data classification. This new classifier SVM-KNN (KSVM) takes SVM as a 1NN classifier in which only one representative point is selected for each class. In the class phase, the algorithm computes the distance from the test samples to the optimal hyperplane of SVM in feature space. If the distance is greater than the given threshold, the test sample will be classified on SVM; otherwise, the KNN algorithm will be used. The experiment results show that KSVM has higher classification rate than those of traditional SVM and KNN. And it also suggests a better method for the problem of gene selection.

1. INTRODUCTION

Gene expression data yield a global view of the cell by enabling the measurement of expression levels of thousands of genes simultaneously. When used to compare normal tissues and tissues at various stages of disease, or diseased tissues with different responses to treatment, gene expression data presents opportunities for improved disease diagnosis and a deeper understanding of the molecular basis of observed phenotypes. Several machine learning methods have been applied to gene expression data classification. Ben-Dor has proved that Nearest Neighbor Algorithm (*NN* Algorithm), Boosting Algorithm and clustering-based classification algorithm can all be applied to gene expression data classification [1]. Support Vector Machines (*SVMs*), a classical machine learning algorithm was firstly applied to gene expression data classification by Brown [2]. A principal component and discriminant analysis method of tumor classification was proposed by Xiong M [3]. Using this method, the percentage of correctly classified normal and tumor tissue was 87.0%. But the classifier must acquire 2000 genes of a new sample to determine the type of the sample. Friedman applied Bayesian network method to analysis of gene expression in 1999 [4]. The advantage of this model is that it can be used to describe complicated stochastic process and supply an explicit method for learning in noisy observation environment. But these algorithms have not determined how many genes should be chosen and how to choose these genes to get the highest classification rate.

In this paper, we present a new classifier that combined Support Vector Machine with K Nearest Neighbor algorithm for classification of gene expression data. The experiment results show that *KSVM* has higher classification rate than that of traditional *SVM* and *KNN*. And this new classifier also suggests a better method for the problem of gene selection.

2. REVIEW OF SVM AND KNN

2.1 SUPPORT VECTOR MACHINE

SVM is an algorithm of machine learning, introduced by Vapnik, based on the Structural Risk Minimization principle from Statistical Learning Theory [6]. *SVM* is a method for finding a hyperplane in high dimensional space that separates training samples of each class while maximizing the minimum distance between that hyperplane and any training sample. If the data are not linearly separable, they can be projected onto a higher dimensional 'feature' space in which they are separable. Upon training, the *SVM* identifies those samples that are closest to the hyperplane, and thus which play a greater role in classifying a test sample. However, the classification rate of *SVM* is not very high when samples are close to the hyperplane.

In gene expression data classification problems, we are given l experiments $\{(x_1, y_1),\cdots, (x_l, y_l)\}$. This is called the *training set*, where x_i is a vector corresponding to the expression measurements of the *ith* experiment or sample (this vector has n components, each component is the expression measurement for a gene or EST) and y_i is a binary class label, which will be ± 1. We want to estimate a multivariate function from the training set that will accurately label a new sample, that is $f(x_{new}) = y_{new}$.

2.2 NEAREST NEIGHBOR

Nearest neighbor methods are based on a distance function for pairs of observations, such as the Euclidean distance or one minus the correlation. For the gene expression data considered here, the distance between two samples, with gene expression profiles $x = (x_1,\cdots,x_l)$ and $x' = (x_1',\cdots,x_l')$, is based on the correlation between their two gene expression profiles:

$$r_{x,x'} = \frac{\sum_{j=1}^{l}(x_j - \bar{x})(x_j' - \bar{x}')}{\sqrt{\sum_{j=1}^{l}(x_j - \bar{x})^2}\sqrt{\sum_{j=1}^{l}(x_j' - \bar{x}')^2}}$$

Sensitivity to noise in the data can be greatly reduced by classifying a sample according to the majority class of the N closest training samples, where N > 1. However, most distance metrics are expected to become less sensitive as the dimensionality of the 'noisy' data increases, thus limiting the performance of NN when applied to gene expression data classification.

3. KSVM GENE CLASSIFICATION

3.1 Principle of KSVM

We analyzed the misclassified samples by *SVM* and found that these samples were all close to the hyperplane. Even *SVM* algorithm is in accordance with maximum margin principal, it still can not handle the situation when the examples of the two classes are close to the hyperplane very well. As shown in figure 1, when we employed *SVM* to classify Leukemia data set, three examples were misclassified.

The new classifier that combined Support Vector Machine with K Nearest Neighbor algorithm as an effective methodology for classification of gene expression data, based on taking *SVM* as a 1NN classifier in which only one representative point is selected for each class. *KNN* classifier rates all points as support vectors, so it has higher classification accuracy. As shown in Figure 2, we calculate the distance between a given test sample x with support vectors of two classes – x^+ and x^-. If the distance is greater than the given threshold(x in region II),the test sample would be classified on *SVM*; otherwise, if the distance is no greater than the given threshold(x in region I),the *KNN* algorithm will be applied.

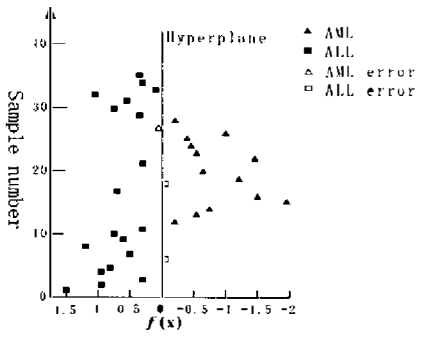

Figure 1. The signed distance, f (x), from the optimal separating hyperplane for the test samples. The black triangles are the correctly labeled *AML* samples. The black rectangles are the correctly labeled *ALL* samples. The white triangle is the misclassified *AML* sample. Two white rectangles are the misclassified *ALL* samples.

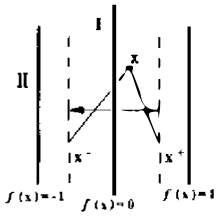

Figure2. Distance from sample x to hyperplane determines which algorithm should be used to classify x.

3.2 KSVM algorithm

Algorithm *KSVM*:

Firstly, use *SVM* algorithm to get relevant support vector and its factor、constant b.

T is the test set, T_{sv} is the support vector set and k is the number of *KNN*.

Step1: if $T \neq \phi$: select $x \in T$; else: algorithm finished.

Step2: $g(x) = \sum_i y_i a_i K(x_i, x) - b$.

Step3: if $|g(x)| > \varepsilon$: $f(x) = \mathrm{sgn}(g(x))$, $output = f(x)$;

Else: pass parameter x 、 T_{sv} 、 k to *KNN* algorithm.

Step4: $T \leftarrow T - \{x\}$, go to Step1.

KNN classification algorithm mentioned in Step3 is the same as normal *KNN* algorithm, it just takes support vector set T_{sv} as representative point set. The difference is that here we compute the distance between test samples and support vectors in feature space instead of original sample space, and its distance formula is not typical Euclidean distance formula but is the following formula:

$$d(x,x_i)\|\phi(x)-\phi(x_i)\|^2 = k(x,x)-2k(x,x_i)+k(x_i,x_i), \quad x_i \in T_{sv} \qquad (1)$$

Similar to *SVM* algorithm, we can use kernel function in formula (1) to solve various problems. Classification threshold ε was always set to round about 1.

4. EXPERIMENTS

In order to compare the performance of *KNN*、*SVM* and *KSVM* algorithms, we use these three algorithms to solve the same gene expression data classification problem. The algorithms are applied to data sets from published cancer gene expression studies. Leukemia data set contains 73 samples: 25 samples of acute myeloid leukemia (AML) and 48 samples of acute lymphoblastic leukemia (ALL). The source of the gene expression measurements was taken from bone marrow samples and peripheral blood samples from acute leukemia patients. Colon data set contains 62 samples of colon epithelial cells taken from colon-cancer patients. 40 of 62 samples are colon cancer samples and the remains are normal samples. Table 1 shows the number of samples in these two data sets and the number of samples used as training data and test data in each data set.

(1) Gene selection

It is important to know which genes are most relevant to the binary classification task and select these genes for a variety of reasons: removing noisy or irrelevant genes might improve the performance of the classifier, a candidate list of important genes can be used to further understand the biology of the disease and design further experiments, and a clinical device recording on the order of tens of genes is much more economical and practical than one requiring thousands of genes.

Data sets		# of Samples	Class -1	Class +1	# of genes
Data set(A) Leukemia	AML *vs* ALL *(train)*	38	27ALL	11AML	7129
	AML *vs* ALL *(test)*	35	21ALL	14AML	7129
Data set(B) Colon	training data	31	11	20	2000
	test data	31	11	20	2000

Table 1. The number of samples in the various data sets

The gene selection problem is an example of what is called *feature selection* in machine learning. Two feature selection approaches are: *signal-to-noise* (*S2N*, also known as *P-metric*) (Golub et al., 1999, Slonim et al., 2000;), and *recursive feature elimination* (*RFE*) (Guyon et al., 2002). In our experiments we use *S2N*. For each gene j, we compute the following statistic:

$$S(j) = \frac{\mu_+(j) - \mu_-(j)}{\sigma_+(j) + \sigma_-(j)} \qquad (2)$$

Where $\mu_+(j)$ and $\mu_-(j)$ are the means of the classes +1 and -1 for the j^{th} gene. Similarly, $\sigma_+(j)$ and $\sigma_-(j)$ are the standard deviations for the two classes for the j^{th} gene. Genes that give the most positive values are most correlated with class +1, and genes that give the most negative values are most correlated with class -1.

Data sets	# of Samples	Method	Errors		
			Class+1	Class-1	Total
Data set (A) Leukemia	49	KNN	1	3	4
		SVM	1	2	3
		KSVM	0	1	1
	999	KNN	1	1	2
		SVM	0	1	1
		KSVM	0	0	0
	7129	KNN	1	1	2
		SVM	1	1	2
		KSVM	0	0	0
Data set (B) Colon	49	KNN	3	2	5
		SVM	2	2	4
		KSVM	1	1	2
	999	KNN	3	1	4
		SVM	1	1	2
		KSVM	0	0	0
	2000	KNN	2	1	3
		SVM	2	2	4
		KSVM	1	0	1

Table 2. Absolute number of errors of the various methods

(2) Experiment results and analysis

For each of these algorithms, we did three experiments: we used top 49 genes from the sample in the first experiment, top 999 genes in the second experiment and all genes in the third. The results are shown in Table 2.

According to the results shown in Table 2, we can make two conclusions:

i) Comparing to traditional *SVM* and KNN algorithms, in the case of using the same gene number, *KSVM* classifier has higher accuracy. The reason is that *KSVM* algorithm obtains more support vectors after training, which means it carries more information.

ii) The number of genes used in the training process has less effect to *KSVM* classifier than the other two classifiers. Owning to the small sample number and high dimension, traditional *SVM* classification has more errors without gene selection. The performance of traditional *SVM* algorithm improves if using gene selection. But to a certain degree, the accuracy of *SVM* algorithm based on the number of genes used [8]. *KSVM* algorithm is less sensitive to the number of genes used in the training process and keeps high accuracy because it carries more information than *SVM* algorithm.

5. CONCLUSIONS

In this paper, we proposed a new classifier that combined Support Vector Machine (*SVM*) with K Nearest Neighbor (*KNN*) as an effective methodology for classificating gene expression data. The results of our experiments show that this new classifier *KSVM* algorithm can not only improve the accuracy of gene selection comparing to those of traditional *SVM* and *KNN* algorithms, but also is less sensitive to the number of genes used. In the future, we will explore the possibility of using only a subset of support vectors for classification, and research that if *KSVM* is available for multiclass classification.

REFERENCES

[1] Ben-Dor, A., Bruhn, L., Friedman, N., Nachman, I., Schummer, M., and Yakhini, Z. (2000).Tissue Classification with Gene Expression Profiles. In: Proceedings of the 4th Annual International Conference on Computational Molecular Biology (RECOMB), Tokyo, Japan: Universal Academy Press.

[2] Brown, M.P.S., Grundy, W., Lin, D., Cristianini, N., Sugnet, C., Furey, T.M., Ares, J., and Haussler, D. Knowledge-based analysis of microarray gene expression data by using support vector machines. PNAS, 97, 262-267.

[3] Xiong M, Jin L, Li WJ et al. Computational methods for gene expression-based tumor classification. Biotechniques, 2000, 29(6):1264

[4] Friedman, N., Linial, M., Nachman, I., and Pe'er, D. 2000. J. Computational Biology 7(3,4), 601–620.

[5] 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, 286:531–537, October 1999.

[6] V,N. Vapnik. The nature of statistical learning theory. Springer,1995.

[7] Li Rong, Ye Shiwei, Shi Zhongzhi. SVM-KNN Classifier - A New Method of Improving the Accuracy of SVM Classifier. Acta Electronicasinica, Vol.30, No.5.

[8] Mukherjee.s, Tamayo, P.Mesirov, J.Slonim, D,Verri,A. and Poggio,T.(1999)Support vector machine classification of microarray data. Technical Report CBCL Paper 182/AI Memo 1676 MIT.

[9] Sheng-Chao Ding; Wei Yuan; Bin Ni; Dong-Li Hu; Juan Liu; Huai-Bei Zhou; Tumor diagnosis with support vector machines; 2003 International Conference on Machine Learning and Cybernetics, Volume:2, 2-5. Nov. 2003 ; Pages: 1264 – 1269

[10] Golub TR, Slonim DK, Tamayo P et al. Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. Science, 1999, 286(5439): 531

[11] sabelle Guyon, Jason Weston, Stephen Barnhill, M.D. and Vladimir Vapnik, Machine Learning, vol.46, 2002, 1-3: 389-422