The Second International Symposium on Optimization and Systems Biology (OSB'08)
Lijiang, China, October 31– November 3, 2008

Incorporating gene similarity into support vector machine for microarray classification and gene selection*

Jun-Yan Tan[1] Zhi-Xia Yang[†2,3]

[1]College of Science, China Agricultural University, Beijing 100083
[2]College of Mathematics and System Science, Xinjiang University, Urumuchi 830046
[3]Academy of Mathematics and Systems Science, CAS, Beijing 100190

Abstract In this paper, we propose a novel method based on support vector machine (SVM) for microarray classification and gene (feature) selection. The proposed method, called similarity-based SVM (SSVM), incorporates the prior knowledge of gene similarity into the standard SVM by combining the standard l_2 norm and the similarity penalty of all the genes. The preliminary experiments show that our method performs better than the standard SVM, $l_2 - l_0$ SVM and SVM-RFE, especially when the features are highly similar.

Keywords Support vector machine; microarray data; gene selection.

1 Introduction

The DNA Microarray technology allows measuring simultaneously the expression level of a great number of genes in tissue samples. However, the microarray data usually contains only a small number of samples. These characteristics raise new challenges for data analysis. In the classification, data overfitting arises when the number of features is much larger than the number of the samples. In order to overcome the risk of overfitting, there are two strategies in general: one is to find ways to reduce the dimensionality of the feature space; another is to use regularization to some extent without requiring space dimensionality reduction. For instance, support vector machine (SVM) is one of the most effective methods by using regularization for microarray classification [1], even it benefits from dimensionality reduction. While, in microarray analysis, researchers are more interested in identifying the genes that are relevant to the cancer, it is desirable to have a tool that can achieve both classification and gene selection. So a major limitation of SVM is that it cannot perform automatic gene selection.

Guyon *et al.* (2002)[1] proposed the SVM-recursive feature elimination (SVM-RFE). The SVM-RFE method ranks all the genes according to some score function and eliminates one or more genes with the lowest scores. This process is repeated until the highest

*This work is supported by the Key Project of the National Natural Science Foundation of China(No. 10631070), the National Natural Science Foundation of China (No.10801112) and the China Postdoctoral Science Foundation funded project(No.20080430573)

†The corresponding author. E-mail: xiyangzhixia@126.com

classification accuracy. Magasarian (1998) [2] and Magasarian (2007) [3] proposed the feature selection via concave minimization (FSV), which can automatically select features by the l_0- norm penalty of the number of features. But their classification accuracy is not very good due to the loss of the maximum margin between two classes samples. Neumann (2005) [4] proposed the $l_2 - l_0$ norm SVM to improve the generalization performance of the classifiers. It combines the l_2 norm and the l_0 norm and performs better in the classification accuracy than the FSV due to the l_2-norm of w in the objective function. Wang (2008) [5] proposed a hybrid huberized support vector machine (HHSVM) which replaced the loss function in the SVM by the huberized hinge loss function.

In this paper, we propose a novel method based on SVM for microarray classification and gene (feature) selection. It incorporates the prior knowledge of gene similarity into the standard SVM. Our method is called similarity-based SVM (SSVM). SSVM automatically selects the minimal genes that are relevant to the class. And the preliminary experiments show that SSVM performs rather nice, particularly when the genes are highly similar.

The rest of the paper is organized as follows. In Section 2, We propose our method after briefly introduce $l_2 - l_0$ SVM. In section 3, our method is tested for both simulation and real microarray datasets. We conclude the paper in section 4.

2 Model

2.1 $l_2 - l_0$ SVM

Neumann (2005) [4] incorporated l_0- norm into the standard SVM and got the optimization problem:

$$\min_{\boldsymbol{w},b,\boldsymbol{\xi}} \quad \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{l}\xi_i + \lambda\|w\|_0, \tag{1}$$

$$\text{s.t.} \quad y_i((w\cdot x_i)+b) \geq 1-\xi_i\ ,\ i=1,\cdots,l\ , \tag{2}$$

$$\xi_i \geq 0\ ,\ i=1,\cdots,l\ . \tag{3}$$

The l_0 penalty tends to shrink the coefficients of the features that are irrelevant to the class label to exactly zero. The l_2 penalty is responsible for the very good SVM classification results. Therefore $l_2 - l_0$ SVM improves the generalization of FSV and selects features simultaneously. The $l_2 - l_0$ SVM selects a fewer features for the classification, but the selected features may be very similar, which means that the selected features are redundant. This will be shown by the numerical experiments in Section 3.

2.2 Similarity-based SVM (SSVM)

Let us turn to our model. Given the training set

$$T = \{(x_1,y_1),\cdots,(x_l,y_l)\} \in (\mathscr{X}\times\mathscr{Y})^l, \tag{4}$$

where $x_i = ([x_i]_1,[x_i]_2,\cdots,[x_i]_n) \in \mathscr{X} \subseteq R^n$ is input and its n components are called "features". For the microarray data, the n features are n gene expression coefficients. $y_i \in \mathscr{Y} = \{-1,1\}$ is output. We pay particular attention to the gene vector

$$g_i = ([x_1]_i,[x_2]_i,\cdots,[x_l]_i)^T, \tag{5}$$

which comprises the i-th feature of all inputs to denote the expression levels of i-th gene in all inputs, where $i = 1, 2, \cdots, n$. Our model is based on similarity among the gene vectors. We incorporate the penalty of similarity among these vectors into the standard SVM and establish the following optimization problem:

$$\min_{w,b,\xi} \quad \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{l}\xi_i + \lambda |w|_*^T G |w|_*, \tag{6}$$

$$\text{s.t.} \quad y_i((w\cdot x_i)+b) \geq 1-\xi_i\ ,\ i = 1, \cdots, l\ , \tag{7}$$

$$\xi_i \geq 0\ ,\ i = 1, \cdots, l\ , \tag{8}$$

where $|w|_* = (|w_1|_*, |w_2|_*, \cdots, |w_n|_*)^T$ and $|w_i|_* = 1$, if $|w_i| > 0$ and 0 otherwise, the matrix $G = (G_{ij})_{n\times n}$ is defined by $G_{i,j} = \mathrm{sim}(i,j)$ if $i \neq j$ and 0 otherwise, where $\mathrm{sim}(i,j)$ stands for the similarity between g_i and g_j. It can be evaluated by the Pearson correlation coefficient or the Euclid distance. $\mathrm{sim}(i,j)$ can also represents the metabolic similarity between gene i and gene j, it can be computed by the GO (Gene Ontology [6]) annotation similarity between gene i and gene j. In this paper, we compute $\mathrm{sim}(i,j) = \frac{1}{\|g_i - g_j\|}$. Thus the similarity penalty of all the genes are the following:

$$|w|_*^T G |w|_* = \sum_{i=1}^{n}\sum_{j=1,(i\neq j)}^{n} \frac{|w_i|_*|w_j|_*}{\|g_i - g_j\|}.$$

The parameter λ in (6) determines the trade-off between the minimization of $|w|_*^T G|w|_*$ and $\frac{1}{2}\|w\|^2 + C\sum_{i=1}^{l}\xi_i$. The minimizing of $|w|_*^T G|w|_*$ tends to shrink the non-zero components of w by considering the similarity between genes. For example, if two vectors g_i and g_j are very similar, i.e. g_i is near to g_j enough, the minimization of $|w|_*^T G|w|_*$ will make $\frac{|w_i|_*|w_j|_*}{\|g_i - g_j\|}$ to be zero, this leads to that either w_i or w_j is equal to zero. This means that at least one of gene i and gene j is removed from n genes. Similarly, if in n vectors, $g_i, i = 1, 2, \cdots, n$, more than two vectors are very similar, the minimization of $|w|_*^T G|w|_*$ will select only one of them and remove the others. When the distance between g_i and g_j is large (they are dissimilar), it may be allowed that both w_i and w_j to be nonzero due to the large denominator in $\frac{|w_i|_*|w_j|_*}{\|g_i - g_j\|}$. This implies that both gene i and gene j will be selected. So, the genes selected by the SSVM are the most dissimilar.

Next, we simplify the problem (6) $\sim$ (8) by some approximation using the strategy in [2]. We approximate$|w|_*$ by $(e - e^{-\alpha|w|})$ and get the optimization problem:

$$\min_{w,b,\xi\ ,v} \quad J(w,b,\xi,v) = \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{l}\xi_i + \lambda(e - e^{-\alpha v})^T G(e - e^{-\alpha v}), \tag{9}$$

$$\text{s.t.} \quad y_i((w\cdot x_i)+b) \geq 1-\xi_i\ ,\ i = 1, \cdots, l\ , \tag{10}$$

$$\xi_i \geq 0\ ,\ i = 1, \cdots, l\ , \tag{11}$$

$$-v \leq w \leq v. \tag{12}$$

In order to simplify the above problem (9) $\sim$ (12) further, the objective function is approximated by its second order Taylor expanded form. This leads to the following algorithm which aims at both classification and gene selection.

Algorithm 1. (Similarity-based SVM)

1. The training set T is given by (4), select the parameter $C > 0$, λ, α, ε, the initial point (w_0, b_0, ξ_0, v_0) and set $k = 0$;
2. Solve the quadratic programming problem

$$\min_{w,b,\xi,v} \quad \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{l}\xi_i + \lambda\alpha v^T(e - e^{-\alpha v_k}) + \lambda v^T M(v_k)v, \tag{13}$$

$$\text{s.t.} \quad y_i((w \cdot x_i) + b) \geq 1 - \xi_i\ ,\ i = 1, \cdots, l\ , \tag{14}$$

$$\xi_i \geq 0\ ,\ i = 1, \cdots, l\ , \tag{15}$$

$$-v \leq w \leq v, \tag{16}$$

   where $M(v_k)$ is the Hessian matrix of $H(v) = (e - e^{-\alpha v})^T G(e - e^{-\alpha v})$ at v_k, $G = (G_{ij})_{n\times n}$ is defined by $G_{i,j} = \frac{1}{\|g_i - g_j\|}$, if $i \neq j$ and 0 otherwise with g_i given by (5), and get the solution$(\overline{w}, \overline{b}, \overline{\xi}, \overline{v})$;
3. If the stopping criterion $|J((\bar{w}, \bar{b}, \bar{\xi}, \bar{v})) - J((w_k, b_k, \xi_k, v_k))| < \varepsilon$, is satisfied, goto step 5; otherwise, set $k = k + 1$, update $(w_k, b_k, \xi_k, v_k) = (\bar{w}, \bar{b}, \bar{\xi}, \bar{v})$, goto step 3;
4. Get the solution $(w^*, b^*, \xi^*, v^*) = (\bar{w}, \bar{b}, \bar{\xi}, \bar{v})$. The decision function is $f(x)$ = sign $((w^* \cdot x) + b^*)$. Feature selection can be achieved by keeping the component of w which satisfies $|w_i^*| > \varepsilon$.

3 Numerical experiments

In this section, both simulation and real data are used to illustrate the SSVM and our method are compared with SVM, $l_2 - l_0$ SVM and SVM-RFE. If two gene's expression levels in the training set T are completely same, we remove one of them from the input of training data. We choose the initial value $w_0 = (0, \cdots, 0), b_0 = 1, \xi_0 = (0, \cdots, 0)$ for both simulation and real data, and $v_0 = (\underbrace{0, \cdots, 0}_{n-1}, e^{-10})$ for the former, $v_0 = \underbrace{(0, \cdots, 0)}_{n}$ for the later. α and ε are set to 5, 10^{-8} respectively for both simulation and real data.

3.1 Simulation

The main purpose of the simulation is to demonstrate that when the features are independent, the SSVM performs slightly better than $l_2 - l_0$ SVM, SVM and SVM-RFE, while the SSVM has more advantage when the features are highly similar.

We first consider the scenario I where all features are independent. We construct the training points with $l = 20$ using the method in[8]. Each input x_i is an $n = 50$ dimensional vector and is generated from $N(0, 1)$. The outputs are determined by the hyperplane $g(x) = 4[x]_1 + 2[x]_2 + 4[x]_3 - 4.8 = 0$. This means that the output of an input x_i is "$+1$" if $g(x_i) > 0$ and is "-1" if $g(x_i) < 0$. The test set is generated in the same way, it contains 20 samples. Therefore the important features are the first three. The features are relevant if they are the subset of the first three and the rest forty seven features are redundant. The parameters λ, C are selected by ten-fold cross validation. Each experiment is repeated 50 times. The average test errors of four methods are listed in Table 1. The prediction error of SSVM is the lowest due to the minimal redundant features. Besides the test

Table 1: Comparison of average test error and feature selection during 50 experiments

	scenario I			scenario II		
	q_{signal}	q_{noise}	test error(%)	q_{signal}	q_{noise}	test error(%)
SVM	3	47	17.9(0.118)	3	47	0
l_2-l_0 SVM	1.78(0.91)	5.42(1.66)	15.6(0.211)	3.6(1.161)	2.84 (2.652)	0
SVM-RFE	2.26(0.89)	5(7.4)	13.6(0.064)	3.02(2.47)	0.22(0.84)	0
SSVM	1.42(0.908)	4.78 (1.91)	11.7(0.045)	5.06(1.766)	0	0

q_{signal} is the number of selected relevant features, q_{noise} is the number of selected noise features. The numbers in the parentheses are the corresponding standard errors.

Table 2: Comparison of selected features in one experiment during scenario II

	relevant feature number	noise feature number	total
SVM	$(1 \sim 10)$	$(20 \sim 50)$	50
l_2-l_0SVM	$(1,34,42,45)$	$(11,21,31,41,44)$	9
SVM-RFE	$(1,2,3,4)$	0	4
SSVM	$(26,27,31,42,43,48)$	0	6

error, feature selection results of four models are also compared. We consider $q_{signal} =$ number of selected relevant features, and $q_{noise} =$ number of selected redundant features. The results are in Table 1. Although SVM, l_2-l_0 SVM and SVM-RFE can select more relevant features than SSVM, they select more redundant features than SSVM.

Now we consider the scenario II where the features are highly similar. We consider $l = 10+10$ and $n = 50$. The first ten dimensions of the input in "$+1$" class are generated from $N(1,0.5)$, the rest are generated by the following way: $g_{10k+i} = g_i + 0.01e$, for $i = 1,\cdots,10, k = 1,\cdots,4$ and e is a vector of all 1. While the first ten dimensions of the input in "-1" class are generated from $N(-1,0.5)$, and the rest are generated using the same way as the "$+1$" class. Obviously, only ten features whose last digits of subscripts vary from 0 to 9 differently are relevant, the remaining forty features are redundant. We also repeat 50 experiments for each method. The performances of four methods are summarized in Table 1. The average test errors of four methods are similar. In the view of feature selection, SSVM performs better because it can select a fewer features and remove all redundant features. Furthermore, the exact features selected by four methods during one experiment are shown in Table 2. We can see that SVM cannot perform feature selection. The l_2-l_0 SVM selects the highly similar features $1,11,21,31,41$(their subscripts have the same last digits), this means the l_2-l_0 SVM cannot remove redundant features effectively. However the features selected by SVM-RFE and SSVM are non-redundant and are all relevant to the class label, because their last digits of subscripts are different from each other.

3.2 Real Data Analysis

We firstly consider the Colon cancer datatset in Alon and Barkai [7]. The training set is T in (1), $l = 62$ (40 colon cancer tumors and 22 normal tissues), $n = 2000$. For SSVM and l_2-l_0 SVM, to reduce the computational cost, the dataset is first shrinked by selecting

Table 3: Results on 100 random splits of the original datasets: the upper part is for the Colon dataset, the lower part is for the Prostate dataset

	SVM	$l_2 - l_0$ SVM	SVM-RFE	SSVM
Colon dataset				
test error(%)	14.65(1.34)	13.9(0.107)	17.1(0.87)	12.2(0.053)
number of genes	All	33.2(6.318)	64	6.3(1.7)
Prostate dataset				
test error(%)	0.109(1.96)	0.063(0.054)	0.069(1.59)	0.041(0.023)
number of genes	All	31.6(5.14)	37.4(41.2)	4.8(1.48)

The numbers in the parentheses are the corresponding standard errors.

Table 4: The most frequently selected genes by the SSVM for the Colon dataset

gene number	selection frequency	gene number	selection frequency	gene number	selection frequency
377	100	249	100	1870	62
1473	36	1993	33	994	30

selection frequency is the number of times the gene was selected out of 100 experiments.

the top 100 informative genes using the method [9]*i.e.* ranking genes by $t-test$ value: $P(j) = \left|\frac{\mu_1(j)-\mu_{-1}(j)}{\sigma_1(j)+\sigma_{-1}(j)}\right|$, $j = 1,\cdots,n$, where $\mu_1(j)$ and $\sigma_1(j)$ are the mean and standard deviation of jth feature of all inputs in "$+1$" class, $\mu_{-1}(j)$ and$\sigma_{-1}(j)$ are those of jth feature of all inputs in and "-1" class. We rank $P(j)\,(j = 1,\cdots,n)$ in descending order and choose the top 100 features.

We randomly split the samples into training and test sets 100 times; for each split, the training set consists of 42 samples(27 cancer samples and 15 normal samples), the rest samples form the test set. Four methods are applied to the training set for each split. The parameters λ and C are chosen by ten-fold cross validation on the training set. The average test errors of the four methods and the number of selected genes during 100 random splits of the original colon cancer dataset are summarized in the upper part of Table 3. We can see that the SSVM has the lowest test error and selects the minimal genes. Table 4 summarizes the genes that are "frequently" selected by SSVM. As we can see, two genes are selected 100 times by the SSVM. The SSVM performs more stable in selecting the most important genes. We can see that the genes selected by the SSVM and the 6 most frequently selected genes in [5] are all in the same clusters if we cluster the 2000 genes into 100 or 200 or 400 clusters. This means the two genes selected by the SSVM is the most important genes. The 249*th* and 377*th* genes can be seen as "seed genes" relevant to the colon cancer, and the other relevant genes can be found via ranking the genes by the similarity to the two genes.

The second dataset we considered is the Prostate dataset [10], which provides the expression levels of 12,600 genes for 50 normal samples and 52 prostate cancer samples.

Table 5: The most frequently selected genes by the SSVM for the Prostate dataset

gene number	selection frequency	gene number	selection frequency	gene number	selection frequency
5886	100	9087	100	11686	38
3526	29	10144	23	5435	20

selection frequency is the number of times the gene was selected out of 100 experiments.

Table 6: Ten-fold cross validation and feature selection for the Leukemia dataset

	SVM	$l_2 - l_0$ SVM	SVM-RFE	SSVM
test error(%)	0.033	6.6(0.405)	0	0
number of genes	All	12(1.699)	60	6.7(1.34)

The numbers in the parentheses are the corresponding standard errors.

We randomly split the dataset into training and test sets with the sample size 68(33 normal samples and 35 prostate cancer samples) and 34 respectively. We repeat it 100 times. The behavior of four methods are listed in the lower part of Table 3, we can see that the SSVM performs better than the other three methods due to the lowest test error and the minimal genes. The frequently selected genes during 100 experiments are listed in Table 5. The 5886 *th* gene and the 9087 *gene* are selected 100 times by SSVM, this means SSVM performs stable in selecting the most important genes.

The third dataset we considered is the Leukemia dataset[9], it contains the expression levels of 7129 genes for 27 patients of acute lymphoblastic leukemia(ALL) and 11 patients of acute myeloid leukemia(AML).Only ten-fold cross validation errors are computed for this datasets, the result is summarized in Table 6. The SSVM selected the minimal genes than the other three models. The test error of SSVM is comparable to the other three methods.

4 Conclusions and the future work

In this paper, we propose the similarity-based support vector machine (SSVM) for microarray classification and gene selection. The SSVM incorporates the prior knowledge of gene similarity into the standard SVM. The numerical experiments show that the new method tends to select the most relevant genes and remove more redundant genes, especially when the genes are highly similar. n the future, it seems interesting to modify the SSVM to solve large scale problem. It is also interesting to incorporate the co-expression network of genes into SVM for gene selection.

Acknowledgement

We would like to thank Professor Naiyang Deng who shares his insights with us in discussions.

References

[1] Guyon I., Weston J., Barnhill S., Vapnik V. Gene selection for cancer classification using support vector machines. *Mach. Learn*. **46:** , 389-422,2002.

[2] Bradley P.S., Mangasarian O.L. Feature selection via concave minimization and support vector machines. *In Proc. 13th ICML*,82-90,1998.

[3] Mangasarian O.L., Wild E.W. Feature selection for nonlinear kernel support vector machines.*IEEE Seventh International Conference on Data Mining (ICDM'07)*,2007.

[4] Neumann J., Schnörr C., Steidl G. Combined SVM-based feature selection and classification. *Mach. Learn.*. **61:** 129-150,2005.

[5] Wang L., Zhu J., Zou H. Hybrid huberized support vector machines for microarray classification and gene selection. *Bioinformatics*, **23:** 2507-2517,2008.

[6] Ashburner M., Ball C.A.., Blake J.A., Botstein D., Butler H., Cherry J.M., Davis A.P., Dolinski K., Dwight S.S., Eppig J.T., Harris M.A., Hill D.P., Issel-Tarver L., Kasarskis A., Lewis S., Matese J.C., Richardson J.E., Ringwald M., Rubin G.M., Sherlock G. Gene ontology: tool for the unification of biology. *The Gene Ontology Consortium*. *Nat. Genet.*,**25:**25-29,2000.

[7] Alon U. , Barkai N. Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon cancer tissues probed by oligonucleotide arrays. *Proc. Natl Acad. Sci. USA*. **96:** 6745-6750,1999.

[8] Zhang H.H., Ahn J., Lin X.D., Park C. Variable selection for svm via shrinkage methods. *Available at http://math.uc.edu/ linxd/paper/9.pdf*

[9] Golub T., Slonim D., Tamayo P., Huard C., Gaasenbeek M., Mesirov J.P., Coller H., Loh M. L., Downing J. R., Caligiuri M. A., Bloomfield C. D., Lander E. S. Molecular classification of cancer: Class discovery and class prediction by gene expression monitoring. *Science*. **286:** 531-537,1999.

[10] Singh D., Febbo P., Ross K., Jackson D., Manola J., Ladd C., Tamayo P., Renshaw A., D'Amico A., Richie J., Lander E., Loda M., Kantoff P., Golub T.,Sellers W. Gene expression correlates of clinical prostate cancer behavior. *cancer cell*,**1:**,203-209,2002.