Support Vector Based T-Score for Gene Ranking

Piyushkumar A. Mundra[1] and Jagath C. Rajapakse[1,2,3]

[1] Bioinformatics Research Center, School of Computer Engineering,
Nanyang Technological University, Singapore
[2] Singapore-MIT Alliance, Singapore
[3] Department of Biological Engineering,
Massachusetts Institute of Technology, USA
asjagath@ntu.edu.sg

Abstract. T-score between classes and gene expressions is widely used for gene ranking in microarray gene expression data analysis. We propose to use only support vector points for computation of t-scores for gene ranking. The proposed method uses backward elimination of features, similar to Support Vector Machine Recursive Feature Elimination (SVM-RFE) formulation, but achieves better results than SVM-RFE and t-score based feature selection on three benchmark cancer datasets.

1 Introduction

Simultaneous measurement of thousands of genes has become possible due to recent advances in DNA microarray technology. Unfortunately, due to high cost of experiments, sample sizes are still very small compared to the number of genes measured. Because of this bottleneck, curse of dimensionality and computational instabilities occur in microarray data analysis, which make it difficult to efficiently extract useful information. To overcome such problems, selection of relevant genes has became extremely important in microarray data analysis.

Various gene selection approaches have been recently proposed by different research groups [1,2,3,4,5,6,7,8,9,10,11]. Gene selection methologies can be broadly classified into two methods: filter methods and wrapper methods [2]. Filter methods evaluate gene subsets by looking at intrinsic characteristics of data with respect to class labels [1]. T-score, P-score, mutual information, euclidean distance, and correlation coefficients are some of the widely used filter criterions [2]. In wrapper approach, the goodness of gene subset is evaluated by estimating the accuracy and the selection is embedded in the specific learning method. Wrapper methods are better in principle but more complex and computationaly expensive. Various algorithms have been developed for gene ranking based on SVM [9, 10, 12]. Support vector machine - recursive feature elimination (SVM-RFE) is one of the widely used wrapper method [12]. SVM-RFE is a multivariate gene ranking method which uses SVM classifier for ranking. SVM-RFE has also been applied to peak selection of mass spectrometry data for cancer classification [13]. Recently, we proposed a linear combination of SVM-RFE with minimum redundancy maximum relevancy based filter criteria to minimize between gene redundancy without affecting classification performance [11].

M. Chetty, A. Ngom, and S. Ahmad (Eds.): PRIB 2008, LNBI 5265, pp. 144–153, 2008.

In filter approach, the standard practice is to consider all the sample points into gene ranking. But, the success of SVM in classification with its excellent generalization capability has proved that only boundary points are important for classification with an optimal margin. We propose a novel method for gene ranking by incorporating t-score in SVM-RFE based ranking to analyze support vector points. In this paper, we investigate the effect of t-score based ranking on classification performance while considering only support vector points. Proposed t-score gene ranking method is formulated in a backward elimination manner as removal of genes from dataset changes support vector points. As seen later, the proposed method showed better performance compared to t-score based or SVM-RFE method on benchmark datasets.

This manuscript is organized as follows: In section 2, we describe the SVM-RFE method and a detailed description of proposed method. Numerical experimental procedures and results are discussed in section 3. Finally, section 4 includes the discussion and conclusion.

2 Method

Let $D = \{x_{ij} : i = 1, 2, ..., n; j = 1, 2, ...m\}$ denotes the microarray gene expression dataset where x_{ij} is the expression measurement of ith gene in jth sample, n represents the total number of measured genes and m denotes the total number of samples. Let $x_j = (x_{1j}, x_{2j}, \ldots, x_{nj})$ be the gene expressions measured in the jth sample. In this paper, we address two class classification of tissue samples in to cancer or benign samples. Let the target class label of jth sample be $y_j \in \{+1, -1\}$ taking values +1 and -1 for being benign and cancerous tissues, respectively.

2.1 Support Vector Machine Recursive Feature Elimination(SVM-RFE)

The objective function for the Support Vector Machines maximize the margin of separation between two classes [14]. The soft-margin SVM is obtained by,

$$\max_\alpha W(\alpha) = \sum_{k=1}^{m} \alpha_k - \frac{1}{2} \sum_{k=1}^{m} \sum_{l=1}^{m} \alpha_k \alpha_l y_k y_l K(x_k, x_l) \tag{1}$$

$$\text{subject to } 0 \leq \alpha_k \leq \zeta, \text{for all } k = 1, \ldots, m \tag{2}$$

$$\text{and } \sum_{k=1}^{m} \alpha_k y_k = 0 \tag{3}$$

where $\{(x_k, y_k) : k = 1, 2, \ldots, m\}$ denotes the training examples. Here, ζ is SVM sensitivity parameter, $K(.,.)$ the Kernel function, and α_k is a parameter obtained by training SVM. SVM formulation only depends on the support vectors to define boundaries as parameters α_k is non-zero only for support vector points.

SVM-RFE technique was developed to rank genes for cancer classification [12]. In SVM-RFE, starting with all genes in the subset, iteratively one can remove gene with least importance for sample classification, given by the weights. This SVM weight vector w is computed using α_k corresponding to support vector points as follows:

$$w = \sum_{k=1}^{m} \alpha_k y_k x_k \tag{4}$$

Support vectors denote data points on the boundaries of and within the separating margins. It can be shown that α_k are zero for non-support vector points. If w_i represents the corresponding component of above weight vector after normalization, the ith gene with smallest ranking score, ${w_i}^2$, is removed from the gene subset. For the computational efficiency, more than one feature can be removed at each step [12] though it may have negative effect on performance of feature selection method if a large portion of features are removed at a time.

2.2 T-Score Based Support Vector Backward Feature Elimination (SV-RFE)

Support vector points represent samples with $0 < \alpha_k \leq \zeta$, i.e., points either lie on the decision boundary or on the wrong side of the margin. In our method, we only concentrate on these points to compute the t-score. The non-support vector points need not be considered for gene ranking. This idea is based on SVM-RFE method where points only with $\alpha_k > 0$ are used for gene ranking.

Let M_+ and M_- subscripts represent set of support vector points corresponding to positive and negative samples. The ranking score for the proposed method is given by [2],

$$r_i = \frac{\left|\mu_{i,M_+} - \mu_{i,M_-}\right|}{\sqrt[2]{\frac{m_{M_+}\sigma^2_{i,M_+} + m_{M_-}\sigma^2_{i,M_-}}{m_{M_+}+m_{M_-}}}} \tag{5}$$

where μ_i and σ_i^2 represent mean and variance of expression values of gene i in respective support vector groups, (M_+ or M_-), m_{M_+} and m_{M_-} denote the number of positive and negative support vector points respectively.

T-statistics compare means of two sets of samples assuming equal variances for both sets. Gene which has higher t-score between the desired and undesired class labels is assumed to have higher class separability. The filter methods utilizing t-statistics have been proven successful in gene selection [1,2]. In standard t-test, all the sample points are considered for score computation. Refering to Eq. (5), instead of taking only M_+ and M_- points (which are support vector points), the previous t-statistics based methods use all points in positive and negative class to compute standard t-score [2].

The psuedocode for t-score based Support Vector Backward Feature Elimination (SV-RFE) is described in Algorithm 1.

Algorithm 1. T-score based Support Vector Backward Feature Elimination

Begin : Ranked gene set R = [ ], and gene subset $S = [1, 2, \ldots, n]$
repeat
Train linear SVM with gene set S in input variable
Obtain the support vector points and compute the ranking score r_i
Select the gene with smallest ranking score $e = \arg\min(r_i)$
Update $R = [e, R]\,; S = S - [e]$
until all genes are ranked
end : output R

Looking from different point of view, the proposed method has some resemblence to original SVM-RFE with certain assumptions. From Eq. (2), it is clear that $\alpha_k \leq \zeta$. After normalizing α vector, this constraint becomes $\alpha_k \leq 1$. Assuming all support vector points have $\alpha_k = 1$ and substituting it in Eq. (4), SVM-RFE weight becomes a simple summation of each gene's expression values. Instead of simple summation, we propose to use statistically more correct t-score based ranking. In a way, proposed method does not use α parameter obtained from SVM learning and in each iteration, model is trained to obtain optimum support vector points. Due to this, our method differs from SVM-RFE significantly. This algorithm is computationally expensive than standard t-score.

3 Experiments and Results

3.1 Data

To evaluate the performance of proposed t-score based SV-RFE method, we performed extensive experiments on three microarray gene expression datasets, namely, Colon [15], Leukemia [1], and Prostate [16] cancer dataset. These are widely used benchmark datasets to evaluate gene ranking methods. In Colon cancer, no separate testing set is available. Hence we divided the original dataset into separate training set and testing set. The number of samples and genes are given in Table 1.

3.2 Preprocessing

To obtain the support vector points, we normalized the training dataset to zero mean and unit variance based on gene expression of a particular gene. These continuous datasets were directly used in SVM-RFE after normalization.

Table 1. Sample Sizes of Three Gene-Expression Datasets

Dataset	# Training	# Testing	# Total Genes
Colon	40	22	2000
Leukemia	38	34	7129
Prostate	102	34	12600

For t-score computation, we use mean centered gene expression dataset (without shifting by unit variance). For t-score based method, we obtain support vector points using zero mean unit variance training data while t-score in each iteration was computed using corresponding sample points in mean centered original gene expression training set.

3.3 Parameter Estimation

Obtaining optimal support vector points is one of the key steps in the proposed method. This depends on sensitivity parameter η in case of linear SVMs. η values were chosen from finite set $\left\{2^{-20}, \ldots, 2^{0}, \ldots, 2^{15}\right\}$ using 10-fold cross-validation (CV). This set was also used for SVM-RFE and test performance evaluation.

CV error is generally employed by either, k-fold CV or *Leave-One-Out*. In present work, we use Matthew's Correlation Coefficient (MCC[1]) with 10-fold cross-validation for training performance evaluation and parameter tuning. MCC was choosen as the error measure because sample size was small and imbalanced in lables in most datasets.

To increase the speed of the numerical simulations with both SVM-RFE and proposed method, we employ following heuristic strategy:

$$\text{Number of genes removed} = \begin{cases} 100 \text{ if } n' \geq 10000 \\ 10 \quad \text{if } 1000 \leq n' < 10000 \\ 1 \quad n' < 1000 \end{cases} \tag{6}$$

where n' is the number of genes in the gene set.

3.4 Performance Evaluation

Ranking of genes in each dataset was obtained using simple t-score, SVM-RFE, and proposed method. Only training data was used to rank the genes using a linear SVM. Using the gene ranking list, we tested gene subsets starting from top ranked gene and then successively adding one gene at a time in testing subset till total number of genes in subset equals 100.

Small sample size in gene expression datasets present a peculiar problem while dividing into training and testing sets. It will not give correct performance evaluation if only one set of testing set is used. This is known as "unfortunate" partitioning of training and testing sets. To solve this "unfortunate" partitioning problem, we merge the training and testing datasets before testing. After that, we employ stratified sampling to partition the total samples into separate training and testing sets by maintaing number of samples in each set as before. Then, the classifier is trained on the training set and tested on the corresponding testing set. This process is followed for 100 times and performance measure such as, test accuracy, sensitivity and specificity were computed for these 100 trials. Finally, total number of genes required for best classification accuracy corresponds to subset with the least average test error.

[1] $MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}}$

Table 2. Performance of t-score, SVM-RFE and Proposed method on Various *Cancer* Datasets

Dataset	Measurement	T-score	SVM-RFE	Proposed Method
Colon	# Genes	95	90	**83**
	Accuracy	88.18 ± 5.29	91.00 ± 5.17	$\mathbf{91.14 \pm 5.22}$
	Sensitivity	82.50 ± 11.92	86.75 ± 10.18	$\mathbf{87.12 \pm 11.16}$
	Specificity	91.43 ± 6.09	$\mathbf{93.43 \pm 5.53}$	$\mathbf{93.43 \pm 5.71}$
Leukemia	# Genes	88	**47**	64
	Accuracy	96.88 ± 3.44	97.88 ± 2.07	$\mathbf{98.41 \pm 1.79}$
	Sensitivity	92.64 ± 8.40	95.00 ± 5.13	$\mathbf{96.21 \pm 4.24}$
	Specificity	99.85 ± 1.11	99.90 ± 0.70	$\mathbf{99.95 \pm 0.50}$
Prostate	# Genes	85	85	**21**
	Accuracy	93.41 ± 3.79	96.24 ± 3.37	$\mathbf{97.18 \pm 2.89}$
	Sensitivity	92.84 ± 4.93	95.88 ± 4.08	$\mathbf{96.88 \pm 3.49}$
	Specificity	95.00 ± 7.80	97.22 ± 5.56	$\mathbf{98.00 \pm 4.57}$

Table 3. Comparison of accuracies with the published results

Method/Dataset	Colon		Leukemia		Prostate	
	Accuracy	# of Genes	Accuracy	# of Genes	Accuracy	# of Genes
Bayes + KNN [8]	90.32	6	100.00	3	94.12	11
Bayes + SVM [8]	87.10	20	**100.00**	**2**	96.08	13
t-test + Fisher Classifier [19]	88.30	...	88.00	...	92.00	...
MMC-RFE + NMC [20]	88.80	100	99.20	100	90.10	**10**
Proposed Method + SVM	**91.14**	83	98.41	64	**97.18**	21

We also compared the results with SVM-RFE method. This method was performed in exactly the same way as that of proposed method except ranking criteria. In all gene selection methods and testing the classifier, we used LIBSVM - 2.84 software [17].

3.5 Results

The proposed method has remarakably good performance than t-score method in all three gene expression dataset. Both sensitivity and specificity are improved in all datasets. Figures 1,2, and 3 represent the average test misclassification error rate in each of the three datasets. Also, except Prostate Cancer dataset, our method needed less number of genes for classification compared to t-score method. The proposed method also have comparable performance with SVM-RFE method.

Table 3 shows a comparison of classification accuracy with other methods available in the literature. As seen in the table, our method performed reasonably well in all three datasets. Classification performance is much better in Prostate cancer dataset. In Leukemia dataset, out method is inferior to Leave one out

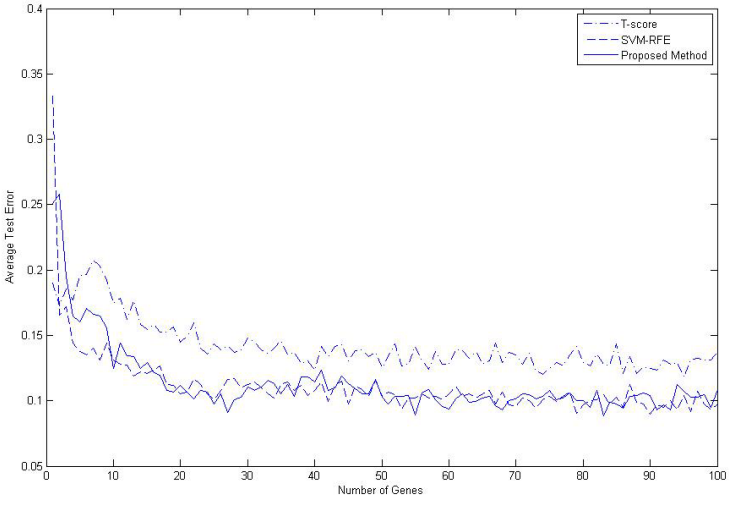

Fig. 1. Average misclassification error rate for all three methods on Colon Cancer Dataset against the number of genes

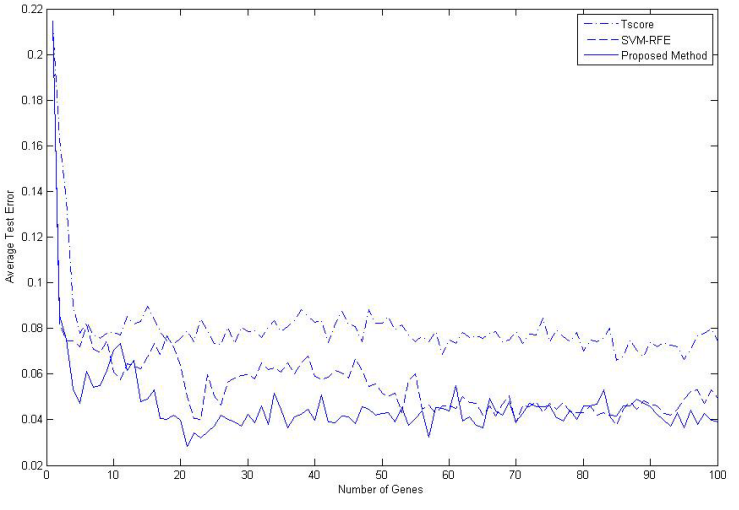

Fig. 2. Average misclassification error rate for all three methods on Prostate Cancer Dataset against the number of genes

(LOO) method but better than both 10-fold and 100-split testing. As compared and discussed in [18], LOO gives optimistic accuracy estimations compared to both k-fold cross validation and bootstrap method.

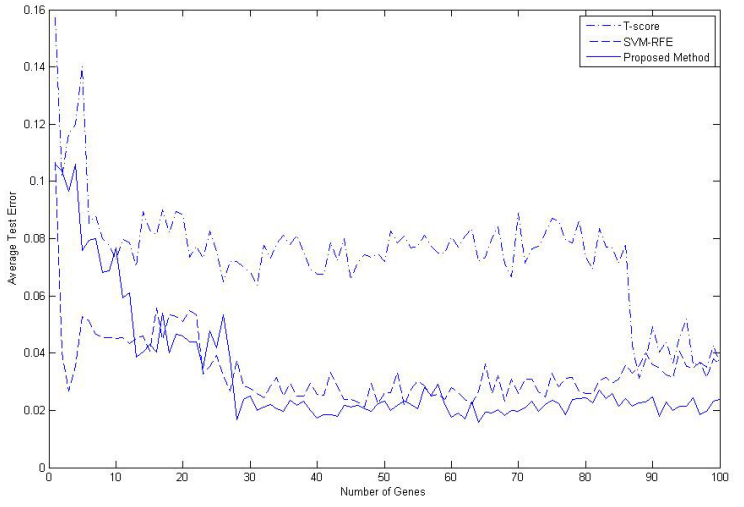

Fig. 3. Average misclassification error rate for all three methods on Leukemia Cancer Dataset against the number of genes

4 Discussion

We propose a support vector based t-score method for gene ranking. We evaluated performance of the proposed method on three benchmark datasets and showed remarkable improvement in accuracy compare to standard t-score. Performance results are quite comparable to SVM-RFE.

In practice, standard t-score based approach considers all the data points in the training set. But as shown in SVM based classification, only the data points which lie on the boundary are important for decision making. Based on success of such strategy, our approach only considers data points obtained from SVM model. Because of only considering support vector points, statistically we lose some degree of freedoms. But as shown in the results, only concentrating on support points improves the classification performance.

Removal of one gene can change support vector points, and hence t-score will change. To incorporate such effect, we use backward elimination based SVM-RFE approach with t-score criteria in gene ranking. This approach is different from standard t-score method where all the genes are ranked in one iteration.

We would like to reemphasize that the proposed method does not use α parameter obtained from SVM models. As discussed in the methods section, if α value is assumed to be 1 for all support vector points, SVM-RFE weight criteria is simple summation of gene expression values. In the proposed method, we use statistical t-score, which ranks genes based on mean and variance of gene expression values in cancerous and benign tissue samples. This results improved the classification performance. Only similarity with SVM-RFE is that, in each iteration, specified numbers of genes were removed and new t-score was calculated

for new SVM model. As number of support vector points change in each iteration, and hence mean and variance of gene, our method formulation indirectly changes univariate t-score into multivariate system. It would be interesting to see if same hypothesis of using only support vectors can be applied with other filter criteria.

In conclusion, we proposed a novel support vector based t-score computation in SVM-RFE formulation. Extensive testing on three benchmark cancer classification gene-expression dataset revealed that proposed method perfoms significantly better than standard t-score appraoch and results are comparable with SVM-RFE.

References

1. Golub, T., Slonim, D., Tamayo, P., Huard, C., Gaasenbeek, M., Mesirov, J., Coller, H., Loh, M., Downing, J., Caligiuri, M., Bloomfield, C., Lander, E.: Molecular classification of cancer: Class discovery and class prediction by gene expression. Science 286, 531–537 (1999)
2. Inza, I., Larranaga, P., Blanco, R., Cerrolaza, A.: Filter versus wrapper gene selection approaches in DNA microarray domains. Artificial Intelligence Medicine 31, 91–103 (2004)
3. Battiti, R.: Using mutual information for selecting features in supervised neural net learning. IEEE Trans Neural Network 5, 537–550 (1994)
4. Liu, X., Krishnan, A., Mondry, A.: An entropy-based gene selection method for cancer classification using microarray data. BMC Bioinformatics 6, 76 (2005)
5. Ding, C., Peng, H.: Minimum redundancy feature selection from microarray gene expression data. J. Bioinformatics Computational Biology 3, 185–205 (2005)
6. Peng, H., Long, F., Ding, C.: Feature selection based on mutual information: Criteria of max-dependency, max-relevance, and min-redundancy. IEEE Trans. Pattern Analysis Machine Intelligence 27, 1226–1237 (2005)
7. Ooi, C., Chetty, M., Teng, S.: Differential prioritization between relevance and redundancy in correlation-based feature selection techniques for multiclass gene expression data. BMC Bioinformatics 7, 320–339 (2006)
8. Zhang, J., Deng, H.: Gene selection for classification of microarray data based on bayes error. BMC Bioinformatics 8, 370 (2007)
9. Rakotomamonjy, A.: Variable selection using svm criteria. J. Machine Learning Research (Special Issue on Variable Selection) 3, 1357–1370 (2003)
10. Kai-Bo, D., Rajapakse, J., Wang, H., Azuaje, F.: Multiple SVM-RFE for gene selection in cancer classification with expression data. IEEE Trans. Nanobioscience 4, 228–234 (2005)
11. Mundra, P., Rajapakse, J.: SVM-RFE with relevancy and redundancy criteria for gene selection. In: Rajapakse, J., Schmidt, B., Volkert, L.G. (eds.) PRIB 2007. LNCS (LNBI), vol. 4774, pp. 242–252. Springer, Heidelberg (2007)
12. Guyon, I., Weston, J., Barhill, S., Vapnik, V.: Gene selection for cancer classification using support vector machines. Machine Learning 46, 389–422 (2002)
13. Rajapakse, J., Kai-Bo, D., Yeo, W.: Proteomic cancer classification with mass spectrometry data. American. J. Pharmacogenomics 5, 281–292 (2005)
14. Hastie, T., Tibshirani, R., Friedman, J.: The Elements of Statistical Learning: Data Mining, Inference, and Prediction. Springer, Heidelberg (2001)

15. Alon, U., Barkai, N., Notterman, D., Gish, K., Ybarra, S., Mack, D., Levine, A.: Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays. PNAS 96, 6745–6750 (1999)
16. 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)
17. Chang, C., Lin, C.: Libsvm: A library for support vector machines (2001), `www.csie.ntu.edu.tw/~cjlin/libsvm`
18. Azuaze, F.: Genomic data sampling and its effect on classification performance assessment. BMC Bioinformatics 4, 5 (2003)
19. Lai, C., Reinders, M., van't Veer, L., Wessels, L.: A comparison of univariate and multivariate gene selection techniques for classification of cancer datasets. BMC Bioinformatics 7, 235 (2006)
20. Niijima, S., Kuhara, S.: Recursive gene selection based on maximum margin criterion: a comparison with svm-rfe. BMC Bioinformatics 7, 543 (2006)