Gene Feature Extraction Using T-Test Statistics and Kernel Partial Least Squares

Shutao Li[1], Chen Liao[1], and James T. Kwok[2]

[1] College of Electrical and Information Engineering
Hunan University
Changsha 410082, China
[2] Department of Computer Science
Hong Kong University of Science and Technology
Clear Water Bay, Hong Kong
shutao_li@yahoo.com.cn, lc337199@sina.com, jamesk@cs.ust.hk

Abstract. In this paper, we propose a gene extraction method by using two standard feature extraction methods, namely the T-test method and kernel partial least squares (KPLS), in tandem. First, a preprocessing step based on the T-test method is used to filter irrelevant and noisy genes. KPLS is then used to extract features with high information content. Finally, the extracted features are fed into a classifier. Experiments are performed on three benchmark datasets: breast cancer, ALL/AML leukemia and colon cancer. While using either the T-test method or KPLS does not yield satisfactory results, experimental results demonstrate that using these two together can significantly boost classification accuracy, and this simple combination can obtain state-of-the-art performance on all three datasets.

1 Introduction

Gene expression studies by DNA microarrays provide unprecedented chances because researchers can measure the expression level of tens of thousands of genes simultaneously. Using this microarray technology, a comprehensive understanding of exactly which genes are being expressed in a specific tissue under various conditions can now be obtained [3].

However, since the gene dataset usually includes only a few samples but with thousands or even tens of thousands of genes, such a limited availability of high-dimensional samples is particularly problematic for training most classifiers. As such, oftentimes, dimensionality reduction has to be employed. Ideally, a good dimensionality reduction method should eliminate genes that are irrelevant, redundant, or noisy for classification, while at the same time retain all the highly discriminative genes [11].

In general, there are three approaches to gene (feature) extraction, namely, the filter, wrapper and embedded approaches. In the filter approach, genes are selected according to the intrinsic characteristics. It works as a preprocessing step without the incorporation of any learning algorithm. Examples include the nearest shrunken centroid method, TNoM-score based method and the T-statistics

I. King et al. (Eds.): ICONIP 2006, Part III, LNCS 4234, pp. 11–20, 2006.

method [8]. In the wrapper approach, a learning algorithm is used to score the feature subsets based on the resultant predictive power, and an optimal feature subset is searched for a specific classifier [4]. Examples include recursive feature elimination, and genetic algorithm-based algorithms.

In this paper, we propose a new gene extraction method based on the filter approach. First, genes are preprocessed by the T-test method to filter irrelevant and noisy genes. Then, kernel partial least squares (KPLS) is used to extract features with high information content and discriminative power. The rest of this paper is organized as follows. In Section 2, we first review the T-test method and KPLS. The new gene extraction method is presented in Section 3. Section 4 then presents the experimental results, which is followed by some concluding remarks.

2 Review

In the following, we suppose that a microarray dataset containing n samples is given, with each sample x represented by the expression levels of m genes.

2.1 T-Test Method

The method is based on the t-statistics [7]. Denote the two classes as positive $(+)$ class and negative $(-)$ class. For each feature x_j, we compute the mean μ_j^+ (respectively μ_j^-) and standard deviation δ_j^+ (respectively δ_j^-) for the $+$ class (respectively, $-$ class) samples. Then a score $T(x_j)$ can be obtained as:

$$T(x_j) = \frac{|\mu_j^+ - \mu_j^-|}{\sqrt{\frac{(\delta_j^+)^2}{n_+} + \frac{(\delta_j^-)^2}{n_-}}},$$

where n_+ and n_- are the numbers of samples in the positive and negative classes respectively.

2.2 Kernel Partial Least Squares (KPLS)

Given a set of input samples $\{x_i\}_{i=1}^n$ (where each $x_i \in \mathbb{R}^m$) and the corresponding set of outputs $\{y_i\}_{i=1}^n$ (where $y_i \in \mathbb{R}$). Here, only one-dimensional output is needed because only two-class classification is considered. With the use of a kernel, a nonlinear transformation of the input samples $\{x_i\}_{i=1}^n$ from the original input space into a feature space $\mathcal{F}$ is obtained, i.e. mapping $\phi : x_i \in \mathbb{R}^m \to \phi(x_i) \in \mathcal{F}$. The aim of KPLS is then to construct a linear PLS model in this kernel-induced feature space $\mathcal{F}$. Effectively, a nonlinear kernel PLS in the original input space is obtained and the mutual orthogonality of the score vectors can be retained.

Let Φ be the $n \times m'$ matrix of input samples in the feature space $\mathcal{F}$, and its ith row be the vector $\phi(x_i)^T$. Let m' be the dimensionality of $\phi(x_i)$, which can be infinite. Denote ϕ' the $n \times m'$ deflated dataset and Y' the $n \times 1$ deflated class label. Then the rule of deflation is

$$\phi' = \phi - t(t^T\phi), \qquad (1)$$
$$Y' = Y - t(t^TY).$$

Here, t is the score vector (component) which is obtained in the following way. Let w and c be the weight vectors. The process starts with random initialization of the Y-score u and then iterates the following steps until convergence:

1: $w = X^Tu/(u^Tu)$;
2: $\|w\| \to 1$;
3: $t = Xw$;
4: $c = Y^Tt/t^Tt$;
5: $u = Yc/(c^Tc)$;
6: Repeat steps 1.-5.

The process is iterated for Fac times. Subsequently, the deflated dataset can be obtained from the original dataset and the PLS component, while the deflated class label be obtained from the original class labels and the PLS component.

Denote the sequence of t's and u's obtained $n \times 1$ vectors $t_1, t_2, \ldots, t_{Fac}$ and $u_1, u_2, \ldots, u_{Fac}$, respectively. Moreover, let $T = [t_1, t_2, \ldots, t_{Fac}]$ and $U = [u_1, u_2, \ldots, u_{Fac}]$. The "kernel trick" can then be utilized instead of explicitly mapping the input data, and results in: $K = \Phi\Phi^T$, where K stands for the $n \times n$ kernel matrix: $K(i,j) = k(x_i, x_j)$, where k is the kernel function. K can now be directly used in the deflation instead of ϕ, as

$$K' = (I_n - tt^T)K(I_n - tt^T). \qquad (2)$$

Here, K' is the deflated kernel matrix and I_n is n-dimensional identity matrix. Now Eq.(2) takes the place of Eq.(1). So deflated kernel matrix is obtained by the original kernel matrix and the PLS component. In kernel PLS, the assumption that the variables of X have zero mean in linear PLS should also hold. The procedure must be applied to centralize the mapped data in the feature space $\mathcal{F}$ as:

$$K = (I_n - \frac{1}{n}1_n1_n^T)K(I_n - \frac{1}{n}1_n1_n^T).$$

Here, 1_n is the $n \times 1$ vector with all elements equal to one. Given a set of test samples $\{z_i\}_{i=1}^n$ (where $z_i \in \mathbb{R}^m$), its projection into the feature space is

$$D_p = K_tU(T^TKU)^{-1},$$

where $D_p = [d_1, d_2, \ldots, d_{n_i}]^T$ is a $n_t \times p$ matrix, the columns of D_p are the p KPLS components and the rows of D_p are the n_t test samples in the reduced-dimensional space. K_t is the $n_t \times n$ kernel matrix defined on the test set such that $K_t(i,j) = k(z_i, x_j)$. T^TKU is an upper triangular matrix and thus invertible. The centralized test set kernel Gram matrix K_t can be calculated by [10,9]

$$K_t = (K_t - \frac{1}{n}1_n1_n^T)K(I_n - \frac{1}{n}1_n1_n^T).$$

3 Gene Extraction Using T-Test and KPLS

While one can simply use the T-test method or KPLS described in Section 2 for gene extraction, neither of them yields satisfactory performance in practice[1]. In this paper, we propose using the T-test and KPLS in tandem in performing gene extraction. Its key steps are:

1: (Preprocessing using T-test): Since the samples are divided into two classes, one can compute the score for each gene by using the T-statistics. Those genes with scores greater than a predefined threshold T are considered as discriminatory and are selected. On the other hand, those genes whose scores are smaller than T are considered as irrelevant/noisy and are thus eliminated.
2: (Feature extraction using KPLS): The features extracted in the first step are further filtered by using KPLS.
3: (Training and Classification): Using the features extracted, a new training set is formed which is then used to train a classifier. The trained classifier can then be used for predictions on the test set.

A schematic diagram of the whole process is shown in Figure 1.

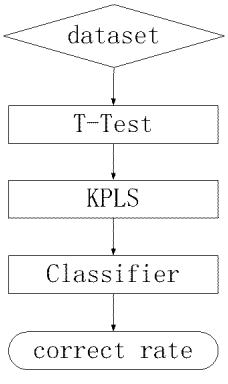

Fig. 1. The combined process of gene extraction and classification

4 Experiments

4.1 Setup

In this section, we evaluate the performance of the proposed gene selection method on three benchmark datasets:

1. Breast cancer dataset: It contains 7,129 genes and 38 samples. 18 of these samples are ER+ (estrogen receptor) while the remaining 20 are ER− [12].

[1] This will be experimentally demonstrated on several benchmark datasets in Section 4.

Table 1. Parameter used in the classifiers

	breast cancer	leukemia	colon cancer
K in K-NN	3	13	14
number of hidden units in NN	3	3	3
soft-margin parameter (C) in SVM	1	10	100

Table 2. Testing accuracies (%) when either the T-test or KPLS is used

		breast cancer	leukemia	colon cancer
T-test only	$T = 1000$	86.8	97.2	82.3
	$T = 2200$	76.3	93.1	79.0
	$T = 2500$	65.8	95.8	82.3
KPLS only	$\gamma = 2$	89.5	93.1	88.7
	$\gamma = 5$	89.5	93.1	85.5

Table 3. Testing accuracy (%) on the breast cancer dataset

T	γ	Fac	K-NN	NN	SVM
500	100	10	94.7	97.4	97.4
		15	94.7	97.4	97.4
		20	76.3	97.4	97.4
	200	10	94.7	100.0	100.0
		15	94.7	100.0	100.0
		20	84.2	100.0	100.0
	300	10	94.7	100.0	100.0
		15	92.1	100.0	100.0
		20	86.8	100.0	100.0
1000	100	10	92.1	100.0	100.0
		15	89.5	100.0	100.0
		20	94.7	100.0	100.0
	200	10	94.7	100.0	100.0
		15	97.4	100.0	100.0
		20	97.4	100.0	100.0
	300	10	92.1	100.0	100.0
		15	92.1	100.0	100.0
		20	97.4	100.0	100.0
1500	100	10	81.6	86.8	86.8
		15	84.2	86.8	84.2
		20	81.6	89.4	86.8
	200	10	81.6	84.2	84.2
		15	81.6	86.8	81.6
		20	84.2	86.8	84.2
	300	10	81.6	81.6	84.2
		15	81.6	81.6	84.2
		20	81.6	84.2	84.2

Table 4. Testing accuracy (%) on the leukemia dataset

T	γ	Fac	K-NN	NN	SVM
2000	100	10	97.2	98.6	98.6
		15	95.8	98.6	98.6
		20	97.2	98.6	98.6
	200	10	97.2	98.6	98.6
		15	98.6	98.6	98.6
		20	95.8	98.6	98.6
	300	10	97.2	98.6	98.6
		15	98.6	98.6	98.6
		20	98.6	98.6	98.6
2500	100	10	94.5	98.6	98.6
		15	95.8	98.6	98.6
		20	91.7	98.6	98.6
	200	10	94.4	100.0	100.0
		15	97.2	100.0	100.0
		20	83.3	100.0	100.0
	300	10	94.4	100.0	100.0
		15	94.4	100.0	100.0
		20	83.3	100.0	100.0
3000	100	10	95.8	100.0	100.0
		15	90.3	100.0	100.0
		20	86.1	100.0	100.0
	200	10	95.8	100.0	100.0
		15	90.3	100.0	100.0
		20	79.2	100.0	100.0
	300	10	95.8	98.6	98.6
		15	90.3	98.6	98.6
		20	76.4	98.6	98.6

2. Leukemia dataset: It contains 7,129 genes and 72 samples. 47 of these samples are of Acute Myeloid Leukemia (AML) and the remaining 25 are of Acute Lymphoblastic Leukemia (ALL) [5].
3. Colon cancer dataset: It contains 2,000 genes and 62 samples. 22 of these samples are of normal colon tissues and the remaining 40 are of tumor tissues [1].

Using the genes selected, the following classifiers are constructed and compared in the experiments:

1. K-nearest neighbor classifier (k-NN).
2. Feedforward neural network (NN) with a single layer of hidden units. Here, we use the logistic function for the hidden units and the linear function for the output units. Back-propagation with adaptive learning rate and momentum is used for training.
3. Support vector machine (SVM). In the experiments, the linear kernel is always used.

Table 5. Testing accuracy (%) on the colon cancer dataset

T	γ	Fac	K-NN	NN	SVM
1700	100	2	87.1	88.7	90.3
		5	88.7	88.7	88.7
		10	87.1	88.7	88.7
	200	2	87.1	88.7	90.3
		5	88.7	88.7	88.7
		10	87.1	88.7	88.7
	300	2	87.1	90.3	90.3
		5	88.7	82.3	90.3
		10	83.9	90.3	90.3
2200	100	2	87.1	91.9	90.3
		5	91.9	87.1	90.3
		10	90.3	91.9	91.9
	200	2	87.1	88.7	90.3
		5	91.9	87.1	90.3
		10	90.3	90.3	90.3
	300	2	87.1	90.3	90.3
		5	91.9	87.1	90.3
		10	87.1	90.3	90.3
2500	100	2	88.7	88.7	88.7
		5	87.1	79.0	87.1
		10	82.3	82.3	85.5
	200	2	88.7	87.1	88.7
		5	87.1	85.5	87.1
		10	80.7	83.9	80.7
	300	2	88.7	90.3	88.7
		5	87.1	83.9	87.1
		10	82.3	83.9	85.5

Each of these classifiers involves some parameters. The parameter settings used on the different datasets are shown in Table 1. Because of the small training set size, leave-one-out (LOO) cross validation is used to obtain the testing accuracy. Both gene selection and classification are put together in each LOO iteration, i.e., they are trained on the training subset and then the performance of the classifier with the selected features is assessed with the left out examples.

4.2 Results

There are three adjustable parameters in the proposed method:

1. The threshold T associated with the T-test method;
2. The width parameter γ in the Gaussian kernel

$$k(x, y) = \exp(-\|x - y\|^2/\gamma),$$

   used in KPLS;
3. The number (Fac) of score vectors used in KPLS.

Table 6. Testing accuracies (%) obtained by the various methods as reported in the literature

classifier	breast cancer	leukemia	colon cancer
Adaboost (decision stumps) [2]	-	95.8	72.6
SVM (quadratic kernel) [2]	-	95.8	74.2
SVM (linear kernel) [2]	97.4	94.4	77.4
RVM (linear kernel) [6]	94.7	94.4	80.6
RVM (no kernel) [6]	89.5	97.2	88.7
logistic regression (no kernel) [6]	-	97.2	71.0
sparse probit regression (quadratic kernel) [6]	-	95.8	84.6
sparse probit regression (linear kernel) [6]	97.4	97.2	91.9
sparse probit regression (no kernel) [6]	84.2	97.2	85.5
JCFO (quadratic kernel) [6]	-	98.6	88.7
JCFO (linear kernel) [6]	97.4	**100.0**	**96.8**
proposed method	**100.0**	**100.0**	91.9

As a baseline, we first study the individual performance of using either the T-test method and KPLS for gene extraction. Here, only the SVM is used as the classifier. As can be seen from Table 2, the accuracy is not high. Moreover, the performance is not stable when different parameter settings are used.

We now study the performance of the proposed method that uses both the T-test method and KPLS in tandem. Testing accuracies, at different parameter settings, on the three datasets are shown in Tables 3, 4 and 5, respectively. As can be seen, the proposed method can reach the best classification performance of 100% on both the breast cancer and leukemia datasets. On the colon cancer dataset, it can also reach 91.9%.

Besides, on comparing the three classifiers used, we can conclude that the neural network can attain the same performance as the SVM. However, its training time is observed to be much longer than that of the SVM. On the other hand, the K-NN classifier does not perform as well in our experiments.

We now compare the performance of the proposed method with those of the other methods as reported in the literature. Note that all these methods are evaluated using leave-one-out cross-validation and so their classification accuracies can be directly compared. As can be seen in Table 6, the proposed method, which attains the best classification accuracy (of 100%) on both the breast cancer and leukemia datasets, outperforms most of the methods. Note that the *Joint Classifier and Feature Optimization* (JCFO) method [6] (using the linear kernel) can also attain 100% on the Leukemia dataset. However, JCFO relies on the Expectation-Maximization (EM) algorithm [6] and is much slower than the proposed method.

5 Conclusions

In this paper, we propose a new gene extraction scheme based on the T-test method and KPLS. Experiments are performed on the breast cancer, leukemia and colon cancer datasets. While the use of either the T-test method or KPLS for gene extraction does not yield satisfactory results, the proposed method, which uses both the T-test method and KPLS in tandem, shows superior classification performance on all three datasets. The proposed gene extraction method thus proves to be a reliable gene extraction method.

Acknowledgment

This paper is supported by the National Nature Science Foundation of China (No. 6040204) and Program for New Century Excellent Talents in University.

References

1. U. Alon, N. Barkai, D.A. Notterman, K. Gish, S. Ybarra, D. Mack, and A.J. Levine. Broad patterns of gene expression revealed by clustering of tumor and normal colon tissues probed by oligonucleotide arrays. *Proceedings of the National Academy of Science*, 96:6745–6750, 1999.
2. A. Ben-Dor, L. Bruhn, N. Friedman, I. Nachman, M. Schummer, and Z. Yakhini. Tissue classification with gene expression profiles. In *Proceedings of the Fourth Annual International Conference on Computational Molecular Biology*, pages 54–64, 2000.
3. H. Chai and C. Domeniconi. An evaluation of gene selection methods for multi-class microarray data classification. In *Proceedings of the Second European Workshop on Data Mining and Text Mining for Bioinformatics*, pages 3–10, Pisa, Italy, September 2004.
4. K. Duan and J.C. Rajapakse. A variant of SVM-RFE for gene selection in cancer classification with expression data. In *Proceedings of the IEEE Symposium on Computational Intelligence in Bioinformatics and Computational Biology*, pages 49–55, 2004.
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(5439):531–537, 1999.
6. B. Krishnapuram, L. Carin, and A. Hartemink. Gene expression analysis: Joint feature selection and classifier design. In B. Schölkopf, K. Tsuda, and J.-P. Vert, editors, *Kernel Methods in Computational Biology*, pages 299–318. MIT, 2004.
7. H. Liu, J. Li, and L. Wong. A comparative study on feature selection and classification methods using gene expression profiles and proteomic patterns. *Genome Informatics*, 13:51–60, 2002.
8. B. Ni and J. Liu. A hybrid filter/wrapper gene selection method for microarray classification. In *Proceedings of International Conference on Machine Learning and Cybernetics*, pages 2537–2542, 2004.
9. R. Rosipal. Kernel partial least squares for nonlinear regression and discrimination. *Neural Network World*, 13(3):291–300, 2003.

10. R. Rosipal, L.J. Trejo, and B. Matthews. Kernel PLS-SVC for linear and nonlinear classification. In *Proceedings of the Twentieth International Conference on Machine Learning*, pages 640–647, Washington, D.C., USA, August 2003.
11. Y. Tang, Y.-Q. Zhang, and Z. Huang. FCM-SVM-RFE gene feature selection algorithm for leukemia classification from microarray gene expression data. In *Proceedings of IEEE International Conference on Fuzzy Systems*, pages 97–101, 2005.
12. M. West, C. Blanchette, H. Dressman, E. Huang, S. Ishida, R. Spang, H. Zuzan, J.A. Olson Jr., J.R. Marks, and J.R. Nevins. Predicting the clinical status of human breast cancer by using gene expression profiles. *Proceedings of the National Academy of Science*, 98(20):11462–11467, 2001.