

Data-dependent Kernel Machines for Microarray Data Classification

Huilin Xiong, Ya Zhang, and Xue-Wen Chen, *Senior Member, IEEE*

Abstract

One important application of gene expression analysis is to classify tissue samples according to their gene expression levels. Gene expression data are typically characterized by high dimensionality and small sample size, which makes the classification task quite challenging. In this paper, we present a data-dependent kernel for microarray data classification. This kernel function is engineered so that the class separability of the training data is maximized. A bootstrapping-based resampling scheme is introduced to reduce the possible training bias. The effectiveness of this adaptive kernel for microarray data classification is illustrated with a k-Nearest Neighbor (KNN) classifier. Our experimental study shows that the data-dependent kernel leads to a significant improvement in the accuracy of KNN classifiers. Furthermore, this kernel-based KNN scheme has been demonstrated to be competitive to, if not better than, more sophisticated classifiers such as Support Vector Machines (SVMs) and the Uncorrelated Linear Discriminant Analysis (ULDA) for classifying gene expression data.

Index Terms

Microarray data analysis, cancer classification, kernel machines, kernel optimization, bootstrapping resampling.

I. INTRODUCTION

Manuscript received March 15, 2006; revised August 31, 2006; accepted October 23, 2006.

Huilin Xiong, Ya Zhang, and Xue-wen Chen are with Department of Electrical Engineering and Computer Science, University of Kansas, Kansas, USA 66045. Emails: {hlxiong, yazhang, xwchen}@ittc.ku.edu.

MICROARRAY technology, which simultaneously measures the expression levels of tens of thousands of genes, represents an efficient way to characterize cells at the molecular level. In recent years, microarray technology has found many important applications, ranging from fundamental biological studies to clinical studies. For example, genes' expressions are monitored at different conditions or different cellular stages to reveal the functions of genes [1] as well as their regulatory interactions [2]. Gene expression of disease tissues may be use to gain a better understanding of many diseases such as different type of cancers [3], [4]. Gene expression also reveals cellular responses to drug treatment [5]. For example, microarray experiments have been increasingly used to identify drug responsive genes [6], predict treatment outcomes, and select potential drug targets [7].

Compared with small-scale gene expression study (e.g. northern blotting analysis) where only a few genes' expression levels are measured, microarray experiments are high-throughput in nature. Typically, thousands of genes are involved in a single microarray experiment. The large volume of microarray data makes manual analysis of the data impractical, if not impossible. Computer-aided data analysis, especially machine learning techniques, has been widely used for gene expression analysis. In general, three types of machine learning techniques have been applied to microarray data analysis: clustering, classification, and feature selection. Clustering techniques [8] group genes according to similarity in their expression profiles, aiming at discovering genes that are co-regulated or members of the same regulatory network. Examples of clustering techniques are hierarchical clustering [9], self-organizing maps [10], and k-means clustering [11]. Classification techniques are generally applicable when we have a known set of samples (with their gene expression profiles) and want to predict some type of membership for new unknown samples based on their expression profiles. We defer a detailed discussion of microarray classification to latter part of the paper. Feature selection methods [12], [13], [14] are usually used in combination with classification techniques, where the aim is to choose a small subset of features (genes) as the most distinctive characteristics among different types of samples.

In this study, we investigate the problem of microarray classification where gene expression profiles are used as the basis for classifying different types of samples. For

the rest of the paper, we will use cancer tissue classification as an illustrating example. Specifically, given a collection of gene expression profiles for tissue samples belonging to various cancer types, our goal is to build a classifier to automatically determine the cancer type to a new sample at high precision. Classifying cancer tissue based on their gene expression profiles have the promise of providing more reliable means to diagnose and predict various types of cancers. However, it is worthy noting that the classification scheme we proposed here is designed for general microarray classification and should by no means be limited to cancer tissue classification tasks.

Gene expression data are typically characterized by high dimensionality (i.e. a large number of genes) and small sample size. The curse of dimensionality is one of the major challenges in microarray data classification. Kernel tricks represent one way to cope with the curse of dimensionality in many machine learning tasks. A kernel function implicitly maps data from input space into a new feature space. The mapping is implicit because the inner product of two feature mappings is evaluated without knowing the actually feature mapping. Due to their appealing features, kernel methods have attracted a lot of attentions in pattern recognition and machine learning community. Many application-specific kernels have been engineered for various computational biology and bioinformatics applications [15].

In the literature, a number of methods have been applied or developed to classify microarray data [3], [4], [16], [17], [18], [19], [20]. These methods include k-nearest-neighbor (KNN), boosting [18], linear discriminant analysis (LDA) [17], [21], and SVMs. Most of them require a predefined similarity or distance metric, and their performances rely largely on how well the metric reflect the real relationship among samples. Popularly used metrics include Euclidean distance, Manhattan distance, and Pearson-correlation. However, in practice, it is desirable that the metric be data-dependent or adaptive to the input data. The notion of adaptive distance metric, which depends on the local feature of the data, is introduced in [22], where the authors call their local LDA-based metric as discriminant adaptive nearest neighbor (DANN) metric. However, this metric does not address the issue for the curse of dimensionality and is not applicable directly to the microarray data classification problem, since the so-called “within sum-of-square matrix” [22] is always singular in the case of gene expression data.

Kernel function can be thought of as a similarity measure between the input objects. Adaptive similarity metrics may be achieved through kernel design. In this paper, we present an adaptive similarity metric for microarray data classification. This similarity metric is obtained by optimizing a data-dependent kernel, aiming at increasing the class separability of the training data. We illustrate the effectiveness of this adaptive metric associated with the k -Nearest Neighbor (KNN) classifier for cancer classification. Considering the high dimensionality and relatively small sample size in gene expression data, a bootstrapping-based resampling scheme is introduced to reduce the possible training bias. The proposed kernel-based KNN algorithm, denoted by KerNN, is applied to several sets of cancer expression data. The experimental results show that KerNN always achieves remarkable improvement over the conventional KNN method with Euclidean distance metric in classifying gene expression data. Compared with several other well-known methods, the KerNN classifier achieves comparable results, if not better, on nine public microarray data sets.

The rest of the paper is organized as follows. Section II provides a brief review of some related work. In Section III, after introducing a data-dependent kernel model, we develop an efficient way to optimize the kernel. The kernel metric derived from the optimized kernel is formulated. Section IV presents the experimental results on nine publicly available gene expression data sets. Section V gives some detailed information about the experimental settings and parameter tuning, illustrates that the kernel optimization can substantially improve the class separability of the data. Finally, Section VI concludes the paper.

II. RELATED WORK

Numerous methods for tissue classification using gene expression data have been reported in literature. In this section, we briefly review four well-known methods, which is used to compare with our method.

K-Nearest Neighbor (KNN): The KNN method is the simplest, yet useful approach to general pattern classification. Its error rate has been proven to be asymptotically at most twice that of the Bayesian error rate [23]. However, its performance deteriorates dramatically when the input data set has a relatively low local relevance [24].

Diagonal Linear Discriminant Analysis (DLDA): DLDA is the simplest case of the maximum likelihood discriminant rule, in which the class densities are supposed to have the same diagonal covariance matrix. In the special case of binary classification, the DLDA scheme can be viewed as the “weighted voting scheme” proposed by Golub *et al.* in [19].

Linear Discriminant Analysis (LDA): LDA is an important method in general pattern classification. The classical LDA method works by searching the most discriminatory projection directions of the input data and classifies the data in the projected space. A major problem in employing the classical LDA scheme to classify gene expression data is that the so-called scatter matrices are always singular, due to the nature of high dimensionality and relatively small sample size in microarray data. To handle this problem, generalized linear discriminant analysis (GLDA) [25] and uncorrelated linear discriminant analysis (ULDA) [21] are developed recently by using more delicate matrix techniques to modify the classic LDA into a more general version. The ULDA scheme is applied to classify microarray data in [21].

Support vector machines (SVM): The support vector machines work by searching a hyperplane that separates the classes of the input data with the maximum margin. SVM has been recognized as the most powerful classifier in various applications of pattern classification. It has been shown to perform well in a wide range of tasks in computational biology, including but not limited to classifying gene expression data [26], detecting remote protein homologies [27], recognizing translation initiation sites [28], functional classification of promoter regions [29], prediction of protein function from phylogenetic profiles [30], protein subcellular localization prediction [31], recognizing splice sites [32], predicting signal peptide cleavage sites [33], discovering functional RNAs in prokaryotes [34], secondary structure prediction [35], and prediction of protein-protein interactions [36]. In this paper, we follow the way in [37] to implement SVM algorithm.

III. KERNEL-BASED ADAPTIVE DISTANCE METRIC LEARNING

A. Data-dependent kernel model

Let $\{x_i, \zeta_i\}$ ($i = 1, 2, \dots, m$) be m d -dimensional training samples of the given gene expression data, where $\zeta_i = \pm 1$ represent the class labels of the samples. We engineer

a data-dependent kernel to capture the relationship among the data in this classification task. This data-dependent kernel is formulated as,

$$k(x, y) = q(x)q(y)k_0(x, y) \quad (1)$$

where $x, y \in \mathbf{R}^d$, $k_0(x, y)$, called the basic kernel, may be any kernel function such as Gaussian kernel or a polynomial kernel, and $q(\cdot)$, the factor function, takes the form of

$$q(x) = \alpha_0 + \sum_{i=1}^m \alpha_i k_1(x, x_i) \quad (2)$$

in which $k_1(x, x_i) = e^{-\gamma_1 \|x - x_i\|^2}$, α_i 's are the combination coefficients. This kernel model was first introduced in [38], and called ‘‘conformal transformation of a kernel’’. However, the authors in [38] did not consider as to how to optimize the kernel model.

Let the kernel matrices corresponding to $k(x, y)$ and $k_0(x, y)$ be K and K_0 . It is easy to verify $K = [q(x_i)q(x_j)k_0(x_i, x_j)]_{m \times m} = QK_0Q$, where Q is a diagonal matrix with diagonal elements $q(x_1), q(x_2), \dots, q(x_m)$. Let us denote the vectors $(q(x_1), q(x_2), \dots, q(x_m))^T$ and $(\alpha_0, \alpha_1, \alpha_2, \dots, \alpha_m)^T$ by q and α , respectively. We have $q = K_1\alpha$, where K_1 is an $m \times (m + 1)$ matrix

$$K_1 = \begin{pmatrix} 1 & k_1(x_1, x_1) & \cdots & k_1(x_1, x_m) \\ 1 & k_1(x_2, x_1) & \cdots & k_1(x_2, x_m) \\ \vdots & \vdots & \ddots & \vdots \\ 1 & k_1(x_m, x_1) & \cdots & k_1(x_m, x_m) \end{pmatrix}$$

B. Kernel optimization

The optimization of the data-dependent kernel in (1) is to set the value of combination coefficient vector α so that the class separability of the training data in mapped feature space is maximized. For this purpose, Fisher scalar is adopted as the objective function of our kernel optimization. Fisher scalar measures the class separability of the training data in the mapped feature space and is formulated as

$$J = \frac{\text{tr}(S_b)}{\text{tr}(S_w)} \quad (3)$$

where S_b represents the ‘‘between-class scatter matrix’’, and S_w ‘‘within-class scatter matrix’’.

Suppose that the training data are grouped according to their class labels, i.e., the first m_1 data belong to one class, and the remaining m_2 data belong to the other class ($m_1 + m_2 = m$). Then the basic kernel matrix K_0 can be partitioned as

$$K_0 = \begin{pmatrix} K_{11}^0 & K_{12}^0 \\ K_{21}^0 & K_{22}^0 \end{pmatrix}$$

where the sizes of the submatrices K_{11}^0 , K_{12}^0 , K_{21}^0 , and K_{22}^0 respectively are $m_1 \times m_1$, $m_1 \times m_2$, $m_2 \times m_1$, and $m_2 \times m_2$. A close relation between the class separability measure J and the kernel matrices has been established [39].

$$J(\alpha) = \frac{\alpha^T M_0 \alpha}{\alpha^T N_0 \alpha} \quad (4)$$

where $M_0 = K_1^T B_0 K_1$, $N_0 = K_1^T W_0 K_1$, in which

$$B_0 = \begin{pmatrix} \frac{1}{m_1} K_{11}^0 & 0 \\ 0 & \frac{1}{m_2} K_{22}^0 \end{pmatrix} - \frac{1}{m} K_0$$

$$W_0 = \text{diag}(k_{11}^0, k_{22}^0, \dots, k_{mm}^0) - \begin{pmatrix} \frac{1}{m_1} K_{11}^0 & 0 \\ 0 & \frac{1}{m_2} K_{22}^0 \end{pmatrix}$$

It is easy to verify that, if matrix N_0 is nonsingular, the optimal α that maximizes the $J(\alpha)$ in (4) is the eigenvector corresponding to the maximum eigenvalue of the system

$$M_0 \alpha = \lambda N_0 \alpha$$

Unfortunately, in practice, especially in analyzing gene expression data, the matrix N_0 is frequently singular, which makes the eigen-decomposition method not applicable. A gradient-based learning algorithm was proposed in [40] to solve the optimization problem. However, this approach is generally associated with two disadvantages: 1) two extra parameters (the learning rate and the total iteration number) are introduced and need to be specified in advance; and 2) the solution may be trapped in a local optimal point.

An alternative way to handle this problem is to introduce a regulation parameter $\mu > 0$, and substitute the matrix N_0 by $N_0 + \mu I$, where I denotes the unit matrix. Therefore, the optimal α that maximizes the class separability measure $J(\alpha)$ is determined by the eigenvector corresponding the maximum eigenvalue of the system

$$M_0 \alpha = \lambda(N_0 + \mu I) \alpha \quad (5)$$

C. kernel distance metric and data resampling

Given two samples $x, y \in \mathbf{R}^d$, the inner product is defined as: $x \cdot y = \langle x, y \rangle = k(x, y)$; therefore, their derived distance can be calculated

$$\begin{aligned} d(x, y) &= \langle x, x \rangle + \langle y, y \rangle - 2 \langle x, y \rangle \\ &= k(x, x) + k(y, y) - 2k(x, y) \end{aligned}$$

Using our data-dependent kernel model, the distance can be expressed as

$$\begin{aligned} d(x, y) &= q^2(x) + q^2(y) - 2q(x)q(y)k_0(x, y) \\ &= [q(x) - q(y)]^2 + 2q(x)q(y)(1 - k_0(x, y)) \end{aligned}$$

where we assume that the basic kernel function satisfy: $k_0(x, x) = 1$.

The optimized kernel maximizes the class separability of the training data in the feature space. Assuming the testing data have the same distribution as the training data, intuitively, this kernel metric should adapt better to the labeled data than the Euclidean metric. Hence classification methods based on this data-dependent kernel are expected to perform significantly better than their counterpart with the Euclidean metric. In the cases that we have enough training samples, the above intuition stays true. However, for the task of microarray data classification, changes are we will get a very small set of training samples in high dimensions. Because the kernel optimization is performed with the training data only, in the case of small training set, it is possible that the kernel optimization increases the class separability of the training data remarkably, however, does not improve, sometimes even decrease, that of the test data. This training bias is essentially caused by some outliers in the training data. In the case of small training set, the effect of one outlier on training could be significant. To reduce the adverse effect of the training bias, a bootstrapping-based resampling scheme, called disturbed resampling, is interlaced in our KerNN scheme. The disturbed resampling is essentially a scheme of “bootstrapping with noise” [40].

Suppose that $\{x_i, \zeta_i\}$ ($i = 1, 2, \dots, m$) are the training data ($\zeta_i = \pm 1$), we construct a new set of training data $\{y_i, \xi_i\}$ ($i = 1, 2, \dots, 3m$)

$$y_i = \begin{cases} x_i & \text{if } 1 \leq i \leq m \\ x_r + \varepsilon & \text{if } i > m \end{cases} \quad (6)$$

in which x_r is a sample randomly selected from $\{x_i\}$ with replacement, and ε represents a random disturbance with uniform or normal distribution, that is, $\varepsilon \sim N(0, \sigma^2)$. The class labels are determined as

$$\xi_i = \begin{cases} \zeta_i & \text{if } 1 \leq i \leq m \\ \zeta_r & \text{if } i > m \end{cases}$$

IV. EXPERIMENTS

We performed a set of experiments to compare the performances of our KerNN scheme to some well-known classification algorithms, i.e., KNN, DLDA, ULDA, and SVM¹. Nine publicly available microarray data sets are chosen for this purpose. In the experiments, each data set is first normalized to zero mean and unity variance at the gene direction. The data set is then randomly partitioned into two disjoint subsets with equal number of samples, one for training, and the other for testing.

A. Data sets

- 1) *ALL-AML Leukemia Data*: This data set is available at <http://sdmc.lit.org.sg/GEDatasets/>. It contains 72 samples of human acute leukemia. These samples are categorized as acute lymphoblastic leukemia (ALL) and the acute myeloid leukemia (AML), representing two different types of leukemia. Each sample contains the expression levels of 7129 genes. For the detailed information, one can refer to [19].
- 2) *Embryonal Tumors of Central Nervous System (CNS)*: This data set is available at <http://sdmc.lit.org.sg/GEDatesets/>. It contains 60 patient samples, of which 21 are survivors of a treatment, and 39 are failures. Expression levels of 7129 genes are contained in the data set. More information about this data set may be found in [41].
- 3) *Breast Cancer Data*: The data are available at http://mgm.duke.edu/genome/dna_micro/work/. The expression matrix monitors 7129 genes in 49 breast tumor samples. There are two response variables respectively describing the status of the estrogen receptor (ER) and the lymph nodal (LN) status. According to the ER status, 25 samples are ER+, whereas the remaining 24 samples are ER-. Based on the LN

¹We only consider Gaussian kernel function in the proposed and SVM algorithms.

variable, there are 25 positive samples and 24 negative samples. The detailed information about this data set can be found in [20].

- 4) *Colon Tumor Data*: This data set is downloaded from <http://sdmc.lit.org.sg/GEDatasets/>. It contains 62 samples collected from colon-cancer patients. Among them, 40 samples are from tumor tissues, and 22 are from healthy parts of the colons of the same patients. Expression levels are measured for 2000 genes. Additional information about this data set may be obtained from [42].
- 5) *Lung Cancer Data*: This data set is downloaded from <http://sdmc.lit.org.sg/GEDatasets/>. It contains 181 tissue samples, which are classified into two classes: malignant pleural mesothelioma (MPM) and adenocarcinoma (ADCA). Each sample is described by the expression level of 12533 genes. More information about this data set can be found [43].
- 6) *Lymphoma Data*: The data set is available at <http://www.broad.mit.edu/mpr/lymphoma>. It contains 77 tissue samples, of which 58 are diffuse large B-cell lymphomas (DL-BCL) and the remaining are follicular lymphomas (FL). Each sample is represented by the expression levels of 7129 genes. The detailed information about this data set can be found in [44].
- 7) *Ovarian Cancer Data*: This data set, available at <http://sdmc.lit.org.sg/GEDatasets/>, is to distinguish ovarian cancer from non-cancer. It contains 253 samples, and each sample has 15154 features. More details can be found in [45].
- 8) *Prostate Cancer Data*: This data set is downloaded from <http://www.broad.mit.edu/>. It contains the gene expression levels of 12600 genes for 52 prostate tumor samples and 50 normal prostate samples. One can refer [3] for the details about this data set.

The basic information about these data sets is summarized in Table I.

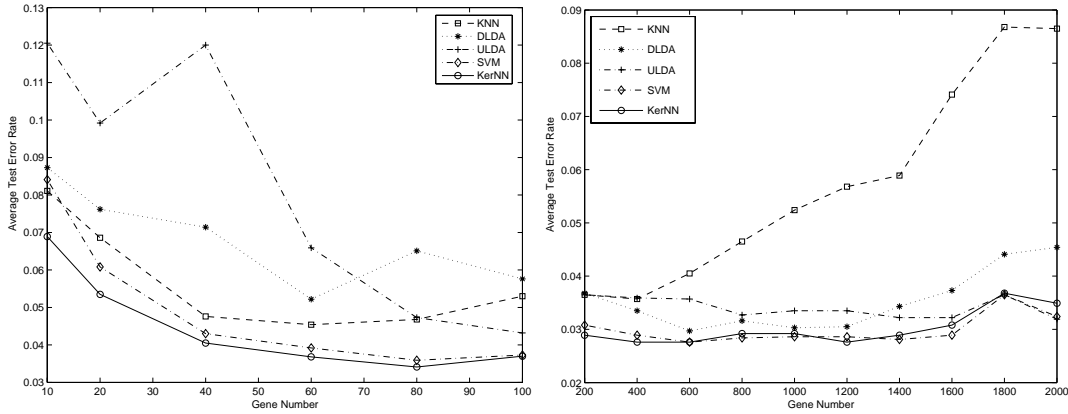
B. Gene selection

Because the sample size is much smaller than the dimensionality, too many genes may not be helpful, sometimes may even be harmful, for the class discrimination. Selecting the most discriminatory genes and removing the rest not only reduce the computational complexity, but also substantially improve the performances of many classifiers.

TABLE I

THE BASIC INFORMATION ABOUT THE GENE EXPRESSION DATA SETS

| | sample size | number of genes |
|------------------|-------------|-----------------|
| <i>ALL-MAL</i> | 72 | 7129 |
| <i>Breast-ER</i> | 49 | 7129 |
| <i>Breast-LN</i> | 49 | 7129 |
| <i>CNS</i> | 60 | 7129 |
| <i>Colon</i> | 62 | 2000 |
| <i>Lung</i> | 181 | 12533 |
| <i>Lymphoma</i> | 77 | 7129 |
| <i>Ovarian</i> | 253 | 15154 |
| <i>Prostate</i> | 102 | 12600 |

Fig. 1. The average test error rate as a function of feature number for the *ALL-MAL* microarray data

In this paper, we use the BW ratio in [17], which is essentially a Fisher discriminant measure, to select genes. For a gene j , the BW score (or ratio) on gene j is calculated as

$$g(j) = \frac{\sum_{k=1}^2 m_k (\bar{x}_k(j) - \bar{x}(j))^2}{\sum_{k=1}^2 \sum_{i \in C_k} (x_i(j) - \bar{x}_k(j))^2} \quad (7)$$

where C_k denotes the index set of the k -th class ($k = 1, 2$), m_k is the number of samples in C_k ($\sum_{k=1}^2 m_k = m$), and $\bar{x}_k(j)$ and $\bar{x}(j)$ represent the average expression levels within the k -th class and of the entire training samples on gene j , respectively.

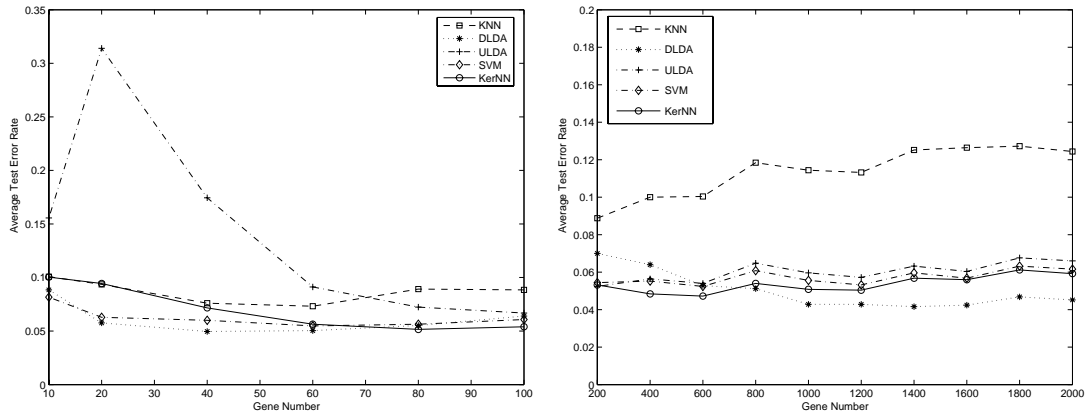


Fig. 2. The average test error rate as a function of the selected gene number for the *Breast-ER* microarray data

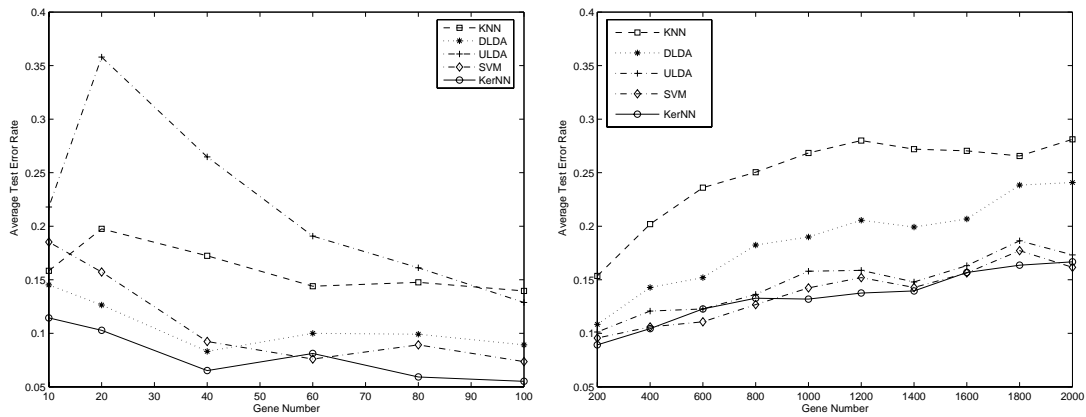


Fig. 3. The average test error rate as a function of the selected gene number for the *Breast-LN* microarray data

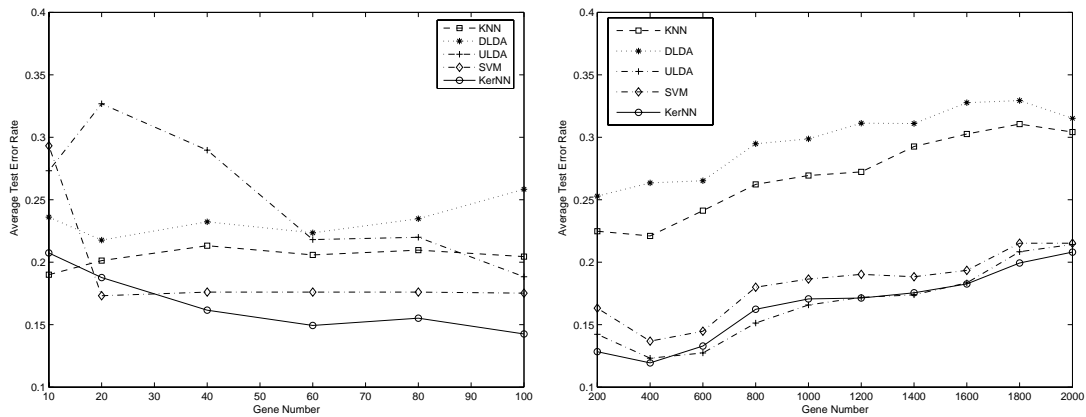


Fig. 4. The average test error rate as a function of the selected gene number for the *CNS* microarray data

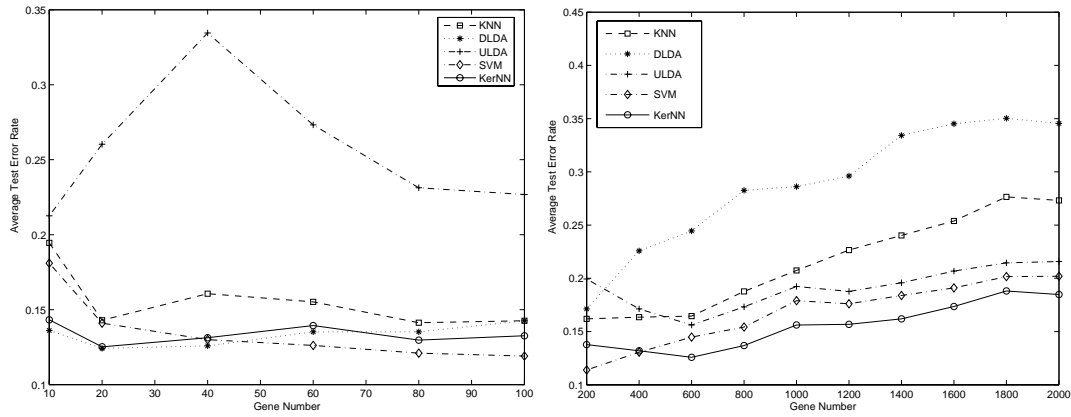


Fig. 5. The average test error rate as a function of the selected gene number for the *Colon* microarray data

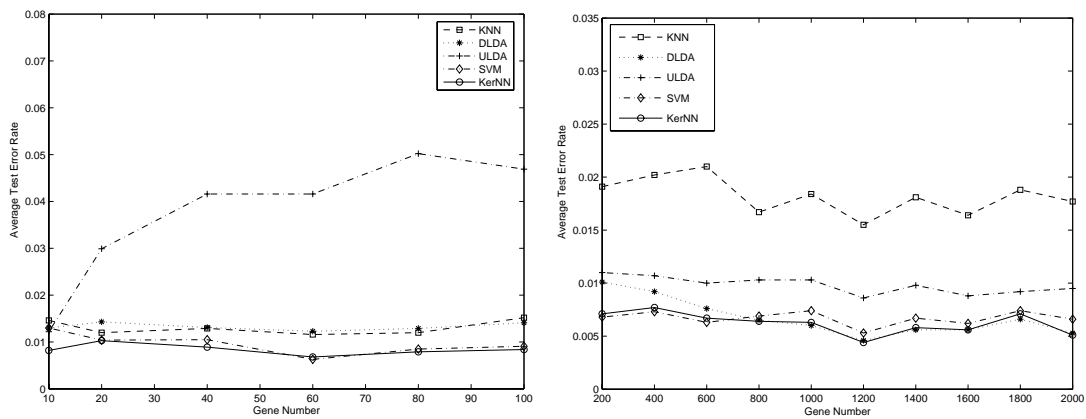


Fig. 6. The average test error rate as a function of the selected gene number for the *Lung* microarray data

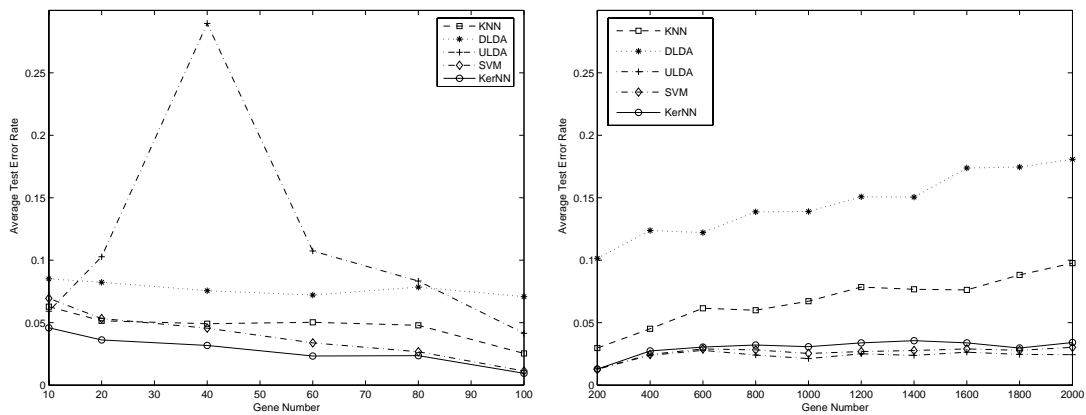


Fig. 7. The average test error rate as a function of the selected gene number for the *Lymphoma* microarray data

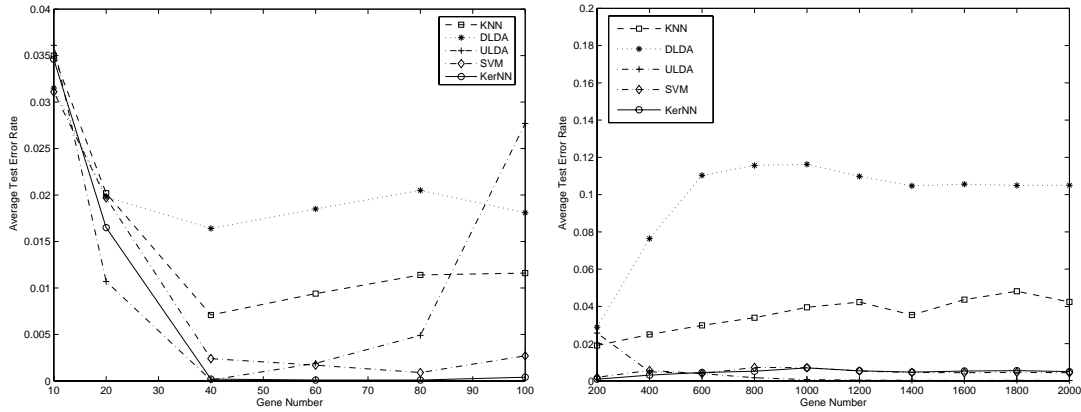


Fig. 8. The average test error rate as a function of the selected gene number for the *Ovarian* microarray data

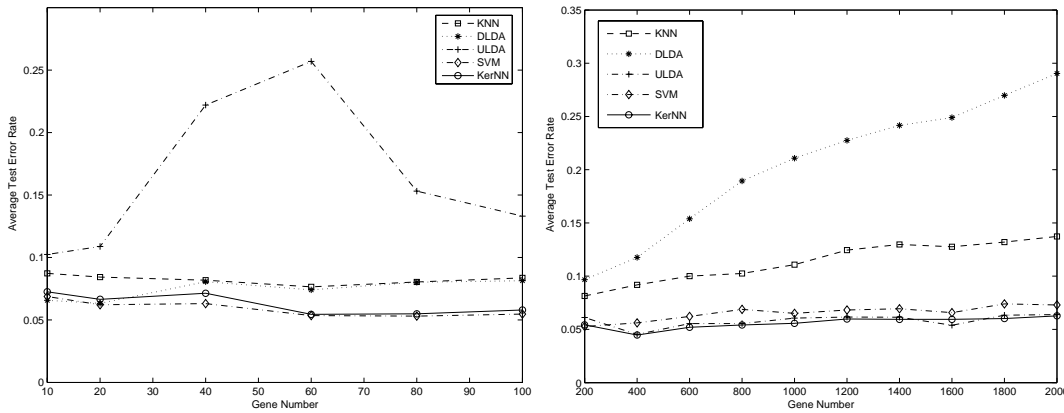


Fig. 9. The average test error rate as a function of the selected gene number for the *Prostate* microarray data

C. Experimental results

We now present our experimental results for the five classifiers (KerNN, KNN, DLDA, ULDA, and SVM) with the nine data sets. In the experiments, the classification for KNN, ULDA, and KerNN is performed by the K-nearest-neighbor (KNN) algorithm with $K=3$.

1) *Comparisons with different number of genes:* To investigate the influence of the selected gene number N_f on the five classification algorithms, we compare their performances when different number of genes are used in the classification. For each data set, experiments are carried out with N_f respectively set to 10, 20, 40, 60, 80, 100, 200, 400, 600, 800, 1000, 1200, 1400, 1600, 1800, and 2000. For each value of N_f , we choose the N_f most discriminatory genes according to the BW score calculated by (7). Each experiment is repeated 100 times, and the average test error rates and their standard

TABLE II

COMPARISONS IN TERMS OF THE AVERAGE TEST ERROR RATES(%) AND THE STANDARD VARIANCE (IN PARENTHESE)

| | KNN | DLDA | ULDA | SVM | KerNN |
|------------------|--------------|--------------------|--------------------|---------------------|---------------------|
| <i>ALL-AML</i> | 5.78 (3.58) | 4.16 (2.57) | 6.92 (4.13) | 5.24 (3.20) | 5.14 (3.10) |
| <i>Breast-ER</i> | 10.32 (5.12) | 7.04 (5.63) | 9.76 (6.86) | 7.36 (4.07) | 7.84 (5.11) |
| <i>Breast-LN</i> | 14.16 (5.89) | 9.74 (5.61) | 15.52 (8.35) | 9.76 (6.91) | 8.08 (5.08) |
| <i>Colon</i> | 17.03 (4.66) | 14.00 (4.59) | 20.90 (8.04) | 13.55 (5.05) | 12.90 (4.47) |
| <i>CNS</i> | 21.94 (5.37) | 23.61 (6.47) | 17.10 (8.65) | 15.94 (7.24) | 16.02 (6.36) |
| <i>Lung</i> | 1.56 (1.20) | 1.23 (0.91) | 1.87 (1.59) | 1.21 (0.81) | 0.84 (0.75) |
| <i>Lymphoma</i> | 4.10 (3.88) | 8.36 (3.90) | 5.59 (7.27) | 3.38 (3.33) | 3.38 (2.95) |
| <i>Ovarian</i> | 1.15 (1.30) | 1.98 (1.05) | 0.39 (0.77) | 0.44 (1.02) | 0.44 (0.92) |
| <i>Prostate</i> | 8.63 (3.43) | 6.78 (3.00) | 7.57 (5.59) | 6.78 (3.02) | 6.43 (4.10) |

variances over the 100 experiments are reported. The experimental results for the nine data sets are shown in Fig. 1 to Fig. 9, respectively, where the horizontal axis represents the number of the selected genes and the vertical axis corresponds to the average test error rates of the classifiers over 100 experiments.

As can be seen from these plots, our kernel-based KerNN scheme significantly outperforms the conventional KNN classifier; moreover, compared with the other methods, KerNN performs favorably in most cases. Different from the ULDA scheme, which usually performs poorly in the case of small gene number (see Fig.1(a) to Fig.9(a)), and the DLDA scheme, whose performance often degrade in the case of relatively large gene number (see Fig.3(b), Fig.4(b), Fig.5(b), Fig.7(b), and Fig.9(b)), the proposed KerNN scheme perform more stable when the number of selected genes changes.

2) *Comparisons with the number of genes determined by cross validation:* This time, we use the leave-one-out cross validation to choose the number of genes. Considering there are two other parameters need to be tuned for the SVM and KerNN classifiers, to reduce the computational complexity involving in the cross validation, we choose the gene number only from {10, 20, 40, 60, 80, 100, 200, 400, 600, 800, 1000, 1200, 1400, 1600, 1800, 2000}. Given a training/testing split of a microarray data set, for each

classifier, after the number of genes is determined on the training data, the error rates on the testing data is calculated.

Table II presents the average test error rates and the standard variances over 100 trials for the five classifiers on the nine microarray data sets. To visualize the distributions of the test error rates of different algorithms, we present the boxplots of the test error rates in Fig. 10 to Fig. 13. It can be seen that, in five out of the nine data sets, KerNN algorithm achieves the lowest average test error rates. On the other four data sets, the performances of KerNN are almost close to the best. More importantly, the fact that the proposed KerNN classifier significantly outperforms the KNN classifier with the Euclidean metric suggests that the kernel optimization induces a more adaptive distance metric than the Euclidean metric to the gene expression data.

V. DISCUSSION

Here we have presented a novel cancer classification method based on data-dependent kernel. The experimental results of nine data sets have demonstrated the robustness of our proposed KerNN algorithm in classifying different types of cancer expression data. Although the KerNN algorithm is evaluated favorably in most cases, a drawback of the method is that there are more parameters to tune than the other methods being compared. For DLDA algorithms, no parameter need to be specified. The number of nearest neighbors K are specified as 3 for KNN and ULDA. For SVM, two parameters, the γ in the Gaussian kernel function and the regulation constant C , need to be set in advance. As to the KerNN algorithm, there are four parameters, γ_0 for the basic Gaussian kernel $k_0(x, y)$ in (1), γ_1 for the function $k_1(x, a_i)$ in (2), the regulation parameter μ in (5), and the parameter σ in the disturbed resampling. To reduce the computational intensity in tuning these parameters, we empirically set $\gamma_0 = \frac{10^{-5}}{N_f}$, $\gamma_1 = \frac{\beta}{N_f}$, where N_f is the number the selected genes, and $\sigma = 0.1$. Therefore, only two parameters β and μ are tuned for the KerNN method.

In the experiments, we employ the leave-one-out technique on the training data to tune these parameters. For SVM classifier, we follow [37] to implement the SVM algorithm, in which the parameter C is chosen from $\{1.0e+00, 1.0e+01, 1.0e+02, 1.0e+03, 1.0e+04, 1.0e+05, 1.0e+06, 1.0e+07\}$ and γ from $\{1.0e-07, 5.0e-07, 1.0e-06, 5.0e-06, 1.0e-05,$

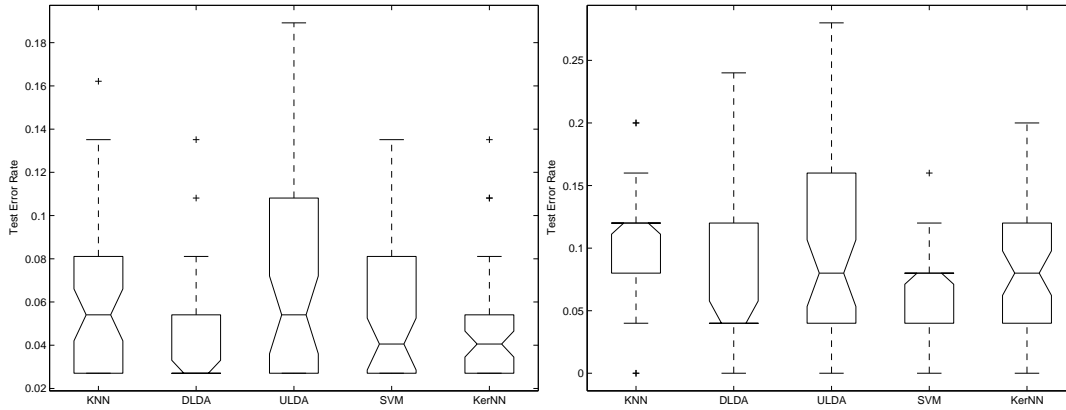


Fig. 10. Boxplots of error rates for *ALL-AML* and *BreastER* data sets.

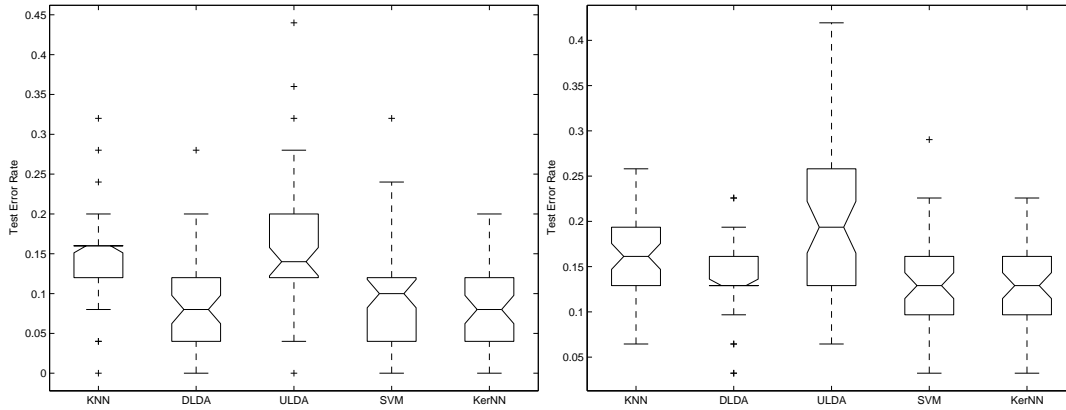


Fig. 11. Boxplots of error rates for *BreastLN* and *Colon* data sets.

$5.0e-05, 1.0e-04, 5.0e-04, 1.0e-03, 5.0e-03, 1.0e-02\}$ using the leave-one-out cross validation. For our KerNN algorithm, the parameters β and μ are selected from a same set $\{0.0001, 0.001, 0.01, 0.1\}$. Note that only the training samples are used for setting parameters. Test samples are independent of this process.

A. The effect of the kernel optimization on class separability

The goal of the kernel optimization is to maximize the measure of class separability (3) of training data. It is certain that the class separability of the training data in the feature space should be remarkably improved with the optimized kernel. However, the question is whether this data-dependent kernel increases the separability of testing data. We here illustrate the effect of the data-dependent kernel on training data and testing data with two dimensional embedding of the data points. The *Prostate* data set with 200 genes

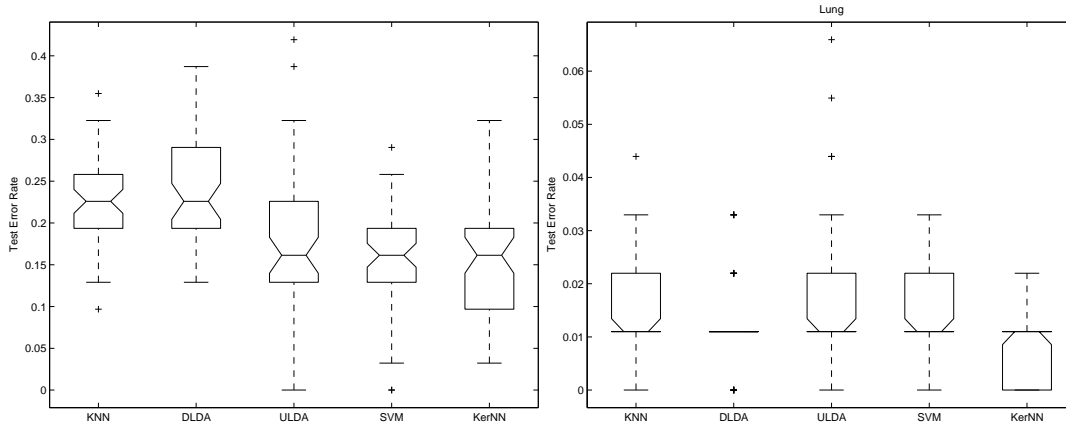


Fig. 12. Boxplots of error rates for *CNS* and *Lung* data sets.

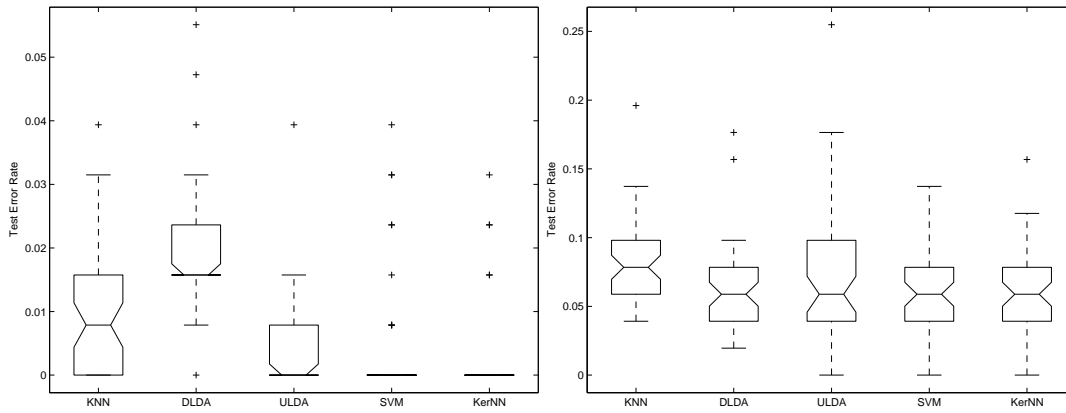
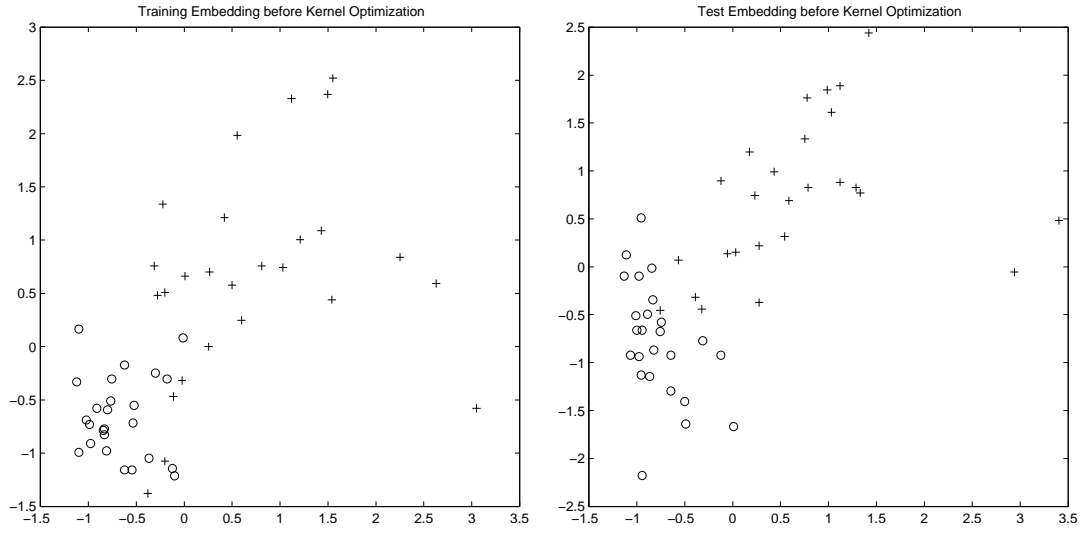


Fig. 13. Boxplots of error rates for *Ovarian* and *Prostate* data sets.

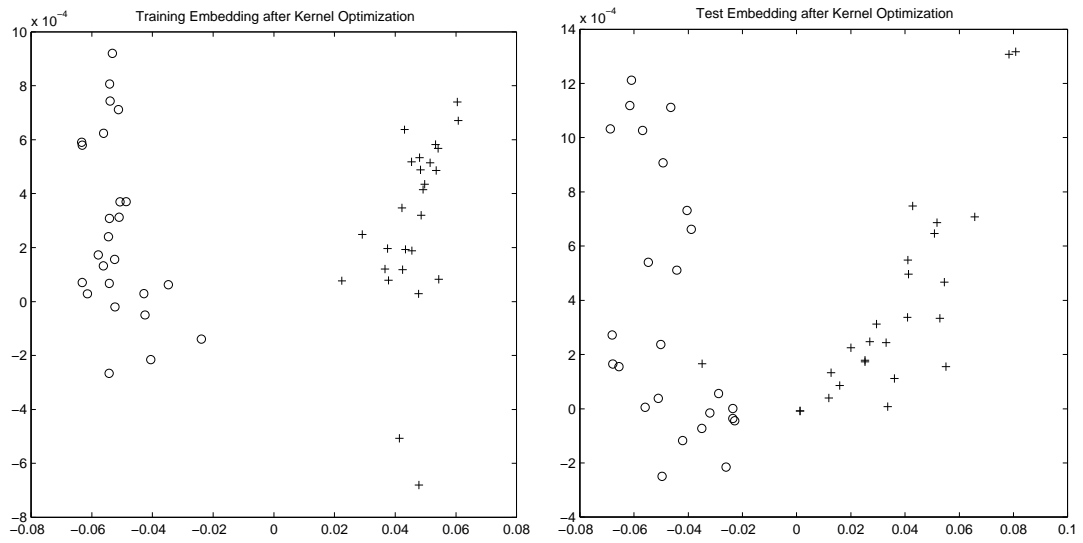
selected ($N_f = 200$) is used as an example to show the effect. Fig. 14 (a) shows the projection of the training data and test data onto their first two significant dimensions as determined by multidimensional scaling (MDS) technique [46], [47]. Fig. 14 (b) illustrates the corresponding projection with the optimized data-dependent kernel function. These figures indicate that the class separability of the test data is substantially improved together with that of the training data, although the kernel optimization is only performed with the training data.

B. The effect of the disturbed resampling

Due to the lack of enough training samples, the procedure of the kernel optimization may lead to serious bias in training the classifier. As discussed in Section III-C, to reduce this bias, the strategy of disturbed resampling is adopted. In this section, we demonstrate



(a)



(b)

Fig. 14. Kernel optimization improves the class separability for both training and test data. (a) Two-dimensional embedding of the training and test data. (b) The corresponding two-dimensional embedding after the optimized kernel function is used.

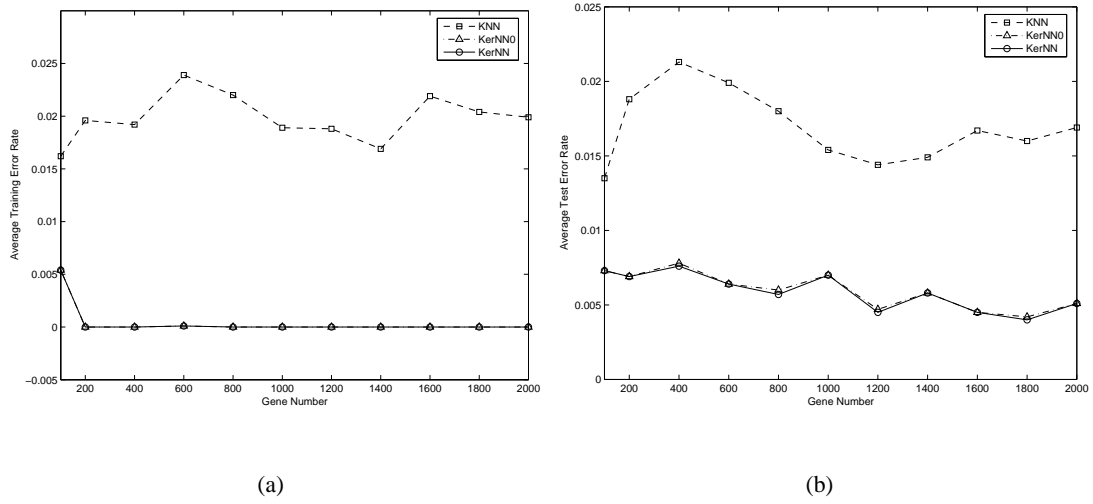


Fig. 15. The effect of adopting the technique of disturbed resampling on the *Lung* data set. (a) Results on the training data. (b) Results on the test data.

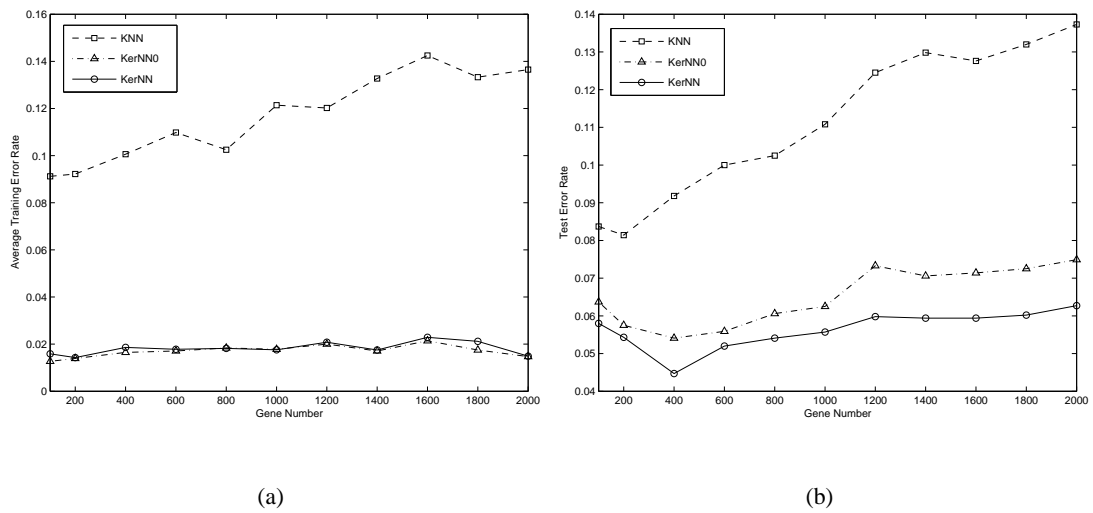


Fig. 16. The effect of adopting the technique of disturbed resampling on the *Prostate* data set. (a) Results on the training data. (b) Results on the test data.

that effectiveness of this strategy.

In the cases where there are relatively large training samples, the adverse effect of the training bias usually can be ignored, and the kernel-based KNN classifier without using the strategy of disturbed resampling, denoted by KerNN0, can achieve as good classification results as those of the KerNN scheme, on both the training and test data. Fig. 15 demonstrates this point on the *Lung* data set, which contains 181 samples. On

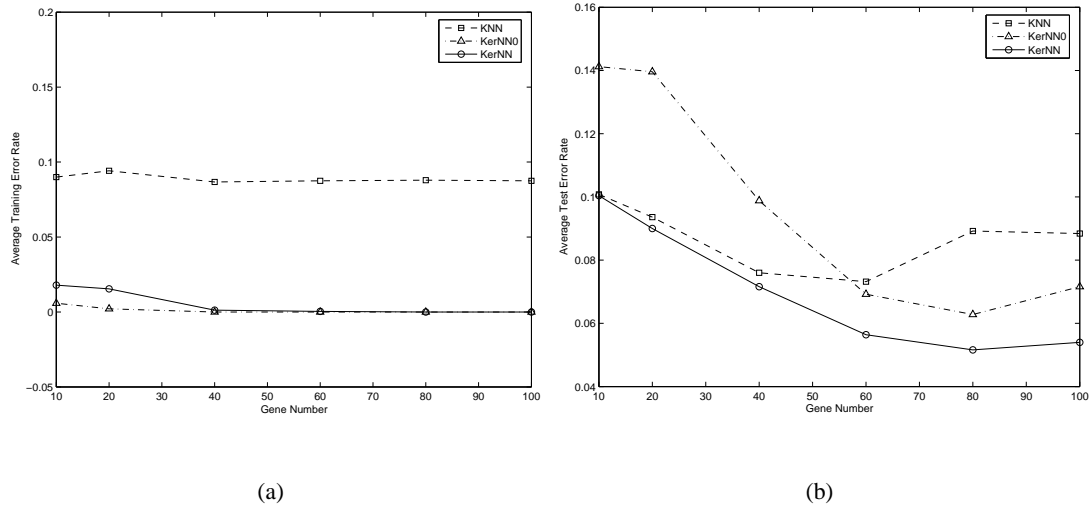


Fig. 17. The effect of adopting the technique of disturbed resampling on the *Breast-ER* data set. (a) Results on the training data. (b) Results on the test data.

the other hand, in the case of relatively small sample size, the KerNN0 algorithm has a relatively higher risk being impaired by the training bias than the KerNN scheme. We illustrate this point with the *Prostate* data set (Fig.16), which includes 102 samples, and with the *Breast-ER* data set (Fig.17), which contains only 49 samples in total. It can be seen from Fig.16(b) that the performance of the kernel-based KNN classifier can be significantly improved when the disturbed resampling technique is adopted in the scheme. In Fig. 17, we see that, although KerNN0 works quite well on the training data, its performance degrades remarkably on the test data (even worse than KNN in some cases, see Fig.17(b)).

VI. CONCLUSIONS

In this paper, a novel data-dependent kernel is proposed for cancer classification. This data-dependent kernel is learned through maximizing the class separability of the training data in the kernel-induced feature space. Our experimental results show that this optimized kernel also helps to increase the class separation for the testing data. When this data-dependent kernel is incorporated into KNN classifier, a significant improvement of the performance is achieved compared to KNN classifier based on Euclidean distance metric. This classification scheme is applied to cancer classification with gene expression data.

The data are characterized by high dimensionality and small sample size. This kernel-based method, together with a disturbed resampling strategy, is demonstrated to be capable of alleviating the problem of overfitting. Compared with several other cancer classification schemes including SVM, ULDA, DLDA and KNN, the K-nearest-neighbor classifier based on the data-dependent kernel achieves competitive performance, if not the best.

ACKNOWLEDGEMENTS

This investigation was partially supported by the U. S. Army Research Laboratory and the U. S. Army Research Office under contract number DAAD19-03-1-0123. The authors would like to thank the editor and anonymous reviewers for their insightful comments on the manuscripts.

REFERENCES

- [1] A. Schulze, J. Downward. **Navigating gene expression using microarrays—a technology review.** *Nat Cell Biol.* 2001, **3**(8):E190-195.
- [2] E. Keedwell, A. Narayanan. **Discovering Gene Networks with a Neural-Genetic Hybrid.** *IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB)* 2005, **2**(3):231-242.
- [3] D. Singh, P.G. Febbo, K. Ross, D.G. Jackson, J. Manola, C. Ladd, P. Tamayo, A.A. Renshaw, A.V. D’Amico, J.P. Richie, E.S. Lander, M. Loda, P.W. Kantoff, T.R. Golub, W.R. Sellers. **Gene expression correlations of clinical prostate cancer behavior.** *Cancer Cell* 2004, **1**:203-209.
- [4] L.J. van’t Veer, H. Dai, *et.al.*. **Gene expression profiling predicts clinical outcome of breast cancer.** *Nature* 2002, **419**:530-536.
- [5] K. M. Borgwardt, S. V. N. Vishwanathan, and H. Kriegel. **Class Prediction from Time Series Gene Expression Profiles Using Dynamical Systems Kernels.** *Pacific Symposium on Biocomputing 2006* **11**:547-558.
- [6] M. Wilson, J. DeRisi, H. H. Kristensen, P. Imboden, S. Rane, P. O. Brown, G. K. Schoolnik. **Exploring drug-induced alterations in gene expression in Mycobacterium tuberculosis by microarray hybridization.** *Proc Natl Acad Sci U S A.* 1999, **96**(22):12833-12838.
- [7] W. E. Evans and R. K. Guy. **Gene expression as a drug discovery tool** *Nat Genet* 2004, **36**(3):214-215.
- [8] R. Sharan and R. Shamir. *Current Topics in Computational Molecular Biology*, chapter Algorithmic approaches to clustering gene expression data, pages 269–300. The MIT Press, 2002.
- [9] M. B. Eisen, P. T. Spellman, P. O. Brown, and D. Botstein. **Cluster analysis and display of genome-wide expression patterns.** *Proc. Natl Acad. Sci. USA*, **95**:14863–14868, 1998.
- [10] P. Toronen, M. Kolehmainen, G. Wong, and E. Castren. **Analysis of gene expression data using self-organizing maps.** *FEBS Letters*, **451**:142–146, 1999.
- [11] S. Tavazoie, J. D. Hughes, M. J. Campbell, R. J. Cho, and G. M. Church. **Systematic determination of genetic network architecture.** *Nature Genet*, **22**:281–285, 1999.
- [12] P. Langley. **Selection of relevant features in machine learning.** *Proceedings of the AAAI Fall Symposium on Relevance.* AAAI Press.

- [13] R. Kohavi and G. John. **Wrapper for feature subset selection.** *Artificial Intelligence*, 97, 273-324.
- [14] E.P. Xing, M.I. Jordan, and R.M. Karp. **Feature Selection for High-Dimensional Genomic Microarray Data.** In *Proceedings of the Eighteenth International Conference in Machine Learning, ICML2001.*
- [15] *Kernel methods in computational biology.* edited by B. Scholkopf, K. Tsuda, and J.-P. Vert. The MIT Press, 2004.
- [16] A. Ben-Dor, L. Bruhn, N. Friedman, I. Nachman, M. Schummer, and Z. Yakhini. **Tissue classification with gene expression profiles.** *J. Computational Biology* 2000, 7:559-584.
- [17] S. Dudoit, J. Fridlyand, and T.P. Speed. **Comparison of discrimination method for the classification of tumor using gene expression data.** *J. Am. Statistical Assoc.* 2002, 97:77-87.
- [18] M. Dettling and P. Bühlmann. **Boosting for tumor classification with gene expression data.** *Bioinformatics* 2003, 19:1061-1069.
- [19] T.R. Golub, D.K. Slonim, P. Tamayo, C. Huard, M. Gassenbeek, 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* 1999, 286:531-537.
- [20] B. West, C. Blanchette, H. Dressman, E. Huang, and *et.al.* **Predicting the clinical status of human breast cancer by using gene expression profiles.** *Proc. Natl. Acad. Sci. USA.* 2001, 98:11462-11467.
- [21] J. Ye, T. Li, T. Xiong, and R. Janardan. **Using uncorrelated discriminant analysis for tissue classification with gene expression data.** *IEEE/ACM Trans. on Computational Biology and Bioinformatics* 2004, 1:181-190.
- [22] T. Hastie and R. Tibshirani. **Discriminant Adaptive Nearest Neighbor Classification.** *IEEE Trans. on Pattern Analysis and Machine Intelligence* 1996, 18:607-615.
- [23] R.O. Duda, P.E. Hart, and D.G. Stork. *Pattern Classification (Second Edition).* A Wiley-Interscience Publication, 2000.
- [24] J.H. Friedman. **Flexible metric nearest neighbor classification.** Technical report, Dept. of Statistics, Stanford University, 1994.
- [25] P. Howland and H. Park. **Generalizing Discriminant Analysis Using the Generalized Singular Value Decomposition.** *IEEE Trans. on Pattern Analysis and Machine Intelligence* 2004, 26:995-1006.
- [26] T.S. Furey, N. Cristianini, N. Duffy, D.W. Bednarski, M. Schummer, and D. Haussler. **Support vector machine classification and validation of cancer tissue samples using microarray expression data.** *Bioinformatics* 2000, 16:906-914.
- [27] T. Jaakkola, M. Diekhans, and D. Haussler. **Using the Fisher kernel method to detect remote protein homologies.** In *Proceedings of the 7th International Conference on Intelligent Systems for Molecular Biology.* AAAI Press, Menlo Park, CA.
- [28] A. Zien, G. Rätsch, S. Mika, B. Schölkopf, C. Lemmen, A. smola, T. Lengauer, and K. Müller. **Engineering support vector machine kernels that recognize translation initiation sites.** *Bioinformatics* 2000, 16:799-807.
- [29] P. Pavlidis, T. S. Furey, M. Liberto, and W. N. Grundy. **Promoter region-based classification of genes.** *Proceedings of the Pacific Symposium on Biocomputing* 2001, 151-163.
- [30] J.-P. Vert. **A tree kernel to analyze phylogenetic profiles.** *Bioinformatics* 2002, 18:S276-S284.
- [31] S. Hua and Z. Sun. **Support vector machine approach for protein subcellular localization prediction.** *Bioinformatics* 2001, 17(8):721-728.
- [32] S. Degroeve, B. De Baets, Y. Van de Peer, and P. Rouz. **Feature subset selection for splice site prediction.** *Bioinformatics* 2002, 18:S75-S83.

- [33] J.-P. Vert. **Support vector machine prediction of signal peptide cleavage site using a new class of kernels for strings.** *Proceedings of the Pacific Symposium on Biocomputing 2002*, 649-660.
- [34] R. J. Carter, I. Dubchak, and S. R. Holbrook. **A computational approach to identify genes for functional RNAs in genomic sequences.** *Nucleic Acids Research 2001*, 29(19):3928-3938.
- [35] S. Hua and Z. Sun. **A novel method of protein secondary structure prediction with high segment overlap measure: support vector machine approach.** *Journal of Molecular Biology 2001*, 308:397-407.
- [36] J. R. Bock and D. A. Gough. **Predicting protein-protein interactions from primary structure.** *Bioinformatics 2001*, 17:455-460.
- [37] G. C. Cawley. MATLAB support vector machine toolbox [<http://theoval.sys.uea.ac.uk/~gcc/svm/ toolbox>]. University of East Anglia, School of Information Systems, Norwich, Norfolk, U.K. NR4 7TJ, 2000.
- [38] S. Amari and S. Wu. **Improving support vector machine classifiers by modifying kernel functions.** *Neural Networks 1999*, 12:783-789.
- [39] H. Xiong, M.N.S. Swamy, and M.O. Ahmad. **Optimizing the data-dependent kernel in the empirical feature space.** *IEEE Trans. on Neural Networks 2005*, 16:460-474.
- [40] Y. Raviv and N. Intrator. **Bootstrapping with noise: An efficient regularization technique.** *Connection Science*, vol.8, 1996, pp.355-372.
- [41] S.L. Pomeroy, P. Tamayo, M. Gaasenbeek, L.M. Sturla, M. Angelo, M.E. McLaughlin, J.Y.H. Kim, L.C. Goumnerova, P.M. Black, C. Lau, J.C. Allen, D. Zagzag, J.M. Olson, T. Curran, C. Wetmore, J.A. Biegel, T. Poggio, S. Mukherjee, R. Rifkin, A. Califano, G. Stolovitzky, D.N. Louis, J.P. Mesirov, E.S. Lander and T.R. Golub. **Prediction of central nervous system embryonal tumor outcome based on gene expression.** *Letters to Nature, Nature 2002*, 415:436-442.
- [42] 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 analysis of tumor and normal colon tissue probed by oligonucleotide arrays.** *Proc. Natl. Acad. Sci. USA 1999*, 96:6745-6750.
- [43] G.J. Gordon, R.V. Jenson, L.-L. Hsiao, S.R. Gullans, J.E. Blumenstock, S. Ramaswamy, W.G. Richards, D.J. Sugarbaker and R. Bueno. **Translation of microarray data into clinically relevant cancer diagnostic tests using gene expression ratios in lung cancer and mesothelima.** *Cancer Research 2002*, 62:4936-4967.
- [44] M.A. Shipp, K.N. Ross, P. Tamayo, A.P. Weng, J.L. Kutok, R.C.T. Aguiar, M. Gaasenbeek, M. Angelo, M. Reich, G.S. Pinkus, T.S. Ray, M.A. Koval, K.W. Last, A. Norton, T.A. Lister, J. Mesirov, D.S. Neuberg, E.S. Lander, J.C. Aster, T.R. Golub. **Diffuse Large B-Cell Lymphoma Outcome Prediction by Gene Expression Profiling and Supervised Machine Learning.** *Nature Medicine 2002*, 8:68-74.
- [45] E.F. Petricoin, A.M. Ardekanl, B.A. Hitt, P.J. Levine, V.A. Fusaro, S.M. Steinberg, G.B. Mills, C. Simone, D.A. Fishman, E.C. Kohn, and L.A. Liotta. **Use of Proteomic Patterns in Serum to Identify Ovarian Cancer.** *The Lancet 2002*, 359:572-577.
- [46] D.W. Wichern and R.A. Johnson. *Applied Multivariate Statistical Analysis*. Prentice Hall, 5th edition, 2002.
- [47] E. Pekalska, P. Paclik, and Robert P.W. Duin. **A generalized kernel approach to dissimilarity-based classification.** *Journal of Machine Learning Research 2001*, 2:175-211.
- [48] C. Leslie and R. Kuang. **Fast String Kernels Using Inexact Matching for Protein Sequences.** *Journal of Machine Learning Research 2004* 5:1435-1455.



Huilin Xiong received his Ph.D. degree in Pattern Recognition and Intelligent Control from the Institute of Pattern Recognition and Artificial Intelligence, Huazhong University of Science and Technology, Wuhan, China, in 2000. His research interests include bioinformatics, pattern recognition, machine learning, and image processing. Currently, he is a postdoctoral researcher in the Department of Electrical Engineering and Computer Science, University of Kansas, Kansas, USA.



Ya Zhang received her B.S. degree from Tsinghua University, China in 2000, and Ph.D. degree in Information Sciences and Technology from the Pennsylvania State University in 2005. Since August 2005, she has been an assistant professor in the Department of Electrical Engineering and Computer Science at the University of Kansas. Her research interests include bioinformatics, computation biology, machine learning, and data mining with application to life sciences. Especially, she is interested in using computational methods to solve puzzles in functional genomics and proteomics.



Xue-wen Chen Xue-wen Chen, IEEE senior member, received the PhD degree from Carnegie Mellon University, Pittsburgh, USA in 2001. He then spent about one year as a postdoctoral at the University of Illinois at Urbana-Champaign. He is currently an assistant professor of computer science at the University of Kansas, Lawrence, USA. He is also a member in Kansas Masonic Cancer Research Institute. His research interest includes bioinformatics and machine learning. Much of his work addresses two core problems in learning: analyzing large scale dataset and learning from high-dimensions. His current research is focused on developing computational methods such as kernel based classifiers and feature selection for genomic and proteomic data analysis.