International Journal of Computational Intelligence Systems, Vol. 9, No. 6 (2016) 1041-1054

Manifold Regularized Proximal Support Vector Machine via Generalized Eigenvalue

Jun Liang *, Fei-yun Zhang , Xiao-xia Xiong , Xiao-bo Chen , Long Chen , Guo-hui Lan

Automotive Engineering Research Institute, Jiangsu University, Zhenjiang, 212013, P. R. China

Received 8 April 2014

Accepted 11 April 2016

Abstract

Proximal support vector machine via generalized eigenvalue (GEPSVM) is a recently proposed binary classification technique which aims to seek two nonparallel planes so that each one is closest to one of the two datasets while furthest away from the other. In this paper, we proposed a novel method called Manifold Regularized Proximal Support Vector Machine via Generalized Eigenvalue (MRGEPSVM), which incorporates local geometry information within each class into GEPSVM by regularization technique. Each plane is required to fit each dataset as close as possible and preserve the intrinsic geometric structure of each class via manifold regularization. MRGEPSVM is also extended to the nonlinear case by kernel trick. The effectiveness of the method is demonstrated by tests on some examples as well as on a number of public data sets. These examples show the advantages of the proposed approach in both computation speed and test set correctness.

Keywords: Support vector machines; Generalized eigenvalues; Locality preserving projections; Manifold regularization.

1. Introduction

Standard SVMs[1], which are powerful tools for data classification and regression, have come to play a very dominant role in machine learning and pattern recognition community. The approach is systematic and motivated by Statistical Learning Theory[2], which seeks the optimal separating hyperplane by minimizing structural risk instead of empirical risk. In binary classification, it assigns the point to one of the two disjoint half spaces in either the original input space or a higher dimensional feature space. In order to find the optimal separating hyperplane, SVMs need to solve a quadratic programming problem (QPP) which is time-consuming in large sized samples.

Recently, several simpler classification methods were proposed[3,4]. Opposite to SVMs, in binary classification, they all aim at finding two planes so that each plane was closest to the points of its own class and furthest away from the points of the other class. This idea not only leads to faster and simpler algorithms than the conventional SVMs but also avoids the unbalanced sample distribution problem in SVMs. Among these methods, generalized eigenvalue proximal SVM (GEPSVM)[5,6,7] does binary classification by formulating two eigenvalue problems instead of quadratic programming problem to generate two nonparallel planes which make it extremely fast in real-world applications. The eigenvectors corresponding to the smallest eigenvalues determines the above two planes, as shown in Fig. 1.

* Corresponding author. E-mail:liangjun@ujs.edu.cn.

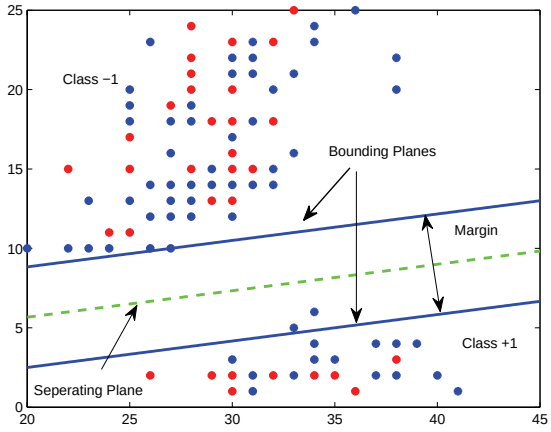

Fig. 1. The examples of standard SVM

The last decade has witnessed the emergence of manifold learning as a vigorous paradigm for dimensionality reduction and data visualization. The basic assumption of manifold learning is that though probably each data point consists thousands of features, it may actually be sampled from a low dimensional manifold embedded in a high dimensional space. The popular manifold learning methods, including Locally Linear Embedding (LLE)[8], ISOMAP[9], Laplacian Eigenmap[10] and their various extensions[11,12], have gained successful applications in many fields. In addition to the applications in dimensionality reductions, manifold regularization[13] has been proposed as a new form of regularization that allows us to exploit the geometry of the distribution of the data points. It has also been used in SVMs, e.g., Laplacian SVMs[14,15], to take advantage of unlabeled data points. In the literature, manifold learning, especially locality preserving projections criteria[16], are widely used to reduce dimensionality of the original data in feature extraction or penalize classifier using unlabeled data samples in semi-supervised learning. However, there is little evidence which has shown whether supervised classifier can benefit from manifold structure directly from labeled samples. Recently, the idea of manifold learning has been applied in GEPSVM by considering the local information within each class to eliminate the influence of outliers[17].

With the geometric intuition of GEPSVM and the essence of manifold learning, we argue that manifold regularization is particularly suitable for GEPSVM and its variants. In contrast with conventional SVMs, GEPSVM and other recently proposed classifiers, e.g., Twin SVMs[4,18], are all based on seeking planes best fitting the data points of the same class rather than just separating points of different classes. Twin SVMs are non-parallel planar classifiers whose target is to construct a classification hyper plane for two kinds of data, which makes the sample of each super plane distance as close as possible, and the distance of the samples as far as possible. In other words, they are not only "discriminative" based approach but also "expressive" based approach. Therefore, it's natural to make attempts to combine manifold structure with these methods. Another reason is that the numerators of the objective functions of GEPSVM have similar formulations as least square regression which is known to tend to overfit data points. That's why Tikhonov regularization terms[19] are integrated in GEPSVM to enhance its generalization ability and avoid overfitting. But they merely penalize the numerators whereas ignoring the denominators. Furthermore, in many cases, manifold regularization is more powerful as it takes the intrinsic geometry structure of the data points into account. In this paper, manifold regularization, specifically locality preserving projections (LPP)[20,21], is firstly extended in our context and then incorporated into classical GEPSVMs. Locality Preserving Projections (LPP) are linear projective maps that arise by solving a variation problem that optimally preserves the neighborhood structure of the data set[22,23]. LPP should be seen as an alternative to Principal Component Analysis (PCA)—a classical linear technique that projects the data along the directions of maximal variance. When the high dimensional data lies on a low dimensional manifold embedded in the ambient space, the Locality Preserving Projections are obtained by finding the optimal linear approximations to the Eigen functions of the Laplace Beltrami operator on the manifold[24,25]. LPP shares many of the data representation properties of nonlinear techniques such as Laplacian Eigen maps or Locally Linear Embedding. Yet LPP is linear and more crucially is defined everywhere in ambient space rather than just on the training data points. LPP may be conducted in the original space or in the reproducing kernel Hilbert space into which

data points are mapped. This gives rise to kernel LPP. We should emphasize that the purpose of manifold regularization in our context is different from the classical one whose primary aim is to utilize the distribution information of unlabeled samples. Unlike semi-supervised[26] learning, the intention of the proposed approach is to demonstrate how manifold structure of labeled samples is also beneficial to supervised learning even when unlabeled samples are available. Twin SVM is a non-parallel planar classifier shown in Fig. 2, its target is to construct a classification hyper plane for two kinds of data, which makes the sample of each super plane distance as close as possible, and the distance of the samples as far as possible.

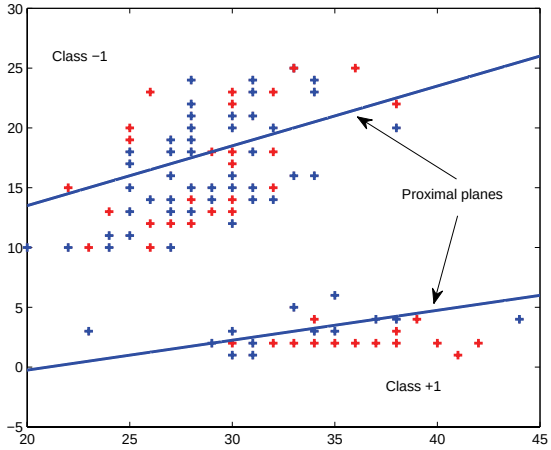

Fig. 2. The examples of Twin SVM

In this paper, MRGEPSVM is derived by designing a modified locality preserving projections criteria and incorporating it into GEPSVM. Unlike conventional LPP, we project the data points onto planes and require the local geometry structure to be preserved. It demonstrates that manifold regularization is suitable for GEPSVM and its variants.

The remainder of this paper is organized as follows: In section 2, we give a brief overview of GEPSVM followed by the construction of local preserving projections criteria in section 3. Section 4 presents manifold regularized GEPSVM based on the novel regularization. In section 5, detailed experimental results are given and we conclude this paper in section 6.

2. A brief review of GEPSVM

Supposed we are given m_1 training samples belongs to class 1 and m_2 training samples belongs to class 2 in the n-dimensional real space, with $m_1 + m_2 = m$. Let A and B be two matrices whose rows are samples of class 1 and class 2 respectively. Let w_1 and w_2 be the normal to the two planes respectively, GEPSVM seeks two nonparallel planes given by:

$$\begin{aligned} x^T w_1 + b_1 &= 0 \\ x^T w_2 + b_2 &= 0 \end{aligned} \tag{1}$$

so that each one is close to the points of one class but far away from the points of the other. This idea yields the following optimization problems.

$$\begin{cases} \min\limits_{w_1, b_1 \neq 0} \dfrac{\|Aw_1 + e_1 b_1\|^2 / \|w_1\|^2}{\|Bw_1 + e_2 b_1\|^2 / \|w_1\|^2} \\ \min\limits_{w_2, b_2 \neq 0} \dfrac{\|Bw_2 + e_2 b_2\|^2 / \|w_2\|^2}{\|Aw_2 + e_1 b_2\|^2 / \|w_2\|^2} \end{cases} \tag{2}$$

where $Bw_1 + e_2 b_1 \neq 0$, and $Aw_2 + e_1 b_2 \neq 0$, while e_1 and e_2 are vectors of ones of appropriate dimensionality. In order to achieve good generalization, Tikhonov regularization term 12 is introduced, leading to the following optimization problems.

$$\begin{cases} \min\limits_{w_1, b_1 \neq 0} \dfrac{\|Aw_1 + e_1 b_1\|^2 + \|[w_1; b_1]\|^2}{\|Bw_1 + e_2 b_1\|^2} \\ \min\limits_{w_2, b_2 \neq 0} \dfrac{\|Bw_2 + e_2 b_2\|^2 + \|[w_1; b_1]\|^2}{\|Aw_2 + e_1 b_2\|^2} \end{cases} \tag{3}$$

where $Bw_1 + e_2 b_1 \neq 0$, and $Aw_2 + e_1 b_2 \neq 0$. By defining:

$$\begin{aligned} z_1^T &= \left[w_1^T, b_1\right] \\ z_2^T &= \left[w_2^T, b_2\right] \\ P &= [Ae_1]^T [Ae_1] \\ Q &= [Be_2]^T [Be_2] \end{aligned} \tag{4}$$

The objective functions in (2) can be written as the following Rayleigh quotient problems:

$$\begin{cases} \min\limits_{z_1 \neq 0} \dfrac{z_1^T (P+\delta_1 I) z_1}{z_1^T Q z_1} \\ \min\limits_{z_2 \neq 0} \dfrac{z_2^T (P+\delta_1 I) z_2}{z_2^T Q z_2} \end{cases} \tag{5}$$

The optimal z_1 and z_2, i.e. the coefficients of the two planes, are determined by solving a pair of generalized eigenvalue problems respectively. As to the detailed algorithm of GEPSVM, please refer to Ref. 3.

3. Local Preserving Projections on Plane

In this section, the idea of local preserving projections is extended in the case that we expect the intrinsic geometry structure of original data samples of the same class can be preserved after they are projected to a plane.

Firstly, in order to preserve the original local geometry structure within each class, we need to construct a graph for each class, named as with-in class graph $G^{(c)}(c = 1,2)$. Each vertex in $G^{(c)}$ corrspinds to a sample point which belongs to class c and an edge between a vertex pair is added when the corresponding sample pair is each other's k-nearest neighbors. After the structure of $G^{(c)}$ is fixed, we need to determine its weight matrix $S^{(c)}$. In the literature, there exist several methods for weight calculation, e.g., RBF kernel, inverse Euclidean distance. In our experiments, RBF kernel is selected as follows: $S_{ij}^{(c)} = \exp\left(-\left\|x_i^{(c)} - x_j^c\right\|^2 / \delta\right)$ if $x_i^{(c)}$ and $x_j^{(c)}$ are neighbors and belonging to the same class c , otherwise $S_{ij}^{(c)} = 0$.

After with-in graphs have been constructed in input space, we project the points of each class onto their corresponding fitting planes and extend LPP criteria under this condition. The detailed procedure is given as follows.

Give an arbitrary point x in $\mathbb{R}^n$ and let y be its orthogonal projection onto the fitting plane $w^T x + b = 0$, we can obtain the following equality from Fig. 3.

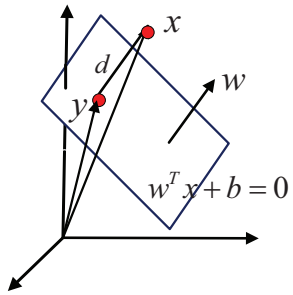

Fig. 3. Illustration of the geometry of projecting x onto $w^T x + b = 0$.

$$y + \frac{w}{\|w\|} d = x \tag{6}$$

After a simple calculation as described in Ref. 15, the value of d can be expressed as:

$$d = \frac{w^T x + b}{\|w\|} \tag{7}$$

On substituting (7) into (6), the orthogonal projection of x can be formulated as:

$$\begin{aligned} y &= x - \frac{w}{\|w\|} d \\ &= x - \frac{\left(w^T x + b\right) w}{\|w\|^2} \\ &= x - \left(w^T x + b\right) \overline{w} \end{aligned} \tag{8}$$

where $\overline{w}$ is defined as:

$$\overline{w} = \frac{w}{\|w\|^2} \tag{9}$$

According to our idea, the local geometry structure within the points of each class should be preserved after the projection. Take class 1 for example, this leads to the following optimization problem.

$$\min_{w_1} \sum_{i,i=1}^{m_1} \left\| y_i^{(1)} - y_j^{(1)} \right\|^2 S_{ij}^{(1)} \tag{10}$$

where $y_i^{(1)}$ is the orthogonal project of point $x_j^{(1)}$, i.e., i-th point of class 1, onto its corresponding fitting plane $w_1^T x + b_1 = 0$. Following (8), $y_i^{(1)}$ can be expressed explicitly in the form:

$$y_i^{(1)} = x_i^{(1)} - \frac{\left(w_1^T x_i^{(1)} + b\right) w_1}{\|w_1\|^2} = x_i^{(1)} - \left(w_1^T x_i^{(1)} + b\right) \overline{w}_1 \tag{11}$$

where:

$$\overline{w}_1 = \frac{w_1}{\|w_1\|^2} \tag{12}$$

Thus, the objective function of (10) is reformulated as:

$$\begin{aligned}
&\sum_{i,i=1}^{m_1} \left\| y_i^{(1)} - y_j^{(1)} \right\|^2 S_{ij}^2 \\
&= \sum_{i,i=1}^{m_1} \left\| x_i^{(1)} - \left(w_1^T x_i^{(1)} + b\right) \overline{w}_1 - x_j^{(1)} + \left(w_1^T x_j^{(1)} + b\right) \overline{w}_1 \right\|^2 S_{ij}^{(1)} \\
&= \sum_{i,i=1}^{m_1} \left\| \left(x_i^{(1)} - x_j^{(1)}\right) - w_1^T \left(x_i^{(1)} - x_j^{(1)}\right) \overline{w}_1 \right\|^2 S_{ij}^{(1)}
\end{aligned} \tag{13}$$

For the sake of simplicity, let us define:

$$d_{ij}^{(1)} = x_i^{(1)} - x_j^{(1)} = A^T t_i - A^T t_j = A^T (t_i - t_j) \tag{14}$$

where l_i is m_1-dimensional unit vector with the i-th element 1, 0 otherwise. Therefore, (13) can be expressed in the form:

$$\begin{aligned}
&\sum_{i,j=1}^{m_1} \left\| y_i^{(1)} - y_j^{(1)} \right\|^2 S_{ij}^{(1)} \\
&= \sum_{i,j=1}^{m_1} \left\| d_{ij}^{(1)} - w_1^T d_{ij}^{(1)} \overline{w}_1 \right\|^2 S_{ij}^{(1)} \\
&= \sum_{i,j=1}^{m_1} \left(d_{ij}^{(1)} - w_1^T d_{ij}^{(1)} \overline{w}_1\right)^T \left(d_{ij}^{(1)} - w_1^T d_{ij}^{(1)} \overline{w}_1\right) S_{ij}^{(1)} \\
&= \sum_{i,j=1}^{m_1} \left({d_{ij}^{(1)}}^T - w_1^T d_{ij}^{(1)} {\overline{w}_1}^T\right) \left(d_{ij}^{(1)} - w_1^T d_{ij}^{(1)} \overline{w}_1\right) S_{ij}^{(1)} \\
&= \sum_{i,j=1}^{m_1} \left({d_{ij}^{(1)}}^T d_{ij}^{(1)} - 2 w_1^T d_{ij}^{(1)} {d_{ij}^{(1)}}^T \overline{w}_1 + \left(w_1^T d_{ij}^{(1)}\right)^2 {\overline{w}_1}^T \overline{w}_1\right) S_{ij}^{(1)}
\end{aligned} \tag{15}$$

By dropping the terms irrelative to optimize variables and making use of the expression of w_1 and $d_{ij}^{(1)}$, i.e., (12) and (14), (15) can be further formulated as following:

$$
\begin{aligned}
&\sum_{i,j=1}^{m_1} \left\| y_i^{(1)} - y_j^{(1)} \right\|^2 S_{ij}^{(1)} \\
&\quad \propto \sum_{i,j=1}^{m_1} \left(-2 w_1^T d_{ij}^{(1)} {d_{ij}^{(1)}}^T \overline{w}_1 + \left(w_1^T d_{ij}^{(1)} \right)^2 {\overline{w}_1}^T \overline{w}_1 \right) S_{ij}^{(1)} \\
&\quad = \sum_{i,j=1}^{m_1} \left(-2 \frac{\left(w_1^T d_{ij}^{(1)} \right)^2}{\|w_1\|^2} + \frac{\left(w_1^T d_{ij}^{(1)} \right)^2}{\|w_1\|^2} \right) S_{ij}^{(1)} \\
&\quad = -\frac{1}{\|w_1\|^2} \sum_{i,j=1}^{m_1} \left(w_1^T d_{ij}^{(1)} \right)^2 S_{ij}^{(1)} \\
&\quad = -\frac{1}{\|w_1\|^2} w_1^T \left(\sum_{i,j=1}^{m_1} d_{ij}^{(1)} {d_{ij}^{(1)}}^T S_{ij}^{(1)} \right) w_1 \\
&\quad = -\frac{1}{\|w_1\|^2} w_1^T A^T \left(\sum_{i,j=1}^{m_1} (t_i - t_j) (t_i - t_j)^T S_{ij}^{(1)} \right) A w_1
\end{aligned} \tag{16}
$$

where e_i is a m_1 dimensional unit vector with the i-th element 1, 0 otherwise. Now, let us derive a more explicit expression for $\sum_{i,j=1}^{m_1} (t_i - t_j)(t_i - t_j)^T S_{ij}^{(1)}$ as:

$$
\begin{aligned}
&\sum_{i,j=1}^{m_1} (t_i - t_j)(t_i - t_j)^T S_{ij}^{(1)} \\
&\quad = \sum_{i,j=1}^{m_1} (t_i - t_j) \left(t_i^T - t_j^T \right) S_{ij}^{(1)} \\
&\quad = \sum_{i,j=1}^{m_1} \left(t_i t_i^T - t_i t_j^T - t_j t_i^T + t_j t_j^T \right) S_{ij}^{(1)} \\
&\quad = 2 \left(\sum_{i=1}^{m_1} t_i t_i^T \sum_{j=1}^{m_1} S_{ij}^{(1)} - \sum_{i,j=1}^{m_1} t_i t_j^T S_{ij}^{(1)} \right) \\
&\quad = 2 \left(D^{(1)} - S^{(1)} \right) \\
&\quad = 2 L^{(1)}
\end{aligned} \tag{17}
$$

where $D^{(1)}$ is a diagonal matrix with its entries being the row sums of $S^{(1)}$, i.e. $D_{ij}^{(1)} = \sum_j S_{ij}^{(1)}$, and $L^{(1)} = D^{(1)} - S^{(1)}$ is the Laplacian matrix of the data points of class 1. On substituting (17) into (16), we have:

$$
\sum_{i,j=1}^{m_1} \left\| y_i^{(1)} - y_j^{(1)} \right\|^2 S_{ij}^{(1)} \propto -\frac{2}{\|w_1\|^2} w_1^T A^T L^{(1)} A w_1 \tag{18}
$$

Considering minimization modus operandi in (10), we finally obtain the following relationship.

$$\begin{cases} \min\limits_{w_1} \sum\limits_{i,j=1}^{m_1} \left\| y_i^{(1)} - y_j^{(1)} \right\|^2 S_{ij}^{(1)} \\ \max\limits_{w_1} \dfrac{2}{\|w_1\|^2} w_1^T A^T L^{(1)} A w_1 \end{cases} \tag{19}$$

Now we should compare our novel LPP criteria (19) with the traditional one usually expressed in the form, and X is the set of x_i.

$$\min \frac{w_1^T X^T L X w_1}{w_1^T X^T D X w_1} \tag{20}$$

which is obtained by optimizing the vector w_1 onto which each point of is X projected meanwhile best preserving the local structure within X. In our context, $X = A$. We can gain some insights into the difference between (19) and (20), i.e., minimization is converted to maximization. This phenomenon is explained as follows. First of all, it can be noticed that w_1 is merely the normal vector of the fitting plane. Furthermore, our goal is to preserve local geometry information within original space after points are projected onto the fitting plane instead of its normal vector. After determining whether the normal vector is orthogonal to its derived plane, it's clear that we reach that goal. A simple geometry example to visually illustrate this phenomenon is given in Fig. 4.

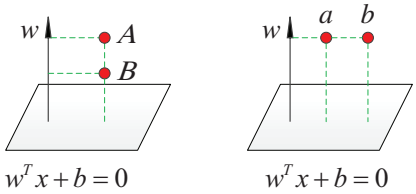

Fig. 4. Illustration of the difference between LPP and the proposed projections

It's obvious that the best vector that preserves local structure of data points corresponds to the worst plane doing so. As Fig. 4 left shows, A, B are projected meanwhile best preserving the local structure. However, in Fig. 4 right, a and b are projected, but they do not preserve the local structure.

4. Incorporate local geometry structure into GEPSVM

We are now in a position to introduce the new criteria (18) into classical GEPSVM. As aforementioned, GEPSVM aims to find two distinct planes which are not parallel to each other so that each closest to the points of one class and furthest away from the other. Recalling the objective function (5) of GEPSVM, it's obvious that its numerator is similar to least square regression, which tends to generate overfitting solutions and that's why we need regularization technique to avoid it. In original GEPSVM, Tikhonov regularization plays the role in tackling this problem. In this paper, besides Tikhonov term, manifold regularization, i.e., criteria (19), is also used to handle it.

Our idea is to seek two planes so that each one is required not only closest to the data set of its own and furthest from the other data set but also to preserve the local geometry structure of them, respectively. To achieve this, (10) is used to penalize the original GEPSVM, leading to the following modified objective functions for class 1.

$$\begin{cases} \min\limits_{w_1,b_1} \dfrac{\|Aw_1 + e_1 b_1\|^2 / \|w_1\|^2}{\|Bw_1 + e_2 b_1\|^2 / \|w_1\|^2} \\ \min\limits_{w_1} \sum\limits_{i,j=1}^{m_1} \left\| y_i^{(1)} - y_j^{(1)} \right\|^2 S_{ij}^{(1)} \end{cases} \tag{21}$$

Making the result of (19), (21) can be expressed as following:

$$\begin{cases} \min\limits_{w_1,b_1} \dfrac{\|Aw_1 + e_1 b_1\|^2 / \|w_1\|^2}{\|Bw_1 + e_2 b_1\|^2 / \|w_1\|^2} \\ \max\limits_{w_1} \dfrac{2}{\|w_1\|^2} w_1^T A^T L^{(1)} A w_1 \end{cases} \tag{22}$$

We can view the second function as manifold regularized and incorporated into the first function as in (23). At the same time, in order to balance the effect of the two objectives, trade-off factor δ_M is introduced, which leads to the following optimization problems.

$$\begin{aligned}\min_{w_1,b_1} & \frac{\|Aw_1+e_1b_1\|^2/\|w_1\|^2}{\|Bw_1+e_2b_1\|^2/\|w_1\|^2+\delta_M w_1^T A^T L^{(1)} A w_1/\|w_1\|^2} \\ = \min_{w_1,b_1} & \frac{\|Aw_1+e_1b_1\|^2}{\|Bw_1+e_2b_1\|^2+\delta_M w_1^T A^T L^{(1)} A w_1}\end{aligned} \tag{23}$$

Under the definition in GEPSVM, (23) can be written as:

$$\begin{aligned}\min_{w_1,b_1} & \frac{\|Aw_1+e_1b_1\|^2}{\|Bw_1+e_2b_1\|^2+\delta_M \begin{bmatrix} w_1^T & b_1 \end{bmatrix} \begin{bmatrix} A^T L^{(1)} A & 0_n \\ 0_n^T & 0 \end{bmatrix} \begin{bmatrix} w_1 \\ b_1 \end{bmatrix}} \\ & = \frac{z_1^T P z_1}{z_1^T Q z_1+\delta_M z_1^T M^{(1)} z_1} \\ & = \frac{z_1^T P z_1}{z_1^T \left(Q+\delta_M M^{(1)}\right) z_1}\end{aligned} \tag{24}$$

where $\begin{bmatrix} A^T L^{(1)} A & 0_n \\ 0_n^T & 0 \end{bmatrix}$ and 0_n is a vector of zero with order n. By combining (24) with Tikhonov regularization with appropriate trade-off factor δ_T, we obtain the following optimization problems.

$$\min_{z_1 \neq 0} \frac{z_1^T (P+\delta_T I) z_1}{z_1^T \left(Q+\delta_M M^{(1)}\right) z_1} \tag{25}$$

Then, the optimal z_1, i.e. $[w_1; b_1]$, is the eigenvector corresponding to the smallest eigenvalue of the following generalized eigenvalue problem.

$$(P+\delta_T I) z_1 = \lambda \left(Q+\delta_M M^{(1)}\right) z_1 \tag{26}$$

Following the similar procedure above, we define an analogous minimization problem to (25) for determining $Z_2 = [w_2; b_2]$, i.e., the coefficient of the second plane $w_2^T x + b_2 = 0$ by:

$$\min_{z_2 \neq 0} \frac{z_2^T (Q+\delta_T I) z_2}{z_2^T \left(P+\delta_M M^{(1)}\right) z_2} \tag{27}$$

where $M^{(2)} = \begin{bmatrix} B^T L^{(2)} B & 0_n \\ 0_n^T & 0 \end{bmatrix}$, and $L^{(2)} = D^{(2)} - S^{(2)}$.

The minimum of (27) is achieved at the eigenvector corresponding to the smallest eigenvalue of the following generalized eigenvalue problem.

$$(Q+\delta_T I) z_2 = \lambda \left(P+\delta_M M^{(2)}\right) z_2 \tag{28}$$

Once we obtain the eigenvectors corresponding to the smallest eigenvalues of (26) and (28), the two planes are determined at the same time. Therefore, for a new coming sample x, we first calculate the distance from x to the two planes given by (29):

$$\begin{aligned} d_1 &= |x^T w_1 + b_1| / \|w_1\|^2 \\ d_2 &= |x^T w_2 + b_2| / \|w_2\|^2 \end{aligned} \tag{29}$$

Then we classify x as belonging to class 1 if otherwise $d_1 < d_2$ to class 2.

We now describe our simple algorithm as follows for implementing a linear MRGEPSVM.

Algorithm 1: *Linear Manifold Regularized Proximal Support Vector Machine via Generalized Eigenvalue*

Given m_1 samples in $\mathbb{R}^n$ represented by $A \in \mathbb{R}^{m_1 \times n}$ and m_2 samples represented by $B \in \mathbb{R}^{m_2 \times n}$:

(i) Define the augmented matrix P, Q, $M^{(1)}$ and $M^{(2)}$ in feature space

(ii) Solve the eigenvalue problems of (36) and (38);

(iii) For the new coming sample x, compute the distances d_1, d_2, and classify it to the nearest one.

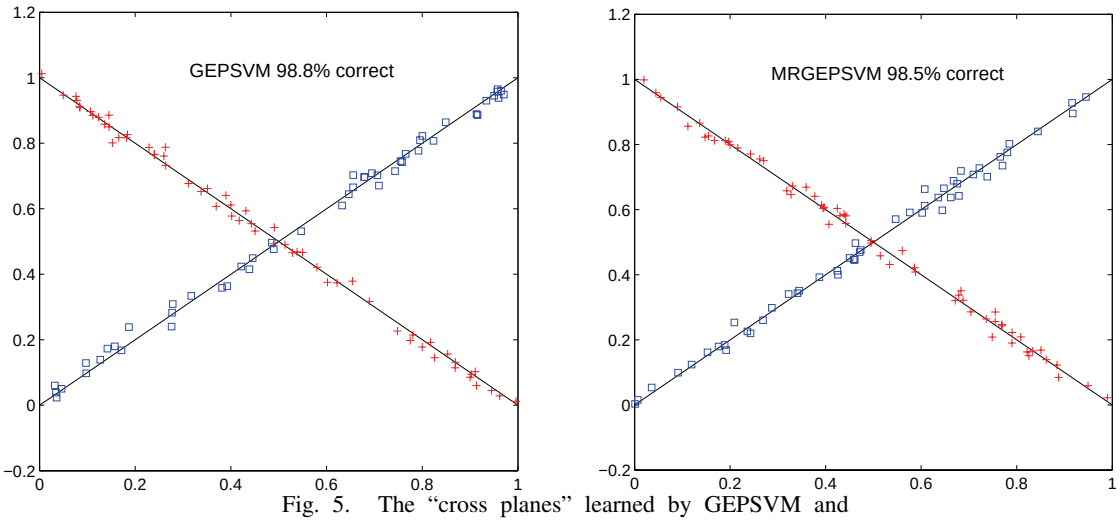

Fig. 5. The "cross planes" learned by GEPSVM and MRGEPSVM together with their correctness on the training data set

Originally, GEPSVM were proposed to tackle XOR problems which is difficult for conventional linear classifications such as SVMs, by which it can obtain nearly 100% correct. A "cross planes" example is show in Fig. 5 for GEPSVM and the proposed MRGEPSVM. We observe that MRGEPSVM obtains nearly the same accuracy as GEPSVM, which implies the ability of GEPSVM to solve XOR problems is not depressed in MRGEPSVM by introducing manifold regularization technique.

5. Nonlinear MRGEPSVM

In this section, we extend our results to nonlinear classifiers. Suppose that the Euclidean space $\mathbb{R}^n$ is mapped to a Hilbert space H, named as feature space instead of input space $\mathbb{R}^n$, through a nonlinear mapping function ϕ: $\mathbb{R}^n \to H$. Let $K(x_i, x_j)$ be kernel function in feature space satisfying $K(x_i, x_j) = \phi(x_i)^T \phi(x_j)$.The data samples A and B are mapped to $\phi(A)$ and $\phi(B)$, respectively. Let the fitting planes in feature space be denoted by:

$$\begin{aligned} \phi\left(x^T\right) w_1 + b_1 = 0 \\ \phi\left(x^T\right) w_2 + b_2 = 0 \end{aligned} \tag{30}$$

therefore, (23) can be expressed in following form in feature space.

$$\min_{w_1,b_1} \frac{\|\phi(A)w_1 + e_1 b_1\|^2}{\|\phi(B)w_1 + e_2 b_1\|^2 + \delta_M w_1^T \phi(A)^T L^{(1)} \phi(A) w_1} \tag{31}$$

A basic principle in feature space is that w_1 can be expressed as the linear combinations of all data samples in H, leading to:

$$w_1 = \phi\left(C^T\right) u_1 \tag{32}$$

where $C^T = \begin{bmatrix} A^T & B^T \end{bmatrix}$, and $u_1 \in \mathbb{R}^m$.

Substituting (32) into (31), we obtain:

$$\begin{aligned} \min_{u_1,b_1} & \frac{\left\|K\left(A, C^T\right) u_1 + e_1 b_1\right\|^2}{\|K(B, C^T) u_1 + e_2 b_1\|^2 + \delta_M u_1^T K(C, A^T) L^{(1)} K(A, C^T) u_1} \\ &= \frac{z_1^T P z_1}{z_1^T Q z_1 + \delta_M z_1^T M^{(1)} z_1} \end{aligned} \tag{33}$$

where:

$$\begin{aligned} P &= \begin{bmatrix} K\left(A, C^T\right) & e_1 \end{bmatrix}^T \begin{bmatrix} K\left(A, C^T\right) & e_1 \end{bmatrix} \\ Q &= \begin{bmatrix} K\left(A, C^T\right) & e_2 \end{bmatrix}^T \begin{bmatrix} K\left(A, C^T\right) & e_2 \end{bmatrix} \\ M^{(1)} &= \begin{bmatrix} K\left(C, A^T\right) L^{(1)} K\left(A, C^T\right) & 0_m \\ 0_m^T & 0 \end{bmatrix} \end{aligned} \tag{34}$$

Similarly, by combining (33) with Tikhonov regularization, we obtain:

$$\min_{z_1 \neq 0} \frac{z_1^T (P + \delta_T I) z_1}{z_1^T \left(Q + \delta_M M^{(1)}\right) z_1} \tag{35}$$

By an entirely similar argument, we can define an analogous minimization problem for the fitting plane of class 2.

$$\min_{z_2 \neq 0} \frac{z_1^T (P + \delta_T I) z_2}{z_2^T \left(Q + \delta_M M^{(2)}\right) z_2} \tag{36}$$

where:

$$M^{(2)} = \begin{bmatrix} K\left(C, B^T\right) L^{(1)} K\left(B, C^T\right) & 0_m \\ 0_m^T & 0 \end{bmatrix} \tag{37}$$

After z_1 and z_2 are solved, two fitting planes in feature space are determined. In order to derive the decision rule in feature space, suppose we gave a new data point $x \in \mathbb{R}^n$ to be classified. Then the distance between its image in feature space, i.e. $\phi(x)$, and the fitting plane of class 1, i.e. $\phi\left(x^T\right) w_1 + b_1 = 0$, is given by:

$$d_1 = \frac{\left|\phi\left(x^T\right) w_1 + b_1\right|}{\|w_1\|^2} \tag{38}$$

By using (32), (38) can be expressed as:

$$\begin{aligned} d_1 &= \left|\phi\left(x^T\right) \phi\left(C^T\right) u_1 + b_1\right| / u_1^T \phi(C) \phi\left(C^T\right) u_1 \\ &= \left|K\left(x^T, C^T\right) u_1 + b_1\right| / u_1^T K\left(C, C^T\right) u_1 \end{aligned} \tag{39}$$

Similarly, the distance between and the fitting plane of class 2, i.e., $\phi\left(x^T\right) w_2 + b_2 = 0$, is given by:

$$d_2 = \left|K\left(x^T, C^T\right) u_2 + b_2\right| / u_2^T K\left(C, C^T\right) u_2 \tag{40}$$

The decision rule for nonlinear MRGEPSVM is the same as its linear version, expect the distances from x to fitting planes need to be calculated by (35) and (36).

We now give an explicit statement for nonlinear MRGEPSVM algorithm.

Algorithm 2: *Nonlinear Manifold Regularized Proximal Support Vector Machine via Generalized Eigenvalue*

Given m_1 samples in $\mathbb{R}^n$ represented by $A \in \mathbb{R}^{m_1 \times n}$ and m_2 samples represented by $B \in \mathbb{R}^{m_2 \times n}$.

(i) Define the matrix P, Q, $M^{(1)}$ and $M^{(2)}$ in input space;

(ii) Solve the eigenvalue problems of (26) and (27);

(iii) For the new coming sample x, compute the distances d_1, d_2, and classify it to the nearest one.

6. Experimental Results

To demonstrate the performance of the proposed method, we conducted experiments on several benchmark datasets from UCI Repository [27]. Optimal values of all parameters involving each method were obtained by using a tuning set comprising of 10 percent of the data set. Table 1 shows the comparison of MRGEPSVM, TSVM, GEPSVM and SVM.

From the experimental results listed in Table 1, we can see that MRGEPSVM gains superior performance than GEPSVM on nearly all datasets. We also performed experiments on several UCI datasets using RBF kernel. The performances of each method are shown in Table 2.

The aforementioned methods have been tested on benchmark data sets that are publicly available. Results regard their performance in terms of classification accuracy. The results regarding the linear kernel have been obtained using the first two repositories. The third one has been used in the non-linear kernel implementation. For each data set, the latter repository offers 100 predefined random splits into training and test sets. For several algorithms, results obtained from each trial, including SVMs,

Table 1. Classification accuracy using linear kernel

	MRGEPSVM	TSVM	GEPSVM	SVM
Hepatitis	85.16 ± 2.16	80.79 ± 3.74	58.29 ± 2.13	80.00 ± 2.03
Ionosphere	87.47 ± 3.38	88.03 ± 4.69	75.19 ± 3.43	86.04 ± 5.91
Heart-statlog	84.82 ± 1.98	84.44 ± 2.62	84.81 ± 3.11	84.07 ± 4.56
Votes	95.40 ± 1.62	96.08 ± 2.73	91.93 ± 3.08	94.50 ± 2.79
Sonar	79.81 ± 1.08	77.25 ± 3.75	66.76 ± 5.44	79.79 ± 2.13
Pima	74.88 ± 3.09	73.70 ± 5.14	74.60 ± 4.16	76.68 ± 3.82

Table 2. Classification accuracy using nonlinear kernel

	MRGEPSVM	TSVM	GEPSVM	SVM
Hepatitis	81.94 ± 3.14	82.67 ± 5.03	78.25 ± 4.86	83.13 ± 3.56
BUPA	66.09 ± 2.71	67.83 ± 7.01	63.80 ± 3.25	58.32 ± 6.83
WPBC	79.72 ± 2.14	81.92 ± 4.17	62.70 ± 3.36	79.92 ± 3.16

are recorded. Execution times and the other accuracy results have been calculated using an Intel Core i7 CPU 3.20 GHz, 8 GB RAM running Windows 8 with Matlab 2013b, during normal daylight operations. In the case of nonlinear kernel, we observe there are still perceptible improvements compared with GEPSVM. Meanwhile the accuracy of MRGEPSVM is close to TSVM and SVM. It seems that manifold structure didn't greatly facilitate the performance in feature space. A possible explanation may lie in the higher dimensionality offset the effect brought by manifold structure within data samples after they are mapped.

Elapsed time are listed in Table 3 and Table 4, by different methods with Gaussian and linear kernel, respectively. In the linear one, MRGEPSVM and GEPSVM outperform TSVM and SVM in all cases. MRGEPSVM is at least twice faster than GEPSVM. When the Gaussian kernel is used, SVMs implementations achieve better performances with respect to the eigenvalues based methods. In all cases, MRGEPSVM is faster than GEPSVM.

USPS database is a handwritten numeral recognition database provided by the United States postal service, which includes 10 kinds of gray images from 0 to 9, where the gray value has been normalized, with each figure containing 1100 images of 16×16 pixels. Table 5 is the average accuracy and standard deviation of different algorithms using USPS dataset.

In this experiment, we randomly selected 110 image data sets from each class of samples, and randomly selected 10% and 20% of the data as the training set to verify the MRGEPSVM algorithm. The penalty parameters C of SVM and TWSVM, as well as the regularization terms of MRGEPSVM and GEPSVM, are selected from the collection $\{2^i|-9,\ldots,9\}$. The value k of KNN is selected from the collection $\{3,4,\ldots,20\}$.

The experimental set-up is meant to simulate a real-world situation: we considered binary classification problems due to the splits of the training data, where all of one driver cases were labeled and all the rest were left unlabeled. The test set is composed of entirely new drivers, forming the separate group.

In Fig. 6, we compare the error rates of 30 binary classification problems of GEPSVM, MRGEPSVM algorithm. We train on the same driver from a training set of examples, and test on the remaining five set of samples. We considered the task of classifying the driver whether he meets obstacles as a binary classification problem.

Table 3. Elapsed time in seconds using Gaussian kernel

Dataset	MRGEPSVM	GEPSVM	TSVM	SVM
Votes	1.1473	5.8744	0.4523	0.2022
Sonar	3.8717	5.8947	0.1543	0.4080
Pima	0.0296	0.1143	0.0302	7.1968
Flare-solar	1.9839	16.1654	2.1429	4.4562
Waveform	0.5998	4.480	0.9016	0.2284
Thyroid	0.0246	0.1280	0.0503	0.0718
Heart	0.0361	0.2187	0.0278	0.1732
Banana	0.4989	3.1102	0.0346	1.3505
Breast-cancer	0.0688	0.3544	0.0429	0.1188

Table 4. Elapsed time in seconds using linear kernel

Dataset	MRGEPSVM	GEPSVM	TSVM	SVM
Votes	0.0119	0.0277	0.0024	0.0019
Sonar	0.0364	0.0854	0.0589	0.0395
Pima	0.0015	0.2858	0.6809	0.0013
Flare-solar	0.0158	0.1673	0.1092	0.0893
Waveform	0.0013	0.0934	0.0438	0.0472
Thyroid	0.0011	0.0183	0.00524	0.0018
Heart	0.0012	0.1091	0.0019	0.0011
Banana	0.0024	0.1578	0.0063	0.0038
Breast-cancer	0.0002	0.0158	0.0016	0.0009

Table 5. The average accuracy and standard deviation of different algorithms in USPS dataset

Num.	SVM Test±Std (%)	LPP Test±Std (%)	TSVM Test±Std (%)	GEPSVM Test±Std (%)	MRGEPSVM Test±Std (%)
$l = 10$	77.14 ± 1.97	82.26 ± 1.73	81.12 ± 1.67	81.41 ± 1.85	82.26 ± 1.37
$l = 20$	84.67 ± 0.93	89.19 ± 1.00	89.40 ± 1.09	87.78 ± 1.34	89.97 ± 1.06
$l = 30$	86.07 ± 1.62	91.60 ± 1.07	91.93 ± 0.87	90.51 ± 1.13	92.99 ± 0.98
$l = 40$	88.13 ± 1.13	93.22 ± 1.02	93.49 ± 0.95	91.88 ± 0.99	94.33 ± 0.93
$l = 50$	90.48 ± 1.34	94.50 ± 0.88	94.46 ± 1.05	93.40 ± 1.26	95.50 ± 1.08

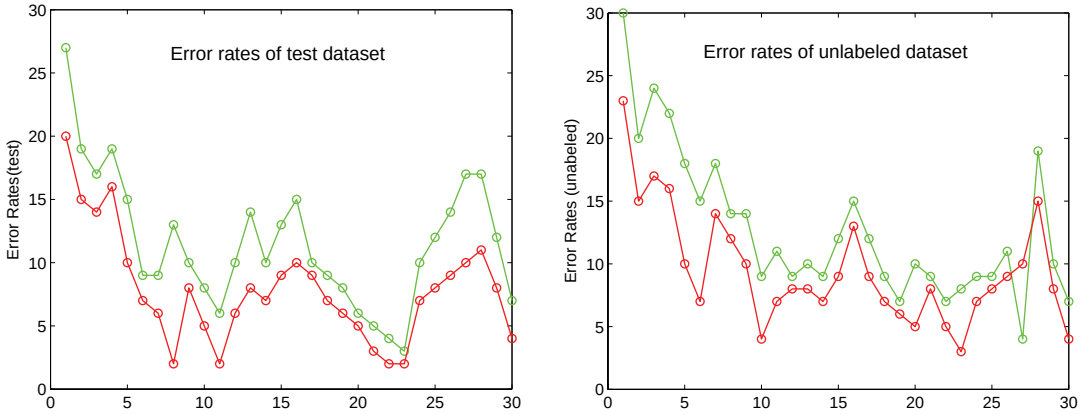

Fig. 6. Error rates of 30 binary classification problems

7. Conclusions

In this paper, a novel proximal support vector machine, called manifold regularized GEPSVM, is presented. Our analysis allows us to see the role of manifold regularization in proximal support vector machine via generalized eigenvalue in a clear way. MRGEPSVM is derived by designing a modified locality preserving projections criteria and incorporating it into GEPSVM. Unlike conventional LPP, we project the data points onto planes and require the local geometry structure to be preserved. In GEPSVM, we also solve a pair of eigenvalue problems to determine the two fitting planes. One advantage of MRGEPSVM is that it demonstrates that manifold regularization is suitable for GEPSVM and its variants. As to future work, we believe the proposed regularization technique may be applied in more powerful classifiers, e.g. Twin SVM, to further boost its performance.

Acknowledgements

This work is partially supported by the National Natural Science Foundation of China (Grant U1564201, No. 51108209, No. 50875112, No. 61573171 and No. 70972048), the Natural Science Foundation of Jiangsu Province (Grant No. BK2010339 and No. BK20140570), the Natural Science Foundation of the Jiangsu Higher Education Institutions of China (Grant No. 10KJD580001), the College Graduate Research and Innovation Program of Jiangsu Province(Dr. Innovation) (Grant No. CXLX11_0593), the Project Funded by the Priority Academic Program Development of Jiangsu Higher Education Institutions(PAPD), the Project Funded by the department of transport information (Grant 2013-364-836-900), the "Six Talent Peak" project in Jiangsu Province (DZXX-048), and the National Statistical Science Research Project (2014596).

References

1. C. J. C. Burges, A Tutorial on Support Vector Machines for Pattern Recognition, *Data Mining and Knowledge Discovery*, 2(2) (1998) 1–43.
2. R. Tempo, G. Calafiore, and F. Dabbene, *Randomized Algorithms for Analysis and Control of Uncertain Systems: With Applications*, (Springer-Verlag, London, 2012), pp. 123-134.
3. O. L. Mangasarian, and E. W. Wild, Multisurface Proximal Support Vector Machine Classification via Generalized Eigenvalues, *Pattern Analysis and Machine Intelligence*, 28(1) (2006) 69–74.
4. R. Khemchandani, and S. Chandra, Twin Support Vector Machines for Pattern Classification, *Pattern Analysis and Machine Intelligence*, 29(5) (2007) 905–910.
5. Y. Zhang, Z. Dong, S. Wang, et al., Preclinical Diagnosis of Magnetic Resonance (MR) Brain Images via Discrete Wavelet Packet Transform with Tsallis Entropy and Generalized Eigenvalue Proximal Support Vector Machine (GEPSVM), *Entropy*, 17(4) (2015) 1795–1813.
6. F. Dufrenois, and J. C. Noyer, Generalized eigenvalue proximal support vector machines for outlier description, in *2015 International Joint Conference on Neural Networks (IJCNN)*, (Killarney, Ireland, 2015), pp. 1–9.

7. W. J. Chen, Y. H. Shao, D. K. Xu, et al., Manifold proximal support vector machine for semi-supervised classification, *Applied intelligence*, 40(4) (2014) 623–638.
8. J. Wang, *Geometric Structure of High-Dimensional Data and Dimensionality Reduction*, (Springer-Verlag, Berlin, 2012), pp. 203–220.
9. Z. Zhang, T. W. S. Chow, and M. Zhao, M-Isomap: Orthogonal Constrained Marginal Isomap for Nonlinear Dimensionality Reduction, *IEEE transactions on cybernetics*, 43(1) (2013) 180–191.
10. X. He, P. Niyogi, Locality Preserving Projections, in *Advances in Neural Information Processing Systems*, Vol. 16 (MIT, 2004), pp. 153–160.
11. H. T. Chen, H. W. Chang, and T. L. Liu, Local Discriminant Embedding and Its Variants, in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, Vol. 2 (IEEE Computer Society, Washington DC, 2005), pp. 846–853.
12. C. Chen, L. Zhang, J. Bu, et al., Constrained Laplacian Eigenmap for dimensionality reduction, *Neurocomputing*, 73(4) (2010) 951–958.
13. S. Sun, Multi-view Laplacian Support Vector Machines, in *Advanced Data Mining and Applications*, (Springer-Verlag, Berlin, 2011), pp. 209–222.
14. C. M. Bishop, *Pattern Recognition and Machine Learning*, (Springer-Verlag, New York, 2006).
15. M. Belkin, P. Niyogi, Laplacian Eigenmaps for Dimensionality Reduction and Data Representation, *Neural Computation*, 15(6) (2003) 1373–1396.
16. A. N. Tikhonov, and V. Y. Arsenin, *Solutions of ill-posed problems*, (VH Winston and Sons, Washington DC, 1977).
17. X. Yang, S. Chen, B. Chen, et al., Proximal support vector machine using local information, *Neurocomputing*, 73(1) (2009) 357–365.
18. Y. H. Shao, C. H. Zhang, X. B. Wang, et al., Improvements on Twin Support Vector Machines, *IEEE Transactions on Neural Networks*, 22(6) (2011) 962–968.
19. D. Zhang, F. Wang, C. Zhang, et al., Multi-View Local Learning. in *Proceedings of the 23rd AAAI Conference on Artificial Intelligence*, Vol. 2 (AAAI, 2008), pp. 752–757.
20. J. Cheng, Q. Liu, H. Lu, et al., Supervised kernel locality preserving projections for face recognition, *Neurocomputing*, 67 (2005) 443–449.
21. Z. Zheng, X. Huang, Z. Chen, et al., Regression analysis of locality preserving projections via sparse penalty, *Information Sciences*, 303 (2015) 1–14.
22. M. Wu, and B. Schölkopf, A Local Learning Approach for Clustering, in *Advances in Neural Information Processing Systems*. Vol. 19 (MIT, 2006), pp. 1529–1536.
23. M. Wu, K. Yu, S. Yu, et al., Local Learning Projections, in *Proceedings of the 24th Annual International Conference on Machine Learning (ICML 2007)*, (ACM, Corvallis, 2007), pp. 1039-1046.
24. T. Joachims, Training linear SVMs in linear time, in *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, (ACM, New York, 2006), pp. 217-226.
25. S. Shalev-Shwartz, Y. Singer, N. Srebro, et al., Pegasos: primal estimated sub-gradient solver for SVM, *Mathematical Programming*, 127(1) (2011) 3–30.
26. D. Cai, X. He, and J. Han, Semi-supervised Discriminant Analysis, in *2007 IEEE 11th International Conference on Computer Vision*, (IEEE, Rio de Janeiro, 2007), pp. 1–7.
27. M. Lichman, UCI Machine Learning Repository, (University of California, Irvine, 2013), http://archive.ics.uci.edu/ml.