Twin Support Vector Machines for Pattern Classification

Jayadeva, *Senior Member*, *IEEE*,
R. Khemchandani, *Student Member*, *IEEE*,
and
Suresh Chandra

Abstract—We propose Twin SVM, a binary SVM classifier that determines two nonparallel planes by solving two related SVM-type problems, each of which is smaller than in a conventional SVM. The Twin SVM formulation is in the spirit of proximal SVMs via generalized eigenvalues. On several benchmark data sets, Twin SVM is not only fast, but shows good generalization. Twin SVM is also useful for automatically discovering two-dimensional projections of the data.

Index Terms—Support vector machines, pattern classification, machine learning, generalized eigenvalues, eigenvalues, eigenvectors.

1 Introduction

THE last decade has witnessed the evolution of Support Vector Machines (SVMs) as a powerful paradigm for pattern classification and regression [1], [2], [3], [4]. SVMs emerged from research in statistical learning theory on how to regulate the trade-off between structural complexity and empirical risk. One of the most popular SVM classifiers is the "maximum margin" one that attempts to reduce generalization error by maximizing the margin between two disjoint half planes [1], [2], [3], [4]. The resulting optimization task involves the minimization of a convex quadratic function subject to linear inequality constraints.

Recently, Mangasarian and Wild [5] proposed a nonparallel plane classifier for binary data classification, which they termed the generalized eigenvalue proximal support vector machine (GEPSVM). In this approach, data points of each class are proximal to one of two nonparallel planes. The nonparallel planes are eigenvectors corresponding to the smallest eigenvalues of two related generalized eigenvalue problems.

In this paper, we propose a new nonparallel plane classifier, termed as the Twin Support Vector Machine (TWSVM) for binary data classification. TWSVMs also aim at generating two nonparallel planes such that each plane is closer to one of the two classes and is as far as possible from the other. However, the formulation of TWSVMs is totally different from that of GEPSVMs and is very much in line with standard SVMs. However, TWSVMs differ from SVMs in one fundamental way. In TWSVMs, we solve a pair of quadratic programming problems (QPPs), whereas, in SVMs, we solve a single QPP. In SVMs, the QPP has all data points in the constraints, but, in TWSVMs, they are distributed in the sense that patterns of one class give the constraints of the other QPP and vice versa. This strategy of solving two smaller sized QPPs, rather than one large QPP, makes TWSVMs work faster than standard SVMs.

In practice, there are often situations where patterns belonging to one class play a more significant role in classification. Traditionally, such problems have been solved by fuzzy SVMs, e.g., Lin and Wang [6], and fuzzy proximal SVMs [7], where patterns of the more important class are assigned higher membership values. Our formulation of TWSVMs can also handle such prefential classification problems by solving only one smaller sized SVM.

The paper is organized as follows: Section 2 briefly dwells on SVMs and also introduces the notation used in the rest of the paper. Section 3 discusses generalized eigenvalue proximal support vector machines. Section 4 introduces linear Twin Support Vector Machines, while, in Section 5, we extend TWSVMs for nonlinear kernels. Section 6 deals with experimental results and Section 7 contains concluding remarks.

2 Support Vector Machines

Let the patterns to be classified be denoted by a set of m row vectors $A_i (i = 1, 2, \ldots, m)$ in the n-dimensional real space $\mathbf{R}^n$, where $A_i = (A_{i1}, A_{i2}, \ldots, A_{in})^T$. Also, let $y_i \in \{1, -1\}$ denote the class to which the ith pattern belongs. We first consider the case when the patterns belonging to the two classes are strictly linearly separable. Then, we need to determine $w \in \mathbf{R}^n$ and $b \in \mathbf{R}$ such that

$$A_i w \geq 1 - b \quad \text{for} \quad y_i = 1 \text{ and } A_i w \leq -1 - b \quad \text{for} \quad y_i = -1. \tag{1}$$

The plane described by

$$w^T x + b = 0 \tag{2}$$

lies midway between the bounding planes given by

$$w^T x + b = 1 \quad \text{and} \quad w^T x + b = -1, \tag{3}$$

and separates the two classes from each other with a margin of $\frac{1}{\|w\|_2}$ on each side. In other words, the margin of separation between the two classes is given by $\frac{2}{\|w\|_2}$. Here, $\|w\|_2$ denotes the L_2 norm of a vector w. Data samples which lie on the planes given by (3) are termed as support vectors. The maximum margin classifier, which is the standard SVM, is obtained by maximizing this margin and is equivalent to the following problem

$$\begin{aligned} (SVM1) \quad & \underset{w,b}{Min} \quad \frac{1}{2} w^T w \\ & subject\ to\ A_i w \geq 1 - b \quad \text{for} \quad y_i = 1 \quad \text{and} \quad A_i w \leq -1 - b \\ & \text{for} \quad y_i = -1. \end{aligned} \tag{4}$$

When the two classes are not strictly linearly separable, there will be an error in satisfying the inequalities (1) for some patterns and we can modify (1) to

$$\begin{aligned} & A_i w + q_i \geq 1 - b \quad \text{for} \quad y_i = 1 \text{ and } A_i w - q_i \leq -1 - b \\ & \text{for} \quad y_i = -1, \ \ q_i \geq 0, i = 1, 2 \ldots, m, \end{aligned} \tag{5}$$

where q_i denotes the error variable associated with the ith data sample. In this case, the classifier is termed as a "soft margin" one, and it approximately classifies points into two classes with some error. The classification of a given test sample x is obtained by determining the sign of $w^T x + b$. The soft margin depends on the value of the nonnegative error variables q_i. In this case, one needs to choose a trade-off between the margin and the error and the standard SVM formulation for classification of the data points with a linear kernel is given by

$$\begin{aligned} (SVM2) \qquad & \underset{w,b,q}{Min} \quad c\, e^T q + \frac{1}{2} w^T w \\ & subject\ to \\ & A_i w + q_i \geq 1 - b \quad \text{for} \quad y_i = 1, \\ & A_i w - q_i \leq -1 - b \quad \text{for} \quad y_i = -1, \\ & q_i \geq 0, \qquad i = 1, 2 \ldots, m. \end{aligned} \tag{6}$$

- *Jayadeva is with the Department of Electrical Engineering, Indian Institute of Technology, Hauz-Khas, New-Delhi-110016, India. E-mail: jayadeva@ee.iitd.ac.in.*
- *R. Khemchandani and S. Chandra are with the Department of Mathematics, Indian Institute of Technology, Hauz-Khas, New-Delhi-110016, India. E-mail: reshmaiitd@gmail.com, chandras@maths.iitd.ac.in.*

Manuscript received 2 Dec. 2005; revised 26 May 2006; accepted 26 Sept. 2006; published online 18 Jan. 2007.
Recommended for acceptance by Y. Amit.
For information on obtaining reprints of this article, please send e-mail to: tpami@computer.org, and reference IEEECS Log Number TPAMI-0673-1205.
Digital Object Identifier no. 10.1109/TPAMI.2007.1068.

0162-8828/07/$25.00 © 2007 IEEE Published by the IEEE Computer Society

Here, c denotes a scalar whose value determines the trade-off; a larger value of c emphasizes the classification error, while a smaller one places more importance on the classification margin.

In practice, rather than solving (SVM1) and (SVM2), we solve their dual problems to get the appropriate hard or soft margin classifier. The case of nonlinear kernels is handled on lines similar to linear kernels [8].

3 Generalized Eigenvalue Support Vector Machine Classifier

In this section, we give a brief outline of GEPSVMs [5]. Here, data points belonging to classes 1 and -1 are represented by matrices A and B, respectively. Let the number of patterns in classes 1 and -1 be given by m_1 and m_2, respectively. Therefore, the sizes of matrices A and B are $(m_1 \times n)$ and $(m_2 \times n)$, respectively. The GEPSVM classifier aims to determine two nonparallel planes

$$x^T w^{(1)} + b^{(1)} = 0 \quad \text{and} \quad x^T w^{(2)} + b^{(2)} = 0, \tag{7}$$

so as to minimize the Euclidean distance of the planes from the data points of classes 1 and -1, respectively. This leads to the following optimization problem:

$$\underset{w,\, b \neq 0}{Min} \quad \frac{\|Aw + eb\|^2 / \ \|[w, b]^T\|^2}{\|[Bw + eb\|^2 / \ \|[w, b]^T\|^2}, \tag{8}$$

where e is a vector of ones of appropriate dimension and $\|\cdot\|$ denotes the L_2 norm. It is implicitly assumed that $(w, b) \neq 0 \Rightarrow Bw + eb \neq 0$ [5]. On simplification, we obtain

$$\underset{w,\, b \neq 0}{Min} \quad \|Aw + eb\|^2 / \ \|Bw + eb\|^2. \tag{9}$$

The optimization problem (9) can be regularized by introducing a Tikhonov regularization term [9] as follows:

$$\underset{w,\, b \neq 0}{Min} \quad \left(\|Aw + eb\|^2 + \quad \delta \left\| [w, b]^T \right\|^2 \right) / \ \|Bw + eb\|^2 , \tag{10}$$

where $\delta > 0$. This, in turn, leads to the Rayleigh Quotient of the form

$$\underset{z \neq 0}{Min} \quad z^T G z / \ z^T H z, \tag{11}$$

where G and H are symmetric matrices in $\mathbf{R}^{(n+1)\times(n+1)}$ defined as

$$\begin{aligned} G &:= [A \quad e]^T \times [A \quad e] + \delta \times I \quad \text{for some } \delta > 0, \\ H &:= [B \quad e]^T \times [B \quad e], \quad \text{and } z := [w, \ b]^T. \end{aligned} \tag{12}$$

Using the well-known properties of the Rayleigh Quotient ([5], [10]), the solution of (11) is obtained by solving the generalized eigenvalue problem

$$Gz = \mu H z, \quad z \neq 0, \tag{13}$$

where the global minimum of (11) is achieved at an eigenvector corresponding to the smallest eigenvalue μ_{min} of (13). Therefore, if z_1 denotes the eigenvector corresponding to μ_{min}, then $[w^{(1)}, b^{(1)}]^T = z_1$ determines the plane $x^T w^{(1)} + b^{(1)} = 0$ that is close to data points of class 1. Next, we define another minimization problem analogous to (8) by interchanging the roles of A and B. The eigenvector z_2 corresponding to the smallest eigenvalue of the second generalized eigenvalue problem will yield the plane $x^T w^{(2)} + b^{(2)} = 0$, which is close to points of class -1.

4 Twin Support Vector Machines

In this section, we introduce a novel approach to SVM classification which we have termed as Twin Support Vector Machines (TWSVMs). As mentioned earlier, TWSVMs are similar to GEPSVMs in spirit, as they also obtain nonparallel planes around which the data points of the corresponding class get clustered. However, they are based on an entirely different formulation. In fact, each of the two quadratic programming problems in the TWSVM pair has the formulation of a typical SVM, except that not all patterns appear in the constraints of either problem at the same time.

The TWSVM classifier is obtained by solving the following pair of quadratic programming problems

$$\begin{aligned} (TWSVM1) \quad \underset{w^{(1)},\, b^{(1)},\, q}{Min} \quad & \frac{1}{2}(Aw^{(1)} + e_1 b^{(1)})^T (Aw^{(1)} + e_1 b^{(1)}) + c_1 e_2^T q \\ \textit{subject to} \quad & -(Bw^{(1)} + e_2 b^{(1)}) + q \geq e_2, \quad q \geq 0, \quad \text{and} \end{aligned} \tag{14}$$

$$\begin{aligned} (TWSVM2) \quad \underset{w^{(2)},\, b^{(2)},\, q}{Min} \quad & \frac{1}{2}(Bw^{(2)} + e_2 b^{(2)})^T (Bw^{(2)} + e_2 b^{(2)}) + c_2 e_1^T q \\ \textit{subject to} \quad & (Aw^{(2)} + e_1 b^{(2)}) + q \geq e_1, \quad q \geq 0, \end{aligned} \tag{15}$$

where $c_1, c_2 > 0$ are parameters and e_1 and e_2 are vectors of ones of appropriate dimensions.

The algorithm finds two hyperplanes, one for each class, and classifies points according to which hyperplane a given point is closest to. The first term in the objective function of (14) or (15) is the sum of squared distances from the hyperplane to points of one class. Therefore, minimizing it tends to keep the hyperplane close to points of one class (say class 1). The constraints require the hyperplane to be at a distance of at least 1 from points of the other class (say class -1); a set of error variables is used to measure the error wherever the hyperplane is closer than this minimum distance of 1. The second term of the objective function minimizes the sum of error variables, thus attempting to minimize misclassification due to points belonging to class -1.

In a nutshell, TWSVMs are comprised of a pair of quadratic programming problems such that, in each QPP, the objective function corresponds to a particular class and the constraints are determined by patterns of the other class. Thus, TWSVMs give rise to two smaller sized QPPs. In TWSVM1, patterns of class 1 are clustered around the plane $x^T w^{(1)} + b^{(1)} = 0$. Similarly, in TWSVM2, patterns of class -1 cluster around the plane $x^T w^{(2)} + b^{(2)} = 0$. We observe that TWSVM is approximately four times faster than the usual SVM. This is because the complexity of the usual SVM is no more than m^3, and TWSVM solves two problems, namely, (14) and (15), each of which is roughly of size $(m/2)$. Thus, the ratio of runtimes is approximately

$$\left[(m^3) / \left(2 \times \left(\frac{m}{2} \right)^3 \right) \right] = 4.$$

The Lagrangian corresponding to the problem TWSVM1 (14) is given by

$$\begin{aligned} L(w^{(1)}, b^{(1)}, q, \alpha, \beta) = & \frac{1}{2}(Aw^{(1)} + e_1 b^{(1)})^T (Aw^{(1)} + e_1 b^{(1)}) \\ & + c_1 e_2^T q - \alpha^T (-(Bw^{(1)} + e_2 b^{(1)}) + q - e_2)) - \beta^T q, \end{aligned} \tag{16}$$

where $\alpha = (\alpha_1, \alpha_2 \ldots \alpha_{m_2})^T$ and $\beta = (\beta_1, \beta_2 \ldots \beta_{m_2})^T$ are the vectors of Lagrange multipliers. The Karush-Kuhn-Tucker (K.K.T) necessary and sufficient optimality conditions [11] for (TWSVM1) are given by

$$A^T (Aw^{(1)} + e_1 b^{(1)}) + B^T \alpha = 0, \tag{17}$$

$$e_1^T (Aw^{(1)} + e_1 b^{(1)}) + e_2^T \alpha = 0, \tag{18}$$

$$c_1 e_2 - \alpha - \beta = 0, \tag{19}$$

$$-(Bw^{(1)} + e_2 b^{(1)}) + q \geq e_2, \quad q \geq 0, \tag{20}$$

$$\alpha^T (-(Bw^{(1)} + e_2 b^{(1)}) + q - e_2) = 0, \quad \beta^T q = 0, \tag{21}$$

$$\alpha \geq 0, \quad \beta \geq 0. \tag{22}$$

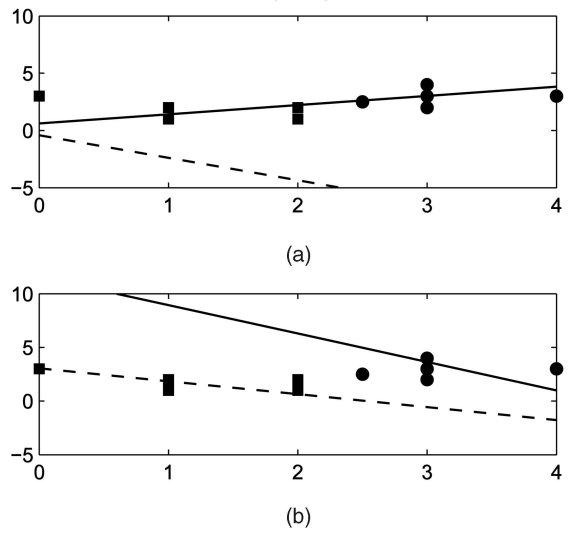

Fig. 1. (a) GEPSVM Classifier and (b) TWSVM Classifier. Points of class 1 are represented by a • and those of class -1 by a ■.

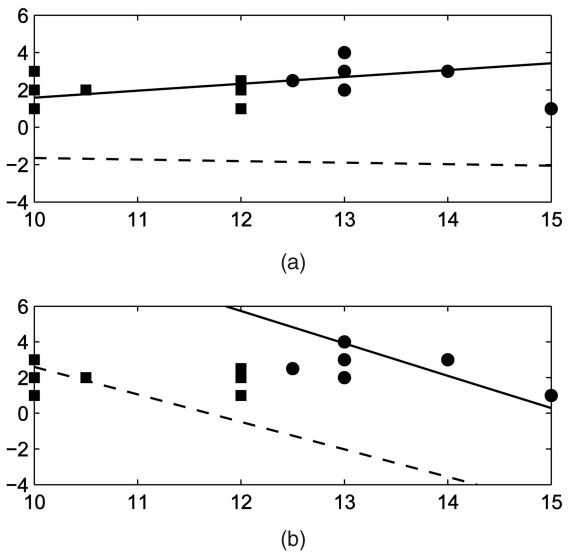

Fig. 2. (a) GEPSVM Classifier and (b) TWSVM Classifier. Points of class 1 are represented by a • and those of class -1 by a ■.

Since $\beta \geq 0$, from (19) we have

$$0 \leq \alpha \leq c_1. \tag{23}$$

Next, combining (17) and (18) leads to

$$[A^T \quad e_1^T][A \quad e_1]\left[w^{(1)}, b^{(1)}\right]^T + [B^T \quad e_2^T]\alpha = 0. \tag{24}$$

We define

$$H = [A \quad e_1], \quad G = [B \quad e_2], \tag{25}$$

and the augmented vector $u = \left[w^{(1)}, b^{(1)}\right]^T$. With these notations, (24) may be rewritten as

$$H^T H u + G^T \alpha = 0, \quad \text{i.e.,} \quad u = -(H^T H)^{-1} G^T \alpha. \tag{26}$$

Although $H^T H$ is always positive semidefinite, it is possible that it may not be well conditioned in some situations. On the lines of the regularization term introduced in Ridge Regression approaches such as [12], we introduce a regularization term ϵI, $\epsilon > 0$, to take care of problems due to possible ill-conditioning of $H^T H$. Here, I is an identity matrix of appropriate dimensions. Therefore, (26) gets modified to

$$u = -(H^T H + \epsilon I)^{-1} G^T \alpha. \tag{27}$$

However, in the following, we shall continue to use (26) with the understanding that, if need be, (27) is to be used for the determination of u.

Using (16) and the above K.K.T conditions, we obtain the Wolfe dual [11] of TWSVM1 as follows:

$$\begin{aligned} (DTWSVM1) \quad & \max_{\alpha} \quad e_2^T \alpha - \frac{1}{2} \alpha^T G (H^T H)^{-1} G^T \alpha \\ & \textit{subject to} \quad 0 \leq \alpha \leq c_1. \end{aligned} \tag{28}$$

Similarly, we consider TWSVM2 and obtain its dual as

$$\begin{aligned} (DTWSVM2) \quad & \max_{\gamma} \quad e_1^T \gamma - \frac{1}{2} \gamma^T P (Q^T Q)^{-1} P^T \gamma \\ & \textit{subject to} \quad 0 \leq \gamma \leq c_2. \end{aligned} \tag{29}$$

Here, $P = [A \quad e_1]$, $Q = [B \quad e_2]$, and the augmented vector $v = \left[w^{(2)}, b^{(2)}\right]^T$, which is given by

$$v = (Q^T Q)^{-1} P^T \gamma. \tag{30}$$

In the above discussion, the matrices $H^T H$ and $Q^T Q$ are matrices of size $(n+1) \times (n+1)$, where, in general, n is much smaller in comparison to the number of patterns of classes 1 and -1.

Once vectors u and v are known from (26) and (30), the separating planes

$$x^T w^{(1)} + b^{(1)} = 0 \quad \text{and} \quad x^T w^{(2)} + b^{(2)} = 0 \tag{31}$$

are obtained. A new data sample $x \in \mathbf{R}^n$ is assigned to class r ($r = 1, 2$), depending on which of the two planes given by (31) it lies closest to, i.e.,

$$x^T w^{(r)} + b^{(r)} = \min_{l=1,2} \quad |x^T w^{(l)} + b^{(l)}|, \tag{32}$$

where $|\cdot|$ is the perpendicular distance of point x from the plane $x^T w^{(l)} + b^{(l)} = 0$, $l = 1, 2$.

From the Karush-Kuhn-Tucker conditions (17), (18), (19), (20), (21), (22), and (23), we observe that patterns of class -1 for which $0 < \alpha_i < c_1 \ \ (i = 1, 2, \ldots m_2)$ lie on the hyperplane given by $x^T w^{(1)} + b^{(1)} = 0$. Taking motivation from standard SVMs, we can define such patterns of class -1 as support vectors of class 1 *with respect to* class -1 as they play an important role in determining the required plane. A similar observation holds for the problem TWSVM2.

At this stage, we give two simple examples to visually illustrate TWSVM and GEPSVM. Figs. 1 and 2 illustrate the classifiers obtained for the two examples by using GEPSVM and TWSVM, respectively. The data consists of points in $\mathbf{R}^2$. Points of class 1 are represented by a • and those of class -1 by a ■. The training set accuracy for TWSVM is 100 percent in both the examples, whereas, for GEPSVM, it is 70 percent and 61.53 percent, respectively.

5 The Nonlinear Kernel Classifier

In order to extend our results to nonlinear classifiers, we consider the following kernel-generated surfaces instead of planes.

$$K(x^T, C^T) u^{(1)} + b^{(1)} = 0, \quad \text{and} \quad K(x^T, C^T) u^{(2)} + b^{(2)} = 0, \tag{33}$$

$$\text{where} \quad C^T = [A \quad B]^T, \tag{34}$$

and K is an appropriately chosen kernel. Note that the planes (31) can be obtained as a special case of (33), by using a linear kernel

TABLE 1
Test Set Accuracy with a Linear Kernel

Data Set	TWSVM	GEPSVM	SVM
Heart-statlog (270×14)	84.44±4.32	84.81±3.87	84.07±4.40
Heart-c (303×14)	83.80±5.53	84.44±5.27	82.82±5.15
Hepatitis (155×19)	80.79±12.24	58.29±19.07	80.00±8.30
Ionosphere (351×34)	88.03±2.81	75.19±5.50	86.04 ±2.37
Sonar (208×60)	77.26±10.10	66.76±10.75	79.79±5.31
Votes (435×16)	96.08±3.29	91.93±3.18	94.50±2.71
Pima-Indian(768×8)	73.70±3.97	74.60±5.07	76.68±2.90
Australian (690×14)	85.80±5.05	85.65±4.60	85.51±4.58
CMC (1473×9)	67.28±2.21	65.99±2.30	67.82±2.63

Accuracies have been indicated as percentages.

$K(x^T, C^T) = x^T C$, and by defining $w^{(1)} = C^T u^{(1)}$ and $w^{(2)} = C^T u^{(2)}$. In line with the arguments in Section 4, we construct an optimization problem KTWSVM1 as follows:

$$(KTWSVM1) \quad \underset{u^{(1)},\, b^{(1)},\, q}{Min} \quad \frac{1}{2}\|(K(A, C^T)u^{(1)} + e_1 b^{(1)})\|^2 + c_1 e_2^T q$$
$$subject\ to \quad -(K(B, C^T)u^{(1)} + e_2 b^{(1)}) + q \geq e_2, \quad q \geq 0, \tag{35}$$

where $c_1 > 0$ is a parameter. Next, we define a Lagrangian L as follows:

$$\begin{aligned} L(u^{(1)}, b^{(1)}, q, \alpha, \beta) = {} & \frac{1}{2}\|(K(A, C^T)u^{(1)} + e_1 b^{(1)})\|^2 \\ & + c_1 e_2^T q - \alpha^T(-(K(B, C^T)u^{(1)} + e_2 b^{(1)}) \\ & + q - e_2) - \beta^T q. \end{aligned} \tag{36}$$

We obtain the K.K.T. conditions for (KTWSVM1) as

$$K(A, C^T)^T(K(A, C^T)u^{(1)} + e_1 b^{(1)}) + K(B, C^T)^T \alpha = 0, \tag{37}$$
$$e_1^T(K(A, C^T)u^{(1)} + e_1 b^{(1)}) + e_2^T \alpha = 0, \tag{38}$$
$$c_1 e_2 - \alpha - \beta = 0, \tag{39}$$
$$-(K(B, C^T)u^{(1)} + e_2 b^{(1)}) + q \geq e_2, \quad q \geq 0, \tag{40}$$
$$\alpha^T(-(K(B, C^T)u^{(1)} + e_2 b^{(1)}) + q - e_2) = 0, \quad \beta^T q = 0, \tag{41}$$
$$\alpha \geq 0, \quad \beta \geq 0. \tag{42}$$

Combining (37) and (38), we obtain

$$\begin{bmatrix} K(A, C^T)^T & e_1^T \end{bmatrix} \begin{bmatrix} K(A, C^T) & e_1 \end{bmatrix} \begin{bmatrix} u^{(1)}, & b^{(1)} \end{bmatrix}^T + \begin{bmatrix} K(B, C^T)^T & e_2^T \end{bmatrix} \alpha = 0. \tag{43}$$

Let

$$S = \begin{bmatrix} K(A, C^T) & e_1 \end{bmatrix}, \; R = \begin{bmatrix} K(B, C^T) & e_2 \end{bmatrix},$$

and the augmented vector $z = \begin{bmatrix} u^{(1)}, & b^{(1)} \end{bmatrix}^T$. Then, (43) can be rewritten as

$$S^T S z + R^T \alpha = 0, \quad \text{i.e.,} \quad z = -(S^T S)^{-1} R^T \alpha. \tag{44}$$

The Wolfe dual of (KTWSVM1) is given by

$$(KDTWSVM1) \quad \underset{\alpha}{Max} \quad e_2^T \alpha - \frac{1}{2}\alpha^T R(S^T S)^{-1} R^T \alpha$$
$$subject\ to \quad 0 \leq \alpha \leq c_1. \tag{45}$$

TABLE 2
Percentage Test Set Accuracy with an RBF Kernel

Data Set	TWSVM	SVM	GEPSVM
Hepatitis	82.67±10.04	83.13±11.25	78.25±11.79
WPBC	81.92±8.98	79.92±9.18	62.7*
BUPA liver	67.83±6.49	58.32±8.20	63.8*
Votes	94.72±4.72	94.94±4.33	94.2*

Accuracies have been indicated as percentages.
* *Testing accuracy figures have been obtained from [5].*

In a similar manner, by reversing the roles of $K(A, C^T)$ and $K(B, C^T)$ in (35), we obtain the optimization problem (KTWSVM2) and its dual (KDTWSVM2) for the plane $K(x^T, C^T)u^{(2)} + b^{(2)} = 0$ as follows:

$$(KTWSVM2) \quad \underset{u^{(2)},\, b^{(2)},\, q}{Min} \quad \frac{1}{2}\|(K(B, C^T)u^{(2)} + e_2 b^{(2)})\|^2 + c_2 e_1^T q$$
$$subject\ to \quad (K(A, C^T)u^{(2)} + e_1 b^{(2)}) + q \geq e_1, \quad q \geq 0, \tag{46}$$

where $c_2 > 0$ is a parameter.

$$(KDTWSVM2) \quad \underset{\gamma}{Max} \quad e_1^T \gamma - \frac{1}{2}\gamma^T L(N^T N)^{-1} L^T \gamma$$
$$subject\ to \quad 0 \leq \gamma \leq c_2. \tag{47}$$

Here, $L = [K(A, C^T) \quad e_1]$, $N = [K(B, C^T) \quad e_2]$, and the augmented vector $z_2 = \begin{bmatrix} u^{(2)}, b^{(2)} \end{bmatrix}^T$ is given by $z_2 = (N^T N)^{-1} L^T \gamma$.

Once (KDTWSVM1) and (KDTWSVM2) are solved to obtain the surfaces (33), a new pattern $x \in \mathbf{R}^n$ is assigned to class 1 or class -1 in a manner similar to the linear case.

In practice, if the number of patterns in classes 1 or -1 is large, then the rectangular kernel technique [13] can be applied to reduce the dimensionality of (KTWSVM1) and (KTWSVM2). Further, as in the linear case, we will introduce a regularization term ϵI, $\epsilon > 0$, while inverting $(S^T S)$ in (44) and (45). This allows us to use the Sherman-Morrison-Woodbury formula [14] for matrix inversion and, hence, we need to invert a matrix of a lower order m_1, instead of order m.

6 Experimental Results

The Twin Support Vector Machine (TWSVM), GEPSVM, and SVM data classification methods were implemented by using MATLAB 7 [15] running on a PC with an Intel P4 processor (3 GHz) with 1 GB RAM. The methods were evaluated on data sets from the UCI Machine Learning Repository [16]. Generalization error was determined by following the standard tenfold cross-validation methodology [17].

Table 1 summarizes TWSVM performance on some benchmark data sets available at the UCI machine learning repository [16]. The table compares the performance of the TWSVM classifier with that of SVM [8] and GEPSVM [5]. Optimal values of c_1 and c_2 were obtained by using a tuning set comprising of 10 percent of the data set. Table 2 compares the performance of the TWSVM classifier with that of SVM [8] and GEPSVM [5] using an RBF kernel. In case of the RBF kernel, we have employed a rectangular kernel [13] using 80 percent of the data. Table 3 compares the training time for 10-fold, of Gunn SVM [8] with that of TWSVM. The TWSVM training time has been determined for two cases: The first when an executable file is used and, second, when a dynamic linked library (DLL) file is used. The table indicates that TWSVM is not just effective, but also almost four times faster than a conventional SVM, because it solves

TABLE 3
Training Times (in Seconds)

Data Set	TWSVM (EXE file)	TWSVM (DLL file)	SVM (DLL file)
Hepatitis (155×19)	4.37	4.85	12.7
Sonar (208×60)	4.62	6.64	24.9
Heart-statlog (270×14)	4.72	11.3	50.9
Heart-c (303×14)	8.37	14.92	68.2
Ionosphere (351×34)	9.93	25.9	102.2
Votes (435×16)	12.8	45.8	189.4
Australian (690×14)	37.4	142.1	799.2
Pima-Indian (768×8)	56.9	231.5	1078.6
CMC (1473×9)	63.4	1737.9	6827.8

two quadratic programming problems of a smaller size instead of a single QPP of a very large size.

The nonparallel plane linear kernel classifier obtained from TWSVM can also be used to automatically discover a two-dimensional projection of the given data. Figs. 3 and 4 show two-dimensional scatter plots of the test data (comprised of 10 percent of data) for the Australian Credit and Pima Indian data sets, respectively. The plots have been obtained by plotting points with coordinates (c_i, d_i), where c_i and d_i are the respective distances of a test pattern x_i from the two hyperplanes given in (31), i.e., $c_i = |x_i^T w^{(1)} + b^{(1)}|$ and $d_i = |x_i^T w^{(2)} + b^{(2)}|$. In Figs. 3 and 4, the point x_i has been assigned to class 1 if the value of d_i is less than c_i and vice versa. In the figures, each sample is plotted as a "o" if its class label is 1, while it is plotted as a "+" if its class label is -1. Hence, the clusters of points indicate how well classification criterion is able to discriminate between the two classes. From Figs. 3 and 4, we observe that, in the case of the Australian Credit database, the two classes are well separated, while, in the case of the Pima Indian data set, the projections of the two classes are less distinct. This is also borne out by the test set accuracy for the two data sets.

7 Concluding Remarks

In this paper, we have proposed an SVM approach to data classification, termed TWSVM. In TWSVMs, we solve two quadratic programming problems of a smaller size instead of a large sized one as we do in traditional SVMs. This makes TWSVM almost four times faster than a standard SVM classifier. Furthermore, in contrast to a single hyperplane as given by traditional SVMs, TWSVMs yield two nonparallel planes such that each plane is close to one of the two data sets and is distant from the other data set. In terms of generalization, TWSVMs compare favorably with SVM and GEPSVM.

The formulation of TWSVM is also attractive for handling preferential classification problems that have traditionally been handled by the FSVM [6] and FPSVM [7] approaches. Here, we observe that TWSVM requires solving only one quadratic problem which corresponds to the important class.

When TWSVMs are used with a nonlinear kernel, the two classification problems require the inversion of matrices of order $(m_1 + 1)$ and $(m_2 + 1)$, where m_1 and m_2 are the number of patterns of classes 1 and -1, respectively. In many instances, $m_1 \gg m_2$ and a classifier may be obtained very rapidly by solving the smaller problem. This is particularly interesting for unbalanced data sets,

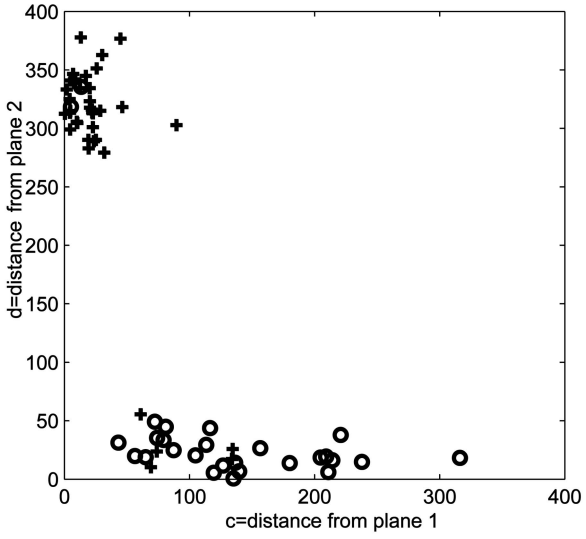

Fig. 3. Two-dimensional projection for test points from the Australian Credit data set.

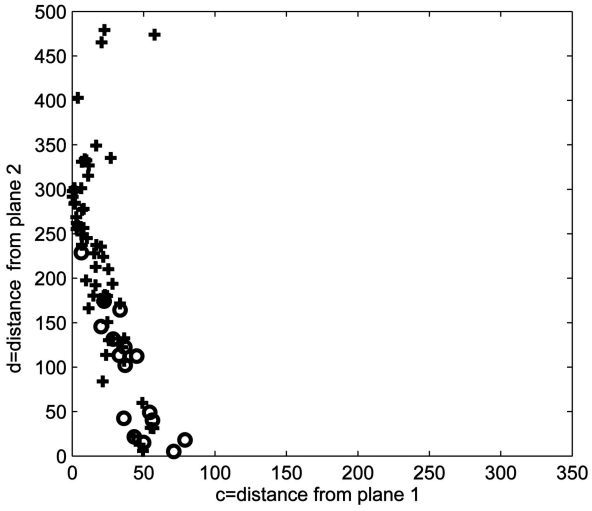

Fig. 4. Two-dimensional projection for test points from the Pima Indians data set.

e.g., in medical databases, where the number of disease-free examples may far out-number instances of the other class.

One significant advantage of TWSVM over GEPSVM is its SVM type formulation, which opens up the possibility of a SMO-type solution for faster computation. Similarly, the TWSVM approach may allow for kernel optimization via semidefinite programming and second order conic programming, as has been demonstrated by Lanckriet et al. [18] for the standard SVM classifier. These certainly seem to be promising areas for future research. Another important line of work that immediately suggests itself is to analyze the statistical properties of TWSVMs and the extension to multicategory classification.

ACKNOWLEDGMENTS

The authors are extremely thankful to the learned referees whose valuable comments have helped to improve the content and presentation of the paper. In particular, they are thankful for the observation regarding the use of TWSVMs to discover two-dimensional projections of the data sets. Also, R. Khemchandani acknowledges the financial support of the Council of Scientific and Industrial Research (India) in the form of a scholarship for pursuing her PhD degree.

REFERENCES

[1] C. Burges, "A Tutorial on Support Vector Machines for Pattern Recognition," *Data Mining and Knowledge Discovery,* vol. 2, pp. 1-43, 1998.
[2] C. Cortes and V.N. Vapnik, "Support Vector Networks," *Machine Learning,* vol. 20, pp. 273-297, 1995.
[3] P.S. Bradley and O.L. Mangasarian, "Massive Data Discrimination via Linear Support Vector Machines," *Optimization Methods and Software,* vol. 13, pp. 1-10, 2000.
[4] V. Cherkassky and F. Mulier, *Learning from Data—Concepts, Theory, and Methods.* John Wiley and Sons, 1998.
[5] O.L. Mangasarian and E.W. Wild, "Multisurface Proximal Support Vector Classification via Generalized Eigenvalues," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 28, no. 1, pp. 69-74, Jan. 2006.
[6] C.-F. Lin and S.-D. Wang, "Fuzzy Support Vector Machines," *IEEE Trans. Neural Networks,* vol. 13, no. 2, pp. 464-471, Mar. 2002.
[7] Jayadeva, R. Khemchandani, and S. Chandra, "Fast and Robust Learning through Fuzzy Linear Proximal Support Vector Machines," *Neurocomputing,* vol. 61, pp. 401-411, 2004.
[8] S.R. Gunn, "Support Vector Machines for Classification and Regression," technical report, School of Electronics and Computer Science, Univ. of Southampton, Southampton, U.K., 1998, http://www.isis.ecs.soton.ac.uk/resources/svminfo/.
[9] A.N. Tikhonov and V.Y. Arsenin, *Solution of Ill Posed Problems.* John Wiley and Sons 1977.
[10] B.N. Parlett, *The Symmetric Eigenvalue Problem.* SIAM, 1998.
[11] O.L. Mangasarian, *Nonlinear Programming.* SIAM, 1994.
[12] C. Saunders, A. Gammerman, and V. Vovk, "Ridge Regression Learning Algorithm in Dual Variables," *Proc. 15th Int'l Conf. Machine Learning,* pp. 515-521, 1998, http://citeseer.csail.mit.edu/saunders98ridge.html.
[13] G. Fung and O.L. Mangasarian, "Proximal Support Vector Machines," *Proc. Seventh Int'l Conf. Knowledge Discovery and Data Mining,* pp. 77-86, 2001.
[14] G.H. Golub and C.F. Van Loan, *Matrix Computations,* third ed. The John Hopkins Univ. Press, 1996.
[15] http://www.mathworks.com, 2007.
[16] C.L. Blake and C.J. Merz, "UCI Repository for Machine Learning Databases," Dept. of Information and Computer Sciences, Univ. of California, Irvine, http://www.ics.uci.edu/~mlearn/MLRepository.html, 1998.
[17] R.O. Duda, P.R. Hart, and D.G. Stork, *Pattern Classification,* second ed.. John Wiley and Sons, 2001.
[18] G.R.G. Lanckriet, N. Cristianini, P. Bartlett, L. El Ghaoui, and M.I. Jordan, "Learning the Kernel Matrix with Semidefinite Programming," *J. Machine Learning Research,* vol. 5, pp. 27-72, 2004.