Multi-view Twin Support Vector Machines

Xijiong Xie and Shiliang Sun [1]
Department of Computer Science and Technology, East China Normal University, 500 Dongchuan Road, Shanghai 200241, China

Abstract. Twin support vector machines are a recently proposed learning method for binary classification. They learn two hyperplanes rather than one as in conventional support vector machines and often bring performance improvements. Multi-view learning is concerned about learning from multiple distinct feature sets, which aims to exploit distinct views to improve generalization performance. In this paper, we propose multi-view twin support vector machines by solving a pair of quadratic programming problems. This paper gives a detailed derivation of the Lagrange dual optimization formulation. The linear multi-view twin support vector machines are further generalized to the nonlinear case by the kernel trick. Experimental results demonstrate that our proposed methods are effective.

Keywords. Twin support vector machines, multi-view learning, quadratic programming, Lagrange dual optimization

1. Introduction

Support vector machines (SVMs) are a powerful tool for pattern classification and regression [1,2,3,4], which are based on the principled idea of structural risk minimization in statistical learning theory. Compared with other machine learning methods such as artificial neural networks [5], SVMs own a better generalization guarantee. SVMs find the best tradeoff between the model complexity and the learning ability according to the limited example information. They can learn a nonlinear decision function which is linear in a potentially high-dimensional feature space by the use of the kernel trick [6]. So far SVMs have been successfully applied to a variety of practical problems such as object detection, text categorization, bioinformatics and image classification, etc.

Recently, Mangasarian and Wild [7] proposed generalized eigenvalue proximal SVMs (GEPSVMs) for binary classification. Instead of finding a single hyperplane as in SVMs, GEPSVMs find two nonparallel hyperplanes such that each hyperplane is as close as possible to examples from one class and as far as possible to examples from the other class. The two hyperplanes are obtained by eigenvectors corresponding to the smallest eigenvalues of two related generalized eigenvalue problems. Jayadeva et al. [8] proposed a refined nonparallel hyperplane classifier called twin SVMs (TSVMs) in the same spirit of GEPSVMs, which aim to generate two nonparallel hyperplanes such that one of the hyperplanes is closer to one class and has a certain distance to the other class. TSVMs

[1] Corresponding Author. Tel.: +86-21-54345183; Fax: +86-21-54345119; Email:*slsun@cs.ecnu.edu.cn.*

have become a popular method in machine learning because of their high classification accuracy and low computational complexity [9].

Multi-modal datasets are very common in practice because of the use of different measuring methods (e.g., infrared and visual cameras), or of different media (e.g., text, video and audio) [10]. A typical example is given by web pages. Web pages can be represented by a vector for the words in the web page text, and another vector for the words in the anchor text of a hyper-link. Multi-view learning (MVL) is an emerging direction which considers learning with multiple feature sets to improve the generalization performance. The main challenge of MVL is to develop effective algorithms to combine multiple views simultaneously. SVM-2K is a successful combination of MVL and SVMs, which combines the maximum margin and multi-view regularization principles to leverage two views to improve the classification accuracy [11]. Farquhar et al. [11] provided a theoretical analysis to illuminate the source and extent of advantage, showing a significant reduction in the Rademacher complexity of the corresponding function class [12]. Sun and Shawe-Taylor [13] proposed sparse multi-view SVMs which use a squared ε-insensitive loss. Simultaneously, they characterized the generalization error of sparse multi-view SVMs in terms of the margin bound and derived the empirical Rademacher complexity of the considered function class [14]. In this paper, we proposed multi-view twin support vector machines (MvTSVMs) which are the first TSVMs applied to MVL. MvTSVMs combine two views by introducing the constraint of similarity between two one-dimensional projections identifying two distinct TSVMs from two feature spaces.

The remainder of this paper proceeds as follows. Section 2 reviews related work including SVMs, TSVMs and SVM-2K. Section 3 thoroughly introduces our proposed MvTSVMs. Section 4 extends our method to kernel MvTSVMs. After reporting experimental results in Section 5, we give conclusions and future work in Section 6.

2. Related work

2.1. SVMs and TSVMs

Suppose there are m examples represented by matrix A with the ith row $A_i\,(i = 1,2,\cdots,m)$ being the ith example. Let $y_i \in \{1,-1\}$ denote the class to which the ith example belongs. For simplicity, we only review the linearly separable case. Then, $w \in R^d$ and $b \in R$ need to satisfy

$$y_i(A_i w + b) \geq 1. \tag{1}$$

The hyperplane described by $w^\top x + b = 0$ lies midway between the bounding hyperplanes given by $w^\top x + b = 1$ and $w^\top x + b = -1$. The margin of separation between the two classes is given by $\frac{2}{\|w\|}$, where $\|w\|$ denotes the ℓ_2 norm of w. Support vectors are those training examples lying on the above two hyperplanes. The standard SVMs [1] are obtained by solving the following problem

$$\begin{aligned} &\min_{w,b} \ \frac{1}{2} w^\top w \\ &\text{s.t.} \ \ \forall i : y_i(A_i w + b) \geq 1. \end{aligned} \tag{2}$$

The decision function is

$$f(x) = \text{sign}(w^\top x + b). \tag{3}$$

Then we introduce TSVMs [8]. Suppose examples belonging to classes 1 and -1 are represented by matrices A_+ and B_-, and the size of A_+ and B_- are $(m_1 \times d)$ and $(m_2 \times d)$, respectively. We define two matrices A, B and four vectors v_1, v_2, e_1, e_2, where e_1 and e_2 are vectors of ones of appropriate dimensions and

$$A = (A_+, e_1),\ B = (B_-, e_2),\ v_1 = \begin{pmatrix} w_1 \\ b_1 \end{pmatrix},\ v_2 = \begin{pmatrix} w_2 \\ b_2 \end{pmatrix}. \tag{4}$$

TSVMs obtain two nonparallel hyperplanes

$$w_1^\top x + b_1 = 0 \ \text{ and } \ w_2^\top x + b_2 = 0 \tag{5}$$

around which the examples of the corresponding class get clustered. The classifier is given by solving the following QPPs separately
(TSVM1)

$$\begin{aligned} \min_{v_1, q_1} \ & \frac{1}{2}(Av_1)^\top (Av_1) + c_1 e_2^\top q_1 \\ \text{s.t.} \ & -Bv_1 + q_1 \succeq e_2,\ q_1 \succeq 0, \end{aligned} \tag{6}$$

(TSVM2)

$$\begin{aligned} \min_{v_2, q_2} \ & \frac{1}{2}(Bv_2)^\top (Bv_2) + c_2 e_1^\top q_2 \\ \text{s.t.} \ & Av_2 + q_2 \succeq e_1,\ q_2 \succeq 0, \end{aligned} \tag{7}$$

where c_1, c_2 are nonnegative parameters and q_1, q_2 are slack vectors of appropriate dimensions. The label of a new example x is determined by the minimum of $|x^\top w_r + b_r|$ $(r = 1, 2)$ which are the perpendicular distances of x to the two hyperplanes given in (5).

2.2. SVM-2K

Suppose we are given two views of the same data. One view is represented by a feature projection ϕ_A with the corresponding kernel K_A and the other view is represented by a feature projection ϕ_B with the corresponding kernel K_B. Then the two views' data are given by a set $S = \{(\phi_A(x_1), \phi_B(x_1)), \cdots, (\phi_A(x_n), \phi_B(x_n))\}$. SVM-2K [11] combines the two views by introducing the constraint of similarity between two one-dimensional projections identifying two distinct SVMs from the two feature spaces:

$$|\langle w_A, \phi_A(x_i)\rangle + b_A - \langle w_B, \phi_B(x_i)\rangle - b_B| \leq \eta_i + \varepsilon \tag{8}$$

where w_A, b_A, (w_B, b_B) are the weight and threshold of the first (second) SVMs. The SVM-2K method has the following optimization for classifier parameters w_A, b_A, (w_B, b_B)

$$\begin{aligned}
\min_{w_A,w_B,q_{1i},q_{2i},\eta_i} \quad & \frac{1}{2}\|w_A\|^2+\frac{1}{2}\|w_B\|^2+c_1\sum_{i=1}^{n}q_{1i}+c_2\sum_{i=1}^{n}q_{2i}+D\sum_{i=1}^{n}\eta_i \\
\text{s.t.} \quad & |\langle w_A,\phi_A(x_i)\rangle+b_A-\langle w_B,\phi_B(x_i)\rangle-b_B|\le\eta_i+\varepsilon, \\
& y_i(\langle w_A,\phi_A(x_i)\rangle+b_A)\ge 1-q_{1i}, \\
& y_i(\langle w_B,\phi_B(x_i)\rangle+b_B)\ge 1-q_{2i}, \\
& q_{1i}\ge 0, q_{2i}\ge 0, \eta_i\ge 0, \text{all for } 1\le i\le n,
\end{aligned} \tag{9}$$

where D, c_1, c_2, ε are nonnegative parameters and q_{1i}, q_{2i}, η_i are slack vectors of appropriate dimensions. Let $\hat{w}_A$, $\hat{w}_B$, $\hat{b}_A$, $\hat{b}_B$ be the solution to this optimization problem. The final SVM-2K decision function is $f(x)=\frac{1}{2}(\langle\hat{w}_A,\phi_A(x)\rangle+\hat{b}_A+\langle\hat{w}_B,\phi_B(x)\rangle+\hat{b}_B)$.
The dual formulation of the above optimization problem can be written as

$$\begin{aligned}
\max_{\xi_i^A,\xi_j^A,\xi_i^B,\xi_j^B,\alpha_i^A,\alpha_i^B} \quad & -\frac{1}{2}\sum_{i,j=1}^{n}(\xi_i^A\xi_j^AK_A(x_i,x_j)+\xi_i^B\xi_j^BK_B(x_i,x_j))+\sum_{i=1}^{n}(\alpha_i^A+\alpha_i^B) \\
\text{s.t.} \quad & \xi_i^A=\alpha_i^Ay_i-\beta_i^++\beta_i^-, \\
& \xi_i^B=\alpha_i^By_i+\beta_i^+-\beta_i^-, \\
& \sum_{i=1}^{n}\xi_i^A=\sum_{i=1}^{n}\xi_i^B=0, \\
& 0\le\beta_i^+,\beta_i^-,\beta_i^++\beta_i^-\le D, \\
& 0\le\alpha_i^{A/B}\le c_{1/2},
\end{aligned} \tag{10}$$

where α_i^A, α_i^B, β_i^+,β_i^- are the vectors of Lagrange multipliers. Here we let $\varepsilon=0$. The prediction function for each view is given by

$$f_{A/B}(x)=\sum_{i=1}^{n}\xi_i^{A/B}K_{A/B}(x_i,x)+b_{A/B}. \tag{11}$$

3. MvTSVMs

Now we extend TSVMs to MvTSVMs. On one view, positive examples are represented by $A_1^{'}$ and negative examples are represented by $B_1^{'}$. On the other view, positive examples are represented by $A_2^{'}$ and negative examples are represented by $B_2^{'}$. For simplicity, suppose that all e are vectors of ones of appropriate dimensions and

$$\begin{aligned}
& A_1=(A_1^{'},e), B_1=(B_1^{'},e), A_2=(A_2^{'},e), B_2=(B_2^{'},e), \\
& v_1=\begin{pmatrix}w_1\\b_1\end{pmatrix},\ v_2=\begin{pmatrix}w_2\\b_2\end{pmatrix}, u_1=\begin{pmatrix}w_3\\b_3\end{pmatrix},\ u_2=\begin{pmatrix}w_4\\b_4\end{pmatrix},
\end{aligned} \tag{12}$$

where (w_1,b_1) and (w_2,b_2) are classifier parameters of $+1$ class, and (w_3,b_3) and (w_4,b_4) are classifier parameters of -1 class. The optimization problems for MvTSVMs are written as

$$
\begin{aligned}
\min_{v_1,v_2,q_1,q_2,\eta} \quad & \frac{1}{2}\|A_1v_1\|^2+\frac{1}{2}\|A_2v_2\|^2+c_1e_2^\top q_1+c_2e_2^\top q_2+De_1^\top\eta \\
\text{s.t.} \quad & |A_1v_1-A_2v_2| \preceq \eta, \\
& -B_1v_1+q_1 \succeq e_2, \\
& -B_2v_2+q_2 \succeq e_2, \\
& q_1 \succeq 0, q_2 \succeq 0, \\
& \eta \succeq 0,
\end{aligned} \tag{13}
$$

$$
\begin{aligned}
\min_{u_1,u_2,k_1,k_2,\zeta} \quad & \frac{1}{2}\|B_1u_1\|^2+\frac{1}{2}\|B_2u_2\|^2+d_1e_1^\top k_1+d_2e_1^\top k_2+He_2^\top\zeta \\
\text{s.t.} \quad & |B_1u_1-B_2u_2| \preceq \zeta, \\
& -A_1u_1+k_1 \succeq e_1, \\
& -A_2v_2+k_2 \succeq e_1, \\
& k_1 \succeq 0, k_2 \succeq 0, \\
& \zeta \succeq 0,
\end{aligned} \tag{14}
$$

where e_1 and e_2 are vectors of ones of appropriate dimensions, v_1, v_2, u_1, u_2 are classifier parameters, c_1, c_2, d_1, d_2, D, H are nonnegative parameters, and q_1, q_2, η, ζ, k_1, k_2 are slack vectors of appropriate dimensions.

The Lagrangian of the optimization problem (13) is given by

$$
\begin{aligned}
L = & \frac{1}{2}\|A_1v_1\|^2+\frac{1}{2}\|A_2v_2\|^2+c_1e_2^\top q_1+c_2e_2^\top q_2+De_1^\top\eta-\beta_1^\top(\eta-A_1v_1+A_2v_2) \\
& -\beta_2^\top(A_1v_1-A_2v_2+\eta)-\alpha_1^\top(-B_1v_1+q_1-e_2)-\alpha_2^\top(-B_2v_2+q_2-e_2) \\
& -\lambda_1^\top q_1-\lambda_2^\top q_2-\sigma^\top\eta,
\end{aligned} \tag{15}
$$

where α_1, α_2, β_1, β_2, λ_1, λ_2 and σ are the vectors of Lagrange multipliers.

We take partial derivatives of the above equation and let them be zero

$$
\begin{aligned}
\frac{\partial L}{\partial v_1} &= A_1^\top A_1v_1+A_1^\top\beta_1-A_1^\top\beta_2+B_1^\top\alpha_1=0, \\
\frac{\partial L}{\partial v_2} &= A_2^\top A_2v_2-A_2^\top\beta_1+A_2^\top\beta_2+B_2^\top\alpha_2=0, \\
\frac{\partial L}{\partial q_1} &= c_1e_2-\alpha_1-\lambda_1=0, \\
\frac{\partial L}{\partial q_2} &= c_2e_2-\alpha_2-\lambda_2=0, \\
\frac{\partial L}{\partial \eta} &= De_1-\beta_1-\beta_2-\sigma=0.
\end{aligned} \tag{16}
$$

From the above equations, we obtain

$$v_1 = (A_1^\top A_1)^{-1}[A_1^\top(\beta_2 - \beta_1) - B_1^\top \alpha_1], \tag{17}$$

$$v_2 = (A_2^\top A_2)^{-1}[A_2^\top(\beta_1 - \beta_2) - B_2^\top \alpha_2]. \tag{18}$$

It follows that

$$\begin{aligned} L = {} & (\alpha_1 + \alpha_2)^\top e_2 - \frac{1}{2}[(\beta_2 - \beta_1)^\top A_1 - \alpha_1^\top B_1] \\ & (A_1^\top A_1)^{-1}[A_1^\top(\beta_2 - \beta_1) - B_1^\top \alpha_1] - \frac{1}{2}[(\beta_1 - \beta_2)^\top A_2 \\ & - \alpha_2^\top B_2](A_2^\top A_2)^{-1}[A_2^\top(\beta_1 - \beta_2) - B_2^\top \alpha_2]. \end{aligned} \tag{19}$$

Therefore, the dual optimization formulation is

$$\begin{aligned} \min_{\xi_1,\xi_2,\alpha_1,\alpha_2} \quad & \frac{1}{2}\xi_1^\top (A_1^\top A_1)^{-1}\xi + \frac{1}{2}\xi_2^\top (A_2^T A_2)^{-1}\xi_2 - (\alpha_1 + \alpha_2)^\top e_2 \\ \text{s.t.} \quad & \xi_1 = A_1^\top(\beta_2 - \beta_1) - B_1^\top \alpha_1, \\ & \xi_2 = A_2^\top(\beta_1 - \beta_2) - B_2^\top \alpha_2, \\ & 0 \preceq \beta_1, \beta_2, \beta_1 + \beta_2 \preceq De_1, \\ & 0 \preceq \alpha_{1/2} \preceq c_{1/2} e_2. \end{aligned} \tag{20}$$

Similarly, we obtain the other dual problem

$$\begin{aligned} \min_{\rho_1,\rho_2,\lambda_1,\lambda_2} \quad & \frac{1}{2}\rho_1^\top (B_1^\top B_1)^{-1}\rho_1 + \frac{1}{2}\rho_2^\top (B_2^\top B_2)^{-1}\rho_2 - (\lambda_1 + \lambda_2)^\top e_1 \\ \text{s.t.} \quad & \rho_1 = B_1^\top(\gamma_2 - \gamma_1) - A_1^\top \lambda_1, \\ & \rho_2 = B_2^\top(\gamma_1 - \gamma_2) - A_2^\top \lambda_2, \\ & 0 \preceq \gamma_1, \gamma_2, \gamma_1 + \gamma_2 \preceq He_2, \\ & 0 \preceq \lambda_{1/2} \preceq d_{1/2} e_1 \end{aligned} \tag{21}$$

and

$$u_1 = (B_1^\top B_1)^{-1}[B_1^\top(\gamma_2 - \gamma_1) - A_1^\top \lambda_1], \tag{22}$$

$$u_2 = (B_2^\top B_2)^{-1}[B_2^\top(\gamma_1 - \gamma_2) - A_2^\top \alpha_2]. \tag{23}$$

For an example x with $x_1^{'}$ and $x_2^{'}$, if $\frac{1}{2}(|x_1^\top v_1| + |x_2^\top v_2|) \leq \frac{1}{2}(|x_1^\top u_1| + |x_2^\top u_2|)$, where $x_1 = (x_1^{'}, 1)$ and $x_2 = (x_2^{'}, 1)$, it is classified to class $+1$, otherwise class -1. For clarity, we explicitly state our linear twin support vector machines algorithm in Algorithm 1.

4. Kernel MvTSVMs

In this part, we extend MvTSVMs to the nonlinear case. The kernel-generated hyperplanes are:

Algorithm 1 Multi-view twin support vector machines

1: **Input:** $A_1^{'}$, $A_2^{'}$, $B_1^{'}$, $B_2^{'}$.
2: Obtain A_1, A_2, B_1, B_2 using (12).
3: Select penalty parameters c_1, c_2, D, d_1, d_2 and H. Usually these parameters are selected based on cross-validation.
4: Determine parameters of two decision functions (v_1, v_2) and (u_1, u_2) using (17), (18), (22), (23).
5: Calculate the decision function values $\frac{1}{2}(|x_1^\top v_1| + |x_2^\top v_2|)$ and $\frac{1}{2}(|x_1^\top u_1| + |x_2^\top u_2|)$ for a new example x with two views $x_1^{'}$ and $x_2^{'}$, where $x_1 = (x_1^{'}, 1)$ and $x_2 = (x_2^{'}, 1)$.
6: Assign the example to class $+1$ or -1 based on the minimum of the decision function values $\frac{1}{2}(|x_1^\top v_1| + |x_2^\top v_2|)$ and $\frac{1}{2}(|x_1^\top u_1| + |x_2^\top u_2|)$.

$$\begin{aligned} &K\{x_1^\top, C_1^\top\}w_1 + b_1 = 0, \quad K\{x_2^\top, C_2^\top\}w_2 + b_2 = 0, \\ &K\{x_1^\top, C_1^\top\}w_3 + b_3 = 0, \quad K\{x_2^\top, C_2^\top\}w_4 + b_4 = 0, \end{aligned} \tag{24}$$

where K is a chosen kernel function which is defined by $K\{x_i, x_j\} = (\Phi(x_i), \Phi(x_j))$. $\Phi(\cdot)$ is a nonlinear mapping from a low-dimensional feature space to a high-dimensional feature space. $\Phi(\cdot)$ is a nonlinear mapping from a low-dimensional feature space to a high-dimensional feature space. C_1 and C_2 denote training examples from view 1 and training examples from view 2 respectively, that is, $C_1 = (A_1^{'\top}, B_1^{'\top})^\top$, $C_2 = (A_2^{'\top}, B_2^{'\top})^\top$.

The optimization problems for kernel MvTSVMs are written as

$$\begin{aligned} \min_{w_1,w_2,b_1,b_2,q_1,q_2,\eta} \quad & \frac{1}{2}\|K\{A_1^{'}, C_1^\top\}w_1 + e_1 b_1\|^2 + \frac{1}{2}\|K\{A_2^{'}, C_1^\top\}w_2 + e_1 b_2\|^2 + c_1 e_2^\top q_1 \\ & + c_2 e_2^\top q_2 + D e_1^\top \eta \\ \text{s.t.} \quad & |K\{A_1^{'}, C_1^\top\}w_1 + e_1 b_1 - K\{A_2^{'}, C_2^\top\}w_2 - e_1 b_2| \preceq \eta, \\ & -K\{B_1^{'}, C_1^\top\}w_1 - e_2 b_1 + q_1 \succeq e_2, \\ & -K\{B_2^{'}, C_2^\top\}w_2 - e_2 b_2 + q_2 \succeq e_2, \\ & q_1 \succeq 0, q_2 \succeq 0, \\ & \eta \succeq 0, \end{aligned} \tag{25}$$

$$\begin{aligned} \min_{w_3,w_4,b_3,b_4,k_1,k_2,\zeta} \quad & \frac{1}{2}\|K\{B_1^{'}, C_1^\top\}w_3 + e_2 b_3\|^2 + \frac{1}{2}\|K\{B_2^{'}, C_2^\top\}w_4 + e_2 b_4\|^2 + d_1 e_1^\top k_1 \\ & + d_2 e_1^\top k_2 + H e_2^\top \zeta \\ \text{s.t.} \quad & |K\{B_1^{'}, C_1^\top\}w_3 + e_2 b_3 - K\{B_2^{'}, C_2^\top\}w_4 - e_2 b_4| \preceq \zeta, \\ & -K\{A_1^{'}, C_1^\top\}w_3 - e_1 b_3 + k_1 \succeq e_1, \\ & -K\{A_2^{'}, C_2^\top\}w_4 - e_1 b_4 + k_2 \succeq e_1, \\ & k_1 \succeq 0, k_2 \succeq 0, \\ & \zeta \succeq 0, \end{aligned} \tag{26}$$

where e_1 and e_2 are vectors of ones of appropriate dimensions, w_1, w_2, w_3, w_4, b_1, b_2, b_3, b_4 are classifier parameters, c_1, c_2, d_1, d_2, D, H are nonnegative parameters, q_1, q_2, η, ζ, k_1, k_2 are slack vectors of appropriate dimensions.

The Lagrangian of the optimization problem (25) is given by

$$\begin{aligned} L = {} & \frac{1}{2}\|K\{A_1^{'}, C_1^{\top}\}w_1 + e_1 b_1\|^2 + \frac{1}{2}\|K\{A_2^{'}, C_2^{\top}\}w_2 + e_1 b_2\|^2 + c_1 e_2^{\top} q_1 + c_2 e_2^{\top} q_2 \\ & + D e_1^{\top}\eta - \beta_1^{\top}(\eta - K\{A_1^{'}, C_1^{\top}\}w_1 - b_1 + K\{A_2^{'}, C_2^{\top}\}w_2 + b_2) \\ & - \beta_2^{\top}(K\{A_1^{'}, C_1^{\top}\}w_1 + e_1 b_1 - K\{A_2^{'}, C_2^{\top}\}w_2 - e_1 b_2 + \eta) \\ & - \alpha_1^{\top}(-K\{B_1^{'}, C_1^{\top}\}w_1 - e_2 b_1 + q_1 - e_2) \\ & - \alpha_2^{\top}(-K\{B_2^{'}, C_2^{\top}\}w_2 - e_2 b_2 + q_2 - e_2) \\ & - \lambda_1^{\top} q_1 - \lambda_2^{\top} q_2 - \sigma^{\top}\eta, \end{aligned} \tag{27}$$

where α_1, α_2, β_1, β_2, λ_1, λ_2 and σ are the vectors of Lagrange multipliers.

We take partial derivatives of the above equation and let them be zero

$$\begin{aligned} \frac{\partial L}{\partial w_1} &= K\{A_1^{'}, C_1^{\top}\}^{\top}(K\{A_1^{'}, C_1^{\top}\}w_1 + e_1 b_1) + K\{A_1^{'}, C_1^{\top}\}^{\top}\beta_1 \\ &\quad - K\{A_1^{'}, C_1^{\top}\}^{\top}\beta_2 + K\{B_1^{'}, C_1^{\top}\}^{\top}\alpha_1 = 0, \\ \frac{\partial L}{\partial b_1} &= e_1^{\top}(K\{A_1^{'}, C_1^{\top}\}w_1 + e_1 b_1) + e_1^{\top}\beta_1 - e_1^{\top}\beta_2 + e_2^{\top}\alpha_1 = 0, \\ \frac{\partial L}{\partial w_2} &= K\{A_2^{'}, C_2^{\top}\}^{\top}(K\{A_2^{'}, C_2^{\top}\}w_2 + e_1 b_2) - K\{A_2^{'}, C_2^{\top}\}^{\top}\beta_1 \\ &\quad + K\{A_2^{'}, C_2^{\top}\}^{\top}\beta_2 + K\{B_2^{'}, C_2^{\top}\}^{\top}\alpha_2 = 0, \\ \frac{\partial L}{\partial b_2} &= e_1^{\top}(K\{A_2^{'}, C_2^{\top}\}w_2 + e_1 b_2) - e_1^{\top}\beta_1 + e_1^{\top}\beta_2 + e_2^{\top}\alpha_2 = 0, \\ \frac{\partial L}{\partial q_1} &= c_1 e_2 - \alpha_1 - \lambda_1 = 0, \\ \frac{\partial L}{\partial q_2} &= c_2 e_2 - \alpha_2 - \lambda_2 = 0, \\ \frac{\partial L}{\partial \eta} &= D e_1 - \beta_1 - \beta_2 - \sigma = 0. \end{aligned} \tag{28}$$

Let

$$\begin{aligned} & E_1 = (K\{A_1^{'}, C_1^{\top}\}, e_1), F_1 = (K\{B_1^{'}, C_1^{\top}\}, e_2), \\ & E_2 = (K\{A_2^{'}, C_2^{\top}\}, e_1), F_2 = (K\{B_2^{'}, C_2^{\top}\}, e_2), \\ & v_1 = \begin{pmatrix} w_1 \\ b_1 \end{pmatrix}, \; v_2 = \begin{pmatrix} w_2 \\ b_2 \end{pmatrix}. \end{aligned} \tag{29}$$

From the above equations, we obtain

$$v_1 = (E_1^\top E_1)^{-1}[E_1^\top(\beta_2 - \beta_1) - F_1^\top \alpha_1], \tag{30}$$

$$v_2 = (E_2^\top E_2)^{-1}[E_2^\top(\beta_1 - \beta_2) - F_2^\top \alpha_2]. \tag{31}$$

It follows that

$$\begin{aligned} L = {} & (\alpha_1 + \alpha_2)^\top e_2 - \frac{1}{2}[(\beta_2 - \beta_1)^\top E_1 - \alpha_1^\top F_1] \\ & (E_1^\top E_1)^{-1}[E_1^\top(\beta_2 - \beta_1) - F_1^\top \alpha_1] - \frac{1}{2}[(\beta_1 - \beta_2)^\top E_2 \\ & - \alpha_2^\top F_2](E_2^\top E_2)^{-1}[E_2^\top(\beta_1 - \beta_2) - F_2^\top \alpha_2]. \end{aligned} \tag{32}$$

Therefore, the dual optimization formulation is

$$\begin{aligned} \min_{\xi_1,\xi_2,\alpha_1,\alpha_2} \quad & \frac{1}{2}\xi_1^\top (E_1^\top E_1)^{-1}\xi_1 + \frac{1}{2}\xi_2^\top (E_2^\top E_2)^{-1}\xi_2 - (\alpha_1 + \alpha_2)^\top e_2 \\ \text{s.t.} \quad & \xi_1 = E_1^\top(\beta_2 - \beta_1) - F_1^\top \alpha_1, \\ & \xi_2 = E_2^\top(\beta_1 - \beta_2) - F_2^\top \alpha_2, \\ & 0 \preceq \beta_1, \beta_2, \beta_1 + \beta_2 \preceq De_1, \\ & 0 \preceq \alpha_{1/2} \preceq c_{1/2} e_2. \end{aligned} \tag{33}$$

Similarly we obtain the other dual problem

$$\begin{aligned} \min_{\rho_1,\rho_2,\lambda_1,\lambda_2} \quad & \frac{1}{2}\rho_1^\top (F_1^\top F_1)^{-1}\rho_1 + \frac{1}{2}\rho_2^\top (F_2^\top F_2)^{-1}\rho_2 - (\lambda_1 + \lambda_2)^\top e_1 \\ \text{s.t.} \quad & \rho_1 = F_1^\top(\gamma_2 - \gamma_1) - E_1^\top \lambda_1, \\ & \rho_2 = F_2^\top(\gamma_1 - \gamma_2) - E_2^\top \lambda_2, \\ & 0 \preceq \gamma_1, \gamma_2, \gamma_1 + \gamma_2 \preceq He_2, \\ & 0 \preceq \lambda_{1/2} \preceq d_{1/2} e_1 \end{aligned} \tag{34}$$

and the augmented vectors $u_1 = \begin{pmatrix} w_3 \\ b_3 \end{pmatrix}$, $u_2 = \begin{pmatrix} w_4 \\ b_4 \end{pmatrix}$ are given by

$$u_1 = (B_1^\top B_1)^{-1}[B_1^\top(\gamma_2 - \gamma_1) - A_1^\top \lambda_1], \tag{35}$$

$$u_2 = (B_2^\top B_2)^{-1}[B_2^\top(\gamma_1 - \gamma_2) - A_2^\top \lambda_2]. \tag{36}$$

Suppose an example x has two views x_1 and x_2. If $\frac{1}{2}(|K\{x_1^\top, C_1^\top\}w_1 + b_1| + |K\{x_2^\top, C_2^\top\}w_2 + b_2|) \leq \frac{1}{2}(|K\{x_1^\top, C_1^\top\}w_3 + b_3| + |K\{x_2^\top, C_2^\top\}w_4 + b_4|)$, it is classified to class $+1$, otherwise class -1. For clarity, we explicitly state our kernel twin support vector machines algorithm in Algorithm 2.

Algorithm 2 Kernel multi-view twin support vector machines

1: **Input:** $A_1^{'}$, $A_2^{'}$, $B_1^{'}$, $B_2^{'}$.
2: Choose a kernel function K.
3: Obtain E_1, E_2, F_1, F_2 using (29).
4: Select penalty parameters c_1, c_2, D, d_1, d_2 and H. Usually these parameters are selected based on cross-validation.
5: Determine parameters of two decision functions (v_1, v_2) and (u_1, u_2) using (30), (31), (35), (36).
6: Calculate the decision function values $\frac{1}{2}(|K\{x_1^\top, C_1^\top\}w_1 + b_1| + |K\{x_2^\top, C_2^\top\}w_2 + b_2|)$ and $\frac{1}{2}(|K\{x_1^\top, C_1^\top\}w_3 + b_3| + |K\{x_2^\top, C_2^\top\}w_4 + b_4|)$ for a new example x with x_1 and x_2.
7: Assign the example to class $+1$ or -1 based on the minimum of the decision function values $\frac{1}{2}(|K\{x_1^\top, C_1^\top\}w_1 + b_1| + |K\{x_2^\top, C_2^\top\}w_2 + b_2|)$ and $\frac{1}{2}(|K\{x_1^\top, C_1^\top\}w_3 + b_3| + |K\{x_2^\top, C_2^\top\}w_4 + b_4|)$.

5. Experiments

In this section, we evaluate our proposed MvTSVMs on three real-world datasets. Two datasets are from UCI Machine Learning Repository: ionosphere classification and handwritten digits classification. The other dataset is about advertisement classification. Details about the three datasets are listed in Table 1.

Table 1. Datasets.

Name	Attributes	Instances	Classes
Ionosphere	34	351	2
Handwritten digits	649	2000	10
Advertisement	587/967	3279	2

5.1. Ionosphere

The ionosphere dataset was collected by a system in Goose Bay, Labrador. This system consists of a phased array of 16 high-frequency antennas with a total transmitted power on the order of 6.4 kilowatts. The targets were free electrons in the ionosphere. "Good" radar returns are those showing evidence of some type of structure in the ionosphere. "Bad" returns are those that do not and their signals pass through the ionosphere. It includes 351 instances in total which are divided into 225 "Good" (positive) instances and 126 "Bad" (negative) instances.

In our experiments, we regard original data as the first view. Then we capture 99% of the data variance while reducing the dimensionality from 34 to 21 with PCA and regard the resultant data as the second view. We use five-fold cross-validation to get the average classification accuracy rates and use a grid search strategy to select best parameters for all involved methods in the region $[2^{-7}, 2^7]$ with exponential growth 0.5. Linear kernel is chosen for the dataset. From the experimental results in Table 2, we can find that our method MvTSVMs performs better than all the other methods. SVM-2K performs nearly as well as single-view TSVM2, though still behaves worse than MvTSVMs.

Table 2. Classification accuracies and standard deviations (%) on Ionosphere.

Method	single-view TSVM1	single-view TSVM2	SVM-2K	MvTSVMs
Accuracy	86.62±2.11	88.88±3.12	88.31±6.02	**90.59**±4.60

Table 3. Classification accuracies and standard deviations (%) on Handwritten digits.

Method	single-view TSVM1	single-view TSVM2	SVM-2K	MvTSVMs
Accuracy	78.75±3.64	94.00±4.79	94.75±3.26	**96.70**±2.22

Table 4. Classification accuracies and standard deviations (%) on Advertisement.

Method	single-view TSVM1	single-view TSVM2	SVM-2K	MvTSVMs
Accuracy	93.60±1.95	90.20±1.79	92.60±2.88	**96.40**±1.95

5.2. Handwritten digits

This dataset consists of features of handwritten digits $(0 \sim 9)$ extracted from a collection of Dutch utility maps. It consists of 2000 examples (200 examples per class) with view 1 being the 76 Fourier coefficients and view 2 being the 64 Karhunen-Love coefficients of each example image. Because TSVMs are designed for binary classification while handwritten digits dataset contains 10 classes. We choose pairs $(1, 7)$ for the experiment. Gaussian kernel is chosen for the dataset. The experimental setting is the same as the above experiment. From the experimental results in Table 3, we can conclude that MvTSVMs is superior to single-view methods and SVM-2K.

5.3. Advertisement

The dataset consists of 3279 examples including 459 ads images (positive examples) and 2820 non-ads images (negative examples). The first view describes the image itself (words in the images URL, alt text and caption), while the other view contains all other features (words from the URLs of the pages that contain the image and the image points to). Here, we randomly select 500 examples therein to form the used data set. Gaussian kernel is chosen for the dataset. The experiment setting is the same as the above two experiments. From the experimental results in Table 4, we can find that our method MvTSVMs performs better than all the other methods. SVM-2K performs better than single-view TSVM2, though still behaves worse than MvTSVMs and single-view TSVM1.

6. Conclusion and Future Work

In this paper, we have proposed a novel classification method called multi-view twin support vector machines, which combine two views by introducing the constraint of similarity between two one-dimensional projections identifying two distinct TSVMs from two feature spaces. MvTSVMs construct a decision function by solving two quadratic

programming problems. We provide their dual formulation making use of Lagrange dual optimization techniques. Experimental results on multiple real-world datasets indicate that MvTSVMs are superior to single-view TSVMs and SVM-2K in classification performance. It would be interesting for future work to extend MvTSVMs to semi-supervised learning, which considers to use both labeled and unlabeled examples for classification.

Acknowledgements

This work is supported by the National Natural Science Foundation of China under Projects 61370175 and 61075005, and Shanghai Knowledge Service Platform Project (No. ZF1213).

References

[1] J. Shawe-Taylor, S. Sun, A review of optimization methodologies in support vector machines, Neurocomputing, **74**(2011), 3609-3618.

[2] V. Vapnik, The nature of statistical learning theory, New York: Springer-Verlag, 1995.

[3] N. Christianini, J. Shawe-Taylor, An introduction to support vector machines, Cambridge: Cambridge University Press, 2002.

[4] C. Burges, A tutorial on support vector machines for pattern recognition. Data Mining Knowledge Discovery, **2**(1998), 121-167.

[5] B. Ripley, Pattern recognition and neural networks, Cambridge: Cambridge University Press, 1996.

[6] B. Scholkopf, A. Smola, Learning with Kernels, Cambridge: MIT Press, 2003.

[7] O. Mangasarian, E. Wild, Multisurface proximal support vector machine classification via generalized eigenvalues, IEEE Transactions on Pattern Analysis and Machine Intelligence, **28**(2006), 69-74.

[8] R. Jayadeva, S. Khemchandani, Chandra, Twin support vector machines for pattern classification, IEEE Transactions on Pattern Analysis and Machine Intelligence, **74**(2007), 905-910.

[9] S. Ghorai, A. Mukherjee, P. Dutta, Nonparallel plane proximal classifier, Signal Processing, **89**(2009), 510-522.

[10] S. Sun, A survey of multi-view machine learning, Neural Computing and Applications, **23**(2013), 2031-2038.

[11] J. Farquhar, D. Hardoon, J. Shawe-Taylor, S. Szedmak, Two view learning: SVM-2K, theory and practice, Advances in Neural Information Processing Systems, **18**(2006), 355-362.

[12] P. Bartlett, S. Mendelson, Rademacher and Gaussian complexities: risk bounds and structural results, Journal of Machine Learning Research, **3**(2002), 463-482.

[13] S, Sun. J. Shawe-Taylor, Sparse semi-supervised learning using conjugate functions, Journal of Machine Learning Research, **11**(2010), 2423-2455.

[14] S. Sun, Multi-view Laplacian support vector machines, Lecture Notes in Computer Science, **7121**(2011), 209-222.