Applied Intelligence manuscript No.
(will be inserted by the editor)

Multi-view Laplacian Twin Support Vector Machines

Xijiong Xie · Shiliang Sun

Received: date / Accepted: date

Abstract Twin support vector machines are a recently proposed learning method for pattern classification. They learn two hyperplanes rather than one as in usual support vector machines and often bring performance improvements. Semi-supervised learning has attracted great attention in machine learning in the last decade. Laplacian support vector machines and Laplacian twin support vector machines have been proposed in the semi-supervised learning framework. In this paper, inspired by the recent success of multi-view learning we propose multi-view Laplacian twin support vector machines, whose dual optimization problems are quadratic programming problems. We further extend them to kernel multi-view Laplacian twin support vector machines. Experimental results demonstrate that our proposed methods are effective.

Keywords Twin support vector machines · Laplacian support vector machines · Laplacian twin support vector machines · semi-supervised learning · multi-view learning

Xijiong Xie
Department of Computer Science and Technology, East China Normal University, 500 Dongchuan Road, Shanghai 200241, P.R. China
E-mail: xjxie11@gmail.com

Shiliang Sun
Department of Computer Science and Technology, East China Normal University, 500 Dongchuan Road, Shanghai 200241, P.R. China (Corresponding Author)
E-mail: slsun@cs.ecnu.edu.cn

1 Introduction

Support vector machines (SVMs) are a state-of-the-art tool for pattern classification and regression problems [1–3], which originate from the idea of structural risk minization in statistical learning theory. SVMs can learn a nonlinear decision function which is linear in a potentially high-dimensional feature space [4] with the aid of the kernel trick. In practice, SVMs have been applied to a variety of domains such as object detection, text categorization, bioinformatics and image classification, etc.

In order to reduce the computational cost of SVMs, proximal support vector machines (PSVMs) [5] have been proposed. Compared with SVMs, PSVMs solve a linear equation with time complexity $O(d^3)$ (d is the dimension of the examples) while SVMs solve the convex optimization problem. In essence, PSVMs classify the examples by a hyperplane on the premise of guaranteeing the maximum margin. Mangasarian and Wild [6] proposed generalized eigenvalue proximal SVMs (GEPSVMs) which are an extension of PSVMs for binary classification. Instead of finding a single hyperplane as in PSVMs, 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. [7] proposed another nonparallel hyperplane classifier called twin SVMs (TSVMs), 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. The formulation of TSVMs is different from that of GEPSVMs and is similar to SVMs. TSVMs solve a pair of quadratic programming problems (QPPs), whereas SVMs solve a single QPP. This strategy of solving two smaller sized QPPs rather than one large QPP makes TSVMs work faster than standard SVMs [7]. Experimental results [8] show that nonparallel hyperplane classifiers given by TSVMs can indeed improve the performance of conventional SVMs [9–14].

In many machine learning tasks [15–18], labeled examples are often difficult and expensive to obtain, while unlabeled examples may be relatively easy to collect. Semi-supervised learning has attracted a great deal of attention in the last decade to deal with this situation. It can be superior to the performance of the counterpart supervised learning approaches if the unlabeled data are properly used. Some extensions of SVMs and TSVMs from supervised learning to semi-supervised learning have been proposed, e.g., transductive SVMs, semi-supervised support vector machines, Laplacian support vector machines (LapSVMs), Laplacian twin support vector machines (LapTSVMs) [19–24]. LapTSVMs [24] are a successful combination of semi-supervised learning and TSVMs, which are a generalized framework of twin support vector machines for learning from labeled and unlabeled data. By choosing appropriate parameters, LapTSVMs can degenerate to TSVMs [25, 26]. Experimental results showed that LapTSVMs are superior to LapSVMs and TSVMs in classification accuracy and the training time is more economical than LapSVMs and TSVMs.

In many real-world applications, multi-modal data are very common because of the use of different measuring methods (e.g., infrared and visual cameras), or of different media (e.g., text, video and audio) [27]. For example, 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. In content-based web-image retrieval, an image can be simultaneously described by visual features and the text surrounding the image. Multi-view learning (MVL) is an emerging direction which aims to improve classifiers

by leveraging the complementarity and consistency among distinct views [28–30]. The theories on MVL can be classified to four categories which are canonical correlation analysis, effectiveness of co-training, generalization error analysis for co-training and generalization error analysis for other MVL approaches [27].

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 classification performance [31]. Farquhar et al. [31] have provided a theoretical analysis to illuminate the effectiveness of SVM-2K, showing a significant reduction in the Rademacher complexity of the corresponding function class. Sun and Shawe-Taylor characterized the generalization error of multi-view sparse SVMs [32] and multi-view LapSVMs (MvLapSVMs) [33] in terms of the margin bound and derived the empirical Rademacher complexity of the considered function classes [34]. MvLapSVMs integrate three regularization terms respectively on function norm, manifold and multi-view regularization in the objective function. However, there is no existing multi-view extension for LapTSVMs although LapTSVMs are superior to LapSVMs. In this paper, we extend LapTSVMs to our new frameworks named by multi-view Laplacian twin support vector machines (MvLapTSVMs) which combine two views by introducing the constraint of similarity between two one-dimensional projections identifying two distinct TSVMs from two feature spaces. Compared to MvLapSVMs, there are two main differences. First, LapSVMs and LapTSVMs are different in the principle though they commonly use the manifold regularization term for semi-supervised learning. MvLapSVMs are based on LapSVMs while MvLapTSVMs are based on LapTSVMs. Second, MvLapTSVMs combine two views in the constraints rather than in the objective function. Experimental results validate that our proposed methods are effective.

The remainder of this paper proceeds as follows. Section 2 briefly reviews related work including SVMs, TSVMs, LapSVMs, LapTSVMs and SVM-2K. Section 3 introduces our proposed linear MvLapTSVMs and kernel MvLapTSVMs. After reporting experimental results in Section 4, we give conclusions in Section 5.

2 Related work

In this section, we briefly review SVMs, TSVMs, LapSVMs, LapTSVMs and SVM-2K. They constitute the foundation of our subsequent proposed methods.

2.1 SVMs and TSVMs

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

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

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

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

The decision function is

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

Then we introduce TSVMs [7]. Suppose examples belonging to classes 1 and -1 are represented by matrices A_+ and B_-, and the size of A_+ and B_- are $(l_1 \times d)$ and $(l_2 \times d)$, 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

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

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

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

(TSVM2)

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

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 LapSVMs

LapSVMs combine manifold regularization and SVMs [22]. Suppose $x_1, \cdots, x_{l+u} \in R^d$ represent a set of examples including l labeled examples and u unlabeled examples. $W_{(l+u)\times(l+u)}$ represents the similarity of every pair of examples

$$W_{ij} = exp(- \parallel x_i - x_j \parallel^2 / 2\sigma^2), \tag{8}$$

where σ is a scale parameter. The manifold regularization can be written as

$$\begin{aligned} Reg(f) &= \frac{1}{2}\sum_{i,j=1}^{l+u} W_{ij}(f(x_i)-f(x_j))^2 \\ &= \sum_{i=1}^{l+u}(\sum_{j=1}^{l+u} W_{ij})f^2(x_i) - \sum_{i,j=1}^{l+u} W_{ij}f(x_i)f(x_j) \\ &= \mathbf{f}^\top(\mathbf{V}-\mathbf{W})\mathbf{f} = \mathbf{f}^\top\mathbf{L}\mathbf{f}, \end{aligned} \tag{9}$$

where function $f : R^d \to R$ and $\mathbf{f}$=$[f(x_1), \cdots, f(x_{l+u})]$. The matrix V is diagonal with the ith diagonal entry $V_{ii} = \sum_{j=1}^{l+u} W_{ij}$. The matrix $L = V - W$ and L is the graph Laplacian of W. LapSVMs have the following optimization problem

$$\begin{aligned} \min_{f\in\mathcal{H}} \ & \frac{1}{l}\sum_{i=1}^{l}(1-y_i f(x_i))_+ + \gamma_A\|f\|^2 \\ & + \frac{\gamma_I}{(u+l)^2}\sum_{i,j=1}^{l+u} W_{ij}(f(x_i)-f(x_j))^2, \end{aligned} \tag{10}$$

where $\mathcal{H}$ is the RKHS induced by a kernel. γ_A and γ_I are respectively ambient and intrinsic regularization coefficients.

2.3 LapTSVMs

The square loss function and hinge loss function are used for TSVMs from supervised learning to semi-supervised learning. LapTSVMs [24] are similar to LapSVMs in the sense of manifold regularization. The optimization problems of LapTSVMs can be written as

$$\begin{aligned} \min_{w_1,b_1,\xi} \ & \frac{1}{2}\|A_+w_1 + e_1b_1\|^2 + c_1e_2^\top\xi + \frac{1}{2}c_2(\|w_1\|^2 + b_1^2) \\ & + \frac{1}{2}c_3(w_1^\top M^\top + e^\top b_1)L(Mw_1 + eb_1) \\ \text{s.t.} \ & -(B_-w_1 + e_2b_1) + \xi \succeq e_2,\ \xi \succeq 0, \end{aligned} \tag{11}$$

$$\begin{aligned} \min_{w_2,b_2,\eta} \ & \frac{1}{2}\|B_-w_2 + e_2b_2\|^2 + c_1e_1^\top\eta + \frac{1}{2}c_2(\|w_2\|^2 + b_2^2) \\ & + \frac{1}{2}c_3(w_2^\top M^\top + e^\top b_2)L(Mw_2 + eb_2) \\ \text{s.t.} \ & -(A_+w_2 + e_1b_2) + \eta \succeq e_1,\ \eta \succeq 0, \end{aligned} \tag{12}$$

where M includes all of labeled data and unlabeled data. L is the graph Laplacian. e_1, e_2 and e are vectors of ones of appropriate dimensions. w_1, b_1, w_2, b_2 are classifier parameters. c_1, c_2 and c_3 are nonnegative parameters. ξ and η are slack vectors of appropriate dimensions. The dual problem of (11) and (12) respectively can be written as

$$\begin{aligned} \max_{\alpha} \ & e_2^\top\alpha - \frac{1}{2}\alpha^\top G(H^\top H + c_2I + c_3J^\top LJ)^{-1}G^\top\alpha \\ \text{s.t.} \ & 0 \preceq \alpha \preceq c_1e_2, \end{aligned} \tag{13}$$

$$\begin{aligned} &\max_{\beta} \ e_1^\top \beta - \frac{1}{2}\beta^\top H(G^\top G + c_2 I + c_3 J^\top L J)^{-1} H^\top \beta \\ &\text{s.t. } 0 \preceq \beta \preceq c_1 e_1, \end{aligned} \tag{14}$$

where

$$\begin{aligned} &v_1 = \begin{pmatrix} w_1 \\ b_1 \end{pmatrix}, \ v_2 = \begin{pmatrix} w_2 \\ b_2 \end{pmatrix}, \\ &H = (A_+, e_1), \ J = (M, e), \ G = (B_-, e_2). \end{aligned} \tag{15}$$

α and β are the vectors of nonnegative Lagrange multipliers. I is an identity matrix of appropriate dimensions. v_1, v_2 can be obtained simultaneously

$$v_1 = -(H^\top H + c_2 I + c_3 J^\top L J)^{-1} G^\top \alpha, \tag{16}$$

$$v_2 = -(G^\top G + c_2 I + c_3 J^\top L J)^{-1} H^\top \beta. \tag{17}$$

According to matrix theory, it can be easily proved that $H^\top H + c_2 I + c_3 J^\top L J$ is a positive definite matrix. LapTSVMs obtain two nonparallel hyperplanes

$$w_1^\top x + b_1 = 0 \quad \text{and} \quad w_2^\top x + b_2 = 0. \tag{18}$$

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 (18).

2.4 SVM-2K

Suppose that we are given two views of the same data, view 1 is represented by a feature projection ϕ_A with the corresponding kernel function k_A and view 2 is represented by a feature projection ϕ_B with the corresponding kernel function k_B. Then the two-view 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 [31] combines the two views by introducing the constraint of similarity between two one-dimensional projections identifying two distinct SVMs from the two feature spaces:

$$|\langle w_A, \phi_A(x_i)\rangle + b_A - \langle w_B, \phi_B(x_i)\rangle - b_B| \le \eta_i + \epsilon \tag{19}$$

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

$$\begin{aligned} \min_{w_A, w_B, q_{1i}, q_{2i}, \eta_i} \quad & \frac{1}{2}\|w_A\|^2 + \frac{1}{2}\|w_B\|^2 + c_1 \sum_{i=1}^{n} q_{1i} + c_2 \sum_{i=1}^{n} q_{2i} + D \sum_{i=1}^{n} \eta_i \\ \text{s.t.} \quad & |\langle w_A, \phi_A(x_i)\rangle + b_A - \langle w_B, \phi_B(x_i)\rangle - b_B| \le \eta_i + \epsilon, \\ & y_i(\langle w_A, \phi_A(x_i)\rangle + b_A) \ge 1 - q_{1i}, \\ & y_i(\langle w_B, \phi_B(x_i)\rangle + b_B) \ge 1 - q_{2i}, \\ & q_{1i} \ge 0, q_{2i} \ge 0, \eta_i \ge 0, \text{all for } 1 \le i \le n, \end{aligned} \tag{20}$$

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

$$\begin{aligned}
\min_{\xi_i^A,\xi_j^A,\xi_i^B,\xi_j^B,\alpha_i^A,\alpha_i^B} \quad & \frac{1}{2}\sum_{i,j=1}^{n}(\xi_i^A\xi_j^A k_A(x_i,x_j) + \xi_i^B\xi_j^B k_B(x_i,x_j)) - \sum_{i=1}^{n}(\alpha_i^A + \alpha_i^B) \\
\text{s.t.} \quad & \xi_i^A = \alpha_i^A y_i - \beta_i^+ + \beta_i^-, \\
& \xi_i^B = \alpha_i^B y_i + \beta_i^+ - \beta_i^-, \\
& \sum_{i=1}^{n}\xi_i^A = \sum_{i=1}^{n}\xi_i^B = 0, \\
& 0 \le \beta_i^+, \beta_i^-, \beta_i^+ + \beta_i^- \le D, \\
& 0 \le \alpha_i^{A/B} \le c_{1/2},
\end{aligned} \tag{21}$$

where α_i^A, α_i^B, β_i^+, β_i^- are the vectors of nonnegative Lagrange multipliers and we have taken $\epsilon = 0$. The prediction function for each view is given by

$$f_{A/B}(x) = \sum_{i=1}^{n}\xi_i^{A/B}k_{A/B}(x_i,x) + b_{A/B}. \tag{22}$$

3 Our proposed methods

3.1 Linear MvLapTSVMs

In this part, we extend LapTSVMs to multi-view learning. Here on view 1, positive examples are represented by $A_1^{'}$ and negative examples are represented by $B_1^{'}$. On view 2, positive examples are represented by $A_2^{'}$ and negative examples are represented by $B_2^{'}$. The optimization problems of linear MvLapTSVMs can be written as

$$\begin{aligned}
\min_{w_1,w_2,b_1,b_2,q_1,q_2,\eta} \quad & \frac{1}{2}\|A_1^{'}w_1 + e_1b_1\|^2 + \frac{1}{2}\|A_2^{'}w_2 + e_1b_2\|^2 + c_1e_2^{\top}q_1 + c_2e_2^{\top}q_2 \\
& + \frac{1}{2}c_3(\|w_1\|^2 + b_1^2 + \|w_2\|^2 + b_2^2) \\
& + \frac{1}{2}c_4[(w_1^{\top}M_1^{'\top} + e^{\top}b_1)L_1(M_1^{'}w_1 + eb_1) \\
& + (w_2^{\top}M_2^{'\top} + e^{\top}b_2)L_2(M_2^{'}w_2 + eb_2)] + De_1^{\top}\eta \\
\text{s.t.} \quad & |A_1^{'}w_1 + e_1b_1 - A_2^{'}w_2 - e_1b_2| \preceq \eta, \\
& -B_1^{'}w_1 - e_2b_1 + q_1 \succeq e_2, \\
& -B_2^{'}w_2 - e_2b_2 + q_2 \succeq e_2, \\
& q_1 \succeq 0,\ q_2 \succeq 0, \\
& \eta \succeq 0,
\end{aligned} \tag{23}$$

$$
\begin{aligned}
\min_{w_3,w_4,b_3,b_4,q_3,q_4,\zeta} \quad & \frac{1}{2}\|B_1^{'}w_3 + e_2b_3\|^2 + \frac{1}{2}\|B_2^{'}w_4 + e_2b_4\|^2 + c_1e_1^{\top}q_3 + c_2e_1^{\top}q_4 \\
& + \frac{1}{2}c_3(\|w_3\|^2 + b_3^2 + \|w_4\|^2 + b_4^2) \\
& + \frac{1}{2}c_4[(w_3^{\top}M_1^{'\top} + e^{\top}b_3)L_1(M_1^{'}w_3 + eb_3) \\
& + (w_4^{\top}M_2^{'\top} + e^{\top}b_4)L_2(M_2^{'}w_4 + eb_4)] + He_2^{\top}\zeta \\
\text{s.t.} \quad & |B_1^{'}w_3 + e_2b_3 - B_2^{'}w_4 - e_2b_4| \preceq \zeta, \\
& - A_1^{'}w_3 - e_1b_3 + q_3 \succeq e_1, \\
& - A_2^{'}w_4 - e_1b_4 + q_4 \succeq e_1, \\
& q_3 \succeq 0,\ q_4 \succeq 0, \\
& \zeta \succeq 0,
\end{aligned}
\tag{24}
$$

where $M_1^{'}$ includes all of labeled data and unlabeled data from view 1. $M_2^{'}$ includes all of labeled data and unlabeled data from view 2. L_1 is the graph Laplacian of view 1 and L_2 is the graph Laplacian of view 2. e_1, e_2 and e are vectors of ones of appropriate dimensions. w_1, b_1, w_2, b_2, w_3, b_3, w_4, b_4 are classifier parameters. c_1, c_2, c_3 and c_4 are nonnegative parameters. q_1, q_2, q_3, q_4, η and ζ are slack vectors of appropriate dimensions.

The Lagrangian of the optimization problem (23) is given by

$$
\begin{aligned}
L = & \frac{1}{2}\|A_1^{'}w_1 + e_1b_1\|^2 + \frac{1}{2}\|A_2^{'}w_2 + e_1b_2\|^2 + c_1e_2^{\top}q_1 + c_2e_2^{\top}q_2 + \frac{1}{2}c_3(\|w_1\|^2 + b_1^2 \\
& + \|w_2\|^2 + b_2^2) + \frac{1}{2}c_4[(w_1^{\top}M_1^{'\top} + e^{\top}b_1)L_1(M_1^{'}w_1 + eb_1) + (w_2^{\top}M_2^{'\top} + e^{\top}b_2) \\
& L_2(M_2^{'}w_2 + eb_2)] + De_1^{\top}\eta - \beta_1^{\top}(\eta - A_1^{'}w_1 - e_1b_1 + A_2^{'}w_2 + e_1b_2) \\
& - \beta_2^{\top}(A_1^{'}w_1 + e_1b_1 - A_2^{'}w_2 - e_1b_2 + \eta) - \alpha_1^{\top}(-B_1^{'}w_1 - e_2b_1 + q_1 - e_2) \\
& - \alpha_2^{\top}(-B_2^{'}w_2 - e_2b_2 + q_2 - e_2) - \lambda_1^{\top}q_1 - \lambda_2^{\top}q_2 - \sigma^{\top}\eta,
\end{aligned}
\tag{25}
$$

where α_1, α_2, β_1, β_2, λ_1, λ_2 and σ are the vectors of nonnegative Lagrange multipliers. We take partial derivatives of the above equation and let them be zero

$$
\begin{aligned}
\frac{\partial L}{\partial w_1} &= A_1^{'\top}(A_1^{'}w_1 + e_1b_1) + c_3w_1 + c_4M_1^{'\top}L_1(M_1^{'}w_1 + eb_1) \\
&\quad + A_1^{'\top}\beta_1 - A_1^{'\top}\beta_2 + B_1^{'\top}\alpha_1 = 0, \\
\frac{\partial L}{\partial b_1} &= e_1^{\top}(A_1^{'}w_1 + e_1b_1) + c_3b_1 + c_4e^{\top}L_1(M_1^{'}w_1 + eb_1) \\
&\quad + e_1^{\top}\beta_1 - e_1^{\top}\beta_2 + e_1^{\top}\alpha_1 = 0, \\
\frac{\partial L}{\partial w_2} &= A_2^{'\top}(A_2^{'}w_2 + e_1b_2) + c_3w_2 + c_4M_2^{'\top}L_2(M_2^{'}w_2 + eb_2) \\
&\quad - A_2^{'\top}\beta_1 + A_2^{'\top}\beta_2 + B_2^{'\top}\alpha_2 = 0, \\
\frac{\partial L}{\partial b_2} &= e_1^{\top}(A_2^{'}w_2 + e_1b_2) + c_3b_2 + c_4e^{\top}L_2(M_2^{'}w_2 + eb_2) \\
&\quad - e_1^{\top}\beta_1 + e_1^{\top}\beta_2 + e_2^{\top}\alpha_2 = 0,
\end{aligned}
\tag{26}
$$

$$\frac{\partial L}{\partial q_1} = c_1 e_2 - \alpha_1 - \lambda_1 = 0,$$
$$\frac{\partial L}{\partial q_2} = c_2 e_2 - \alpha_2 - \lambda_2 = 0,$$
$$\frac{\partial L}{\partial \eta} = De_1 - \beta_1 - \beta_2 - \sigma = 0.$$

We define

$$\begin{aligned} &A_1 = (A_1^{'},\, e_1),\ A_2 = (A_2^{'},\, e_1),\ B_1 = (B_1^{'},\, e_2),\ B_2 = (B_2^{'},\, e_2), \\ &J_1 = (M_1^{'},\, e),\ J_2 = (M_2^{'},\, e),\ v_1 = \begin{pmatrix} w_1 \\ b_1 \end{pmatrix},\ v_2 = \begin{pmatrix} w_2 \\ b_2 \end{pmatrix}. \end{aligned} \tag{27}$$

From the above equations, we obtain

$$A_1^\top A_1 v_1 + c_3 v_1 + c_4 J_1^\top L_1 J_1 v_1 + A_1^\top \beta_1 - A_1^\top \beta_2 + B_1^\top \alpha_1 = 0, \tag{28}$$

$$A_2^\top A_2 v_2 + c_3 v_2 + c_4 J_2^\top L_2 J_2 v_2 - A_2^\top \beta_1 + A_2^\top \beta_2 + B_2^\top \alpha_2 = 0. \tag{29}$$

It follows that

$$v_1 = (A_1^\top A_1 + c_3 I + c_4 J_1^\top L_1 J_1)^{-1}[A_1^\top (\beta_2 - \beta_1) - B_1^\top \alpha_1], \tag{30}$$

$$v_2 = (A_2^\top A_2 + c_3 I + c_4 J_2^\top L_2 J_2)^{-1}[A_2^\top (\beta_1 - \beta_2) - B_2^\top \alpha_2]. \tag{31}$$

We substitute (30), (31) into (25) and get

$$\begin{aligned} L = (\alpha_1 + \alpha_2)^\top e_2 - \frac{1}{2}[(\beta_2 - \beta_1)^\top A_1 - \alpha_1^\top B_1](A_1^\top A_1 + c_3 I \\ + c_4 J_1^\top L_1 J_1)^{-1}[A_1^\top (\beta_2 - \beta_1) - B_1^\top \alpha_1] - \frac{1}{2}[(\beta_1 - \beta_2)^\top A_2 \\ - \alpha_2^\top B_2](A_2^\top A_2 + c_3 I + c_4 J_2^\top L_2 J_2)^{-1}[A_2^\top (\beta_1 - \beta_2) - B_2^\top \alpha_2]. \end{aligned} \tag{32}$$

Therefore, the dual optimization formulation is

$$\begin{aligned} \min_{\xi_1, \xi_2, \alpha_1, \alpha_2} \quad & \frac{1}{2}\xi_1^\top (A_1^\top A_1 + c_3 I + c_4 J_1^\top L_1 J_1)^{-1} \xi_1 + \frac{1}{2}\xi_2^\top (A_2^\top A_2 + c_3 I + c_4 J_2^\top L_2 J_2)^{-1} \xi_2 \\ & - (\alpha_1 + \alpha_2)^\top e_2 \\ \text{s.t.} \quad & \xi_1 = A_1^\top (\beta_2 - \beta_1) - B_1^\top \alpha_1, \\ & \xi_2 = A_2^\top (\beta_1 - \beta_2) - B_2^\top \alpha_2, \\ & 0 \preceq \beta_1, \beta_2, \beta_1 + \beta_2 \preceq De_1, \\ & 0 \preceq \alpha_{1/2} \preceq c_{1/2} e_2. \end{aligned} \tag{33}$$

Applying the same techniques to (24), we obtain its corresponding dual optimization formulation as

$$\begin{aligned} \min_{\rho_1, \rho_2, \omega_1, \omega_2} \quad & \frac{1}{2}\rho_1^\top (B_1^\top B_1 + c_3 I + c_4 J_1^\top L_1 J_1)^{-1} \rho_1 + \frac{1}{2}\rho_2^\top (B_2^\top B_2 + c_3 I + c_4 J_2^\top L_2 J_2)^{-1} \rho_2 \\ & - (\omega_1 + \omega_2)^\top e_1 \\ \text{s.t.} \quad & \rho_1 = B_1^\top (\gamma_2 - \gamma_1) - A_1^\top \omega_1, \\ & \rho_2 = B_2^\top (\gamma_1 - \gamma_2) - A_2^\top \omega_2, \\ & 0 \preceq \gamma_1, \gamma_2, \gamma_1 + \gamma_2 \preceq He_2, \\ & 0 \preceq \omega_{1/2} \preceq c_{1/2} e_1, \end{aligned} \tag{34}$$

where 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

$$u_1 = (B_1^\top B_1 + c_3 I + c_4 J_1^\top L_1 J_1)^{-1}[B_1^\top(\gamma_2 - \gamma_1) - A_1^\top \omega_1], \tag{35}$$

$$u_2 = (B_2^\top B_2 + c_3 I + c_4 J_2^\top L_2 J_2)^{-1}[B_2^\top(\gamma_1 - \gamma_2) - A_2^\top \omega_2]. \tag{36}$$

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.

Now we compare SVM-2K and MvLapTSVMs. SVM-2K is a multi-view supervised learning method for SVMs while MvLapTSVMs are multi-view semi-supervised learning methods for TSVMs. Suppose the number of samples from either class is equal to $l/2$. SVM-2K solves a single QPP and has the computational complexity of $O((2l)^3)$, while MvLapTSVMs solve a pair of QPPs and have the computational complexity of $O(2l^3)$. About hyper-parameter selection, SVM-2K needs three hyper-parameters to select, and MvLapTSVMs need five hyper-parameters to select. Therefore, MvLapTSVMs are more efficient for multi-view learning in computational complexity.

3.2 Kernel MvLapTSVMs

Now we extend the linear MvLapTSVMs to the nonlinear case. The kernel-induced hyperplanes are:

$$\begin{aligned} &K\{x_1^\top, C_1^\top\}\lambda_1 + b_1 = 0, \quad K\{x_2^\top, C_2^\top\}\lambda_2 + b_2 = 0, \\ &K\{x_1^\top, C_1^\top\}\lambda_3 + b_3 = 0, \quad K\{x_2^\top, C_2^\top\}\lambda_4 + b_4 = 0, \end{aligned} \tag{37}$$

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. C_1 and C_2 denote training examples from view 1 and 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 can be written as

$$\begin{aligned} \min_{\lambda_1,\lambda_2,b_1,b_2,q_1,q_2,\eta} \quad & \frac{1}{2}\|K\{A_1^{'}, C_1^\top\}\lambda_1 + e_1 b_1\|^2 + \frac{1}{2}\|K\{A_2^{'}, C_2^\top\}\lambda_2 + e_1 b_2\|^2 \\ & + c_1 e_2^\top q_1 + c_2 e_2^\top q_2 + \frac{1}{2}c_3(\lambda_1^\top K_1 \lambda_1 + b_1^2 + \lambda_2^\top K_2 \lambda_2 + b_2^2) \\ & + \frac{1}{2}c_4[(\lambda_1^\top K_1 + e^\top b_1)L_1(K_1\lambda_1 + e b_1) \\ & + (\lambda_2^\top K_2 + e^\top b_2)L_2(K_2\lambda_2 + e b_2)] + D e_1^\top \eta \\ \text{s.t.} \quad & |K\{A_1^{'}, C_1^\top\}\lambda_1 + e_1 b_1 - K\{A_2^{'}, C_2^\top\}\lambda_2 - e_1 b_2| \preceq \eta, \\ & - K\{B_1^{'}, C_1^\top\}\lambda_1 - e_2 b_1 + q_1 \succeq e_2, \\ & - K\{B_2^{'}, C_2^\top\}\lambda_2 - e_2 b_2 + q_2 \succeq e_2, \\ & q_1 \succeq 0,\ q_2 \succeq 0, \\ & \eta \succeq 0, \end{aligned} \tag{38}$$

$$
\begin{aligned}
\min_{\lambda_3,\lambda_4,b_3,b_4,q_3,q_4,\zeta} \quad & \frac{1}{2}\|K\{B_1^{'},C_1^{\top}\}\lambda_3+e_2b_3\|^2+\frac{1}{2}\|K\{B_2^{'},C_2^{\top}\}\lambda_4+e_2b_4\|^2 \\
& +c_1e_1^{\top}q_3+c_2e_1^{\top}q_4+\frac{1}{2}c_3(\lambda_3^{\top}K_1\lambda_3+b_3^2+\lambda_4^{\top}K_2\lambda_4+b_4^2) \\
& +\frac{1}{2}c_4[(\lambda_3^{\top}K_1+e^{\top}b_3)L_1(K_1\lambda_3+eb_3) \\
& +(\lambda_4^{\top}K_2+e^{\top}b_4)L_2(K_2\lambda_4+eb_4)]+He_2^{\top}\zeta \\
\text{s.t.} \quad & |K\{B_1^{'},C_1^{\top}\}\lambda_3+e_2b_3-K(B_2^{'},C_2^{\top})\lambda_4-e_2b_4|\preceq\zeta, \\
& -K\{A_1^{'},C_1^{\top}\}\lambda_3-e_1b_3+q_3\succeq e_1, \\
& -K\{A_2^{'},C_2^{\top}\}\lambda_4-e_1b_4+q_4\succeq e_1, \\
& q_3\succeq 0,\ q_4\succeq 0, \\
& \zeta\succeq 0,
\end{aligned}
\tag{39}
$$

where K_1 represents kernel matrix of view 1 and K_2 represents kernel matrix of view 2. L_1 is the graph Laplacian of view 1 and L_2 is the graph Laplacian of view 2. e_1, e_2 and e are vectors of ones of appropriate dimensions. λ_1, b_1, λ_2, b_2, λ_3, b_3, λ_4, b_4 are classifier parameters. c_1, c_2, c_3 and c_4 are nonnegative parameters. q_1, q_2, q_3, q_4, η and ζ are slack vectors of appropriate dimensions.

The Lagrangian of the optimization problem (38) is given by

$$
\begin{aligned}
L= & \frac{1}{2}\|K\{A_1^{'},C_1^{\top}\}\lambda_1+e_1b_1\|^2+\frac{1}{2}\|K\{A_2^{'},C_2^{\top}\}\lambda_2+e_1b_2\|^2+c_1e_2^{\top}q_1+c_2e_2^{\top}q_2 \\
& +\frac{1}{2}c_3(\lambda_1^{\top}K_1\lambda_1+b_1^2+\lambda_2^{\top}K_2\lambda_2+b_2^2)+\frac{1}{2}c_4[(\lambda_1^{\top}K_1+e^{\top}b_1)L_1(K_1\lambda_1+eb_1) \\
& +(\lambda_2^{\top}K_2+e^{\top}b_2)L_2(K_2\lambda_2+eb_2)]+De_1^{\top}\eta-\beta_1^{\top}(\eta-K\{A_1^{'},C_1^{\top}\}\lambda_1 \\
& -e_1b_1+K\{A_2^{'},C_2^{\top}\}\lambda_2+e_1b_2)-\beta_2^{\top}(K\{A_1^{'},C_1^{\top}\}\lambda_1+e_1b_1-K\{A_2^{'},C_2^{\top}\}\lambda_2 \\
& -e_1b_2+\eta)-\alpha_1^{\top}(-K\{B_1^{'},C_1^{\top}\}\lambda_1-e_2b_1+q_1-e_2)-\alpha_2^{\top}(-K\{B_2^{'},C_2^{\top}\}\lambda_2 \\
& -e_2b_2+q_2-e_2)-\xi_1^{\top}q_1-\xi_2^{\top}q_2-\sigma^{\top}\eta,
\end{aligned}
\tag{40}
$$

where α_1, α_2, β_1, β_2, ξ_1, ξ_2 and σ are the vectors of nonnegative Lagrange multipliers.

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

$$
\begin{aligned}
\frac{\partial L}{\partial\lambda_1} &= K\{A_1^{'},C_1^{\top}\}^{\top}(K\{A_1^{'},C_1^{\top}\}\lambda_1+e_1b_1)+c_3K_1\lambda_1+c_4K_1L_1(K_1\lambda_1+eb_1) \\
&\quad +K\{A_1^{'},C_1^{\top}\}^{\top}\beta_1-K\{A_1^{'},C_1^{\top}\}^{\top}\beta_2+K\{B_1^{'},C_1^{\top}\}^{\top}\alpha_1=0, \\
\frac{\partial L}{\partial b_1} &= e_1^{\top}(K\{A_1^{'},C_1^{\top}\}\lambda_1+e_1b_1)+c_3b_1+c_4e^{\top}L_1(K_1\lambda_1+eb_1) \\
&\quad +e_1^{\top}\beta_1-e_1^{T}\beta_2+e_2^{\top}\alpha_1=0, \\
\frac{\partial L}{\partial\lambda_2} &= K\{A_2^{'},C_2^{\top}\}^{\top}(K\{A_2^{'},C_2^{\top}\}\lambda_2+e_1b_2)+c_3K_2\lambda_2+c_4K_2L_2(K_2\lambda_2+eb_2) \\
&\quad -K\{A_2^{'},C_2^{\top}\}^{\top}\beta_1+K\{A_2^{'},C_2^{\top}\}^{\top}\beta_2+K\{B_2^{'},C_2^{\top}\}^{\top}\alpha_2=0, \\
\frac{\partial L}{\partial b_2} &= e_1^{\top}\{K(A_2^{'},C_2^{\top}\}\lambda_2+e_1b_2)+c_3b_2+c_4e^{\top}L_2(K_2\lambda_2+eb_2) \\
&\quad -e_1^{\top}\beta_1+e_1^{\top}\beta_2+e_2^{\top}\alpha_2=0,
\end{aligned}
\tag{41}
$$

$$\begin{aligned}
&\frac{\partial L}{\partial q_1} = c_1 e_2 - \alpha_1 - \xi_1 = 0,\\
&\frac{\partial L}{\partial q_2} = c_2 e_2 - \alpha_2 - \xi_2 = 0,\\
&\frac{\partial L}{\partial \eta} = De_1 - \beta_1 - \beta_2 - \delta = 0.
\end{aligned}$$

Let

$$\begin{aligned}
&H_\phi = (K\{A_1^{'}, C_1^\top\}, e_1),\ G_\phi = (K\{B_1^{'}, C_1^\top\}, e_2),\ O_\phi = \left(\begin{smallmatrix} K_1 & 0 \\ 0 & 1 \end{smallmatrix}\right),\\
&J_\phi = (K_1,\, e),\ Q_\phi = (K\{A_2^{'}, C_2^\top\}, e_1),\ P_\phi = (K\{B_2^{'}, C_2^\top\}, e_2),\\
&U_\phi = \left(\begin{smallmatrix} K_2 & 0 \\ 0 & 1 \end{smallmatrix}\right),\ F_\phi = (K_2,\, e),\ \theta_1 = \begin{pmatrix} \lambda_1 \\ b_1 \end{pmatrix},\ \theta_2 = \begin{pmatrix} \lambda_2 \\ b_2 \end{pmatrix}.
\end{aligned} \tag{42}$$

From the above equations, we obtain

$$H_\phi^\top H_\phi \theta_1 + c_3 O_\phi \theta_1 + c_4 J_\phi^\top L_1 J_\phi \theta_1 + H_\phi^\top \beta_1 - H_\phi^\top \beta_2 + G_\phi^\top \alpha_1 = 0, \tag{43}$$

$$Q_\phi^\top Q_\phi \theta_2 + c_3 U_\phi \theta_2 + c_4 F_\phi^\top L_2 F_\phi \theta_2 - Q_\phi^\top \beta_1 + Q_\phi^\top \beta_2 + P_\phi^\top \alpha_1 = 0. \tag{44}$$

It follows that

$$\theta_1 = (H_\phi^\top H_\phi + c_3 O_\phi + c_4 J_\phi^\top L_1 J_\phi)^{-1}[H_\phi^\top (\beta_2 - \beta_1) - G_\phi^\top \alpha_1], \tag{45}$$

$$\theta_2 = (Q_\phi^\top Q_\phi + c_3 U_\phi + c_4 F_\phi^\top L_2 F_\phi)^{-1}[Q_\phi^\top (\beta_1 - \beta_2) - P_\phi^\top \alpha_2]. \tag{46}$$

We substitute (45), (46) into (40) and get

$$\begin{aligned}
L = {} & (\alpha_1 + \alpha_2)^\top e_2 - \frac{1}{2}[(\beta_2 - \beta_1)^\top H_\phi - \alpha_1^\top G_\phi](H_\phi^\top H_\phi + c_3 O_\phi + c_4 J_\phi^\top L_1 J_\phi)^{-1}\\
& [H_\phi^\top (\beta_2 - \beta_1) - G_\phi^\top \alpha_1] - \frac{1}{2}[(\beta_1 - \beta_2)^\top Q_\phi - \alpha_2^\top F_\phi](Q_\phi^\top Q_\phi + c_3 U_\phi\\
& + c_4 F_\phi^\top L_2 F_\phi)^{-1}[Q_\phi^\top (\beta_1 - \beta_2) - P_\phi^\top \alpha_2].
\end{aligned} \tag{47}$$

Therefore, the dual optimization formulation is

$$\begin{aligned}
\min_{\xi_1,\xi_2,\alpha_1,\alpha_2} \quad & \frac{1}{2}\xi_1^\top (H_\phi^\top H_\phi + c_3 O_\phi + c_4 J_\phi^\top L_1 J_\phi)^{-1}\xi_1 + \frac{1}{2}\xi_2^\top (Q_\phi^\top Q_\phi + c_3 U_\phi + c_4 F_\phi^\top L_2 F_\phi)^{-1}\xi_2\\
& - (\alpha_1 + \alpha_2)^\top e_2\\
\text{s.t.} \quad & \xi_1 = H_\phi^\top (\beta_2 - \beta_1) - G_\phi^\top \alpha_1,\\
& \xi_2 = Q_\phi^\top (\beta_1 - \beta_2) - P_\phi^\top \alpha_2,\\
& 0 \preceq \beta_1, \beta_2, \beta_1 + \beta_2 \preceq De_1,\\
& 0 \preceq \alpha_{1/2} \preceq c_{1/2} e_2.
\end{aligned} \tag{48}$$

Table 1 Datasets.

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

Correspondingly, the dual optimization formulation for (39) is

$$\begin{aligned}\min_{\rho_1,\rho_2,\omega_1,\omega_2}\quad & \frac{1}{2}\rho_1^\top(G_\phi^\top G_\phi + c_3O_\phi + c_4J_\phi^\top L_1J_\phi)^{-1}\rho_1 + \frac{1}{2}\rho_2^\top(P_\phi^\top P_\phi + c_3U_\phi + c_4F_\phi^\top L_2F_\phi)^{-1}\rho_2 \\ & -(\omega_1+\omega_2)^\top e_1 \\ \text{s.t.}\quad & \rho_1 = G_\phi^\top(\gamma_2-\gamma_1) - H_\phi^\top\omega_1, \\ & \rho_2 = P_\phi^\top(\gamma_1-\gamma_2) - Q_\phi^\top\omega_2, \\ & 0 \preceq \gamma_1,\gamma_2,\gamma_1+\gamma_2 \preceq He_2, \\ & 0 \preceq \omega_{1/2} \preceq c_{1/2}e_1,\end{aligned} \tag{49}$$

where the augmented vectors $\pi_1 = \begin{pmatrix}\lambda_3\\ b_3\end{pmatrix}$, $\pi_2 = \begin{pmatrix}\lambda_4\\ b_4\end{pmatrix}$ are given by

$$\pi_1 = (G_\phi^\top G_\phi + c_3O_\phi + c_4J_\phi^\top L_1J_\phi)^{-1}[G_\phi^\top(\gamma_2-\gamma_1) - H_\phi^\top\omega_1], \tag{50}$$

$$\pi_2 = (P_\phi^\top P_\phi + c_3U_\phi + c_4F_\phi^\top L_2F_\phi)^{-1}[P_\phi^\top(\gamma_1-\gamma_2) - Q_\phi^\top\omega_2]. \tag{51}$$

Suppose an example x has two views x_1 and x_2. If $\frac{1}{2}(|K\{x_1^\top, C_1^\top\}\lambda_1+b_1|+|K\{x_2^\top, C_2^\top\}\lambda_2+b_2|) \le \frac{1}{2}(|K\{x_1^\top, C_1^\top\}\lambda_3+b_3| + |K\{x_2^\top, C_2^\top\}\lambda_4+b_4|)$, it is classified to class $+1$, otherwise class -1.

4 Experimental results

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

4.1 Ionosphere

The ionosphere dataset [1] 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

[1] http://archive.ics.uci.edu/ml/datasets/Ionosphere

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

Method	$l = 70$, $u = 70$	$l = 70$, $u = 140$
single-view LapTSVM1	88.17±1.61	88.45±1.84
single-view LapTSVM2	88.17±2.75	88.73±2.64
SVM-2K	83.38±3.78	83.38±3.78
MvTSVMs	87.32±2.23	87.32±2.23
MvLapTSVMs	**89.01**±1.84	**89.29**±1.60

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 compare MvLapSVMs with single-view LapTSVMs (LapTSVM1 means using the LapTSVMs method to deal with one view data and LapTSVM2 means using the LapTSVMs method to deal with the other view data), SVM-2K and multi-view TSVMs (MvTSVMs)[2]. The result of experiment varies by use of different size of unlabeled data. We select regularization parameters from the range $[2^{-7}, 2^{7}]$ with exponential growth 0.5. The linear kernel is chosen for the dataset. We select 70 labeled and 70 unlabeled examples as the training set (i.e., $l = 70, u = 70$). The unlabeled examples are randomly selected from both classes. The size of the test data is 71. The result is in second column in Table 2. Then we select 70 labeled and 140 unlabeled examples as the training set (i.e., $l = 70, u = 140$). The unlabeled examples are randomly selected from both classes. The size of the test data is 71. The result is in third column. Each experiment is repeated five times. Experiment result is in Table 2.

4.2 Handwritten digits

The handwritten digits dataset [3] 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-Loève coefficients of each example image.

In this experiment, we compare MvLapSVMs with single-view LapTSVMs, SVM-2K and MvTSVMs. Because TSVMs are designed for binary classification while handwritten digits contains 10 classes, we use three pairs $(1, 7)$, $(2, 4)$ and $(3, 9)$ for binary classification. We select regularization parameters from the range $[2^{-7}, 2^{7}]$ with exponential growth 0.5. We select 160 labeled and 160 unlabeled examples as the training set (i.e., $l = 160, u = 160$). Half of the unlabeled data come from one class and the other half come from the other class. The size of the test data is 80. The Gaussian kernel is chosen for the dataset. Each experiment is repeated five times. Experiment result is in Table 3.

[2] We do not detail the MvTSVMs here. They are supervised extensions of TSVMs to multi-view learning.

[3] https://archive.ics.uci.edu/ml/datasets/Multiple+Features

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

Method	$2 \sim 4$	$1 \sim 7$	$3 \sim 9$
single-view LapTSVM1	88.75±2.17	91.25±2.50	98.00±1.43
single-view LapTSVM2	81.5±7.78	81.00±4.95	78.00±6.22
SVM-2K	94.00±1.63	93.20±2.26	94.50±3.14
MvTSVMs	94.00±2.71	95.00±0.88	96.00±1.05
MvLapTSVMs	**97.75**±1.63	**98.75**±0.88	**98.25**±1.43

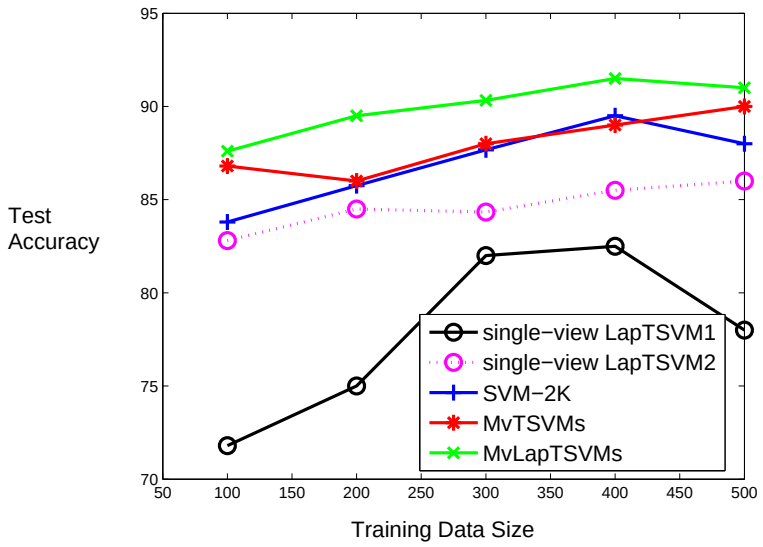

Fig. 1 Classification accuracies (%) of four methods on Advertisement.

4.3 Advertisement

The advertisement dataset [4] [35] consists of 3279 examples including 459 ads images (positive examples) and 2820 non-ads images (negative examples). One 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).

In this experiment, we randomly select 700 examples therein to form the used dataset. We select regularization parameters from the range $[2^{-7}, 2^{7}]$ with exponential growth 0.5. The Gaussian kernel is chosen for the dataset. We select $u = 100$ unlabeled data. The unlabeled examples are randomly selected from both classes. Each experiment is repeated five times. Experiment result is in Figure 1.

[4] http://archive.ics.uci.edu/ml/datasets/Internet+Advertisements

4.4 Analysis of the results

MvLapTSVMs can obtain good performance by combining two views in the constraints and are better than the corresponding single-view LapTSVMs. The second row, third row and sixth row in Table 2 show that MvLapTSVMs are superior to single-view LapTSVMs with the same labeled examples and different unlabeled examples. Similarly, the second row, third row and sixth row in Table 3 show that MvLapTSVMs are superior to single-view LapTSVMs in different digit pairs classification problems. From Figure 1 with varying training sizes, we can conclude that our method MvLapTSVMs are superior to single-view LapTSVMs. MvLapTSVMs can also exploit the usefulness of unlabeled examples to improve the classification accuracy comparable to supervised learning such as MvTSVMs and SVM-2K. The fourth row, fifth row and sixth row in Table 2 show that MvLapTSVMs are superior to MvTSVMs and SVM-2K with the same labeled examples and different unlabeled examples. Similarly, the fourth row, fifth row and sixth row in Table 3 show that MvLapTSVMs are superior to MvTSVMs and SVM-2K in different digit pairs classification problems. From Figure 1 with varying training sizes, MvLapTSVMs are superior to MvTSVMs and SVM-2K.

5 Conclusion

In this paper, we extended LapTSVMs to multi-view learning and proposed a new framework called MvLapTSVMs which combine two views by introducing the constraint of similarity between two one-dimensional projections identifying two distinct TSVMs from two feature spaces. MvLapTSVMs construct a decision function by solving two quadratic programming problems. We provide their dual formulation making use of Lagrange dual optimization techniques. MvLapTSVMs were further extended to their kernel version. Experimental results on real datasets indicate that the multi-view LapTSVMs are better than the corresponding single-view and supervised learning methods.

Acknowledgement

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. Shawe-Taylor J, Sun S (2011) A review of optimization methodologies in support vector machines. Neurocomputing 74(17):3609-3618
2. Vapnik V(1995) The nature of statistical learning theory. Springer-Verlag, New York
3. Christianini N, Shawe-Taylor J (2002) An introduction to support vector machines. Cambridge University Press, Cambridge
4. Scholkopf B, Smola A (2003) Learning with kernels. MIT Press, Cambridge
5. Fung G, Mangasarian O (2001) Proximal support vector machines. In Proceedings of the 7th international conference knowledge discovery and data mining, pp 77-86
6. Mangasarian O, Wild E (2006) Multisurface proximal support vector machine classification via generalized eigenvalues. IEEE Transactions on Pattern Analysis and Machine Intelligence 28(1):69-74

7. Jayadeva K, Khemchandani R, Chandra S (2007) Twin support vector machines for pattern classification. IEEE Transactions on Pattern Analysis and Machine Intelligence 29(5):905-910
8. Ghorai S, Mukherjee A, Dutta P (2009) Nonparallel plane proximal classifier. Signal Processing 89(4): 510-522
9. Shao Y, Chen W, Deng N (2013) Nonparallel hyperplane support vector machine for binary classification problems. Information Sciences. doi:10.1016/j.ins.2013.11.003
10. Shao Y, Wang Z, Chen W, Deng N (2013) Least squares twin parametric-margin support vector machines for classification. Applied Intelligence 39(3): 451-464
11. Xu Y, Guo R (2014) An improved ν-twin support vector machine. Applied Intelligence. doi:10.1007/s10489-013-0500-2
12. Chen W, Shao Y, Xu D, Fu Y (2013) Manifold proximal support vector machine for semi-supervised classification. Applied Intelligence. doi:10.1007/s10489-013-0491-z
13. Yang Z (2013) Nonparallel hyperplanes proximal classifiers based on manifold regularization for labeled and unlabeled examples. International Journal of Pattern Recognition and Artificial Intelligence 27(5):1-19
14. Shao Y, Deng N (2012) A coordinate descent margin based-twin support vector machine for classification. Neural Networks 25:114-121
15. Chapelle O, Schölkopf B, Zien A (2010) Semi-supervised learning. MIT Press, Massachusetts
16. Zhu X (2008) Semi-supervised learning literature survey. Technical Report 1530, Department of Computer Sciences University of Wisconsin Madison
17. Zhu X, Ghahramani Z, Lafferty J (2006) Semi-supervised learning using Gaussian fields and harmonic functions. In Proceedings of the 20th international conference on machine learning, pp 912-919
18. Zhou Z, Zhan D, Yang Q (2007) Semi-supervised learning with very few labeled training example. In Proceedings of the 22nd AAAI conference on artificial intelligence, pp 675-680
19. Joachims T (1999) Transductive inference for text classification using support vector machines. In Proceedings of the 16th international conference on machine learning, pp 200-209
20. Bennett K, Demiriz A (1999) Semi-supervised support vector machines. Advances in Neural Information Processing Systems 11:368-374
21. Fung G, Mangasarian O (2001) Semi-supervised support vector machines for unlabeled data classification. Optimization Method and Software 15:29-44
22. Belkin M, Niyogi P, Sindhwani V (2006) Manifold regularization: a geometric framework for learning from labeled and unlabeled examples. Journal of Machine Learning Research 7:2399-2434
23. Melacci S, Beklin M (2011) Laplacian support vector machines trained in the primal. Journal of Machine Learning Research 12:1149-1184
24. Qi Z, Tian Y, Shi Y (2012) Laplacian twin support vector machine for semi-supervised classification. Neural Networks 35:46-53
25. Shao Y, Zhang C, Wang X, Deng N (2011) Improvements on twin support vector machines. IEEE Transactions on Neural Networks 22(6):962-968
26. Ding S, Zhao Y, Qi B, Huang H (2012) An overview on twin support vector machines. Artificial Intelligence Review. doi:10.1007/s10462-012-9336-0
27. Sun S (2013) A survey of multi-view machine learning. Neural Computing and Applications 23:2031-2038
28. Blum A, Mitchell T (1998) Combining labeled and unlabeled data with co-training. In Proceedings of the 11 th annual conference on computational learning theory, pp 92-100
29. Sindhwani V, Rosenberg D (2008) An RKHS for multi-view learning and manifold co-regularization. In Proceedings of the 25th international conference on machine learning, pp 976-983
30. Sindhwani V, Niyogi P, Belkin M (2005) A co-regularization approach to semi-supervised learning with multiple views. In Proceedings of the workshop on learning with multiple views, 22nd ICML, pp 824-831
31. Farquhar J, Hardoon D, Shawe-Taylor J, Szedmak S (2006) Two view learning: SVM-2K, theory and practice. Advances in Neural Information Processing Systems 18:355-362
32. Sun S, Shawe-Taylor J (2010) Sparse semi-supervised learning using conjugate functions. Journal of Machine Learning Research 11:2423-2455
33. Sun S (2011) Multi-view Laplacian support vector machines. Lecture Notes in Computer Science 7121:209-222
34. Bartlett P, Mendelson S (2002) Rademacher and Gaussian complexities: risk bounds and structural results. Journal of Machine Learning Research 3:463-482
35. Kushmerick N (1999) Learning to remove internet advertisement. In Proceedings of third annual conference on Autonomous Agents, pp 175-181