This article appeared in a journal published by Elsevier. The attached copy is furnished to the author for internal non-commercial research and education use, including for instruction at the authors institution and sharing with colleagues.

Other uses, including reproduction and distribution, or selling or licensing copies, or posting to personal, institutional or third party websites are prohibited.

In most cases authors are permitted to post their version of the article (e.g. in Word or Tex form) to their personal website or institutional repository. Authors requiring further information regarding Elsevier's archiving and manuscript policies are encouraged to visit:

http://www.elsevier.com/copyright
Laplacian twin support vector machine for semi-supervised classification

Zhiquan Qi\textsuperscript{a}, Yingjie Tian\textsuperscript{a,\ast}, Yong Shi\textsuperscript{a,b,\ast\ast}

\textsuperscript{a} Research Center on Fictitious Economy & Data Science, Chinese Academy of Sciences, Beijing 100190, China
\textsuperscript{b} College of Information Science & Technology, University of Nebraska at Omaha, Omaha, NE 68182, USA

A R T I C L E I N F O

Article history:
Received 27 February 2012
Received in revised form 24 May 2012
Accepted 30 July 2012

Keywords:
Semi-supervised classification
Laplacian
Twin support vector machine
Multi-class classification

A B S T R A C T

Semi-supervised learning has attracted a great deal of attention in machine learning and data mining. In this paper, we have proposed a novel Laplacian Twin Support Vector Machine (called Lap-TSVM) for the semi-supervised classification problem, which can exploit the geometry information of the marginal distribution embedded in unlabeled data to construct a more reasonable classifier and be a useful extension of TSVM. Furthermore, by choosing appropriate parameters, Lap-TSVM degenerates to either TSVM or TBSVM. All experiments on synthetic and real data sets show that the Lap-TSVM's classifier combined by two nonparallel hyperplanes is superior to Lap-SVM and TSVM in both classification accuracy and computation time.

1. Introduction

Semi-Supervised Learning (SSL) has attracted an increasing amount of interest in the last decade (Chapelle, Schölkopf, & Zien, 2006; Seeger, 2001; Zhu, 2006). One main reason is that the labeled examples are always rare but there are large amounts of unlabeled examples available in many practical problems. Several novel approaches for making use of the unlabeled data to improve the performance of classifiers have been proposed. Graph based methods are very important branches, where nodes in the graph are the labeled and unlabeled points, and weighted edges reflect the similarities of nodes. The initial assumption of these methods is that all points are located in a low dimensional manifold, and the graph is used for an approximation of the underlying manifold. Neighboring point pairs connected by large weight edges tend to have the same labels and vice versa. By these means, the labels associated with data can be propagated throughout the graph. By using the graph Laplacian, Belkin, Niyogi, and Sindhwani (2006) proposed a novel Laplacian Support Vector Machine (LapSVM). Unlike other methods based on graphs (Belkin & Niyogi, 2002; Joachims, 2003; Zhu, Ghahramani, & Lafferty, 2003), LapSVM is a natural out-of-sample extension, which can classify data that become available after the training process, without having to retrain the classifier or resort to various heuristics (Belkin et al., 2006). A lot of experiments show that LapSVM achieves state of the art performance in semi-supervised classification.

Recently, the research of nonparallel plane classifiers has been a new hot spot. Mangasarian and Wild (2006) first proposed a nonparallel plane classifier, which attempts to generate two nonparallel planes such that each plane is closer to one of two classes and is at least one distance from the other. Motivated by GEPSVM, Jayadeva, Khemchandani, and Chandra (2007) proposed a well known twin support vector machine (TSVM) classifier for binary classification. Experimental results (Jayadeva et al., 2007; Mangasarian & Wild, 2006) show that nonparallel plane classifiers can indeed improve the performance of traditional SVM. Other extensions to TSVM can also be found in Ghorai, Mukherjee, and Dutta (2009), Khemchandani, Jayadeva, and Chandra (2009), Kumar and Gopal (2008, 2009) and Shao, Zhang, Wang, and Deng (2011).

In this paper, inspired by the success of Jayadeva et al. (2007), we propose a novel Laplacian Twin Support Vector Machine (called Lap-TSVM) for the SSL problem (as far as we know, Lap-TSVM is the first twin support vector machine applied in the SSL problem), which can exploit the geometry information of the marginal distribution embedded in unlabeled data to construct a more reasonable classifier and be a useful extension of TSVM. Furthermore, by choosing appropriate parameters, Lap-TSVM degenerates to either TSVM or TBSVM.

The remaining parts of the paper are organized as follows. Section 2 briefly introduces the background of SVM and TSVM; Section 3 describes the details of Lap-TSVM; all public datasets experimental results on synthetic and real datasets are shown in Section 4; in the last section we give the conclusions.
2. Background

2.1. Support vector classification (SVC)

SVMs (Vapnik, 1995, 1996) have been introduced in the framework of structural risk minimization (SRM) and in the theory of VC bounds. For classification about the training data

\[ T = \{(x_1, y_1), \ldots, (x_l, y_l)\} \subseteq (\mathbb{R}^n \times \mathbb{Y})^l, \]

where \( x_i \in \mathbb{R}^n, y_l \in \mathbb{Y} = \{-1, 1\}, i = 1, \ldots, l \). Linear SVM is to solve the following primal QPP

\[
\begin{align*}
\min_{w, b, \xi} & \quad \frac{1}{2} \|w\|^2 + C \sum_i \xi_i, \\
\text{s.t.} & \quad y_i(w^\top x_i + b) \geq 1 - \xi_i, \quad \xi_i \geq 0, \quad i = 1, 2, \ldots, l,
\end{align*}
\]

where \( C \) is a penalty parameter and \( \xi_i \) are slack variables. The goal is to find an optimal separating hyperplane

\[ w^\top x + b = 0, \]

where \( x \in \mathbb{R}^n \). The Wolfe Dual of (2) can be expressed as

\[
\begin{align*}
\min_{\alpha} & \quad \frac{1}{2} \sum_{i=1}^l \sum_{j=1}^l y_i y_j \alpha_i \alpha_j - \sum_i \alpha_i, \\
\text{s.t.} & \quad \sum_i y_i \alpha_i = 0, \\
& \quad 0 \leq \alpha_i \leq C, \quad i = 1, 2, \ldots, l,
\end{align*}
\]

where \( \alpha \in \mathbb{R}^l \) are Lagrangian multipliers. The optimal separating hyperplane of (3) can be given by

\[
w = \sum_{i=1}^l \alpha_i^* y_i x_i, \quad b = \frac{1}{N_w} \left( y - \sum_{i=1}^l \alpha_i^* y_i (x_i \cdot x_i) \right),
\]

where \( \alpha^* \) is the solution of the dual problem (4). \( N_w \) represents the number of support vectors satisfying \( 0 < \alpha < C \). A new sample is classified as \(+1\) or \(-1\) according to the final decision function.

2.2. Twin support vector machine (TSVM)

Consider a binary classification problem of \( m_1 \) positive class and \( m_2 \) negative class. Suppose that data points belonging to the positive class are denoted by \( A \in \mathbb{R}^{m_1 \times n} \), where each row \( A_i \in \mathbb{R}^n \) represents a data point. Similarly, \( B \in \mathbb{R}^{m_2 \times n} \) represents all of the data points belonging to the negative class. For the linear case, the TSVM (Jayadeva et al., 2007) determines two nonparallel hyperplanes:

\[
f_+(x) = w^\top_+ x + b_+ = 0 \quad \text{and} \quad f_-(x) = w^\top_- x + b_- = 0,
\]

where \( w_+, w_- \in \mathbb{R}^n, b_+, b_- \in \mathbb{R} \). Here, each hyperplane is closer to one of the two classes and at least one distance from the other. A new data point is assigned to the positive class or negative class depending upon its proximity to the two nonparallel hyperplanes. Formally, for finding the positive and negative hyperplanes, the TSVM solves the following two Quadratic Programming Problems (QPPs):

\[
\begin{align*}
\min_{w_+, b_+, \xi} & \quad \frac{1}{2} \|Aw_+ + e_+ b_+\|^2 + c_1 e^\top_+ \xi, \\
\text{s.t.} & \quad -(Bw_- + e_- b_-) + \xi \geq e_-, \quad \xi \geq 0,
\end{align*}
\]

and

\[
\begin{align*}
\min_{w_-, b_-, \xi} & \quad \frac{1}{2} \|Bw_- + e_- b_-\|^2 + c_2 e^\top_- \eta, \\
\text{s.t.} & \quad (Aw_- + e_+ b_+) + \eta \geq e_+, \quad \eta \geq 0,
\end{align*}
\]

where \( c_1, c_2 \geq 0 \) are the pre-specified penalty factors, and \( e_+, e_- \) are vectors of proper dimensions. By introducing the Lagrangian multipliers, the Wolfe dual of QPPs (7) and (8) can be represented respectively as follows:

\[
\begin{align*}
\max_{\alpha} & \quad e^\top_+ \alpha - \frac{1}{2} \alpha^\top G(H^\top H)^{-1} G^\top \alpha, \\
\text{s.t.} & \quad 0 \leq \alpha \leq c_1 e_-, \\
\end{align*}
\]

and

\[
\begin{align*}
\max_{\beta} & \quad e^\top_- \beta - \frac{1}{2} \beta^\top P(Q^\top Q)^{-1} P^\top \beta, \\
\text{s.t.} & \quad 0 \leq \beta \leq c_2 e_+,
\end{align*}
\]

where \( G = [B e_-], H = [A e_+], P = [A e_+] \) and \( Q = [B e_-] \). \( \alpha \in \mathbb{R}^{m_1}, \beta \in \mathbb{R}^{m_2} \) are Lagrangian multipliers.

The non-parallel hyperplanes (6) can be obtained from the solutions \( \alpha \) and \( \beta \) of (9) and (10) by

\[
\begin{align*}
& v_1 = -(H^\top H)^{-1} G^\top \alpha, \quad \text{where} \quad v_1 = [w^\top_+ b_+]^\top, \quad v_2 = -(Q^\top Q)^{-1} P^\top \beta, \quad \text{where} \quad v_1 = [w^\top_- b_-]^\top. \\
& \text{A new data point} \ x \in \mathbb{R}^n \ \text{is then assigned to the positive or negative class, depending on which of the two hyperplanes (6) it lies closest to, i.e.} \\
& f(x) = \arg\min_{\pm} d_{\pm}(x),
\end{align*}
\]

where

\[
d_{\pm}(x) = |w^\top_{\pm} x + b_{\pm}|,
\]

and \( |\cdot| \) is the perpendicular distance of point \( x \) from the planes \( w_{\pm}^\top x + b_{\pm} \).

For the nonlinear case, we can refer to Jayadeva et al. (2007).

3. Laplacian twin support vector machine for semi-supervised classification (called Lap-TSVM)

3.1. Semi-supervised learning framework

Regularization (Tikhonov, 1963) is a key technology for obtaining smooth decision functions and thus avoiding overfitting to the training data, which is widely used in machine learning (Belkin et al., 2006; Cucker & Zhou, 2007; Evgeniou, Pontil, & Poggio, 2000; Gnecco & Sanguineti, 2010). Recently, the regularization framework has been recently extended in the SSL field as follows (Belkin et al., 2006; Melacci & Belkin, 2011).

Given a set of labeled data (1) and a set of unlabeled data \( (x_{1i}, \ldots, x_{iu}) \)

\[
(14)
\]

where \( x_{1i} \in \mathbb{R}^n, i = 1, \ldots, u \). Suppose the labeled data are generated according to the distribution \( P \) on \( \mathbb{X} \times \mathbb{Y} \), whereas unlabeled examples are drawn according to the marginal distribution \( P_X \) of \( P \). Labels of samples can be obtained from the conditional probability distribution \( P(y|x) \). The manifold regularization approach exploits the geometry information of the marginal distribution \( P_X \). An important premise of this kind of approach is to assume that the probability distribution of data has the geometric structure of a Riemannian manifold \( M \). The labels of two points that are close in the intrinsic geometry of \( M \) should
be the same or similar. Belkin et al. (2006) applied the intrinsic regularizer \( \|f\|_M^2 \) to describe the constraint above,
\[
\|f\|_M^2 = \sum_{i=1}^{l+u} \sum_{i=1}^{l+u} (f(x_i) - f(x_j))^2 = f^\top L f, \tag{15}
\]
where \( L \) is the graph Laplacian. In practice, choosing exponential weights for the adjacency matrix leads to convergence of the graph Laplacian to the Laplace–Beltrami operator on the manifold (Melacci & Belkin, 2011). For a kernel function \( K(\cdot, \cdot) \), which is associated with a reproducing kernel Hilbert space \( \mathcal{H} \), the decision function can be obtained by minimizing
\[
f^* = \arg\min_{f \in \mathcal{H}} \frac{1}{2} \sum_{i=1}^{l+u} V(x_i, y_i, f) + \gamma_M \|f\|_M^2 + \gamma_W \|f\|_W^2, \tag{16}
\]
where \( f \) is an unknown decision function, \( V \) represents some loss function on the labeled data, and \( \gamma_M \) is the weight of \( \|f\|_M^2 \) and controls the complexity of \( f \) in the Reproducing Kernel Hilbert Space. \( \gamma_W \) is the weight of \( \|f\|_W^2 \) and controls the complexity of the function in the intrinsic geometry of marginal distribution, and \( \|f\|_M^2 \) is able to penalize \( f \) along the Riemann manifold \( \mathcal{M} \). More detailed discussion can be found in Belkin et al. (2006).

### 3.2. Linear Lap-TSVM

Similar to the TSVM, we use the square loss function and hinge loss function for Lap-TSVM. \( V_\gamma(x_i, y_i, f) \) can be expressed as
\[
V_\gamma(x_i, y_i, f) = \max(0, 1 - f(x_i) y_i), \tag{17}
\]
which is an appropriate ones vector. When \( (24) \) or \( (25) \) is used as a penalty item of Eq. (16), we can understand them by means if the neighbor of \( x_i \), \( x_j \) has the higher similarity (\( W_y \) is larger), the difference of \( f(x_i), f(x_j) \) will obtain a big punishment. More intuitively, the smaller \( \|f(x_i) - f(x_j)\| \) is, the more smooth \( f(x) \) in the data adjacency graph is.

Substituting \( (17), (18), (21), (22), (24) \) and \( (25) \) into \( (16) \), the primal problems of Linear Lap-TSVM can be written as
\[
\min_{w_+, b_+, \xi \geq 0} \frac{1}{2} \|Aw_+ + e_+ b_+\|_2^2 + c_1 \xi + c_2 (\|w_+\|_2^2 + b_+), \tag{26}
\]
and
\[
\min_{w_-, b_-, \eta \geq 0} \frac{1}{2} \|Bw_- + e_- b_-\|_2^2 + c_1 \eta + c_2 (\|w_-\|_2^2 + b_-), \tag{27}
\]
\[
\text{s.t. } \begin{align*}
& (Aw_+ + e_+ b_+) \cdot e_+ - \beta^\top \mathbf{e} = 0, \\
& (Bw_- + e_- b_-) \cdot e_- - \beta^\top \mathbf{e} = 0,
\end{align*}
\]
\[
L(\Theta) = \frac{1}{2} \|Aw_+ + e_+ b_+\|_2^2 \Theta \text{ and } \Theta = (\alpha_1, \ldots, \alpha_m, \beta_1, \ldots, \beta_m)^\top \text{ are the Lagrange multipliers. The dual problem can be formulated as}
\]
\[
\max L(\Theta) \tag{28}
\]
\[
\text{s.t. } \nabla_{w_+, b_+} L(\Theta) = 0, \quad \alpha, \beta \geq 0. \tag{29}
\]
From Eq. (29), we get
\[
\nabla_{w_+} L = A^\top (Aw_+ + e_+ b_+) + c_2 w_+ + c_3 M^\top L(Mw_+ + e_+) + B^\top \alpha = 0, \tag{30}
\]
\[
\nabla_{b_+} L = c_2^\top (Aw_+ + e_+ b_+) + c_3 e_+^\top L(Mw_+ + e_+) + e_+^\top \alpha = 0, \tag{31}
\]
\[
\nabla_{\alpha} L = c_2 w_+ - \alpha = 0, \tag{32}
\]
\[
\text{Since } \beta \geq 0, (32) \text{ turns out to be }
\]
\[
0 \leq \alpha \leq c_2 e_+. \tag{33}
\]
Next, combining \( (30) \) and \( (31) \) leads to
\[
\left[ \begin{array}{c}
A^\top \\
\mathbf{e}_+^\top
\end{array} \right] [A e_+] [w_+] + c_2 [w_+] + c_3 M^\top L[M e_+] [w_+] + [\mathbf{e}_+]^\top \alpha = 0.
\]
Let
\[
H = [A e_+], \quad J = [M e_+], \quad G = [B e_-]. \tag{35}
\]
and the augmented vector \( \phi_+ = [w_+^\top \mathbf{b}_+^\top]^\top \). Eq. (34) can be rewritten as
\[
(H^\top H + c_2 J + c_2 J^\top J) \phi_+ + G^\top \alpha = 0,
\]
\[
i.e., \quad \phi_+ = -(H^\top H + c_2 J + c_2 J^\top J)^{-1}(G^\top \alpha). \tag{36}
\]
where $I$ is an identity matrix of appropriate dimensions. According to matrix theory (Gantmacher, 1990), it can be easily proved that $H^T H + c_2 I + c_3 J^T L J$ is a positive definite matrix.

Substituting the above equations into problem (29), we obtain the Wolfe dual of the problem (29) as follows:

$$\max_{\alpha} \ e^T \alpha - \frac{1}{2} \alpha^T G (H^T H + c_2 I + c_3 J^T L J)^{-1} G^T \alpha$$
$$\text{s.t. } 0 \leq \alpha \leq c e.$$  \hspace{1cm} (37)

Similarly, the dual of (27) is

$$\max_{\beta} \ e^T \beta - \frac{1}{2} \beta^T P \left( Q^T Q + c_2 I + c_3 F^T F \right)^{-1} P^T \beta$$
$$\text{s.t. } 0 \leq \beta \leq c_2 e_+,$$  \hspace{1cm} (38)

where

$$Q = [A e_+], \quad F = [M e], \quad P = [B e_+]$$
and the augmented vector $\vartheta_- = [w_- b_-]^T$ is given by

$$\vartheta_- = -(Q^T \varphi + c_2 I + c_3 F^T F)^{-1} P^T \beta,$$  \hspace{1cm} (40)

where $Q^T \varphi + c_2 I + c_3 F^T F$ is a positive definite matrix. Once vectors $\vartheta_+$ and $\vartheta_-$ are obtained from (36) and (40), the separating hyperplanes

$$w_-^T x + b_- = 0, \quad w_-^T x + b_+ = 0$$  \hspace{1cm} (41)
are known. A new data point $x \in \mathbb{R}^p$ is then assigned to the positive or negative class, depending on which of the two hyperplanes it lies closest to, i.e.,

$$f(x) = \arg \min_{\vartheta_+} d_+(x),$$  \hspace{1cm} (42)

where

$$d_+(x) = |w_-^T x + b_+|,$$  \hspace{1cm} (43)

is the perpendicular distance of point $x$ from the planes $w_-^T x + b_+.$

### 3.3 Nonlinear Lap-TSVM

Now we extend the linear Lap-TSVM to the nonlinear case.

The same as in the linear case, the cost function of the errors $V_+(x_i, y_i, f_i)$ and $V_-(x_i, y_i, f_i)$ can be expressed as (17) and (18). The decision function can be written as $f_\lambda(x) = (w_- \cdot \Phi(x)) + b_+,$ where $\Phi(\cdot)$ is a nonlinear mapping from a low dimensional space to a higher dimensional Hilbert space $H.$ According to Hilbert space theory (Schölkopf & Smola, 2002), $w_- \cdot \Phi(x)$ can be expressed as $w_- = \sum_{i=1}^{N} \lambda_i \Phi(x_i) = \Phi(M) \lambda_M.$ So the following kernel-generated hyperplanes are:

$$K(x^T, M^T) \lambda_+ + b_+ = 0,$$
$$K(x^T, M^T) \lambda_- + b_- = 0,$$  \hspace{1cm} (44)

where $K$ is a chosen kernel function: $K(x_i \cdot x_j) = \langle \Phi(x_i) \cdot \Phi(x_j) \rangle.$ By means of the kernel matrix $K$ and relevant coefficients $\lambda_M,$ the regularized term $\|f_\lambda\|^2_M$ and $\|f_\lambda\|^2_{M^T}$ can be expressed as

$$\|f_\lambda\|^2_M = \frac{1}{2} \lambda^T K \lambda_M + b^T_+,$$
$$\|f_\lambda\|^2_{M^T} = \frac{1}{2} (\lambda^T K \lambda_- + b^T_-).$$  \hspace{1cm} (45, 46)

For manifold regularization, on the basis of $f_\lambda = \{f_\lambda(x_1), \ldots, f_\lambda(x_N)\}$ $K \lambda_M + d \in \mathbb{R}^N$ and $\|f_\lambda\|^2_M$ and $\|f_\lambda\|^2_{M^T}$ can be written as

$$\|f_\lambda\|^2_M = f^T_M L f_M = (\lambda^T K + e^T b_-) L (K \lambda_+ + e b_+),$$
$$\|f_\lambda\|^2_{M^T} = f^T_{M^T} L f_{M^T} = (\lambda^T K + e^T b_-) L (K \lambda_- + e b_+).$$  \hspace{1cm} (47, 48)

So the nonlinear optimization problems can be expressed as

$$\min_{\lambda_+ \in \mathbb{R}^M} \frac{1}{2} \|K(A, M^T) \lambda_+ + e_+ b_+\|^2 + c_1 e_+^T \xi +$$
$$\frac{1}{2} c_2 (\lambda^T_+ K \lambda_+ + b^T_+ b_+)^2 + c_1 (\lambda^T_+ K + e^T b_-) L (K \lambda_+ + e b_+),$$  \hspace{1cm} (49)

$$\text{s.t. } - (K(B, M^T) \lambda_+ + e_+ b_+) + \xi \geq e_-, \quad \xi \geq 0,$$  \hspace{1cm} (49)

and

$$\min_{\lambda_- \in \mathbb{R}^M} \frac{1}{2} \|K(B, M^T) \lambda_- + e_- b_-\|^2 + c_1 e_-^T \eta +$$
$$\frac{1}{2} c_2 (\lambda^T_- K \lambda_- + b^T_- b_-)^2 + c_1 (\lambda^T_- K + e^T b_-) L (K \lambda_- + e b_-),$$  \hspace{1cm} (50)

$$\text{s.t. } (K(A, M^T) \lambda_- + e_+ b_-) + \eta \geq e_+, \quad \eta \geq 0.$$  \hspace{1cm} (50)

Define the Lagrangian corresponding to the problem (49) as follows

$$L(\theta) = \frac{1}{2} \|K(A, M^T) \lambda_+ + e_+ b_+\|^2$$
$$+ c_1 e_+^T \xi + \frac{1}{2} c_2 (\lambda^T_+ K \lambda_+ + b^T_+ b_+)$$
$$+ c_1 (\lambda^T_+ K + e^T b_-) L (K \lambda_+ + e b_+),$$
$$\quad - \alpha^T (-(K(B, M^T) \lambda_+ + e_+ b_+) + \xi - e_-) - \beta^T \xi,$$  \hspace{1cm} (51)

where $\theta = (\lambda_+, b_+, \xi, \alpha, \beta).$

The dual problem can be formulated as

$$\max_{\alpha, \beta} \ L(\theta)$$
$$\text{s.t. } \nabla_{\lambda_+} L = K(A, M^T) K(A, M^T) \lambda_+ + e_+ b_+ + c_2 K \lambda_+ + c_4 K \lambda_+ + K(B, M^T) \alpha = 0,$$
$$\nabla_{b_+} L = e_+^T (K(A, M^T) \lambda_+ + e_+ b_+ + c_2 b_+) + c_4 e_+^T L (K \lambda_+ + e b_+),$$
$$\nabla_{\lambda_-} L = c_1 d_+ - \alpha - \beta = 0.$$

Combining (53) and (54) leads to

$$\begin{bmatrix}
K(A, M^T) \\
e_+
\end{bmatrix}
\begin{bmatrix}
\lambda^T_+ \\
b^T_+
\end{bmatrix} =$$
$$+ c_2^T \begin{bmatrix}
K \\
0
\end{bmatrix} \\
\lambda^T_+
\begin{bmatrix}
1 \\
b^T_+
\end{bmatrix}$$
$$+ c_4^T \begin{bmatrix}
K \\
\alpha
\end{bmatrix} \begin{bmatrix}
\lambda^T_+
\\
e^T
\end{bmatrix} + (K(B, M^T))^T \alpha = 0.$$  \hspace{1cm} (56)

Let

$$H_\Phi = [K(A, M^T) e_+] , \quad Q_\Phi = \begin{bmatrix}
K \\
0
\end{bmatrix} \begin{bmatrix}
0 \\
1
\end{bmatrix}$$

and the augmented vector $\rho_+ = (\lambda_+ + b_+)^T.$ Eq. (56) can be rewritten as:

$$H_\Phi^T H_\Phi \rho_+ + c_2 Q_\Phi \rho_+ + c_4 d_\Phi^T L_\Phi \rho_+ + G_\Phi^T \alpha = 0,$$

\hspace{1cm} i.e., $\rho_+ = -(H_\Phi^T H_\Phi + c_2 Q_\Phi + c_4 d_\Phi^T L_\Phi)^{-1}(G_\Phi^T \alpha).$  \hspace{1cm} (58)
Table 1
The percentage of tenfold testing accuracy of Lap-TSVM for two moons and two lines synthetic datasets.

<table>
<thead>
<tr>
<th>Method</th>
<th>Labeled data size (300, 600, 1200, 1800, 2600)</th>
</tr>
</thead>
<tbody>
<tr>
<td>TSVM</td>
<td>91.68 ± 2.25* 92.48 ± 1.88* 93.77 ± 1.45* 96.56 ± 1.33* 97.24 ± 1.41*</td>
</tr>
<tr>
<td>Lap-SVM</td>
<td>94.37 ± 2.14* 95.41 ± 2.32* 96.56 ± 1.12* 97.06 ± 0.89* 97.61 ± 0.98*</td>
</tr>
<tr>
<td>Lap-TSVM</td>
<td>95.53 ± 1.98 96.34 ± 2.54 97.23 ± 2.33 98.65 ± 0.76 98.81 ± 0.66</td>
</tr>
</tbody>
</table>

Table 2
The testing accuracy of ‘5’ vs. ‘8’ dataset. The size of the unlabeled data is 420.

<table>
<thead>
<tr>
<th>Method</th>
<th>Number of unlabeled data (400, 600, 1200, 1500, 2400, 6000)</th>
</tr>
</thead>
<tbody>
<tr>
<td>TSVM</td>
<td>92.34 ± 2.54* 93.56 ± 1.63* 94.95 ± 1.11* 96.24 ± 0.87* 96.45 ± 0.56*</td>
</tr>
<tr>
<td>Lap-SVM</td>
<td>93.57 ± 2.65* 95.35 ± 1.78* 96.45 ± 1.31* 97.55 ± 0.76* 97.86 ± 0.45*</td>
</tr>
<tr>
<td>Lap-TSVM</td>
<td>94.26 ± 1.93 96.12 ± 1.52 97.24 ± 0.98 98.49 ± 0.68 99.45 ± 0.41</td>
</tr>
</tbody>
</table>

Table 3
The testing accuracy of ‘5’ vs. ‘8’ dataset for the differing unlabeled data.

<table>
<thead>
<tr>
<th>Method</th>
<th>Number of unlabeled data (400, 600, 1500, 2400, 6000)</th>
</tr>
</thead>
<tbody>
<tr>
<td>TSVM</td>
<td>92.34 ± 2.54* 93.56 ± 1.63* 94.95 ± 1.11* 96.24 ± 0.87* 96.45 ± 0.56*</td>
</tr>
<tr>
<td>Lap-SVM</td>
<td>93.57 ± 2.65* 95.35 ± 1.78* 96.45 ± 1.31* 97.55 ± 0.76* 97.86 ± 0.45*</td>
</tr>
<tr>
<td>Lap-TSVM</td>
<td>94.26 ± 1.93 96.12 ± 1.52 97.24 ± 0.98 98.49 ± 0.68 99.45 ± 0.41</td>
</tr>
</tbody>
</table>

So the Wolfe dual of the problem (49) is formulated as follows:

\[
\begin{align*}
\max_{\beta} & \quad \frac{1}{2} (\beta^T P_\phi)(Q_\phi Q_\phi + c_2 U_\phi + c_3 F_\phi^T L_\phi F_\phi)^{-1}(P_\phi \beta) \\
\text{s.t.} & \quad \beta \leq 0 \leq \beta c e_+,
\end{align*}
\]

(60)

where

\[
\begin{align*}
Q_\phi &= [K(A, M^T) e_-.] \\
U_\phi &= \begin{bmatrix} \lambda & 0 \\ 0 & 1 \end{bmatrix} \\
P_\phi &= [K(B, M^T) e_+],
\end{align*}
\]

(61)

and the augmented vector \(\rho_- = [\lambda - b_2]^T\), which is given by

\[
\rho_- = -(Q_\phi^T Q_\phi + c_2 U_\phi + c_3 F_\phi^T L_\phi F_\phi)^{-1}(P_\phi^T \alpha).
\]

(62)

Once vectors \(\rho_+\) and \(\rho_-\) are obtained from (58) and (62), a new data point \(x \in \mathbb{R}^n\) is then assigned to the positive or negative class, depending on a manner similar to the linear case.

3.4. Related work and connections to other twin SVMs

As we know, one significant advantage of traditional SVMs (Vapnik, 1995) is the implementation of the structural risk minimization (SRM) principle. However only the empirical risk is considered in the primal problems of TSVM. In order to overcome the shortcoming, Shao et al. (2011) improved TSVM (called TBSVM) by introducing two regularization terms \(\|w_m\|^2 + b_+\) and \(\|w_m\|^2 + b_+\) to make the SRM principle implemented. Consider Lap-TSVM; dropping terms: \(\|w_m\|^2\), our models will degenerate to be the TSVM. Only dropping manifold regularization: \(\|w_m\|^2\), and supposing \(K\) of (45) and (46) is the identity matrix, our models will degenerate to be the TBSVM. Therefore, TSVM and TBSVM are the special cases of our models.

4. Experiment

In this section, we illustrate the effectiveness of the proposed method on synthetic and real datasets. To demonstrate the capabilities of our algorithm, we compare Lap-TSVM with Lap-SVM (Belkin et al., 2006) and Transductive SVM(TSVM) (Vapnik, 1995).
Fig. 2. Samples from ABCDETC dataset.

Our algorithm code is written in MATLAB 2010 on a PC with an Intel Core i5 processor with 2 GB RAM. The testing accuracies of all experiments are computed using standard 10-fold cross validation (Deng & Tian, 2009). $c_1$, $c_2$, $c_3$ and RBF kernel parameter $\sigma$ are all selected from the set $\{2^i | i = -5, \ldots, 5\}$ by 10-fold cross validation on the tuning set comprising of a random 10% of the training data. Once the parameters are selected, the tuning set was returned to the training set to learn the final decision function. Each experiment is repeated 10 times.

4.1. Synthetic data

Similar to Sindhwani, Niyogi, and Belkin (2005), we also use two moons and two lines synthetic data containing 200 examples with only 1 labeled example for each class, which are shown in Fig. 1. We select a linear kernel for the two lines data and a RBF kernel for the two moons data. In order to reflect the performance of the manifold regularization: $\|f\|^2_M$, we fix $c_1 = 4$, $c_2 = 0.024$ and select different $c_3$ to train and test the two synthetic datasets. It is easy to see from Table 1 that, by using the geometric distribution information of labeled data and unlabeled data, the manifold regularization can indeed help the algorithm to seek a more reasonable classifier. With the increase of $c_3$, the accuracy of Lap-TSVM in the two toy data is constantly being improved.

4.2. Handwritten symbol recognition

4.2.1. MNIST Dataset

MNIST Dataset is a handwritten digit dataset with samples from ‘0’ to ‘9’. The size of each sample is $28 \times 28$ pixels. We use digits ‘5’ and ‘8’ to form a binary classification problem. For experiment 1, sizes of the labeled data are separately 300, 600, 1200, 1800 and 2600 (‘5’ and ‘8’ have the same number of samples). Another 420 ‘5’ and ‘8’ are selected as unlabeled data. The test data is 1500. For experiment 2, we selected 500 ‘5’ and ‘8’ as the training set, and another 400, 600, 1500, 2400, and 4000 ‘5’ and ‘8’ as unlabeled data. Test data is 1200. Tables 2 and 3 are the results of experiments 1 and 2 in the case of the RBF kernel.

Fig. 3. The test accuracy and standard deviations of TSVM, Lap-SVM, and Lap-TSVM on ABCDETC dataset for the case of RBF kernel.
In the section, we perform the methods on the UCI datasets (Asuncion & Newman, 2007). All data were scaled such that the features locate in [0, 1] before training. For each dataset, we randomly select the same number of data from different classes to compose a dataset, 35% of each extracted dataset are for training, 30% as unlabeled data, and the others for testing. The results are shown in Table 4.

4.4. Discussion

In order to find out whether Lap-TSVM has better performance than the other algorithms, we perform the t-test on the...
classification results (each experiment is repeated 10 times) to compute the Significance Level (SL) between Lap-TSVM and the other algorithms. The null hypothesis H0 demonstrates that there is no significant difference between the two algorithms tested. The hypothesis H0 is rejected if the SL between two datasets is less than 0.05. In Tables 2–4, “∗” denotes that the SL between SRSVM and the other algorithms is more than 0.05 (in fact, the minimum SL of the three tables is more than 0.3). Fig. 4 gives the t-test results on ABCDETC dataset. From the results of all experiments, Lap-TSVM outperforms TSV and Lap-SVM in most cases (Lap-SVM is slightly better than Lap-TSVM and Lap-SVM in the Credit and German datasets). These show Lap-TSVM’s classifier combined by two nonparallel hyperplanes has better flexibility of the algorithm and predication performance than TSV and Lap-SVM, as is TSV. From Tables 1, 2 and Fig. 3, with the larger of the manifold regularization’s weight or size of unlabeled data, the test accuracy of Lap-TSVM in these datasets is constantly being improved. This shows that the manifold regularization \( \|f-f_0\|_2 \) can indeed improve the performance of the classifier by geometry information embedded in the unlabeled data. Table 4 also indicates Lap-TSVM is faster than Lap-SVM.1 This is because it solves two quadratic programming problems of a smaller size instead of a single one of a very large size.

5. Conclusion

In this paper, we have proposed a novel Laplacian twin support vector machine for a semi-supervised classification problem (Lap-TSVM), which is able to exploit the geometry information of the marginal distribution embedded in unlabeled data to construct a more reasonable classifier. By choosing appropriate parameters of Lap-TSVM, our new model degenerates to either TSV or TBSVM, which is a generalized framework of a twin support vector machine for learning from labeled and unlabeled data. All experiments on synthetic and real datasets show that the Lap-TSVM’s classifier combined by two nonparallel hyperplanes is superior to Lap-SVM and TSV in classification accuracy. Furthermore, Lap-TSVM constructs a decision function by solving two small quadratic programming problems, so its training time is faster than Lap-TSVM and TSV. In the future, one possible work will be extending Lap-TSVM to be an online learning problem. Multi-class classification of Lap-TSVM is also interesting and under our consideration.

Acknowledgments

This work has been partially supported by grants from the National Natural Science Foundation of China (No. 70921061, and No. 10601064), the CAS/SAFEA International Partnership Program for Creative Research Teams, Major International (Regional) Joint Research Project (No. 71110107026), the President Fund of GUCAS, and the National Technology Support Program 2009-BAH42B02.

References


1 TSV is a typical combinatorial optimization problem, which is obviously slower than a quadratic programming problem, so we do not list its training time in Table 4.