PAC-Bayes Bounds for Twin Support Vector Machines

Xijiong Xie, Shiliang Sun*

Department of Computer Science and Technology, East China Normal University, 500 Dongchuan Road, Shanghai 200241, P.R. China

Abstract

Twin support vector machines are regarded as a milestone in the development of support vector machines. Compared to standard support vector machines, they learn two nonparallel hyperplanes rather than one as in standard support vector machines for binary classification, and work faster and sometimes perform better than support vector machines. One of the reasons that support vector machines are widely used is that they are supported by strong statistical learning theory. However, relatively little is known about the theoretical analysis of twin support vector machines. As recent tightest bounds for practical applications, PAC-Bayes bound and prior PAC-Bayes bound are based on a prior and posterior over the distribution of classifiers. In this paper, we study twin support vector machines from a theoretical perspective and use the PAC-Bayes bound and prior PAC-Bayes bound to measure the generalization error bound of twin support vector machines. Experimental results on real-world datasets show better predictive capabilities of the PAC-Bayes bound and prior PAC-Bayes bound for twin support vec-

*Corresponding author. Tel.: +86-21-54345186; fax: +86-21-54345119.
Email address: slsun@cs.ecnu.edu.cn (Shiliang Sun)

tor machines compared to the PAC-Bayes bound and the prior PAC-Bayes bound for support vector machines.

Key words: Twin support vector machine, Support vector machine, PAC-Bayes bounds, Prior PAC-Bayes bounds

---

1. Introduction

Support vector machines (SVMs) [1, 2] have been developed into a powerful tool for pattern classification and regression in machine learning. They have been applied to a variety of practical problems such as object detection, text categorization, bioinformatics and image classification. In order to obtain the best generalization ability, they find the best tradeoff between the model complexity and the learning ability according to the limited example information. They originate from the idea of structural risk minimization in statistical learning theory and output an optimal hyperplane which is obtained by maximizing the margin between two parallel hyperplanes, whose optimization involves the minimization of a quadratic programming (QP) problem. SVMs can also handle the nonlinear problem using the kernel method [3].

Recently, the research of nonparallel hyperplane classifiers has been a new hot spot. At first Mangasarian and Wild [4] proposed a nonparallel hyperplane classifier called generalized eigenvalue proximal SVMs (GEPSVMs) for binary classification. GEPSVMs aim to 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. Then Jayadeva et al. [5] proposed another nonparallel hyperplane classifier called twin support vector machines (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. Experimental results [5] showed that the performance of TSVMs is better than the performance of GEPSVMs. In SVMs, the QP has all examples in constraints while TSVMs solve a pair of QP problems for which examples of one class give the constraints of the other QP and vice versa, so that its time complexity is about $\frac{1}{4}$ of standard SVMs [6]. Experimental results [5] validate that nonparallel hyperplane classifier TSVMs can indeed improve the performance of traditional SVMs.

For the classification problem, a good classifier c is expected to minimize the generalization error which is also called the true risk or the expected loss ($c_D \equiv Pr_{(x,y)\sim D}(c(x) \neq y)$, defined as the probability of misclassifying a pair pattern-label (x, y) selected at random from D). The VC bounds [2] are generally very loose despite their enormous influence on our understanding of learning. Simultaneously, they only consider that their data-dependencies come through the training error of the classifiers. In fact, there exist VC lower bounds that are asymptotically identical to the corresponding upper bounds [17]. This suggests that significantly tighter bounds can only come through extra data-dependent properties such as the distribution of margins achieved by a classifier on the training dataset.

Early bounds are based on covering number computations [17], while

later bounds have considered Rademacher complexity [7]. Among the data-dependent bounds, the tightest bounds appear to be the PAC-Bayes bound [8]. The PAC-Bayes bound is a basic and very general method for data-dependent analysis in machine learning [9, 10, 11, 12, 13, 14, 15, 16, 17]. By now, it has been applied in such diverse areas as supervised learning, unsupervised learning and reinforcement learning, leading to state-of-the-art algorithms and accompanying generalization bounds. The original PAC-Bayes bound uses a Gaussian prior centered at the origin in the weight space. Then the PAC-Bayes bound uses part of the training dataset to compute a more informative prior and compute the bound on the remainder of the examples relative to this prior. This bound is called prior PAC-Bayes bound. Later expectation-prior PAC-Bayes bound [18] was proposed which didn't require the existence of separate dataset. The PAC-Bayes bounds are present for many famous classification methods like SVMs [8], maximum entropy classifiers [19], Gaussian process classification [20] and so on. Although twin support vector machines are a famous classification method and widely applied in practical problems, by now, theoretical analysis on twin support vector machines has not been studied. To justify TSVMs from the perspective of theory, we use the PAC-Bayes bound to analyze the generalization error bound of twin support vector machines. This can also probably motivate new algorithms along the line of TSVMs. Part of this research has been reported in a short conference paper [21]. The PAC-Bayes bound for TSVMs has exactly the same form as the PAC-Bayes bound for SVMs. Except for the above work, we also proposed prior PAC-Bayes bound for twin support

vector machines in this paper.

These bounds can also be applied to other classifiers in the family of TSVMs. The structure of the paper is as follows. After reviewing background knowledge in Section 2, we introduce the PAC-Bayes bound and prior PAC-Bayes bound for twin support vector machines in Section 3. After reporting experimental results in Section 4, we give conclusions and future work in Section 5.

2. Background

In this section, we give a brief review of SVMs, TSVMs and PAC-Bayes bound.

2.1. Support vector machines

SVMs have been introduced in the framework of structural risk minimization and are based on the theory of VC bounds [1, 2]. Consider the following binary classification problem: suppose there are m examples represented by $T = \{(x_1, y_1), ..., (x_m, y_m)\}$. Let x_i denote the ith example and $y_i \in \{1, -1\}$ denote class to which the ith example belongs. First we review the linearly separable case. Classifier parameters $w \in R^d$ and $b \in R$ need to satisfy $y_i(w^\top x_i + b) \geq 1$. 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 $\frac{2}{\|w\|_2}$, where $\|w\|_2$ denotes the L_2 norm of w. Support vectors are those training examples lying on the above two hyperplanes. The standard SVMs are obtained by

solving the following optimization problem

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

The decision function is $f(x) = \text{sign}(w^\top x + b)$, where the sign function represents an indicator function equal to 1 if the argument is nonnegative and equal to -1 if the argument is negative. When the two classes are not strictly linearly separable, classifier parameters w and b need to satisfy $y_i(w^\top x_i + b) \geq 1 - \xi_i$. The optimization problem of (1) can be modified to

$$\begin{aligned} \min_{w,b} \quad & \frac{1}{2} w^\top w + c \sum_{i=1}^{m} \xi_i \\ \text{s.t.} \quad & \forall i : y_i(w^\top x_i + b) \geq 1 - \xi_i, \quad \xi_i \geq 0, \end{aligned} \tag{2}$$

where c is a penalty parameter and ξ_i are the slack variables. The dual optimization problem of (2) can be expressed as

$$\begin{aligned} \min_{\alpha} \quad & \frac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{m} y_i y_j (x_i \cdot x_j) \alpha_i \alpha_j - \sum_{i=1}^{m} \alpha_i \\ \text{s.t.} \quad & \sum_{i=1}^{m} y_i \alpha_i = 0, \\ & 0 \leq \alpha_i \leq c, i = 1, \cdots, m, \end{aligned} \tag{3}$$

where α_i are Lagrangian multipliers. The optimal solution is

$$w = \sum_{i=1}^{m} \alpha_i^* y_i x_i, \;\; b = \frac{1}{N_{sv}} (y_j - \sum_{i=1}^{N_{sv}} \alpha_i^* y_i (x_i \cdot x_j)), \tag{4}$$

where α^* is the solution of the dual optimization problem (3), and N_{sv} represents the number of support vectors satisfying $0 < \alpha < c$. The decision function is $f(x) = \text{sign}(w^\top x + b)$.

2.2. Twin support vector machines

Since TSVMs were proposed, many researchers proposed some improved versions of TSVMs such as twin bounded support vector machine (TBSVMs) [22, 23], CDMTSVMs [24] and sparse TSVMs [25]. The significant advantage of TBSVMs over TSVMs is that the structural risk minimization principle is implemented by introducing the regularization term. The CDMTSVMs using coordinate descent method in TSVMs lead to very fast training. Sparse twin support vector machine classifier in primal space can improve the sparsity and robustness of TSVMs. Researchers also proposed some better optimization methods of TSVMs in [26, 27, 28]. Moreover, least squares twin support vector machines [29], weighted least squares twin support vector machines [30, 31] , knowledge based least squares twin support vector machines [32] and least squares twin parametric-margin support vector machines [33] have been proposed, which can lead to simple and fast algorithms through replacing inequality constraints with equality constraints. Some works [34, 35, 36] commonly attempted to use the centroid of the class to improve TSVMs, such that the examples of one class are closest to its class centroid while the examples of different classes are separated as far as possible. Robust twin support vector machines [37] and centroid twin support vector machines [38] have been proposed to deal with data with measurement noise. Structural twin support vector machines [39] have been proposed considering structural information of data. Probabilistic outputs for twin support vector machines were also proposed to improve the final classifier [40]. There are some papers about extensions of TSVMs to other learning frameworks. For examples,

TSVMs are extended to multitask learning [38], multi-view learning [41], multiple-instance learning [42] and semi-supervised learning [43]. In large data processing, online learning algorithm for least squares twin support vector machines was proposed [44]. TSVMs are also extended to solve regression problem, which are called TSVR [45] and multiclass classification problem by the one-versus-all method [46].

TSVMs [5, 38] seek two nonparallel hyperplanes instead of a single hyperplane as in the case of standard SVMs. The two nonparallel hyperplanes are obtained by solving two QPs of smaller size compared to a single large QP solved by standard SVMs. Consider a binary classification problem, 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. Each row of matrice $A_+(B_-)$ represents one example of d dimension. 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

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

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 two nonparallel hyperplanes is obtained by solving the following two independent QPs separately

(TSVM1)

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

(TSVM2)

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

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).

2.3. PAC-Bayes bound

This section is devoted to a brief review of the PAC-Bayes bound theorem [9]. We first state the general PAC-Bayes result after giving two relevant definitions. Then, we introduce the PAC-Bayes bound and prior PAC-Bayes bound for SVM. Let there be a distribution D defined on a sample space X. Let x denote a random sample X and y $\in \{-1, +1\}$ be the label of x. Moreover, let us consider a distribution Q over the classifiers c. For every classifier c, the following two error measures are defined:

Definition 2.1 (True error). *The true error c_D of a classifier c is defined as the probability of misclassifying a pair pattern-label (x, y) selected at random from D*

$$c_D \equiv Pr_{(x,y)\sim D}(c(x) \neq y) \tag{8}$$

Definition 2.2 (Empirical error). *The empirical error $\hat{c}_S$ of a classifier c on a sample S of size m is defined as the error rate on S*

$$\hat{c}_S \equiv Pr_{(x,y)}(c(x) \neq y) = \frac{1}{m}\sum_{i=1}^{m} I(c(x_i) \neq y_i) \tag{9}$$

where (x, y) comes from S, $I(\cdot)$ represents an indicator function equal to 1 if the argument is true and equal to 0 if the argument is false.

Two error measures on the distribution of classifiers are defined as $Q_D \equiv E_{c\sim Q} c_D$ (the average true error) which means the probability of misclassifying an instance x chosen uniformly from D with a classifier c chosen according to Q and $\hat{Q}_S \equiv E_{c\sim Q}\hat{c}_S$ (the average empirical error) which means the probability of classifier c chosen according to Q misclassifying an instance x chosen from a sample S.

For these two quantities, PAC-Bayes bound on the true error of the distribution of classifiers is given as follows:

Theorem 2.1 (PAC-Bayes bound). *For all prior distributions $P(c)$ over the classifiers c, and for any $\sigma \in (0, 1]$*

$$Pr_{S\sim D^m}\left(\forall Q(c) : KL_+(\hat{Q}_S \parallel Q_D) \leq \frac{KL(Q(c)\|P(c)) + \ln(\frac{m+1}{\delta})}{m}\right) \geq 1 - \delta, \tag{10}$$

where $KL(Q(c)\|P(c)) = E_{c\sim Q} \ln \frac{Q(c)}{P(c)}$ is the Kullback-Leibler divergence, and $KL_+(p\|q) = q \ln \frac{q}{p} + (1 - q) \ln \frac{1-q}{1-p}$ for $p > q$ and 0 otherwise.

The proof of the theorem can be found in [9]. This bound can be generalized to the case of linear classifiers. The m training examples define a linear classifier that can be represented by

$$c_v(x) = \text{sign}(v^\top \phi(x)) \tag{11}$$

where $\phi(x)$ is a nonlinear projection to a certain feature space where the original nonlinear problem can be solved by transforming it to a linear problem, and v is a vector from that feature space that determines the classification hyperplane.

For any vector w ($\|w\| = 1$), a stochastic classifier v is defined in the following way. Assume the prior $P(c_v)$ is a spherical Gaussian with identity covariance matrix centred on the origin, that is $v \sim N(0, I)$. Simultaneously, assume the posterior $Q(c_v) = Q(c_v|w, u)$ is a spherical Gaussian with identity covariance matrix centered along the direction pointed by w at a distance u from the origin, that is $v \sim N(uw, I)$. The generalization performance of the classifier in the form of equation (11) can be bounded as

Theorem 2.2 (PAC-Bayes bound for SVMs). *For all distributions D, for all $\delta \in (0, 1]$, it has*

$$Pr_{S\sim D^m}\left(\forall w, u : KL_+(\hat{Q}_S(w, u) \parallel Q_D(w, u)) \leq \frac{\frac{u^2}{2} + \ln(\frac{m+1}{\delta})}{m}\right) \geq 1 - \delta. \tag{12}$$

Theorem 2.2 is obtained by plugging in the new definition of KL divergence into the result of theorem 2.1. It can be easily proved using a standard expression for the KL divergence between two Gaussians in an N dimensional space,

$$\begin{aligned} KL(N(u_0, \Sigma_0) \parallel N(u_1, \Sigma_1)) = \frac{1}{2}\big(\ln\big(\frac{det\Sigma_1}{det\Sigma_0}\big) + \\ tr(\Sigma_1^{-1}\Sigma_0) + (u_1 - u_0)^\top \Sigma_1^{-1}(u_1 - u_0) - N\big). \end{aligned} \tag{13}$$

So $KL(N(0, I) \parallel N(uw, I)) = \frac{u^2}{2}$. It can be shown (see [9]) that

$$\hat{Q}_S(w, u) = E_m[\tilde{F}(u\gamma(x, y))] \tag{14}$$

where E_m is the average over the m training examples, $\gamma(x, y)$ is the normalised margin of the training examples

$$\gamma(x, y) = \frac{yw^\top \phi(x)}{\|\phi(x)\|\|w\|} \tag{15}$$

and $\tilde{F} = 1 - F$, where F is the cumulative normal distribution

$$F(x) = \int_{-\infty}^{x} \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}} dx. \tag{16}$$

It is observed from that SVMs are computed by the means of the kernel trick. The generalization error of such a classifier can be bounded by at most twice the average true error $Q_D(w, u)$ of the corresponding stochastic classifier in Theorem 2.2. For all u, it has

$$Pr_{(x,y)\sim D}\big(\text{sign}(w^\top \phi(x)) \neq y\big) \leq 2Q_D(w, u). \tag{17}$$

Then we state the prior PAC-Bayes bound and consider learning a different prior by training an SVM on a subset T of the training set containing r training examples [18]. With these r examples, it can learn an (unit and biased) SVM classifier w_r, and form a prior $P(w_r, \eta) \sim N(\eta w_r, I)$ which is a Gaussian distribution with identity covariance matrix centered along w_r at a distance η from the origin.

Theorem 2.3 (Prior PAC-Bayes bound for SVMs [18]). *Let us consider a prior on the distribution of classifiers consisting of a spherical Gaussian with identity covariance centered along the direction given by w_r at a distance η from the origin. Classifier w_r has been learnt from a subset T of r examples a priori separated from a training set S of m examples. Then, for all distributions D, for all $\delta \in (0, 1]$, it has*

$$Pr_{S\sim D^m}\left(\forall w_m, u : KL_+(\hat{Q}_{S\setminus T} \parallel Q_D) \leq \frac{\frac{\|\eta w_r - u w_m\|^2}{2} + \ln(\frac{m-r+1}{\delta})}{m - r}\right) \geq 1 - \delta. \tag{18}$$

where $\hat{Q}_{S\backslash T}$ is a stochastic measure of the empirical error of the classifier on the $m-r$ examples not used to learn the prior. This stochastic error is computed according to equation (14) but averaged over $S\backslash T$.

The KL divergence between prior and posterior is computed as follow:

$$\begin{aligned} &KL(Q(w_m,u)\|P(w_r,\eta)) = KL(N(uw_m,I)\|N(\eta w_r,I)) \\ &= \frac{\|\eta w_r - u w_m\|^2}{2} = \frac{1}{2}(u^2+\eta^2-2u\eta w_r^\top w_m). \end{aligned} \tag{19}$$

3. PAC-Bayes bounds for twin support vector machines

In this section, we introduce our proposed PAC-Bayes bound for twin support vector machines and prior PAC-Bayes bound for twin support vector machines.

3.1. PAC-Bayes bound for twin support vector machines

TSVMs can improve the performance and time complexity compared to SVMs. However, there does not exist formal theoretical analysis about TSVMs. In this section, we attempt to analyze the PAC-Bayes generalization error bound of TSVMs. At first, we analyze the classifier of TSVMs. In order to analyze the PAC-Bayes bound of twin support vector machines, we can rewrite the final decision function of TSVMs as this form

$$\begin{aligned} f(x) = \operatorname{sign}\big(&(\frac{w_2^\top}{\|w_2\|}\operatorname{sign}(w_2^\top x+b_2) - \frac{w_1^\top}{\|w_1\|}\operatorname{sign}(w_1^\top x+b_1))x \\ &+ (\frac{b_2}{\|w_2\|}\operatorname{sign}(w_2^\top x+b_2) - \frac{b_1}{\|w_1\|}\operatorname{sign}(w_1^\top x+b_1))\big). \end{aligned} \tag{20}$$

We define $\bar{w} = (\frac{w_2^\top}{\|w_2\|}\operatorname{sign}(w_2^\top x+b_2) - \frac{w_1^\top}{\|w_1\|}\operatorname{sign}(w_1^\top x+b_1))^\top$ and $\bar{b} = \frac{b_2}{\|w_2\|}\operatorname{sign}(w_2^\top x+b_2) - \frac{b_1}{\|w_1\|}\operatorname{sign}(w_1^\top x+b_1)$, then we can get the final linear

classifier

$$f(x) = \text{sign}(\bar{w}^\top x + \bar{b}). \tag{21}$$

The classifier can also be written as kernelized form

$$c_{\bar{v}}(x) = \text{sign}(\bar{v}^\top \phi(x)). \tag{22}$$

Because different test examples may have different classifier parameters $\bar{w}$ and $\bar{b}$ in TSVMs while they have the same classifier parameters in SVMs. The four decision function forms of TSVMs in details are obtained according to the different values of indicator functions:

1. For training examples satisfying $\text{sign}(w_2^\top x + b_2) >= 0$ and $\text{sign}(w_1^\top x + b_1) >= 0$, their decision function is $f(x) = \text{sign}\big((\frac{w_2^\top}{\|w_2\|} - \frac{w_1^\top}{\|w_1\|})x + (\frac{b_2}{\|w_2\|} - \frac{b_1}{\|w_1\|})\big)$. Let p_1 denote the percentage of the training examples in the whole training set.

2. For training examples satisfying $\text{sign}(w_2^\top x + b_2) >= 0$ and $\text{sign}(w_1^\top x + b_1) < 0$, their decision function is $f(x) = \text{sign}\big((\frac{w_2^\top}{\|w_2\|} + \frac{w_1^\top}{\|w_1\|})x + (\frac{b_2}{\|w_2\|} + \frac{b_1}{\|w_1\|})\big)$. Let p_2 denote the percentage of the training examples in the whole training set.

3. For training examples satisfying $\text{sign}(w_2^\top x + b_2) < 0$ and $\text{sign}(w_1^\top x + b_1) < 0$, their decision function is $f(x) = \text{sign}\big((-\frac{w_2^\top}{\|w_2\|} + \frac{w_1^\top}{\|w_1\|})x + (-\frac{b_2}{\|w_2\|} + \frac{b_1}{\|w_1\|})\big)$. Let p_3 denote the percentage of the training examples in the whole training set.

4. For training examples satisfying $\text{sign}(w_2^\top x + b_2) < 0$ and $\text{sign}(w_1^\top x + b_1) >= 0$, their decision function is $f(x) = \text{sign}\big((-\frac{w_2^\top}{\|w_2\|} - \frac{w_1^\top}{\|w_1\|})x + (-\frac{b_2}{\|w_2\|} - \frac{b_1}{\|w_1\|})\big)$. Let p_4 denote the percentage of the training examples in the whole training set.

Define a vector set $\tilde{w}$ contains $\tilde{w}_i$ ($\|\tilde{w}_i\| = 1$) and a vector $\bar{v}$ has four forms $\bar{v}_i, i = 1, 2, 3, 4$. Let us consider prior classifier $P(c_{\bar{v}})$ to be a spherical Gaussian with identity covariance matrix centred on the origin, that is $\bar{v}_i \sim N(0, I), i = 1, 2, 3, 4$. We choose four posteriors $Q(\tilde{w}_i, u), i = 1, 2, 3, 4$ to be a spherical Gaussian with identity covariance matrix centered along the direction pointed by $\tilde{w}_i, i = 1, 2, 3, 4$ at a distance u from the origin and $p_i, i = 1, 2, 3, 4$ as the corresponding partition percents of the train examples. That is $\bar{v}_i \sim N(u\tilde{w}_i, I), i = 1, 2, 3, 4$. Then we present the PAC-Bayes bound for TSVMs.

Theorem 3.1 (PAC-Bayes bound for TSVMs). *For all distributions D, for all $\delta \in (0, 1]$, we have*

$$\begin{aligned} &Pr_{S\sim D^m}\left(\forall \tilde{w}, u : KL_+(\hat{Q}_S(\tilde{w}, u) \parallel Q_D(\tilde{w}, u)) \leq \frac{\frac{u^2}{2} + \ln(\frac{m+1}{\delta})}{m}\right) \\ &\geq 1 - \delta. \end{aligned} \tag{23}$$

The average KL divergence between prior and posterior is computed as follows:

$$\begin{aligned} \sum_{i=1}^{4} p_i KL(Q(\tilde{w}_i, u) \| P(c)) &= \sum_{i=1}^{4} p_i KL(N(\tilde{w}_i, u) \| N(0, I)) \\ &= \frac{1}{2} \sum_{i=1}^{4} p_i \|u\tilde{w}_i\|^2 = \frac{u^2}{2}. \end{aligned} \tag{24}$$

It can be shown that

$$\hat{Q}_S(\tilde{w}, u) = E_m[\tilde{F}(u\gamma(x, y))], \tag{25}$$

where E_m is the average over the m training examples, $\gamma(x, y)$ is the normalised margin of the training examples

$$\gamma(x, y) = \frac{y\bar{v}^\top \phi(x)}{\|\phi(x)\|\|\bar{v}\|} \tag{26}$$

and $\tilde{F} = 1 - F$, where F is the cumulative normal distribution

$$F(x) = \int_{-\infty}^{x} \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}} dx. \tag{27}$$

It is observed from that TSVMs are also computed by the means of the kernel trick. The generalization error of such a classifier can be bounded by at most twice the average true error $Q_D(\tilde{w}, u)$ of the corresponding stochastic classifier in Theorem 3.1. For all u, we have

$$Pr_{(x,y)\sim D}\big(\text{sign}(\bar{v}^\top \phi(x)) \neq y\big) \leq 2Q_D(\tilde{w}, u). \tag{28}$$

The expression of the PAC-Bayes bound for TSVMs is as same as the one of the prior PAC-Bayes bound for SVMs. Therefore, their main difference is the average empirical error $\hat{Q}_S$.

3.2. Prior PAC-Bayes bounds for twin support vector machines

Then we analysis the prior PAC-Bayes bound for TSVMs. Let us consider four priors on the distribution of classifiers consisting of a spherical Gaussian with identity covariance centered along the direction given by $\tilde{w}_{ri}, i = 1, 2, 3, 4$ at a distance η from the origin. That is $\bar{v}_i \sim N(\eta\tilde{w}_{ri}, I), i = 1, 2, 3, 4$. Classifiers $\tilde{w}_{ri}, i = 1, 2, 3, 4$ has been learnt from a subset T of r examples a priori separated from a training set S of m examples and $p_{ri}, i = 1, 2, 3, 4$ as the corresponding partition percents of the r train examples. We choose four posteriors to be a spherical Gaussian with identity covariance matrix centered along the direction pointed by $\tilde{w}_{mi}, i = 1, 2, 3, 4$ at a distance u from the origin learnt from the rest $m - r$ examples and $p_{mi}, i = 1, 2, 3, 4$

as the corresponding partition percents of the $m-r$ train examples. That is $\bar{v}_i \sim N(u\tilde{w}_{mi}, I), i = 1, 2, 3, 4$. Then we can obtain the prior PAC-Bayes bound for TSVMs.

Theorem 3.2 (Prior PAC-Bayes bound for TSVMs). *for all distributions D, for all $\delta \in (0, 1]$, we have*

$$Pr_{S\sim D^m}\left(\forall \tilde{w}_m, u : KL_+(\hat{Q}_{S\backslash T} \parallel Q_D) \leq \frac{\frac{u^2+\eta^2-2\mu\eta\sum_{i=1}^{4}\sum_{j=1}^{4} p_{ri}p_{mj}\tilde{w}_{ri}^{\top}\tilde{w}_{mj}}{2} + \ln(\frac{m-r+1}{\delta})}{m-r}\right) \geq 1-\delta, \tag{29}$$

here $\hat{Q}_{S\backslash T}$ is a stochastic measure of the empirical error of the classifier on the $m-r$ examples not used to learn the prior. This stochastic error is computed according to equation (14) but averaged over $S\backslash T$. The average KL divergence between prior and posterior is computed as follows:

$$\begin{aligned}\sum_{i=1}^{4}\sum_{j=1}^{4} p_{ri}p_{mj}KL(Q(\tilde{w}_{mj}, u)\|P(\tilde{w}_{ri}, \eta)) &= \frac{1}{2}\sum_{i=1}^{4}\sum_{j=1}^{4} p_{ri}p_{mj}\|\eta\tilde{w}_{ri} - u\tilde{w}_{mj}\|^2 \\ &= \frac{1}{2}(u^2+\eta^2-2\mu\eta\sum_{i=1}^{4}\sum_{j=1}^{4} p_{ri}p_{mj}\tilde{w}_{ri}^{\top}\tilde{w}_{mj}).\end{aligned} \tag{30}$$

The expression of the prior PAC-Bayes bound for TSVMs is quite different from the one of the prior PAC-Bayes bound for SVMs. Their main differences are the average empirical error $\hat{Q}_S$ and KL divergence term.

4. Experimental Results

4.1. Datasets

In this section, we implement experiments of binary classification problems using real-world datasets. Details about the five datasets are given as follows:

Contraceptive Method Choice (CMC). The dataset comes from UCI Machine Learning Repository. It contains 1473 examples and has 9 attributes.

Face Detection. The dataset comes from the MIT CBCL repository. It is a binary classification problem which intends to identify whether a picture is a human face or not. In this experiment, 2000 face and non-face images are used, where half of them are faces and each image is a 19×19 gray picture.

Handwritten Digit Classification. The dataset comes from UCI Machine Learning Repository. The dataset we used here contains 2400 examples of digits 3 and 8 chosen from the MNIST digital images, where half of the data are digit 3 and the image sizes are 28×28.

Pima. The dataset comes from UCI Machine Learning Repository. The dataset contains 768 examples and has 8 attributes.

German Credit. The dataset comes from UCI Machine Learning Repository. The dataset contains 1000 examples and has 20 attributes.

[Table 1 about here.]

[Table 2 about here.]

[Table 3 about here.]

[Table 4 about here.]

[Table 5 about here.]

[Table 6 about here.]

4.2. Experimental Setting

For the comparison between the PAC-Bayes bound of SVMs and the PAC-Bayes bound of TSVMs, we obtain 10 different training/test set partitions with 80% of the examples forming the training dataset and 20% forming the test dataset. We then change the training sizes from 20% to 100% of the formed training datasets. For the comparison between the prior PAC-Bayes bound of SVMs and TSVMs, we obtain 10 different training/test set partitions with 90% of the examples forming the training dataset. We perform experiments with Gaussian RBF kernel. The Gaussian kernel can be written as

$$K(x_i, x_j) = exp(-\frac{\|x_i - x_j\|_2^2}{2\sigma^2}), \tag{31}$$

where σ is the width of the Gaussian kernel. The optimal pair (c, σ) of SVMs is sought by grid search strategy to select best parameters in the region $\{10^{-3}, 10^{-2}, 0.1, 1, 10, 100, 1000\}$ through a five-fold cross-validation. The optimal pair (c_1, c_2, σ) of TSVMs is also sought by grid search strategy to select best parameters in the region $\{10^{-3}, 10^{-2}, 0.1, 1, 10, 100, 1000\}$ through a five-fold cross-validation. In the experiments, we set $\delta = 0.01$. Parameter η need to be fixed in region $[0.1, 100]$. Parameter μ needs to be adjusted in region $[0.1, 100]$ by binary search.

4.3. Experimental Results and Analysis

We show experimental results which compare the PAC-Bayes bounds (Q_D) for TSVMs with the PAC-Bayes bounds for SVMs. The test errors and

PAC-Bayes bounds for SVMs and TSVMs are averaged for 10 times. We complete the average with the standard deviation. The results are shown in Tables 1, 2, 3, 4, 5. "PB-SVM" represents the PAC-Bayes bound for SVMs and "PB-TSVM" represents the PAC-Bayes bound for TSVMs. We also show the difference between "PB-TSVMs" and Error for "TSVMs" called "Gap-TSVM" and the difference between "PB-TSVMs" and "Error for TSVMs" called "Gap-SVM" in the results. The results of Gap-TSVM are less than the ones of Gap-SVM in the most cases on the all dataset. The test errors have little relationship with the PAC-Bayes bounds. They are shown in experiments because we can obtain an important and supplemental conclusion.

From the experimental results, we can find that as the rate of training dataset increases, the bounds for SVMs and TSVMs are much tighter. In Table 2, 4, 5, the bounds for TSVMs are almost tighter than the bounds for SVMs. In Tables 1, 3, the bounds for SVMs and TSVMs are nearly the same. We can also conclude that when the rate of training dataset is low, the performance of TSVMs is not better than SVMs. When the rate of training dataset is high, the performance of TSVMs is better than or close to SVMs. We speculate that when the rate of training dataset is low, TSVMs need more parameters to train and cause over-fitting results. When the rate of training dataset is high, TSVMs are more flexible. The results of Gap-TSVM are less than the ones of Gap-SVM in the most cases on the all dataset. In summary, the experimental results verify the good predictive capabilities of the PAC-Bayes bound for twin support vector machines.

The results of the prior PAC-Bayes bounds is in Tables 6, "PPB-SVM"

represents the prior PAC-Bayes bound for SVMs and "PPB-TSVM" represents the prior PAC-Bayes bound for TSVMs. From the results, we can find the prior PAC-Bayes bounds for TSVMs are almost tighter than the prior PAC-Bayes bounds for SVMs except in German Credit dataset. The results show that the good predictive capabilities PAC-Bayes bound and prior PAC-Bayes bound for TSVMs.

5. Conclusion and Future work

Many practical applications and extended algorithms for twin support vector machines have been proposed. However, there does not exist theoretical justifications on twin support vector machines. In this paper, we use the PAC-Bayes bound and prior PAC-Bayes bound to analyze the generalization error bound of twin support vector machines. Comparative experiments on real-world datasets verify the better predictive capabilities of the PAC-Bayes bound and prior PAC-Bayes bound for twin support vector machines. In the future, we can use other informative priors inspired by [18] to tighten the bounds.

Acknowledgment

The corresponding author Shiliang Sun would like to thank supports from the National Natural Science Foundation of China under Projects 61673179 and 61370175, 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.N. Vapnik, The Nature of Statistical Learning Theory, New York: Springer-Verlag, 1995.

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

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

[5] Jayadeva, S. Khemchandani, Chandra, Twin support vector machines for pattern classification. IEEE Transactions on Pattern Analysis and Machine Intelligence 29 (2007) 905-910.

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

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

[8] A. Ambroladze, E. Parrado-Hernández, J. Shawe-Taylor, Tighter PAC-Bayes bounds. Advanced in Neural Information Processing Systes 473 (2006) 4-28.

[9] J. Langford, Tutorial on practical prediction theory for classification. Jounal of Machine Learning Research 6 (2005) 273-306.

[10] J.F. Roy, M. Marchand, F. Laviolette, A column generation bound minimization approach with PAC-Bayesian generalization guarantees. in: Proceedings of the 19th International Conference on Artificial Intelligence and Statistics, 2016, 12411249.

[11] T. Liu, D. Tao, D. Xu, Dimensionality-dependent generalization bounds for k-dimensional coding schemes. Neural Computation, 8 (2016) 1-34.

[12] A. Tewari, S. Chaudhuri, Generalization error bounds for learning to rank: Does the length of document lists matter? in :Proceedings of the 32nd International Conference on Machine Learning, volume 37 of JMLR Workshop and Conference Proceedings, 2015.

[13] Q. Cao, Z.C. Guo, Y. Ying, Generalization bounds for metric and similarity learning. Machine Learning, 102 (2016) 115-132.

[14] T. Liu, D. Tao, M. Song, S. Maybank, Algorithm-dependent generalization bounds for multi-Task learning. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2016, doi:10.1109/TPAMI.2016.2544314.

[15] J. Langford, J. Shawe-Taylor, PAC-Bayes & Margins. Advances in Neural Information Processing Systems 14 (2002) 423-430.

[16] G. Lever, F. Laviolette, J. Shawe-Taylor, Tighter PAC-Bayes bounds through distribution-dependent priors. Theoretical Computer Science 473 (2013) 4-28.

[17] P. Germain, A. Lacasse, F. Laviolette, M. Marchand, PAC-Bayesian learning of linear classifiers. in: Proceedings of the 26th Annual International Conference on Machine Learning, 2009, pp. 353-360.

[18] E. Parrado-Hernández, A. Ambroladze, J. Shawe-Taylor, S. Sun, PAC-Bayes bounds with data dependent priors. Journal of Machine Learning Research 13 (2012) 3507-3531.

[19] J. Shawe-Taylor, D. R. Hardoon, PAC-Bayes analysis of maximum entropy learning. in: Proceedings of the 12th International Conference on Artificial Intelligence and Statistics, 2009, pp. 480-487.

[20] M. Seeger, PAC-Bayesian generalisation error bounds for Gaussian process classification. Journal of Machine Learning Research 3 (2002) 233-269.

[21] X. Xie, S. Sun, PAC-Bayes analysis for twin support vector machines. in: Proceedings of the 27th International Joint Conference on Neural Networks, 2015, pp. 1-6.

[22] Y. Shao, C. Zhang, X. Wang, N. Deng, Improvements on twin support vector machines. IEEE Transactions on Neural Networks 22 (2011) 962-968.

[23] S. Ding, Y. Zhao, B. Qi, H. Huang, An overview on twin support vector machines. Artificial Intelligence Review 2012.

[24] Y.H. Shao, N.Y. Deng, A coordinate descent margin based-twin support vector machine for classification. Neural Networks 25 (2012) 114-121.

[25] X. Peng, Building sparse twin support vector machine classifiers in primal space. Information Sciences 181 (2011) 3967-3980.

[26] S. Ding, J. Yu, H. Huang, H. Zhao, Twin support vector machines based on particle swarm optimization. Journal of Computers 8 (2013) 3967-3980.

[27] D. Wang, N. Ye, Q. Ye, Twin support vector machines via fast generalized Newton refinement, in: Proceedings of International Conference on Intelligent Human-machine Systems and Cybernetics, 2010, pp. 62-65.

[28] S. Ding, X Zhang, J Yu, Twin support vector machines based on fruit fly optimization algorithm. International Journal of Machine Learning and Cybernetics (2015), doi:10.1007/s13042-015-0424-8.

[29] M.A. Kumar, M. Gopal, Least squares twin support vector machines for pattern classification. Expert Systems with Applications 36 (2009) 7535-7543.

[30] J. Chen, Weighted least squares twin support vector machines for pattern classification, in: Proceedings of the 2nd International Conference on Computer and Automation Engineering, 2010, pp. 242-246.

[31] Y. Xu, X. Lv, Z. Wang, L. Wang, A weighted least squares twin support vector machine. Journal of Information Science and Engineering 30 (2014) 1773-1787.

[32] MA. Kumar, R. Khemchandani, M. Gopal, S. Chandra, Knowledge based least squares twin support vector machines. Information Sciences 180 (2010) 4606-4618.

[33] Y.H. Shao, Z. Wang, W.J. Chen, N.Y. Deng, Least squares twin parametric-margin support vector machines for classification. Applied Intelligence 39 (2013) 451-464.

[34] Y.H. Shao, N.Y. Deng, Z.M. Yang, Least squares recursive projection twin support vector machine for classification. Pattern Recognition 45 (2012) 2299-2307.

[35] Y.H. Shao, Z. Wang, W.J. Chen, N.Y. Deng, A regularization for the projection twin support vector machine. Knowledge-Based Systems 37 (2013) 203-210.

[36] X. Chen, J. Yang, Q. Ye, J. Liang, Recursive projection twin support vector machine via within-class variance minimization. Pattern Recognition 44 (2011) 2643-2655.

[37] Z. Qi, Y. Tian, Y. Shi, Robust twin support vector machine for pattern classification. Pattern Recognition 46 (2014) 305-316.

[38] X. Xie, S. Sun, Multitask centroid twin support vector machines. Neurocomputing 149 (2015) 1085-1091.

[39] Z. Qi, Y. Tian, Y. Shi, Structural twin support vector machine for classification. Knowledge-Based Systems 43 (2013) 74-81.

[40] Y.H. Shao, Y. Tian, Y. Shi, Probabilistic outputs for twin support vector machines. Knowledge-Based Systems 33 (2012) 145-151.

[41] X. Xie, S. Sun, Multi-view Laplacian twin support vector machines. Applied Intelligence 41 (2014) 1059-1068.

[42] Y.H. Shao, Z.X. Yang, X.B. Wang, Multiple instance twin support vector machines. Lect Note Oper Res 12 (2010) 433-442.

[43] Z. Qi, Y. Tian, Y. Shi, Laplacian twin support vector machine for semi-supervised classification. Neural Networks 35 (2012) 46-53.

[44] X.X. Mu, L.Y. Chen, J.T. Li, Online learning algorithm for least squares twin support vector machines. Computer Simulation 29 (2012) 25-28.

[45] X. Peng, TSVR: An efficient twin support vector machine for regression. Neural Networks 23 (2010) 365-372.

[46] J. Xie, K. Hone, W. Xie, X. Gao, Y. Shi, X. Liu, Extending twin support vector machine classifier for multi-category classification problems. Intelligent Data Analysis 17 (2013) 649-664.

List of Tables

Table 1: PAC-Bayes bounds (%) and classification errors (%) on CMC.

Rate	PB-SVMs	Error for SVMs	PB-TSVMs	Error for TSVMs	Gap-SVM	Gap-TSVM
20%	66.01±0.01	35.14±2.35	65.98±0.03	36.00±1.73	30.87	29.98
40%	61.83±0.00	31.55±1.50	61.83±0.01	34.16±2.62	30.28	27.67
60%	59.89±0.00	31.01±1.37	59.89±0.00	29.59±1.43	28.88	30.30
80%	58.69±0.00	28.49±3.79	58.69±0.00	28.98±1.23	30.20	29.71
100%	57.87±0.00	28.52±2.21	57.87±0.00	26.60±1.30	29.35	31.27

Table 2: PAC-Bayes bounds (%) and classification errors (%) on face detection.

Rate	PB-SVMs	Error for SVMs	PB-TSVMs	Error for TSVMs	Gap-SVM	Gap-TSVM
20%	62.52±0.01	3.51±1.12	62.35±0.38	4.22±0.42	59.01	58.13
40%	59.21±0.00	1.43±0.42	58.91±0.27	1.85±0.41	57.78	57.06
60%	57.68±0.00	0.64±0.25	57.52±0.18	0.98±0.27	57.04	56.54
80%	56.74±0.00	0.35±0.13	56.51±0.08	0.38±0.19	56.39	56.13
100%	56.09±0.00	0.12±0.12	55.97±0.11	0.15±0.12	55.97	55.82

Table 3: PAC-Bayes bounds (%) and classification errors (%) on Handwritten Digit Classification.

Rate	PB-SVMs	Error for SVMs	PB-TSVMs	Error for TSVMs	Gap-SVM	Gap-TSVM
20%	61.55±0.01	3.70±0.38	61.46±0.06	4.06±0.81	57.85	57.40
40%	58.49±0.00	2.21±0.26	58.47±0.01	2.59±0.40	56.28	55.88
60%	57.07±0.00	1.47±0.27	57.07±0.01	1.54±0.25	55.60	55.53
80%	56.21±0.00	0.66±0.25	56.21±0.00	0.84±0.26	55.55	55.37
100%	55.61±0.00	0.62±0.15	55.61±0.00	0.44±0.09	54.99	55.17

Table 4: PAC-Bayes bounds (%) and classification errors (%) on Pima.

Rate	PB-SVMs	Error for SVMs	PB-TSVMs	Error for TSVMs	Gap-SVM	Gap-TSVM
20%	68.89±0.01	26.87±2.87	68.81±0.08	33.50±2.24	42.02	35.31
40%	64.06±0.00	24.57±2.15	64.05±0.00	26.71±1.53	39.49	37.34
60%	61.79±0.00	23.81±1.02	61.79±0.00	20.59±1.04	37.98	41.20
80%	60.38±0.00	22.69±0.93	60.38±0.00	15.05±1.23	37.69	45.33
100%	59.40±0.00	22.77±0.70	59.39±0.00	11.68±2.38	36.63	47.71

Table 5: PAC-Bayes bounds (%) and classification errors (%) on German.

Rate	PB-SVMs	Error for SVMs	PB-TSVMs	Error for TSVMs	Gap-SVM	Gap-TSVM
20%	66.91±0.00	26.46±1.68	66.87±0.07	29.59±2.57	40.45	37.28
40%	62.54±0.00	24.21±2.29	62.52±0.04	25.74±1.75	38.33	36.78
60%	60.49±0.00	22.06±1.71	60.47±0.03	21.75±2.53	38.43	38.72
80%	59.22±0.00	20.78±0.75	59.22±0.00	18.95±1.65	38.44	40.27
100%	58.35±0.00	19.83±0.62	58.34±0.00	16.66±1.89	38.52	41.68

Table 6: Prior PAC-Bayes bounds (%).

dataset	PB-SVMs	PB-TSVMs	PPB-SVMs	PPB-TSVMs
CMC	57.39±0.04	57.44±0.01	57.49±0.02	**57.34**±0.85
face detection	41.13±0.20	**37.47**±8.75	39.28±0.24	38.10±8.53
Handwritten Digit	60.29±3.03	55.33±6.34	59.23±1.23	**54.19**±5.71
Pima	58.41±0.76	**58.28**±0.59	59.02±0.13	58.62±0.14
German Credit	57.89±0.00	**57.85**±0.02	58.03±0.01	58.28±0.59