Robust twin support vector machine for pattern classification

Zhiquan Qi, Yingjie Tian*, Yong Shi*

Research Center on Fictitious Economy & Data Science, Chinese Academy of Sciences, Beijing 100190, China

A R T I C L E   I N F O

Article history:
Received 27 December 2011
Received in revised form 22 June 2012
Accepted 27 June 2012
Available online 4 July 2012

Keywords:
Twin support vector machine
Second order cone programming
Robust

A B S T R A C T

In this paper, we proposed a new robust twin support vector machine (called R-TWSVM) via second order cone programming formulations for classification, which can deal with data with measurement noise efficiently. Preliminary experiments confirm the robustness of the proposed method and its superiority to the traditional robust SVM in both computation time and classification accuracy. Remarkably, since there are only inner products about inputs in our dual problems, this makes us apply kernel trick directly for nonlinear cases. Simultaneously we does not need to solve the extra inverse of matrices, which is totally different with existing TW SVMs. In addition, we also show that the TW SVMs are the special case of our robust model and simultaneously give a new dual form of TW SVM by degenerating R-TWSVM, which successfully overcomes the existing shortcomings of TW SVM.

© 2012 Elsevier Ltd. All rights reserved.

1. Introduction

For the last decade, support vector machines (SVMs) [1–3], as powerful tools for pattern classification and regression, have already successfully applied in a wide variety of fields [4–12]. For the standard support vector classification (SVC), the basic idea is to find the optimal separating hyperplane between the positive and negative examples. The optimal hyperplane may be obtained by maximizing the margin between two parallel hyperplanes, which involves the minimization of a quadratic programming problem (QPP). By introducing kernel trick into the dual QPP, SVC can also solve nonlinear classification problem successfully.

Recently, Jayadeva et al. [13] proposed a twin support vector machine (TWSVM) classifier for binary classification, motivated by GEPSVM [14]. TWSVMs generate two nonparallel planes such that each plane is closer to one of two classes and is at least one distance from the other. It is implemented by solving two smaller QPPs rather than a single large QPP, which makes the learning speed of TWSVM is more faster than that of a classical SVM. Experimental results in [13,15] show the superiority of TWSVM over both standard SVM and GEPSVM on UCI datasets. Some extensions to the TWSVM can be found in [15–18].

For the methods mentioned above, the parameters in the training sets are implicitly assumed to be known exactly. However, in real world applications, the parameters have perturbations since they are estimated from the data subject to measurement and statistical errors [19,20]. Goldfarb et al. pointed out that the solutions to optimization problems are typically sensitive to parameter perturbations, errors in the input parameters tend to get amplified in the decision function, which often results in misclassification. For instance, for the fixed examples, original discriminants can correctly separate them (see Fig. 1(a)). When each example is allowed to move in a sphere, original decision function cannot separate samples in the worst case (see Fig. 1(b)). So the goal is to explore a robust model which can deal with data set with measurement or statistical errors (see Fig. 1(c)). There are many methods of constructing the robust SVMs. Bi and Zhang derived a general statistical formulation where unobserved input is modeled as a hidden mixture component [21]. Literatures [20,22–24] employed second order cone programming (SOC) methods to handle the missing and uncertain data; literatures [25–27] construct the robust models by the ramp loss functions. Xu et al. solved the robust classification problem for a class of non-box-typed uncertainty sets, and providing a linkage between robust classification and the standard regularization scheme of SVMs [28,29]. Other related works also can found in [30–34].

In this paper, following the line of the research in [19,20,24,30], we proposed a new robust twin support vector machine for data with measurement errors (called R-TWSVM), which is represented as a second-order cone programming (SOCP) [35]. Second-order cone (SOC) is also called the Lorentz cone:

**Definition 1.1 (Second-order cone).** The cone $K$ is called a second-order cone $L^m$ if

$$K = \begin{cases} 
(u = u_1 \in \mathbb{R} | u_1 \geq 0), & m = 1; \\
(u = (u_1, u_2, \ldots, u_m)^T \in \mathbb{R}^m | u_1 \geq \sqrt{u_2^2 + \cdots + u_m^2}), & m \geq 2.
\end{cases}$$

The SOCP is a special convex optimization problem involving SOC constraints, which can be efficiently solved by interior point
methods. Related work can be found in [20,36–38]. The proposed \(\mathcal{R}\)-TWSVM has the following compelling properties.

- To our knowledge, \(\mathcal{R}\)-TWSVM is the first TWSVM implementation of dealing with data with measurement noise, which is an useful extension of TWSVM.

- We show that the TWSVM and TB SVM [39] (TWSVM with the regular terms) are the special cases of our robust models. This provides an alternative explanation to the success of \(\mathcal{R}\)-TWSVM.

- Although TWSVMs have had great success in classification, they have the following shortcomings: (1) unlike standard SVMs, the dual problem of TWSVMs does not have inner products form about samples, which makes TWSVMs have to use an approximate technology [13] to solve the nonlinear case. This means that TWSVMs need to solve two problems for linear case and two other problems for nonlinear case separately. At the same time, in order to realize the structural risk minimization (SRM) principle, TWSVMs have to add different regularization term for linear and nonlinear cases owing to the reason above [39]. So far, there is not a clear theoretical explanation for these regularization terms, especially for the nonlinear case. (2) Although TWSVMs only solve two smaller QPPs, they have to compute the inverse of matrices,\(^1\) it is in practice intractable or even impossible for a large data set. In this paper, we overcome the shortcomings above successfully. There are only inner products about samples in our dual problems of \(\mathcal{R}\)-TWSVM, this make us not need to add the other regularization term \(\|w\|^2 + b^2\) [39] for nonlinear case. Correspondingly, kernel trick can also be applied to our model directly due to the reason above. In addition, we also give a new dual form of TWSVM by degenerating \(\mathcal{R}\)-TWSVM, which successfully overcomes the shortcomings mentioned above as well.

- Compared with traditional robust optimization models such as [20,22–24], we solve two SOCP problems of a smaller size instead of a large sized one. This makes \(\mathcal{R}\)-TWSVM almost faster than these algorithms above. Theoretical analysis and the results of all the experiments show that the \(\mathcal{R}\)-TWSVM is approximately four times faster than the usual R-SVM [20].

Different with exiting robust SVM models, \(\mathcal{R}\)-TWSVM use two nonparallel hyperplanes for two classes to construct the final decision function, which factually exploits the data's structural information while margin maximizing to the missing and uncertain data. This can improve the model's generalized capability efficiently [14,13,39]. In addition, \(\mathcal{R}\)-TWSVM employs two different loss function: a quadratic loss function making the proximal hyperplane close enough to the class itself, and a soft-margin loss function making the hyperplane as far as possible from the other class, which results that almost all the points in this class and some points in the other class contribute to each final decision function. This makes \(\mathcal{R}\)-TWSVM have stronger insensitivity to missing or uncertain data with label noise.

The remaining content is organized as follows. In Section 2, we briefly introduces the background of SVM and TWSVM. In Section 3, describe the detail of \(\mathcal{R}\)-TWSVM. In Section 4, we show experiments of \(\mathcal{R}\)-TWSVM on various data sets. We conclude this work in Section 5.

2. Background

2.1. Support vector classification (SVC)

For classification about the training data

\[
T = \{(x_1,y_1), \ldots, (x_n,y_n)\} \in (\mathbb{R}^d \times \mathcal{Y})^n,
\]

where \(x_i \in \mathbb{R}^d, y_i \in \mathcal{Y} = \{-1,1\}, i = 1, \ldots, n\). SVM’s linear softmargin algorithm is to solve the following primal QPP:

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

where \(C\) is a penalty parameter and \(\xi_i\) are the slack variables. The goal is to find an optimal separating hyperplane \(w^T \cdot x + b = 0\),

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

\[
\begin{align*}
\max_{\alpha} & \quad \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} y_i y_j \alpha_i \alpha_j \langle x_i, x_j \rangle \\
\text{s.t.} & \quad \sum_{i=1}^{n} y_i \alpha_i = 0, \quad 0 \leq \alpha_i \leq C, \quad i = 1, \ldots, n
\end{align*}
\]

where \(x\) are lagrangian multipliers. The optimal separating hyperplane of (3) can be given by

\[
w = \sum_{i=1}^{n} \alpha_i^* y_i x_i, \quad b = \frac{1}{N_{sv}} \left( \sum_{i=1}^{N_{sv}} \alpha_i^* y_i \langle x_i, x \rangle \right),
\]

where \(\alpha^*\) is the solution of the dual (4), \(N_{sv}\) represents the number of support vectors such that \(0 < \alpha < C\). A new sample is classified as \(+1\) or \(-1\) according to the finally decision function \(f(x) = \text{sgn}(w \cdot x + b)\).

2.2. Twin support vector machine (TWSVM)

Consider a binary classification problem of \(I_1\) positive points and \(I_2\) negative points \((I_1 + I_2 = I)\). Suppose that data points belong to positive class are denoted by \(A_i \in \mathbb{R}^{d \times n}\), where each row \(A_i \in \mathbb{R}^d\) represents a data point. Similarly, \(B \in \mathbb{R}^{d \times n}\) represents all the data.

---

(a) The original examples and discriminants; (b) the effect of measurement noises; (c) the result of robust model.

---

1 Although literature [40] uses the Sherman–Morrison–Woodbury formula [41] to simply the matrix inversion's computational complexity, it is still a difficult task when the dimension or sizes of data is very high.
points belong to negative class. For the linear case, the TWSVM [13] determines two nonparallel hyperplanes:
\[ f_+(x) = w_+^T x + b_+ = 0 \quad \text{and} \quad f_-(x) = w_-^T x + b_- = 0, \]
where \( w_+ \in \mathbb{R}^n, b_+ \in \mathbb{R} \), \( w_- \in \mathbb{R}^n, b_- \in \mathbb{R} \). Here, each hyperplane is closer to one of the two classes and is at least one distance from the other. A new data point is assigned to positive class or negative class depending upon its proximity to the two nonparallel hyperplanes. Formally, for finding the positive and negative hyperplanes, the TWSVM optimizes the following two respective QPPs:
\[
\begin{align*}
\min_{w_+, b_+} & \quad \frac{1}{2} \|Aw_+ + e_+ b_+\|^2 + c_1 \xi_+^2 \\
\text{s.t.} & \quad - (Aw_+ + e_+ b_+) + \xi_+ \geq e_-, \quad \xi_+ \geq 0
\end{align*}
\]
and
\[
\begin{align*}
\min_{w_-, b_-} & \quad \frac{1}{2} \|Aw_- + e_- b_-\|^2 + c_2 \xi_-^2 \\
\text{s.t.} & \quad (Aw_- + e_- b_-) + \xi_- \geq e_+, \quad \xi_- \geq 0,
\end{align*}
\]
where \( c_1, c_2 \geq 0 \) are the pre-specified penalty factors, \( e_+, e_- \) are vectors of ones of appropriate dimensions. By introducing the Lagrangian multipliers, the Wolfe duals of QPPs (7) and (8) can be represented as follows:
\[
\begin{align*}
\max_z & \quad e_+^T z - \frac{1}{2} z^T G (H^T H)^{-1} G z \\
\text{s.t.} & \quad 0 \leq z \leq c_1 e_+, \quad \beta \geq 0,
\end{align*}
\]
and
\[
\begin{align*}
\max_{\beta} & \quad e_-^T \beta - \frac{1}{2} \beta^T (Q^T Q)^{-1} \beta \\
\text{s.t.} & \quad 0 \leq \beta \leq c_2 e_+,
\end{align*}
\]
where \( G = [B e_+, L = [A e_+] P = [A e_+]^T \) and \( Q = [B e_-, \alpha \in \mathbb{R}^n, \beta \in \mathbb{R}^m \) are Lagrangian multipliers.

The non-parallel hyperplanes (6) can be obtained from the solutions \( z \) and \( \beta \) of (9) and (10) by
\[
\begin{align*}
\nu_1 &= - (H^T H)^{-1} G z \quad \text{where} \quad \nu_1 = [w_+^T, b_+]^T, \\
\nu_2 &= - (Q)^{-1} P \beta \quad \text{where} \quad \nu_2 = [w_-^T, b_-]^T.
\end{align*}
\]

For the nonlinear case, we can refer to the literature [13].

3. Robust twin support vector machine (R-TWSVM)

3.1. Linear R-TWSVM

We firstly give the formal representation of robust classification learning problem. Given a training set
\[ T = \{ (x_1, y_1), \ldots, (x_I, y_I) \}, \]
where \( y_i \in \{-1, 1\}, i = 1, \ldots, I \) and input set \( X \) is a sphere within \( r_I \) radius of the \( x_i \) center:
\[ X_i = \{ x_i \subset X | x_i + r_i u_i \}, \quad i = 1, \ldots, I, \quad \| u_i \| \leq 1, \]
where \( X_i \) is the true value of the training data, \( u_i \in \mathbb{R}^n, r_i \) is a given constant. The goal is to induce a real-valued function
\[ y = \text{sgn}(g(x)) \]
to infer the label \( y \) corresponding to any example \( x \) in \( \mathbb{R}^n \) space. Generally, such problem is caused by measurement errors, where \( r_i \) reflects the measurement accuracy.

In order to obtain the optimization decision function of (14), by introducing \( \frac{1}{2} \| w_+ \|^2, (7) \) can be written as the following robust optimization problem:
\[
\begin{align*}
\min_{w_-, b_- \xi} & \quad \frac{1}{2} \|w_+\|^2 + \sum_{i=1}^{l+1} \xi_i \\
\text{s.t.} & \quad - (w_+ \cdot X_i) + b_+ + \sum_{i=1}^{l+1} \xi_i \geq 0, \quad i = 1, \ldots, I
\end{align*}
\]

Since
\[
\min_{y_i, r_i (w \cdot u_i), \| u_i \| \leq 1} = - r_i \| w_+ \|
\]
problem (14) can be converted to
\[
\begin{align*}
\min_{w_+, b_+, \xi_i} & \quad \frac{1}{2} \|w_+\|^2 + \sum_{i=1}^{l+1} \xi_i \geq 0, \quad i = 1, \ldots, I
\end{align*}
\]
By introducing new variables \( t_1, t_2 \) and setting \( \| w_+ \| \leq t_1, \| (w_+ \cdot X_i) + b_+ + \sum_{i=1}^{l+1} \xi_i \| \leq t_2 \).

For replacing \( t_1^2, t_2^2 \) in the objective function (18), we introduce new variables \( u_i, v_i, t_1, t_2 \) and satisfy the linear constraints \( u_i + v_i = 1, i = 1, 2 \) and second order cone constraints \( \sqrt{u_i^2 + v_i^2} \leq u_i \). Therefore, problem (18) can be reformulated as the following second order cone program (SOCP):
\[
\begin{align*}
\min_{\theta_1} & \quad \frac{1}{2} (u_1 - v_1) + \frac{1}{2} (u_2 - v_2) + c_1 \sum_{i=1}^{l+1} \xi_i \\
\text{s.t.} & \quad - (w_+ \cdot X_i) + b_+ - r_i t_1 \geq - \xi_i, \quad i = 1, \ldots, I
\end{align*}
\]
where \( \theta_1 = [w_+^T, b_+^T, \xi_1, \ldots, \xi_{l+1}, u_1, v_1, u_2, v_2] \).

By the optimization theory [42], the dual problem of (19) can be expressed as

\[
\begin{align*}
\max_{\theta_2} & \quad \beta_1 + \beta_2 + \sum_{i=1}^{l+1} \beta_i \\
\text{s.t.} & \quad \beta_1 + z_{u_1} = \frac{1}{\theta_2}, \quad \beta_1 + 2 \beta_1 = - \frac{1}{\theta_2}, \quad \beta_2 + z_{u_2} = \frac{1}{\theta_2}, \quad \beta_2 + 2 \beta_2 = - \frac{1}{\theta_2}, \\
\sum_{i=1}^{l+1} \beta_i - \sum_{i=1}^{l+1} \beta_i = 0, \\
\sum_{i=1}^{l+1} \beta_i x_i - \sum_{j=1}^{l+1} \beta_j x_j \leq \sum_{i=1}^{l+1} r_i x_i - \gamma_i, \\
| \beta_1 | \leq - \gamma_2, \\
| \beta_2 | \leq - \gamma_2, \\
\sqrt{\gamma_1^2 + \gamma_2^2} \leq z_{u_1}, \\
\sqrt{\gamma_1^2 + \gamma_2^2} \leq z_{u_2}, \\
0 \leq \gamma_i \leq c_1, i = 1, \ldots, I.
\end{align*}
\]

where \( \theta_2 = [\xi, \beta_1, \beta_2, \gamma_1, \gamma_2, u_1, v_1, u_2, v_2, x_i] \).
Theorem 3.1. Suppose that $\theta^*_n$ is a solution of the dual problem (20), where $\theta^*_n = [x^*_1, x^*_2, \ldots, x^*_n, z^*_1, z^*_2, \ldots, z^*_n, z^*_n]$. If there exists $0 < \sigma < c_1$, we will obtain the solution $(w^*_n, b^*_n)$ of the primal problem (15):

$$\begin{align*}
w^*_n &= \frac{\gamma^*}{\sum_{i=1}^{l+1} r_i(x_i^* - z_i^*)^2} \left( \sum_{i=1}^{l+1} x_i^* x_i - \sum_{i=1}^{l+1} z_i^* x_i \right), \\
b^*_n &= -1 + \gamma^* r_j - \frac{\gamma^*}{\sum_{i=1}^{l+1} r_i(x_i^* - z_i^*)^2} \left( \sum_{i=1}^{l+1} x_i^*(x_i \cdot x_j) - \sum_{i=1}^{l+1} z_i^*(x_i \cdot x_j) \right).
\end{align*}$$

The proof of Theorem 3.1 can be found in the Appendix. Similarly, the dual of (10) can be written as

$$\max_{\theta_3} \beta_1 + \beta_2 + \sum_{i=1}^{l} x_i$$

s.t. $\beta_1 + z_{ix_i} = \frac{1}{2}$, $\beta_2 + z_{z_i} = -\frac{1}{2}$,

$$-\sum_{i=1}^{l} x_i - \sum_{i=1}^{l} z_i = 0,$$

$$\left\| \sum_{i=1}^{l} x_i x_i + \sum_{i=1}^{l} z_i z_i \right\| \leq \sum_{i=1}^{l} r_i(x_i - z_i^2),$$

$$\sqrt{l^2 + 2z_1^2} \leq z_{ix_i},$$

$$\sqrt{l^2 + 2z_2^2} \leq z_{z_i},$$

$$0 \leq z_i \leq c_2, i = 1, \ldots, l_1,$$

where $\theta_3 = [x^T, \beta_1, \beta_2, x^T, x^T, x^T, x^T]$. The corresponding solution is

$$\begin{align*}
w^*_n &= \frac{\gamma^*}{\sum_{i=1}^{l+1} r_i(x_i^* - z_i^*)^2} \left( -\sum_{i=1}^{l} x_i^* x_i - \sum_{i=1}^{l} z_i^* x_i \right), \\
b^*_n &= -1 + \gamma^* r_j - \frac{\gamma^*}{\sum_{i=1}^{l+1} r_i(x_i^* - z_i^*)^2} \left( -\sum_{i=1}^{l} x_i^*(x_i \cdot x_j) - \sum_{i=1}^{l+1} z_i^*(x_i \cdot x_j) \right).
\end{align*}$$

Once vectors $w_+, b_+$ and $w_-, b_-$ are obtained from (20) and (23), the separating planes $\omega^+, x + b_+ = 0$ and $\omega^-, x + b_- = 0$ are known. A new data point $x \in \mathbb{R}^n$ is then assigned to the positive or negative class, depending on which of the two hyperplanes given by (26) it lies closest to, i.e.

$$f(x) = \text{argmin}_{x_-} d_+(x,x),$$

where

$$d_+(x) = |w^+ x + b_+|, \quad d_-(x) = |w^- x + b_-|,$$

where $| \cdot |$ is the perpendicular distance of point $x$ from the planes $w^+ x + b_+$ and $w^- x + b_-$.  

3.2. Nonlinear $\mathcal{R}$-TWSVM

The above discussion is restricted to the linear case. Here, we will analyze nonlinear $\mathcal{R}$-TWSVM by introducing kernel function $K(x,x') = (\Phi(x) \cdot \Phi(x'))$, and the corresponding transformation:

$$\mathbf{x} = \Phi(x),$$

where $x \in \mathcal{H}$, $\mathcal{H}$ is the Hilbert space. So the training set (12) becomes

$$T = \{(x_i,y_i), \ldots, (x_i,y_i)\},$$

where $X_i = \{\Phi(x_i)\} \in$ is in the sphere of the radius $r$ and the center $x_i$. So when $\|x_i-x_i\| \leq r_i$ and choosing RBF, we have

$$\frac{\|x_i - x_i\|}{(\Phi(x_i) - (\Phi(x_i)) \cdot (\Phi(x_i) - (\Phi(x_i)))} = K(x_i,x_i) = 2 - 2 \exp(-\|x_i-x_i\|^2/2\sigma^2) \leq r_i^2,$$

where $r_i = \sqrt{2 - 2 \exp(-\|x_i-x_i\|^2/2\sigma^2)}$.

Thus $x_i$ becomes a sphere of the center $\Phi(x_i)$ and the radius $r_i$,

$$\mathbf{x} = \{x_i \| x_i - \Phi(x_i) \leq r_i \}.$$  

For nonlinear case of $\mathcal{R}$-TWSVM, $\sum_{i=1}^{l} x_i \Phi(x_i) - \sum_{i=1}^{h} x_i \Phi(x_i)^2$ can be expressed as

$$\sum_{i=1}^{l} x_i \Phi(x_i)^2 - 2 \sum_{i=1}^{h} x_i \Phi(x_i)^2 + \sum_{i=1}^{l} x_i \Phi(x_i)^2$$

Similarly, $\sum_{i=1}^{l} x_i \Phi(x_i)^2$ can be expressed as

$$\sum_{i=1}^{l} x_i \Phi(x_i)^2 + 2 \sum_{i=1}^{h} x_i \Phi(x_i)^2.$$ 

So we can easily obtain the nonlinear $\mathcal{R}$-TWSVM only by taking $K(x,x')$ instead of $(x \cdot x')$ of the optimization problem (20) and (23).

3.3. Discussion

3.3.1. Relationship with R-SVM

Both $\mathcal{R}$-TWSVM and R-SVM use the assumption of the sphere within $r$ to the missing or uncertain data and obtain the final classifier by solving the related SOCP problem. However, there are several differences as follows. (1) $\mathcal{R}$-TWSVM is approximately four times faster than the usual R-SVM. Now suppose the computational complexity is no more than $O(r^4 \log(e^{-1})) [37]$, where $c$ is parameters corresponding to a specific algorithm, and $r$ denotes the number of variables. R-SVM contains $l+4$ variables and each problem of $\mathcal{R}$-TWSVM contains $8 + l/2$ variables. Suppose $l = l = l/2$, the ratio $\rho$ of runtimes can be written as

$$\rho = \frac{(l+4)! \log(e^{-1})}{2(8 + l/2)! \log(e^{-1})} = \frac{(l+4)!}{2(8 + l/2)!}.$$ 

So when $l$ is large, $\rho$ is roughly 4. (2) $\mathcal{R}$-TWSVM uses two nonparallel hyperplanes for two classes to construct the final decision function, which factually exploits the data’s structural information while margin maximizing to the missing and uncertain data. This can directly improve the model’s generalized capability. (3) In fact, R-SVM is only efficient to the data with feature noise, which will lose its robustness when the data is with label noise. Unlike R-SVM, $\mathcal{R}$-TWSVM employs two different loss function: a quadratic loss function making the proximal hyperplane close enough to the class itself, and a soft-margin loss function making the hyperplane as far as possible from the other
class, which results that almost all the points in this class and some points in the other class contribute to each final decision function. This makes $\mathcal{R}$-TWSVM have stronger insensitivity to data with label noise.

### 3.3.2. Relationship with TWSVM

Consider the primal optimization (15) of $\mathcal{R}$-TWSVM, suppose $r_i = 0, i = 1, \ldots, l$, our model will degenerate to the TBSVM; dropping terms: $\frac{1}{2} \| w \|^2$ simultaneously, our model will degenerate to the TWSVM. Therefore, TWSVM and TBSVM are the special cases of our models.

Although TWSVMs have had great success in classification, it exists the following shortcomings: (1) TWSVMs have to solve two problems for linear case and two other problems for nonlinear case separately. In order to realize the structural risk minimization (SRM) principle, TWSVMs have to add different regularization terms for linear and nonlinear cases owing to the reason mentioned in Section 1. So far, there is not a clear theoretical explanation for these regularization terms, especially for the nonlinear case. (2) TWSVMs have to compute the inverse of matrices, it is in practice intractable or even impossible for a large data set by the classical methods.

In this paper, $\mathcal{R}$-TWSVM successfully overcomes the shortcomings above. There are only inner products about samples in the dual problems, this makes us no need to add the other regularization term $\| w \|^2 + b^2$ [39] for nonlinear case and no need to explain its meaning. Correspondingly, kernel trick can also be applied to our model directly due to the reason above. Furthermore, our model does not need to solve the extra inverse of matrices. More importantly, we can derive a new dual form of TWSVM by degenerating $\mathcal{R}$-TWSVM. For simplicity but without loss of generality, we only analyze one of dual problems of $\mathcal{R}$-TWSVM. Consider (20) and suppose $r_i = 0, i = 1, \ldots, l$, the optimization problem will become a standard quadratic programming problem as follows (for simplify, we omit the proof):

$$\max_{\theta} -\frac{1}{2} \left[ \sum_{i=1}^{l} \theta_i x_i \right] - \frac{1}{2} \sum_{i=1}^{l} \theta_i = 0, \quad 0 \leq \theta_i \leq c_i, \quad i = l + 1, \ldots, l,$$

(36)

where $\theta = [\theta_1^T, \theta_1^T]^T$.

The corresponding solution is

$$w^* = -\left( \sum_{l=1}^{1} \theta_1 x_i \right),$$

(37)

$$b^* = -1 + \left( \sum_{l=1}^{1} \theta_1 x_i \right).$$

(38)

According to Eq. (33), the kernel trick can also be applied to the model directly.

For the research of nonparallel classifiers, Mangasarian and Jayadeva et al. did a series of pioneering work. Up to now, hundreds of paper followed have been published. Unfortunately, none of them solves the fundamental disadvantages mentioned above. Encouragingly, the new model derived by $\mathcal{R}$-TWSVM successfully overcomes these shortcomings as well. This is also an important contribution in this paper.

### 4. Experiment

We compared $\mathcal{R}$-TWSVM against TWSVM and R-SVM on various data sets in this section. For simplicity, we set all $r_i$ in (13) to be a constant $r$ and data of all experiments are normalized to $-1$ and $1$. All codes were wrote in MATLAB 2010. The experiment environment: Intel Core i5 CPU, 2 GB memory. The SeDuMi$^2$ software is employed to solve the SOCP problems of T-SVM and $\mathcal{R}$-TWSVM. The “quadpro” function in MATLAB is used to solve the related optimization problems of TWSVM. The testing accuracies for our method are computed using standard 10-fold cross validation. The parameters $c_1, c_2$ and the RBF kernel parameter $\sigma$ are selected from the set $[2^\{i=7, \ldots, 7\}]$ by 10-fold cross validation on the tuning set comprising 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.

#### 4.1. Toy data

To give an intuitive performance of $\mathcal{R}$-TWSVM, we construct two sets of 2-D data: one is to test the influence of $r$ to the accuracy; the other is to show the difference between $\mathcal{R}$-TWSVM and R-SVM when some samples are labeled incorrectly.

The first data is generated randomly from two normal distribution $(u_1 = -0.5, \sigma_1 = 0.4, u_2 = 0.5, \sigma_2 = 0.4)$. The noise $u_i$ is generated randomly from the normal distribution and scaled on the unit sphere. We add many noises for the dataset by $x_i = x_i + r u_i$. From Figs. 2 and 3, it is not difficult to find that $\mathcal{R}$-TWSVM is trying to find a robust classifier which is able to both separate the existing data and all of virtual data included in the circle of radius $r$.

The second data is also generated randomly from two normal distribution mentioned above. Different with the first data, we add the noises of labels for the data randomly. Figs. 4 and 5 show that the $\mathcal{R}$-TWSVM is superior to R-SVM when there are the label errors of data. The main reason is because that the classifier of R-SVM is decided by the support vectors (SVs) and are easily disturbed when SVs are labeled incorrectly.

#### 4.2. Image datasets

Face recognition and image classification are two popular problems in pattern recognition. In this subsection, we test our proposed method in face recognition and image classification. The “1 vs r” method [42] is used to solve the multi-class classification. Because our goal is only to compare the performance between $\mathcal{R}$-TWSVM and other algorithms, all experiment are carried out on raw pixel features.

(1) AR Face Database (see Fig. 6) [43]: there are seven images of each individual: a neutral face, three variations in lighting (from the left, from the right, from both sides) and three variations in expression (smile, frown, scream). We, respectively, take 70 smile, frown, scream’s person aligned and its cropped faces as training set and 30 other persons as testing set. Each image is resized to be $80 \times 60$ pixels. For smile faces, we add Gaussian white noise of mean 0 and variance 0.01, 0.02, 0.03, 0.04; for frown faces, we add “salt and pepper” noise of noise density 0.05, 0.06, 0.07, 0.08; for scream faces, we add multiplicative noise of mean 0 and variance 0.01, 0.02, 0.03, 0.04. Fig. 7 shows the final result.

(2) PASCAL 2006 dataset (see Fig. 8) [44]: it contains 10 object categories (bicycles, buses, cats, cars, cows, dogs, horses,
motorbikes, people, sheep) and 5304 images. We use 2618 images as training data and others for testing. All images are resized to be gray images of $80 \times 100$. We make noises for the dataset by supposing some gray level of the image lost. In practice, for each image, we sort its gray levels by the frequency that each gray level appears and supposing some gray level lost from large to small, of which the related gray value is set to 0. The result can be find in Fig. 9.
For AR Face Database, the accuracy of the scream face is lower than smile and frown faces. This shows that the variation in expression of the face directly leads to the difficulty of classification. From Fig. 7, we also find that ”salt and pepper” noise has lower influence than Gaussian and multiplicative noise. This shows that the degree of the image destructed by “salt and pepper” noise is lower than other noises. For PASCAL 2006 dataset, the noise generated by losing some gray level indeed interferes the accuracy of these algorithms, which has a steadily dropping with the increase of the number of losing gray levels. Especially, results on the two image datasets show that \( R \)-TWSVM has obvious superiority to the other algorithms in

Fig. 3. The performance of \( R \)-TWSVM in the case of RBF case: (a) \( r = 0.02 \), (b) \( r = 0.03 \), (c) \( r = 0.04 \), (d) \( r = 0.05 \), (e) \( r = 0.06 \), (f) \( r = 0.07 \), (g) \( r = 0.08 \), (h) \( r = 0.09 \).
Fig. 4. The performance of R-TWSVM and R-SVM in label’s noise (the cyan dash dot line: the hyperplane of R-SVM; the red and blue solid line: the hyperplanes of R-TWSVM): (a) number of label’s errors $\frac{1}{2}$, (b) number of label’s errors $\frac{1}{3}$, (c) number of label’s errors $\frac{1}{4}$, (d) number of label’s errors $\frac{1}{5}$, (e) number of label’s errors $\frac{1}{6}$, (f) number of label’s errors $\frac{1}{7}$, (g) number of label’s errors $\frac{1}{8}$, (h) number of label’s errors $\frac{1}{9}$. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)
the most cases, which more likely implies that robust model based on nonparallel hyperplanes has better predictive ability than traditional R-SVM.

4.3. UCI datasets

In this subsection, we test our method from the UCI datasets [45]. They are, respectively, “Hepatitis”, “BUPA liver (Bliver)”, “Heart-Statlog (HStatlog)”, “Heart-c”, “Votes” and “WPBC”. For each dataset, we randomly select the same number of samples from different classes to compose a balanced training set and the noise is generated randomly from the normal distribution and scaled on the unit sphere. The results are given in Table 1.

From the results of Table 1, we can find that the performances of the R-TWSVM and robust SVM are better than that of the original TWSVM. This further validates the necessity of designing robust models for the data with noises. Furthermore, the accuracy of R-TWSVM is consistently better than robust SVM, especially for Votes dataset.

This also validates the classifier that uses two nonparallel hyperplanes for two classes has obviously advantages. Firstly, R-TWSVM essentially exploits the data’s structural information while margin maximizing. Secondly, in R-TWSVM, almost all the points in this class and some points in the other class contribute to each final decision function, which makes R-TWSVM have stronger insensitivity to missing or uncertain data with label noise to the missing and uncertain data.

Fig. 10 indicates the training time of R-TWSVM is faster than robust SVM. This is because that it solves two SOCP problems of a smaller size instead of a single SOCP of a very large size.

5. Conclusion

In this paper, we proposed a new Robust Twin Support Vector Machine (called R-TWSVM), which can deal with data with measurement noise efficiently. All experiments show that our method is superior to the traditional robust SVM in both computation time and classification accuracy. Remarkably, since there are only inner products about inputs in our dual problems, this
makes us apply kernel trick directly for nonlinear cases and does not need to solve the extra inverse of matrices, which is different with existing TWSVMs. In addition, we also give a new dual form of TWSVM by degenerating R-TWSVM, which successfully overcomes the existing shortcomings of TWSVM. In the future work, how to further accelerate the algorithm is under our consideration. In addition, the extension to semi-supervised learning and multi-class classification is also interesting.

Acknowledgment

This work has been partially supported by Grants from National Natural Science Foundation of China (Nos. 70921061, 10601064), the CAS/SAFEA International Partnership Program for Creative Research Teams, Major International (Regional) Joint Research Project (No. 71111010726), the President Fund of GUCAS.

Appendix A. The proof of Theorem 3.1

Proof. Introduce the dual problem’s Lagrange function

\[ L(\Theta) = -\beta_1 - \beta_2 - \sum_{i=1}^{l} \zeta_i - \sum_{i=1}^{l} \frac{1}{r_i} (z_i - \gamma_i) \]

\[ -w_i^{\top} \left( \sum_{i=1}^{l} z_i x_i - \sum_{i=1}^{l} \gamma_i x_i \right) - \sum_{i=1}^{l} \zeta_i x_i - u_1 \left( \beta_1 + z_{\text{mi}} - \frac{1}{2} \right) + u_1 \left( \beta_1 + z_{\text{mi}} + \frac{1}{2} \right) \]
\[ u_2 \left( \beta_2 + z_{d0} - \frac{1}{2} \right) + v_2 \left( \beta_2 + z_{d0} + \frac{1}{2} \right) - b \left( \sum_{i=1}^{l} z_i - h_i \right) \]

\[ - \sum_{i=l+1}^{n} \xi_i (C - z_i) - \sum_{i=l+1}^{n} \eta_i z_i - z_1 z_{d0} - z_2 z_1 - z_1 z_{d2} = 0, \]

where \( t_1, t_2, u_1, u_2, v_1, v_2, b, w_1, z_1, z_2, x_1, z_3, x_2, z_2, x_2, z_1, z_{d0}, z_{d2}, z_{d1} \in \mathbb{R} \), \( w_0, \xi, \eta \in \mathbb{R}^n \) are lagrange multipliers.

According to the KKT conditions in the infinite-dimensional space [36], we know that there exist lagrange multipliers satisfying:

\[ \nabla_x L = -1-t_1^* r_1 - (w_* - x_j) - b^* +\xi_j \eta_j^* = 0, \]  

\[ \nabla_{\beta} L = -1 + u_1^* + v_1^* = 0, \]  

\[ \nabla_{\beta} L = -1 + u_2^* + v_2^* = 0, \]  

\[ \nabla_{\gamma_1} L = t_1^* - z_1^* = 0, \]  

\[ \nabla_{\gamma_2} L = t_2^* - z_2^* = 0, \]  

\[ \nabla_{\gamma_3} L = u_1^* - z_1^* = 0, \]  

\[ \nabla_{\gamma_4} L = v_1^* - z_2^* = 0, \]  

\[ \nabla_{\gamma_5} L = u_2^* - z_2^* = 0, \]  

\[ \nabla_{\gamma_6} L = v_2^* - z_2^* = 0, \]  

\[ \nabla_{\gamma_7} L = x_1^* - z_1^* = 0, \]  

\[ \nabla_{\gamma_8} L = x_2^* - z_2^* = 0, \]  

\[ \nabla_{\gamma_9} L = (w_* - x_j) - w_{b1} + b^* = 0, \quad i = 1, \ldots, h, \]

\[ \eta_j^* \geq 0, \quad \xi_j^* \geq 0, \quad i = l_1 + 1, \ldots, l. \]

\[ 0 \leq z_i \leq c_1, \quad \eta_j^* z_j^* \geq 0, \quad \xi_j^* (c_1 - z_j^*) = 0, \quad i = l_1 + 1, \ldots, l. \]

\[ b_1^* + z_1^* = \frac{1}{2}, \]  

\[ b_2^* + z_2^* = -\frac{1}{2}, \]  

\[ b_3^* + z_3^* = \frac{1}{2}, \]  

\[ b_4^* + z_4^* = -\frac{1}{2}, \]  

\[ \sum_{i=1}^{l} z_i - \sum_{i=1}^{l} z_i^* = 0, \]

\[ [t_1^* w_1^*]^T \left[ \sum_{i=l+1}^{n} r_i x_i^* - \gamma_1^* \left( \sum_{i=1}^{h} x_i^* - \sum_{i=1}^{h} x_i \right) \right]^T = 0, \]

\[ [t_1^* w_1^*]^T \in L^{n+1}, \quad t_1^* \in \mathbb{R}, \quad w^* \in \mathbb{R}^n, \]

\[ [t^*_2 w^*_2]^T \in L^{n+1}, \quad t^*_2 \in \mathbb{R}, \quad w^*_2 \in \mathbb{R}^n, \]

\[ [z_1^* z_2^*]^T \in L^3, \quad [z_1^* z_2^*]^T \in L^3, \]

\[ \frac{1}{(\sum_{i=l+1}^{n} r_i x_i^* - \gamma_1^*)} \left( \sum_{i=1}^{h} x_i^* - \sum_{i=1}^{h} x_i \right). \]  

According to (45)-(48), (61) and (62) can be rewritten as:

\[ [u_1^* v_1^* t_1^*]^T \in L^3, \quad [u_2^* v_2^* t_2^*]^T \in L^3, \]

\[ \frac{u_1^* v_1^* t_1^*}{\left( \sum_{i=l+1}^{n} r_i x_i^* - \gamma_1^* \right)} \left( \sum_{i=1}^{h} x_i^* - \sum_{i=1}^{h} x_i \right). \]
By (40)–(62), (84)–(86), we may obtain
\[
\beta_1^* + \beta_2^* = \sum_{i=b+1}^n z_i^2 - \frac{1}{2} (u_1^* - v_1^*)^T + \frac{1}{2} (u_2^* - v_2^*)^T + c_1 \sum_{i=b+1}^n z_i^2.
\] (87)

Eq. (87) shows that the object function value of primal (15) about \(\Theta_1\) is equal to the one of dual (20) about \(\Theta_2\). According to Theorem 4 in [38], we can get \(w_1^* = t_1, w_2^* = t_2, \ldots, w_n^* = t_n\) is the solution to the prival problem. On the other hand, according to duality theory, we can obtain that \(w^*_n\) is the unique solution to primal (15). Hence Eq. (86) is proved.

If there exists a training point \(0 < c_i \leq c_i^*\) classified correctly, then according to (40), we obtain the corresponding
\[
b_i^* = -1 + \frac{1}{n} \sum_{i=1}^n \frac{z_i^2}{z_i^2 - C_1} \left( \sum_{i=1}^n \frac{z_i^2}{z_i^2} - \frac{1}{n} \sum_{i=1}^n \frac{z_i^2}{z_i^2} \right).
\] (88)

References


Zhiquan Qian received his Ph.D. from College of Science, China Agricultural University in 2011. Currently he is engaged in post-doctoral research in Research Center on Fictitious Economy and Data Science, Chinese Academy of Sciences. His research interests include object detection, object tracking, and machine learning.

Yingjie Tian is the Professor of Research Center on Fictitious Economy and Data Science, Chinese Academy of Sciences. He received his first degree in Mathematics (1994), Masters in Applied Mathematics (1997), Ph.D. in Management Science and Engineering. His research interests include support vector machines, optimization theory and applications, data mining.

Yong Shi currently serves as the Executive Deputy Director of Research Center on Fictitious Economy and Data Science, Chinese Academy of Sciences. He is also the Professor of the University of Nebraska, USA. Dr. Shi’s research interests include business intelligence, data mining, and multiple criteria decision making.