Twin SVM with a reject option through ROC curve

Dongyun Lin[a], Lei Sun[b], Kar-Ann Toh[c], Jing Bo Zhang[d], Zhiping Lin[a,*]

[a] *School of Electrical and Electronic Engineering, Nanyang Technological University, 639798, Singapore*
[b] *School of Information and Electronics, Beijing Institute of Technology, Beijing, 100081, PR China*
[c] *School of Electrical and Electronic Engineering, Yonsei University, Seoul 120-749, Korea*
[d] *AEBC, Nanyang Environment and Water Research Institute, Nanyang Technological University, Singapore*

ABSTRACT

This paper proposes a new method which embeds a reject option in twin support vector machine (RO-TWSVM) through the Receiver Operating Characteristic (ROC) curve for binary classification. The proposed RO-TWSVM enhances the classification robustness through inclusion of an effective rejection rule for potentially misclassified samples. The method is formulated based on a cost-sensitive framework which follows the principle of minimization of the expected cost of classification. Extensive experiments are conducted on synthetic and real-world data sets to compare the proposed RO-TWSVM with the original TWSVM without a reject option (TWSVM-without-RO) and the existing SVM with a reject option (RO-SVM). The experimental results demonstrate that our RO-TWSVM significantly outperforms TWSVM-without-RO, and in general, performs better than RO-SVM.

Keywords: Twin SVM, Reject Option, ROC Curve, Binary Classification

1. Introduction

Classifiers incorporating a reject option are adopted in many applications including biometric verification [1], automated medical diagnosis [2], genetic engineering [3] and image categorization [4], etc. The main purpose of embedding a reject option is to improve the robustness of classifiers over uncertainties. Unlike standard classifiers which assign each testing sample to a specific class, classifiers with a reject option withhold or abstain making the decision

[*] Corresponding author: Tel.: +65 6790 6857; Fax.: +65 6793 3318;
e-mail: EZPLIN@ntu.edu.sg (Zhiping Lin)

if the sample is in a region with low classification confidence. Such region is called the reject region. Classifiers with a reject option are thus called abstaining classifiers [5] or selective classifiers [6]. In practice, if a sample is labelled by a classifier with a very low confidence, it is appropriate to identify it as 'rejected' rather than directly assigning it with a categorical label. This is particularly useful in automated medical diagnosis where the consequence of a misclassification can be very harmful [2]. For instance, a wrong diagnosis of a terminal illness in either way could both lead to a disastrous reaction of the patient. For such cases, additional human investigations or observations can be conducted to avoid misclassification. From an automated classification perspective, another classifier or new features can be explored on those rejected samples to increase the classification accuracy. In medical testing, the additional sampling for further testing should not be as costly as that from making decisions with low confidence.

Classification with a reject option has been well discussed in the literature. Among the early works, Chow [7, 8] developed a Bayesian optimal decision rule with a reject option based on posterior probabilities. Given the posterior probabilities, Chow's rule is optimal in view of minimization of the error rate with respect to a predefined rejection rate. A reject option has been embedded in state-of-the-art classification methods such as the support vector machine (SVM) [9, 10]. However, Chow's optimal rejection rule [8] cannot be directly applied to SVM because the original SVM does not output posterior probabilities. Several methods were proposed to embed a reject option by manipulating the cost function of SVM [11, 12, 13]. However, since these methods constructed the reject region during the training phase, the training procedure had to be re-calculated if the cost settings are changed. In [14], Tortorella proposed an elegant method to embed a reject region for SVM (RO-SVM) using the Receiver Operating Characteristic (ROC) curve and proved that this method can be applied to a general dichotomizer [15]. In this method, the reject region is determined after the training procedure. Therefore, it does not require another round of training if the cost settings are changed. In [16], Santos-Pereira and Pires theoretically proved the equivalence between the ROC based reject rule and Chow's optimal reject rule. In [17], the ROC based method is extended to multi-categorical classification. Classifiers with a reject option are recently developed in multi-label classification problems [18].

Twin Support Vector Machine (TWSVM) is a novel SVM variant [19, 20]. Comparing with the original SVM, TWSVM is computationally more efficient [19] and has better generalization performance for imbalance data classification [21]. This is due to the flexibility offered by the nonparallel hyperplanes trained with TWSVM. Motivated by these observations, in this work, we propose a new method of embedding TWSVM with a reject option. Particularly, (i) a TWSVM with a reject option (RO-TWSVM) is proposed through the ROC based method for RO-SVM [14]. (ii) Two synthetic data sets are applied to demonstrate the proposed RO-TWSVM can generate different reject regions for different cost settings and outperform the existing RO-SVM [14] for certain type of data like “Cross Planes” [22]. (iii) Finally, extensive statistical tests, namely, the Wilcoxon rank sum test, the Friedman test, and the Nemenyi post-hoc test are conducted on multiple real-world benchmark data sets to compare the proposed RO-TWSVM with several related methods.

The main novelties of the proposed RO-TWSVM are summarized as follows: (i) The utilization of the distance difference between the sample and the two hyperplanes trained by TWSVM rather than the distance to one hyperplane of the SVM [14] to better measure the classification uncertainty of the sample; (ii) The generation of flexible and robust multidirectional reject regions which are not necessary parallel to the separating hyperplane as that in RO-SVM [14]. These contributions yield a flexible learning mechanism more than a simple combination of [19] (or [20]) and [14].

The remaining parts of this paper are organized as follows: Section 2 briefly reviews the original TWSVM. In Section 3, we present the formulation of the proposed RO-TWSVM. In Section 4, the performance of RO-TWSVM is evaluated on two synthetic data sets. We conduct three statistical tests using multiple real-world benchmark data sets to compare the RO-TWSVM with the competing methods in Section 5. Section 6 concludes our work.

2. Twin support vector machine

Essentially, Twin Support Vector Machine (TWSVM) [19] combines the spirit of the generalized eigenvalue proximal support vector machine (GEPSVM) [22] with the original SVM optimization formulation. It trains for nonparallel hyperplanes around which the samples of the corresponding class are clustered. The resulted hyperplanes have sparse representations similar to that of SVM’s hyperplane [19]. For binary classification, let the positive and negative labelled training samples be represented as the rows of matrices $\boldsymbol{A}$ and $\boldsymbol{B}$, respectively [19]. The dimensions of $\boldsymbol{A}$ and $\boldsymbol{B}$ are $m_1 \times n$ and $m_2 \times n$, where m_1 and m_2 are respectively the number of positive and negative training samples and n denotes the number of features of a sample. Two nonparallel hyperplanes are parameterized as

$$
\begin{aligned}
\boldsymbol{x}^{\boldsymbol{T}}\boldsymbol{w}^{\mathbf{(1)}} + b^{(1)} = 0 \\
\boldsymbol{x}^{\boldsymbol{T}}\boldsymbol{w}^{\mathbf{(2)}} + b^{(2)} = 0
\end{aligned}
\tag{1}
$$

where $\boldsymbol{x}$ denotes the feature vector of a sample, $[\boldsymbol{w}^{\mathbf{(1)}}, b^{(1)}]$ and $[\boldsymbol{w}^{\mathbf{(2)}}, b^{(2)}]$ denote the parameters for the hyperplanes corresponding to the positive and negative classes, respectively.

The two primal quadratic programming problems (QPPs) formulated in [19] are given by

$$
\begin{aligned}
&(TWSVM1) \\
&\underset{\boldsymbol{w}^{\mathbf{(1)}},\, b^{(1)},\, \boldsymbol{q}}{\text{minimize}} \quad \tfrac{1}{2}(\boldsymbol{A}\boldsymbol{w}^{\mathbf{(1)}} + \mathbf{1_1}b^{(1)})^T(\boldsymbol{A}\boldsymbol{w}^{\mathbf{(1)}} + \mathbf{1_1}b^{(1)}) + c_1\mathbf{1_2}^{\boldsymbol{T}}\boldsymbol{q} \\
&\text{subject to} \quad -(\boldsymbol{B}\boldsymbol{w}^{\mathbf{(1)}} + \mathbf{1_2}\mathrm{b}^{(1)}) + \boldsymbol{q} \geq \mathbf{1_2}, \boldsymbol{q} \geq \mathbf{0}
\end{aligned}
\tag{2}
$$

$$\begin{array}{ll} (TWSVM2) & \\ \underset{\boldsymbol{w^{(2)}},\ b^{(2)},\ \boldsymbol{q}}{\text{minimize}} & \frac{1}{2}(\boldsymbol{Bw^{(2)}} + \boldsymbol{1_2}b^{(2)})^T(\boldsymbol{Bw^{(2)}} + \boldsymbol{1_2}b^{(2)}) + c_2\boldsymbol{1_1{}^T q} \\ \text{subject to} & (\boldsymbol{Aw^{(2)}} + \boldsymbol{1_1}\mathrm{b}^{(2)}) + \boldsymbol{q} \geq \boldsymbol{1_1}, \boldsymbol{q} \geq \boldsymbol{0} \end{array} \tag{3}$$

where $\boldsymbol{1_1}$ and $\boldsymbol{1_2}$ are vectors of ones of appropriate dimensions. c_1 and c_2 are regularization parameters. All inequalities between vectors are componentwise. Essentially, the Twin Support Vector Machine (TWSVM) [19] is proposed to solve the binary classification problem by generating two nonparallel hyperplanes such that each hyperplane is closer to one of the two classes and is as far as possible from the other. The first term of the cost function of (2) (similarly for (3)) is formulated to minimize the sum of the squared distances from the hyperplane to samples of one class and the constraints require the hyperplane to be at a distance at least 1 from the samples of the other class. The second term of the cost function of (2) penalizes the violation of such constraints [19]. By Lagrangian formulation for the primal problems (2) and (3) and applying Karush-Kuhn-Tucker (K.K.T) conditions, the Wolfe dual problems [23] are obtained as:

$$\begin{array}{ll} (DTWSVM1) & \\ \underset{\boldsymbol{\alpha}}{\text{maximize}} & \boldsymbol{1_2}^T\boldsymbol{\alpha} - \frac{1}{2}\boldsymbol{\alpha}^T\boldsymbol{G}(\boldsymbol{H}^T\boldsymbol{H})^{-1}\boldsymbol{G}^T\boldsymbol{\alpha} \\ \text{subject to} & 0 \leq \boldsymbol{\alpha} \leq \mathrm{c}_1 \end{array} \tag{4}$$

$$\begin{array}{ll} (DTWSVM2) & \\ \underset{\boldsymbol{\gamma}}{\text{maximize}} & \boldsymbol{1_1}^T\boldsymbol{\gamma} - \frac{1}{2}\boldsymbol{\gamma}^T\boldsymbol{P}(\boldsymbol{Q}^T\boldsymbol{Q})^{-1}\boldsymbol{P}^T\boldsymbol{\gamma} \\ \text{subject to} & 0 \leq \boldsymbol{\gamma} \leq \mathrm{c}_2 \end{array} \tag{5}$$

where $\boldsymbol{\alpha}$ and $\boldsymbol{\gamma}$ are Lagrangian multipliers, $\boldsymbol{H} = \boldsymbol{P} = [\boldsymbol{A} \quad \boldsymbol{1_1}]$ and $\boldsymbol{G} = \boldsymbol{Q} = [\boldsymbol{B} \quad \boldsymbol{1_2}]$ and the inequality constraints in (4) and (5) are for each component of the vectors $\boldsymbol{\alpha}$ and $\boldsymbol{\gamma}$, respectively, i.e., each component of these vectors is between 0 and c_1 (or c_2). When the data are linearly independent, $(\boldsymbol{H}^T\boldsymbol{H})^{-1}$ and $(\boldsymbol{Q}^T\boldsymbol{Q})^{-1}$ are both positive definite. However, they can be ill-conditioned in some situations [19]. In the original formulation of TWSVM, a regularization term is introduced (i.e., make $\boldsymbol{H}^T\boldsymbol{H}$ as $\boldsymbol{H}^T\boldsymbol{H}+\varepsilon\boldsymbol{I}$, $\varepsilon > 0$) to handle the possible ill-conditioning. Solving the optimization problems leads to the nonparallel hyperplanes defined in (1). In the testing phase, a testing sample is assigned to the class whose hyperplane is closer to the sample, *i.e.*,

$$c = \underset{l=1,2}{\arg\min} \frac{\left|\boldsymbol{x}^T\boldsymbol{w^{(l)}} + b^{(l)}\right|}{\left\|\boldsymbol{w^{(l)}}\right\|_2}. \tag{6}$$

TWSVM can be extended to a nonlinear classifier by considering surfaces induced by an appropriate kernel function K:

$$\begin{array}{l} K(\boldsymbol{x}^T, \boldsymbol{C}^T)\boldsymbol{u}^{(1)} + b^{(1)} = 0 \\ K(\boldsymbol{x}^T, \boldsymbol{C}^T)\boldsymbol{u}^{(2)} + b^{(2)} = 0 \end{array} \tag{7}$$

where $\boldsymbol{C} = \begin{bmatrix} \boldsymbol{A} \\ \boldsymbol{B} \end{bmatrix}$. For any $\boldsymbol{A} \in \boldsymbol{R}^{m\times n}$ and $\boldsymbol{B} \in \boldsymbol{R}^{n\times k}$, the kernel function $K(\boldsymbol{A}, \boldsymbol{B})$ maps

$\boldsymbol{R}^{m\times n} \times \boldsymbol{R}^{n\times k}$ into $\boldsymbol{R}^{m\times k}$ [22]. The parameters $\boldsymbol{u}^{(l)}$ and $b^{(l)}$ $(l = 1, 2)$ are determined through quadratic programming. The kernel tricks on TWSVM were well studied in [24].

3. Proposed TWSVM with a reject option (RO-TWSVM) through ROC curve

Essentially, TWSVM encodes discriminative information through distances from samples to different hyperplanes of the corresponding classes. For binary classification, the decision rule is based on such distance difference, *i.e.*,

$$d(\boldsymbol{x}) = d_-(\boldsymbol{x}) - d_+(\boldsymbol{x}) \tag{8}$$

where $d_+(\boldsymbol{x})$ and $d_-(\boldsymbol{x})$ are the distances from the sample $\boldsymbol{x}$ to the positive and the negative hyperplanes, respectively. They are defined as:

$$d_+(\boldsymbol{x}) = \frac{\left|\boldsymbol{x}^T\boldsymbol{w}^{\boldsymbol{(1)}} + b^{(1)}\right|}{\left\|\boldsymbol{w}^{\boldsymbol{(1)}}\right\|_2} \tag{9}$$

$$d_-(\boldsymbol{x}) = \frac{\left|\boldsymbol{x}^T\boldsymbol{w}^{\boldsymbol{(2)}} + b^{(2)}\right|}{\left\|\boldsymbol{w}^{\boldsymbol{(2)}}\right\|_2} \tag{10}$$

where $\left\|\cdot\right\|_2$ is the Euclidean norm and $\boldsymbol{w}^{\boldsymbol{(l)}}, b^{(l)}, l = 1, 2$, are defined in (1).

Without the rejection setting, the original TWSVM classification rule [19] is:

$$\begin{cases} d(\boldsymbol{x}) \geq 0 & \text{positive class} \\ d(\boldsymbol{x}) < 0 & \text{negative class} \end{cases} \tag{11}$$

To enhance the classification robustness of the original TWSVM, we determine a reject region for TWSVM delimited by two thresholds t_1 and t_2 $(t_2 > t_1)$ for $d(\boldsymbol{x})$.

Hence, the classification rule for RO-TWSVM is formulated as

$$\begin{cases} d(\boldsymbol{x}) > t_2 & \text{positive class} \\ d(\boldsymbol{x}) < t_1 & \text{negative class} \\ t_1 \leq d(\boldsymbol{x}) \leq t_2 & \text{rejection} \end{cases} \tag{12}$$

This formulation is reasonable since the reject region is defined as the region between the hyperplanes where samples have similar distances to both hyperplanes. The ambiguity of samples can be reflected by the $d(\boldsymbol{x})$. Fig. 1 gives an illustration of a simple 2D example, where two black dashed-dotted lines represent the hyperplanes (lines) trained by TWSVM. The region delimited by two blue dashed lines is the reject region and the black diamond represents a sample being rejected.

The optimal decision thresholds t_1 and t_2 can be determined following the principle of minimization of the expected cost. The expected cost is defined according to the cost settings shown in Table 1. In Table 1, P and N denote the sets of positive and negative class, respectively. CFP, CFN and CR are the costs of false positive errors, false negative errors and rejection, respectively. CTP and CTN are the costs of true positive and true negative,

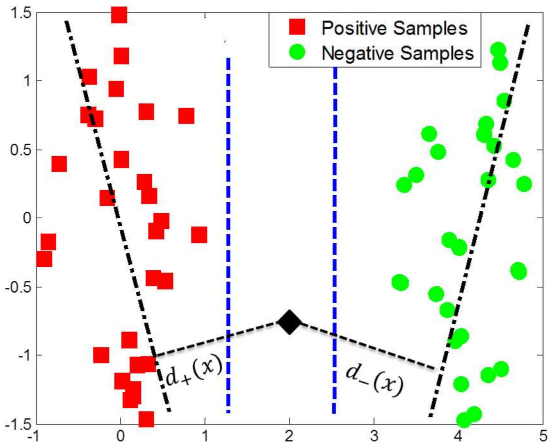


Fig. 1. The reject region for a 2D example. The blue dashed lines delimit the rejection boundaries and the black dashed-dotted lines represent nonparallel hyperplanes trained by TWSVM.

Table 1. Cost matrix.

		Predicted Class		
		N	P	Reject
True Class	N	CTN	CFP	CR
	P	CFN	CTP	

respectively. Normally, CTP and CTN are zero or possibly negative whereas CFP, CFN and CR are positive [14]. The value of CR should be less than CFP and CFN. This condition ensures that embedding a reject option is necessary in terms of reduction of the expected cost. Otherwise, the reject option should not be activated since it costs more than misclassification (CFP and CFN).

Referring to Table 1 and Fig. 1, the expected cost $EC(t_1, t_2)$ can be formulated with thresholds t_1 and t_2 [14]:

$$\begin{aligned} EC(t_1,t_2) &= p(P)\cdot CFN\cdot FNR(t_1) + p(N)\cdot CTN\cdot TNR(t_1) \\ &+ p(P)\cdot CTP\cdot TPR(t_2) + p(N)\cdot CFP\cdot FPR(t_2) \\ &+ p(P)\cdot CR\cdot RP(t_1,t_2) + p(N)\cdot CR\cdot RN(t_1,t_2) \end{aligned} \tag{13}$$

and

$$\begin{array}{ll} TPR(t) = \int_t^{+\infty}\varphi_+(\omega)d\omega; & FNR(t) = \int_{-\infty}^{t}\varphi_+(\omega)d\omega \\ TNR(t) = \int_{-\infty}^{t}\varphi_-(\omega)d\omega; & RN(t_1,t_2) = \int_{t_1}^{t_2}\varphi_-(\omega)d\omega \\ FPR(t) = \int_t^{+\infty}\varphi_-(\omega)d\omega; & RP(t_2,t_2) = \int_{t_1}^{t_2}\varphi_+(\omega)d\omega \end{array} \tag{14}$$

where FNR , TNR, TPR, FPR are the false negative rate, the true negative rate, the true positive rate and the false positive rate while RP and RN are the rejection rates on the positive and the negative samples, respectively. $p(P)$ and $p(N)$ are the prior probabilities of the positive and negative classes, respectively. $\varphi_+(\omega) = p(d(\boldsymbol{x}) = \omega | \boldsymbol{x} \in P)$ and $\varphi_-(\omega) = p(d(\boldsymbol{x}) = \omega | \boldsymbol{x} \in N)$ are class-conditional probability density functions of $d(\boldsymbol{x})$ which is the distance difference from TWSVM defined in (8). In [14], $\varphi_+(\omega)$ and $\varphi_-(\omega)$ are calculated based on the output of SVM, whereas here they are based on the distance difference $d(\boldsymbol{x})$ of TWSVM.

Following the derivations in [14], the expected cost in (13) can be rewritten as:

$$EC(t_1, t_2) = \varepsilon_2(t_2) - \varepsilon_1(t_1) + p(P) \cdot CFN + p(N) \cdot CTN \tag{15}$$

where

$$\begin{aligned} \varepsilon_1(t_1) &= p(P) \cdot CFN^* \cdot TPR(t_1) + p(N) \cdot CTN^* \cdot FPR(t_1) \\ \varepsilon_2(t_2) &= p(P) \cdot CTP^* \cdot TPR(t_2) + p(N) \cdot CFP^* \cdot FPR(t_2) \end{aligned} \tag{16}$$

and

$$\begin{aligned} CTP^* = CTP - CR, \quad CTN^* = CTN - CR \\ CFN^* = CFN - CR, \quad CFP^* = CFP - CR \end{aligned} \tag{17}$$

To minimize $EC(t_1, t_2)$, since the term $p(P) \cdot CFN + p(N) \cdot CTN$ does not depend on t_1 and t_2, minimization of the expected cost $EC(t_1, t_2)$ is equivalent to:

$$\begin{aligned} \underset{t_1, t_2}{\text{minimize}} \quad & \varepsilon_2(t_2) - \varepsilon_1(t_1) \\ \text{subject to} \quad & \mathrm{t_1 < t_2} \end{aligned} \tag{18}$$

To solve problem (18), since $\varepsilon_1(t_1)$ only depends on t_1 and $\varepsilon_2(t_2)$ depends on t_2, we can maximize $\varepsilon_1(t_1)$ and minimize $\varepsilon_2(t_2)$ simultaneously. The optimization problem can be formulated as:

$$\begin{aligned} \underset{t_1}{\text{maximize}} \quad & \varepsilon_1(t_1) \\ \underset{t_2}{\text{minimize}} \quad & \varepsilon_2(t_2) \\ \text{subject to} \quad & \mathrm{t_1 < t_2} \end{aligned} \tag{19}$$

The values of $\varepsilon_1(t_1)$ and $\varepsilon_2(t_2)$ are closely related to TPR and FPR which are the coordinates in the ROC curve. Hence, the optimization problem can be solved geometrically by finding the most appropriate operating points in the ROC curve. Specifically, t_1 and t_2 are determined by searching the tangential intersection points between the level curves $\varepsilon_1(t_1)$ and $\varepsilon_2(t_2)$ with the slopes (defined by (20), (21)) and the convex hull of the ROC curve [14] as shown in Fig. 2. Therefore, they are computed numerically by geometric properties and have no analytic expressions. The two level curve slopes m_1 and m_2 can be calculated based on the level curve (straight line) equations defined in (16) as:.

$$m_1 = -\frac{p(N) \cdot CTN^*}{p(P) \cdot CFN^*} \tag{20}$$

$$m_2 = -\frac{p(N) \cdot CFP^*}{p(P) \cdot CTP^*} \tag{21}$$

In [14], a necessary condition (22) for the existence of reject option is derived from the constraint $t_1 < t_2$ (see Fig. 3):

$$\frac{CTN \cdot CTP - CFN \cdot CFP}{(CTN + CTP) - (CFN + CFP)} > CR \tag{22}$$

Condition (22) is necessary but not sufficient since two level curves may intersect at the same point in the convex hull of the ROC curve (shown in Fig. 4). Therefore, the feasibility of t_1 and t_2 should be checked twice. The first is to confirm if condition (22) is satisfied and the second is to check if $t_1 = t_2$ after geometrical searching procedure.

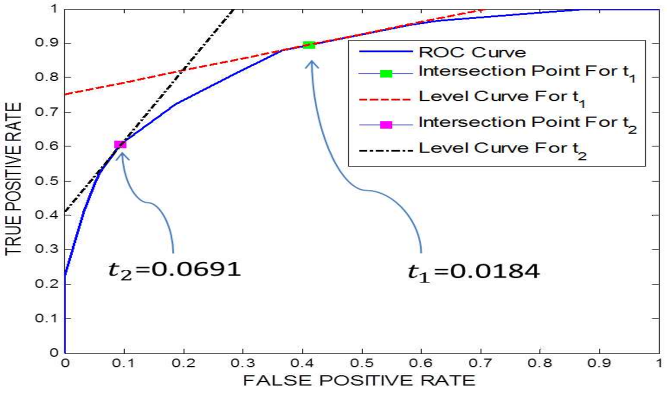


Fig. 2. The case when condition (22) is satisfied and two level curves do not intersect at the same point with the convex hull of the ROC curve.

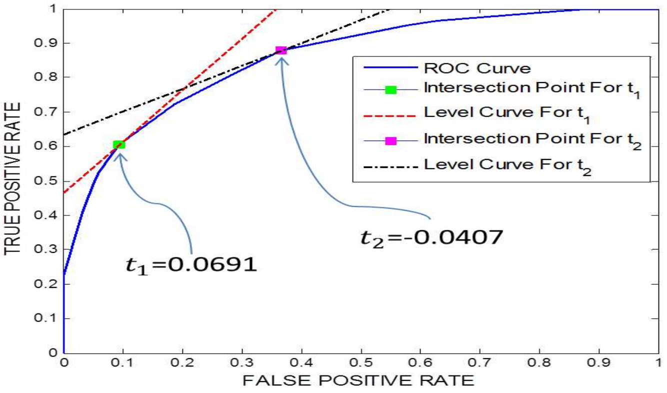


Fig. 3. The case when condition (22) is violated.

Algorithm 1 describes the processing steps of RO-TWSVM. In general, the algorithm consists of two stages. Firstly, a standard TWSVM model is trained over a set of training data. Secondly, a reject region is determined using the validation data based on the ROC curve of the trained TWSVM model and the cost settings.

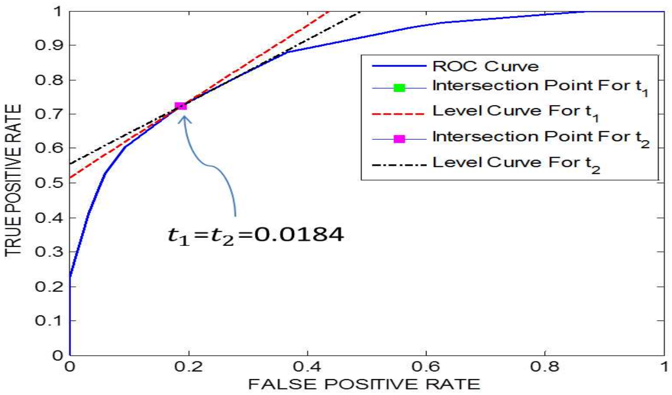


Fig. 4. The case when condition (22) is satisfied but two level curves intersect at the same point with the convex hull of the ROC curve.

In the first stage, since the RO-TWSVM firstly trains an original TWSVM model for classification without cost consideration, a standard cross validation is adopted for kernel selection and parameter tuning. For both the linear kernel and the RBF kernel, the parameters c_1 and c_2 of TWSVM are obtained through a search in the range of $[2^{-8}, 2^{8}]$. For the RBF kernel, the kernel width σ is obtained through a search in the range of $[2^{-8}, 2^{8}]$. These parameters are determined through the 5-fold cross validation (see details in [19, 24]).

In the second stage, to determine the reject region, we firstly check if condition (22) is satisfied, otherwise the reject option is not activated and we record this case as “RO is not applicable”. If condition (22) is satisfied, we find the tangential intersection points between the level curves (defined in (16)) and the convex hull of the ROC curve. From these intersection points we obtain the corresponding thresholds t_1 and t_2. Subsequently, we check if the constraint $t_1 < t_2$ is satisfied. If it is satisfied, the reject region is determined and the rule defined in (12) is applied. Otherwise, the case is also recorded as “RO is not applicable”.

Remark. The ROC based method to determine the optimal thresholds t_1 and t_2 for RO-TWSVM is similar to RO-SVM [14]. On the one hand, some advantages of RO-SVM are retained in the proposed RO-TWSVM, such as the training for TWSVM hyperplanes and the plot of ROC curve are implemented only once since changes made on cost settings, if any, only influence the slopes of the level curves. Hence, excessive re-calculations for the TWSVM hyperplanes and the ROC curve are not required. On the other hand, RO-TWSVM does not assume that samples being rejected should stay close to the separating hyperplane as RO-SVM does. Since we choose the distance difference in TWSVM to measure the uncertainty of a sample, an outlier which is far away from the SVM hyperplane and not rejected by RO-SVM could still be rejected by the proposed RO-TWSVM.

The effectiveness and advantages of RO-TWSVM will be demonstrated through the experiments on both synthetic and real-world benchmark data sets in the subsequent sections.

Algorithm 1: The method description of RO-TWSVM.

```
Input:
Data: $\boldsymbol{D} = \{(\boldsymbol{x_i}, y_i)\}_{i=1}^{N}$;
Costs: $CTN$,$CFP$,$CFN$,$CTP$,$CR$;
Output:
Reject region: $[t_1, t_2]$ or Message: “RO is not applicable”;
if condition (22) is not satisfied then
    return Message: “RO is not applicable”;
else
    Randomly divide $D = \{(\boldsymbol{x_i}, y_i)\}_{i=1}^{N}$ into a training set $P_{\bar{k}}$ and a validation set $P_k$,
      respectively.
    Use the samples in the training set $P_{\bar{k}}$ to train for a TWSVM model: $\boldsymbol{w^{(1)}}$, $\boldsymbol{w^{(2)}}$, $b^{(1)}$, $b^{(2)}$.
    Use the samples in the validation set $P_k$ to generate the ROC curve.
    Determine $t_1$ and $t_2$ geometrically through the ROC curve;
    if $t_1 < t_2$ then
        return Reject region: $[t_1, t_2]$;
    else
        return Message: “RO is not applicable”;
```

(i): TWSVM training consists of kernel selection and regularization parameter tuning. These parameters are determined through the cross validation.

(ii): Dertermine t_1 and t_2 by searching the intersection points between the level curves and the convex hull of the ROC curve. See details in [14].

4. Simulation on synthetic data sets

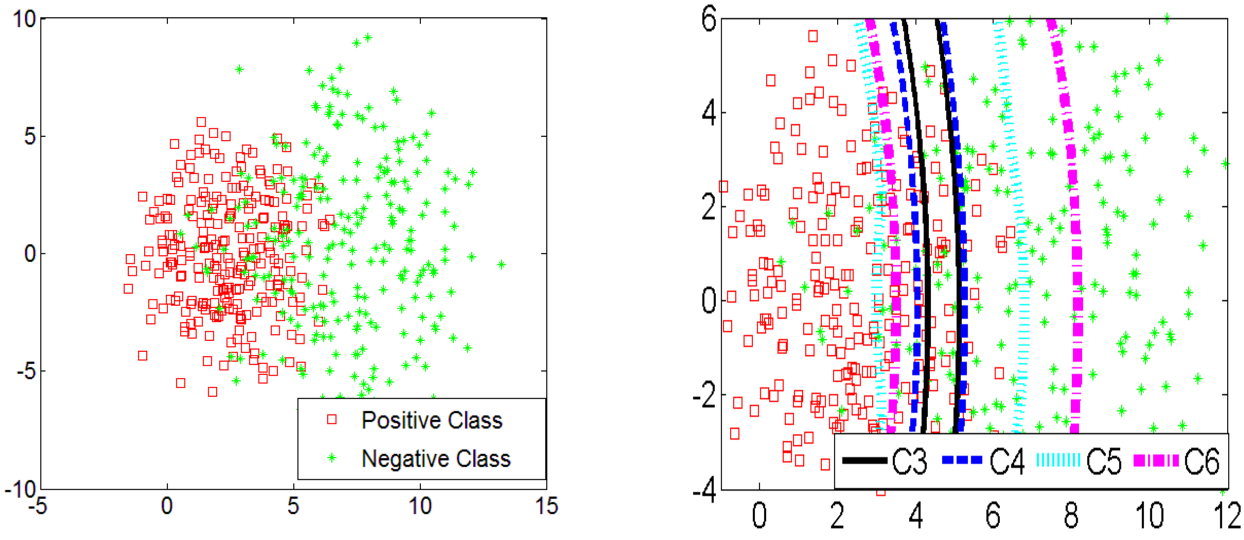
In this section, two synthetic data sets are simulated to observe the behavior of the proposed method. The simulation setup here is similar to that in RO-SVM [14] on synthetic data set 1 [14] to observe the variation of the reject region generated by the proposed RO-TWSVM with respect to various cost settings. TWSVM formulates a hyperplane for samples of each class where better flexibility than that of SVM could be anticipated. Hence, the goal of the second synthetic data set is to illustrate the merit of this flexibility leading to more reasonable reject regions by RO-TWSVM than those by RO-SVM. In our experiments, the prior probabilities for the two classes are assumed to be equal. Hence, 50% of samples are randomly chosen for training and validation in Procedure 1 . The remaining 50% of samples are used for testing. Among the training and validating samples, for the RO-TWSVM formulation described in Algorithm 1, the training set $P_{\bar{k}}$ contains 75% of the samples and the validation set P_k contains the remaining 25% for determining the optimal thresholds t_1 and t_2.

4.1. Synthetic data 1: uniformly distributed data with Gaussian noise

To demonstrate that the reject regions of RO-TWSVM can adapt to different cost settings, the proposed RO-TWSVM is implemented on a synthetic data set according to [14]. As shown in Fig. 5(a), this data set consists of 720 samples being equally distributed between the two classes. The samples are uniformly distributed with unit variance and spherical Gaussian noise along two $2\pi/3$ arcs with radii 6.2 and 10.0 for the positive and negative classes, respectively.

Table 2. The cost models for synthetic data 1.

	CTP	CFP	CTN	CFN	CR
C1	-20	22	-10	18	4
C2	-20	20	-10	18	2
C3	-14	22	-10	24	2
C4	-8	26	-10	28	2
C5	-4	36	-6	120	2
C6	-6	120	-4	36	2



(a)

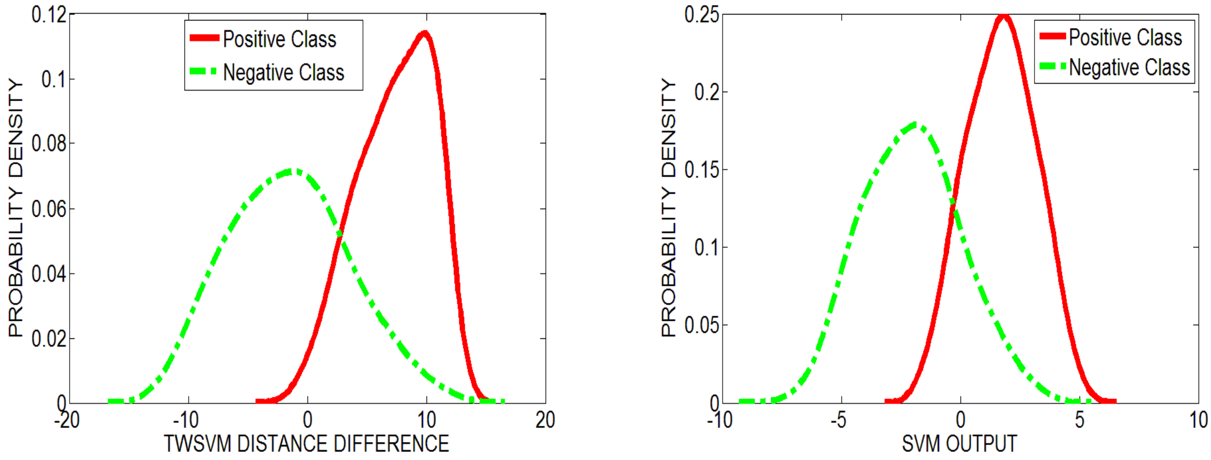
(b)

Fig. 5. (a) Synthetic data set 1, (b) The reject region boundaries of RO-TWSVM for each cost model: C3 (black solid line), C4(blue dashed line), C5 (cyan dotted line) and C6 (magenta dashed-dotted line).

The purpose of this example is to show the variation of reject regions generated by the RO-TWSVM under different cost models. An RBF kernel is adopted and the cost settings applied are shown in Table 2. These settings follow those recommended in [14] for RO-SVM. An illustration with several reject regions corresponding to the cost settings in Table 2 are presented in Fig. 5(b). In Fig. 5(b), there is no reject region generated for cost model C1 and C2. The reasons are that C1 violates condition (22) and the two level curves intersect at the same point for C2. From Table 2, it is observed that C3 and C4 represent the cases that $CTP \approx CTN$ and $CFP \approx CFN$ which correspond to applications with balanced costs. Consequently, for these two cost models, the generated reject regions show no rejection preference to either class. In contrary, C5 and C6 represent applications with imbalanced cost settings. In particular, C5 is for the case where the cost of false negative error is much higher than that of false positive error. As a result, we observe that the reject region for C5 includes more samples from the positive class. However, C6 has an opposite setting where the cost of false positive error is much higher than that of false negative error. In this case, the reject region includes more samples from the negative class. In conclusion, we can see both the sizes and locations of reject regions adapt to changes of cost settings. It should be emphasized that this dynamic behavior is desirable since it reflects that our proposed RO-TWSVM can output an appropriate reject region according to a given cost setting.

Table 3. The expected costs for different cost models for synthetic data 1.

	C3	C4	C5	C6
RO-TWSVM	-7.5125	-4.5750	-1.3667	-0.2750
RO-SVM	-7.4750	-4.5292	-1.2458	-0.2875



(a) (b)

Fig. 6. Approximated class-conditional probability density functions of (a) Distance difference $d(\boldsymbol{x})$ of TWSVM and (b) SVM outputs for synthetic data set 1.

For synthetic data set 1, our proposed RO-TWSVM generates a reject region similar to that of RO-SVM. To establish a quantitative comparison, we calculate the expected costs (defined in (13)) of RO-TWSVM and RO-SVM for C3 through C6. The results of these costs are shown in Table 3. From Table 3, it is observed that RO-TWSVM and RO-SVM produce almost the same costs for the four cost models. Therefore, for this data set, our proposed RO-TWSVM achieves a comparable performance with RO-SVM. In Fig. 6(a) and Fig. 6(b), we plot the class-conditional probability density functions (pdfs) of $d(\boldsymbol{x})$ (defined in (8)) for TWSVM and that of SVM outputs [14], respectively. The overlapping regions of class-conditional pdfs of $d(\boldsymbol{x})$ reflect those samples which have similar distances to both hyperplanes of TWSVM (see Fig. 6(a)). For SVM, it represents samples with small margins to the separating hyperplane (see Fig. 6(b)). Generally, a larger overlapping region of class-conditional pdfs means more potentially misclassified samples. Hence, similar classification costs of RO-TWSVM and RO-SVM can also be observed from similar size of the overlapping regions in the class-conditional pdfs, as shown in Fig. 6(a) and Fig. 6(b), respectively.

4.2. Synthetic data 2: "Cross Planes" data

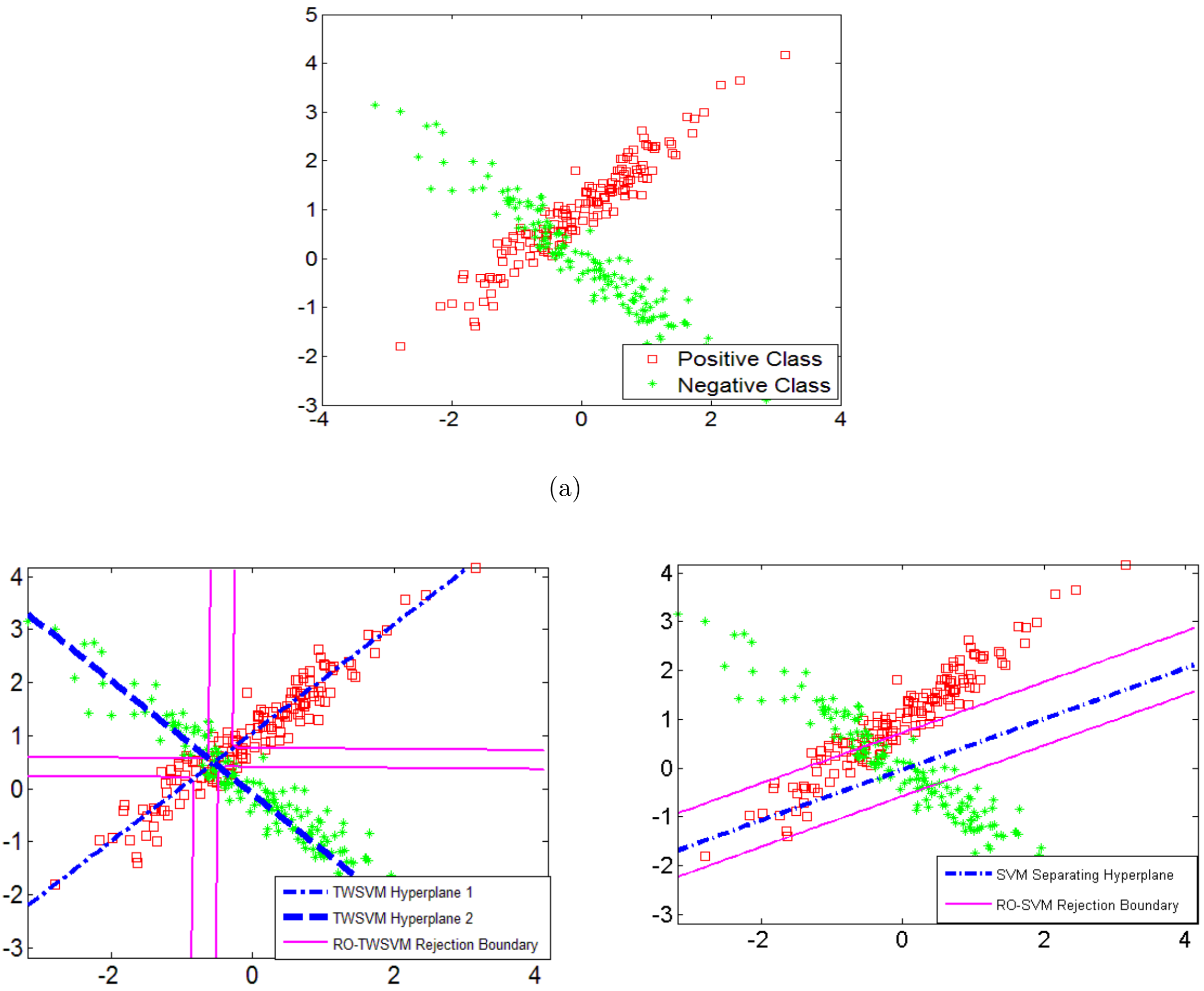
To show that RO-TWSVM has an advantage over RO-SVM [14] for a specific type of data due to the flexibility offered by the nonparallel hyperplanes of TWSVM, we apply these two methods on a synthetic data set called "Cross Planes" which is adopted in some TWSVM related papers [25, 26]. In this data set, two classes of samples are obtained by perturbing samples originally lying on two intersecting planes (lines) as shown in Fig. 7(a).

Table 4. The expected costs for different cost models for "Cross Planes" data set.

	C3	C4	C5	C6
RO-TWSVM	-9.3467	-6.7400	-3.0800	-2.4400
RO-SVM	-3.7867	-0.7533	3.3333	15.5800

In this simulation, a linear kernel is applied and the cost settings follow Table 2 in Section 4.1 (see also [14]). For C1 and C2, no reject region is generated due to violation of condition (22) and two level curves intersecting at the same point, respectively. Since the experimental results are similar for cases C3 through C6, we only analyze the case for C5 by observing the reject regions generated by RO-TWSVM and RO-SVM (shown in Fig. 7(b) and Fig. 7(c)) and the corresponding expected costs recorded in Table 4. From Fig. 7(b), it is noted that RO-TWSVM generates multiple reject regions which cover most of the sample regions with low classification confidence. In contrast, the reject region generated by RO-SVM in Fig. 7(c) is not reasonable as it contains many samples which should be classified correctly with high confidence. From Table 4, we observe that the proposed RO-TWSVM produces a lower expected cost than that of RO-SVM for each of the four cost models. Moreover, for the cases of C5 and C6 which have imbalanced costs associated with misclassification (false positive and false negative), the reject regions generated by RO-TWSVM are more robust since they have smaller fluctuation than RO-SVM in the expected costs. In summary, the proposed RO-TWSVM can generate more reasonable and robust reject region than RO-SVM for data sets like "Cross Planes".

On one hand, this improved performance of RO-TWSVM can be explained by different formulations of SVM and TWSVM. RO-TWSVM trains two nonparallel hyperplanes corresponding to the two classes, respectively. It provides flexibility to fit the data and generate a reasonable reject region accordingly as shown in Fig. 7(b). In contrast, for RO-SVM, the reject region is delimited by two hyperplanes which are parallel to the separating hyperplane [14] (see Fig. 7(c)). From Fig. 7(c), it is observed that the two classes of data in "Cross Planes" cannot be linearly discriminated by one hyperplane from SVM. Therefore, RO-SVM



(a)

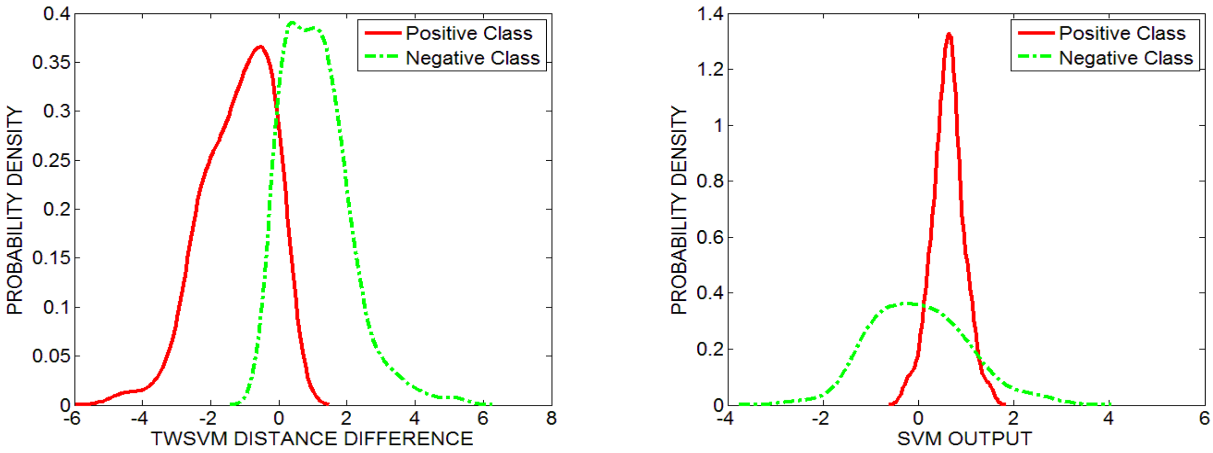
(b)

(c)

Fig. 7. (a) “Cross Planes” data set. (b) The reject region generated by the proposed RO-TWSVM. (c) The reject region generated by RO-SVM [14]**.**

generates an inappropriate reject region which is close to the separating hyperplane. In summary, we conclude that RO-TWSVM outperforms RO-SVM due to its flexibility offered by the nonparallel hyperplanes of TWSVM.

On another hand, the class-conditional pdfs for $d(\boldsymbol{x})$ and SVM outputs are also plotted in Fig. 8. Conclusions can be drawn from a comparison with the original sample location as shown in Fig. 7(a). From Fig. 7(a), it is observed that samples from both the classes in the small “intersection area” are difficult to identify and should be rejected. Fig. 8(a) illustrates that $d(\boldsymbol{x})$ applied by the proposed RO-TWSVM can reflect the samples in the “intersection area” with a small overlapping region of class-conditional pdfs. However, the class-conditional pdfs of SVM outputs cannot reflect this situation and even have a heavy overlapping region (see Fig. 8(b)). As a result, RO-TWSVM rejects samples that are mostly in the “intersection area” and RO-SVM mistakenly rejects a large number of samples which should be classified with high confidence. Hence, the analysis on class-conditional pdfs support the idea that the reject region determined by RO-TWSVM is more reasonable than that of RO-SVM.



(a)

(b)

Fig. 8. Approximated class-conditional probability density functions of (a) Distance difference $d(\boldsymbol{x})$ of TWSVM and (b) SVM outputs for "Cross Planes" data set.

In this section, the proposed RO-TWSVM and RO-SVM are compared on the two synthetic data sets. By means of quantitative comparison based on expected costs and the analysis on the class-conditional pdfs, the major observations are summarized as following:

1. The proposed RO-TWSVM can produce reject regions with different sizes and locations according to various cost settings.
2. For synthetic data set 1 in Section 4.1, RO-TWSVM and RO-SVM generate similar reject regions and achieve similar performance in terms of classification costs.
3. For synthetic data set 2 called "Cross Planes" in Section 4.2, RO-TWSVM generates more reasonable and robust reject regions than RO-SVM due to the flexibility offered by the nonparallel hyperplanes of TWSVM.

5. Experiments on real-world benchmark data sets

To show the effectiveness of the proposed RO-TWSVM on real-world data sets, two groups of experiments are conducted to evaluate the effect of incorporating a reject option in TWSVM in statistical sense. Following [14], the Wilcoxon rank sum tests are applied on 4 real-world data sets to compare the proposed RO-TWSVM with TWSVM without a reject option (TWSVM-without-RO) and the existing RO-SVM [14]. Subsequently, the Friedman and Nemenyi post-hoc tests are conducted on 23 standard binary classification data sets to statistically compare the performance of the proposed RO-TWSVM and competing methods in view of classification costs.

Table 5. Summary of experimental data sets for Wilcoxon rank sum test.

Name	Number of Samples	%Positive	%Negative
Pima	768	34.90%	65.10%
German Credit	1000	30.00%	70.00%
Breast Cancer Wisconsin	683	34.99%	65.01%
Heart Disease Cleveland	297	27.95%	72.05%

Table 6. The cost models for the experiments on real-world data sets.

	CTP	CFP	CTN	CFN	CR
CM1	Unif [-10,0]	Unif [0,50]	Unif [-10,0]	Unif [0,50]	1
CM2	Unif [-10,0]	Unif [0,100]	Unif [-10,0]	Unif [0,50]	1
CM3	Unif [-10,0]	Unif [0,50]	Unif [-10,0]	Unif [0,100]	1
CM4	Unif [-10,0]	Unif [0,50]	Unif [-10,0]	Unif [0,50]	Unif [0,30]

The notation Unif $[a, b]$ denotes the uniform distribution over the interval $[a, b]$.

5.1. Wilcoxon rank sum test

In this subsection, we conduct the Wilcoxon rank sum test on 4 real-world data sets [14] listed in Table 5. The Wilcoxon rank sum test is a non-parametric method for comparison of two classifiers over multiple data sets. In the Wilcoxon rank sum test, the pair of classifiers of interest are compared through counting the number of runs (out of 1000 runs) of wins, looses and ties based on their expected cost results. Following the settings in [14], the cost models (CM) are shown in Table 6. It is discussed in [14] that these 4 cost models are adopted to reduce the bias of experimental comparisons with respect to certain particular cost values. Such reduction is achieved by assuming the costs being generated by different statistical distributions rather than fixed values. In our experiment, we firstly compare the performance between RO-TWSVM and TWSVM-without-RO. Secondly, RO-TWSVM and RO-SVM [14] are compared. The linear and RBF kernels are adopted for both TWSVM and SVM as linear and nonlinear classifiers, respectively. The SVM is implemented using the LIBSVM software package from the public domain [27].

Algorithm 2 lists our experimental steps. The results for each CM are obtained from an experiment of 1000 runs. For each run, 5 repetitions of the experiment are implemented. In each of the 5 repetitions, the entire data set is split into three subsets: a training set, a validation set and a testing set. The training and validation sets which account for 75% of the total samples are applied in Procedure 1 to determine the optimal thresholds t_1 and t_2 for RO-TWSVM. The remaining 25% of the samples are for testing where the expected costs are calculated. Hence, for each run, we obtain 5 expected costs over the testing set. The final cost is obtained as the mean value of these 5 expected costs. RO-SVM is run in a similar fashion.

Table 7 through Table 10 record the experimental results for all cost model and data set pairs for linear and RBF kernel, respectively. Each cell of these tables contains three values

Algorithm 2: The experimental procedure for Wilcoxon rank sum test.

```
Input:
Data: $\boldsymbol{D} = \{(\boldsymbol{x_i}, y_i)\}_{i=1}^{N}$;
Output:
$[Num_H, Num_L, Num_{ID}]$;
Initilization:
$Num_H \leftarrow 0$;
$Num_L \leftarrow 0$;
$Num_{ID} \leftarrow 0$;
for $m = 1 : 1000$ do
    Cost Generation: $CTN$,$CFP$,$CFN$,$CTP$,$CR$;
    $TotalCost \leftarrow 0$;
    $AverageCost \leftarrow 0$;
    for $l = 1 : 5$ do
        Randomly divide $\boldsymbol{D} = \{(\boldsymbol{x_i}, y_i)\}_{i=1}^{N}$ into a set $P_{\bar{k}}$ for training and validation in
          Procedure 1 and a testing set $P_k$, respectively;
        Output $\leftarrow$ Run Procedure 1 with input $(\boldsymbol{x_j}, y_j)$, $j \in P_{\bar{k}}$, $CTN$,$CFP$,$CFN$,$CTP$,$CR$;
        if Output == Message: "RO is not applicable" then
            $Num_{ID} \leftarrow Num_{ID} + 1$;
            break;
        else if Output == reject region $[t_1, t_2]$ then
            Calculate $TPR$, $TNR$, $FPR$, $FNR$, $RP$ and $RN$ over testing samples $(x_j, y_j)$,
              $j \in P_k$;
            Calculate the costs $C$ defined by (13);
            $TotalCost \leftarrow TotalCost + C$;
    $AverageCost = \frac{TotalCost}{5}$;
    if RO-TWSVM produces a higher AverageCost than that of the competing method then
        $Num_H \leftarrow Num_H + 1$;
    else if RO-TWSVM produces a lower AverageCost than that of the competing method
      then
        $Num_L \leftarrow Num_L + 1$;
    else if RO-TWSVM produces an equal AverageCost to that of the competing method then
        $Num_{ID} \leftarrow Num_{ID} + 1$;
return $[Num_H, Num_L, Num_{ID}]$;
```

(i): Num_H, Num_L, Num_{ID} denote the numbers of runs that RO-TWSVM produces higher, lower or indistinguishable (or tie) costs, respectively.

(ii): For the comparison between RO-TWSVM and RO-SVM, if any of them outputs "RO is not applicable", we just record this case as the indistinguishable costs.

indicating the number of runs that RO-TWSVM produces higher, lower or indistinguishable (or tie) cost comparing with that of TWSVM-without-RO or RO-SVM. The indistinguishable cost is set for the case that two compared methods produce the same cost or any one of them is recorded as "RO is not applicable" [14]. For example, in Table 7, the cell in the upper left corner of the table indicates that for the Pima data set tested on CM1, we obtain 19 runs that RO-TWSVM produces higher cost than TWSVM-without-RO, 865 runs that

RO-TWSVM produces lower cost than TWSVM-without-RO and 116 runs that the costs are indistinguishable. Therefore, we can conclude that on the basis of the expected costs, among all 1000 runs, RO-TWSVM outperforms TWSVM-without-RO for 856 runs, TWSVM-without-RO outperforms RO-TWSVM for 19 runs and the costs are indistinguishable for the rest of 116 runs.

Table 7. Comparison between RO-TWSVM and TWSVM-without-RO for linear kernel based on Wilcoxon rank sum test.

	Pima	German Credit	Breast Cancer	Cleveland
CM1	19 **865** 116	81 **798** 121	4 **821** 175	35 **844** 121
CM2	13 **909** 78	45 **860** 95	4 **847** 149	12 **903** 85
CM3	7 **910** 83	35 **894** 71	2 **878** 120	16 **900** 84
CM4	45 241 **714**	124 125 **751**	2 224 **774**	45 207 **748**

(i) Each cell of the table contains three values which indicate the number of runs that RO-TWSVM produces higher, lower or indistinguishable (or tie) cost comparing with that of TWSVM-without-RO or RO-SVM. The second value highlighted in each cell is the number of runs that RO-TWSVM outperforms its competing method. These presentation format follows that used in [14].

Table 8. Comparison between RO-TWSVM and TWSVM-without-RO for RBF kernel based on Wilcoxon rank sum test.

	Pima	German Credit	Breast Cancer	Cleveland
CM1	51 **833** 116	87 **792** 121	3 **797** 200	267 **590** 143
CM2	23 **900** 77	50 **857** 93	5 **814** 181	379 **504** 117
CM3	23 **894** 83	39 **890** 71	3 **862** 135	128 **765** 107
CM4	70 217 **713**	109 139 **752**	6 207 **787**	115 141 **744**

Table 7 and Table 8 record the comparison results between RO-TWSVM and TWSVM-without-RO. From these tables, we observe the number of runs that RO-TWSVM produces a lower classification cost is significantly more than the numbers of the other two outcomes. Specifically, for the cost models CM1 through CM3, the number of runs that RO-TWSVM outperforms TWSVM-without-RO is clearly dominant. For CM4, the cost of rejection follows the distribution $Unif[0,30]$ rather than the fixed value. Therefore it is more probable that condition (22) is not satisfied which leads to the increasing percentage where the comparison result “RO is not applicable” occurs. In summary, RO-TWSVM outperforms TWSVM-without-RO in view of reduction of classification cost.

Table 9. Comparison between RO-TWSVM and RO-SVM [14] for linear kernel based on Wilcoxon rank sum test.

	Pima	German Credit	Breast Cancer	Cleveland
CM1	326	395	**642**	302
	564	**488**	266	**589**
	110	117	92	109
CM2	213	356	**602**	355
	712	**559**	331	**575**
	75	85	67	70
CM3	285	413	**732**	260
	638	**519**	198	**665**
	77	68	70	75
CM4	156	129	223	113
	135	125	59	159
	709	**746**	**718**	**728**

Table 10. Comparison between RO-TWSVM and RO-SVM [14] for RBF kernel based on Wilcoxon rank sum test.

	Pima	German Credit	Breast Cancer	Cleveland
CM1	**458**	127	**507**	230
	432	**756**	401	**661**
	110	117	92	109
CM2	**506**	155	424	292
	419	**760**	**509**	**638**
	75	85	67	70
CM3	**469**	98	**667**	118
	454	**834**	263	**807**
	77	68	70	75
CM4	150	53	126	108
	141	201	156	164
	709	**746**	**718**	**728**

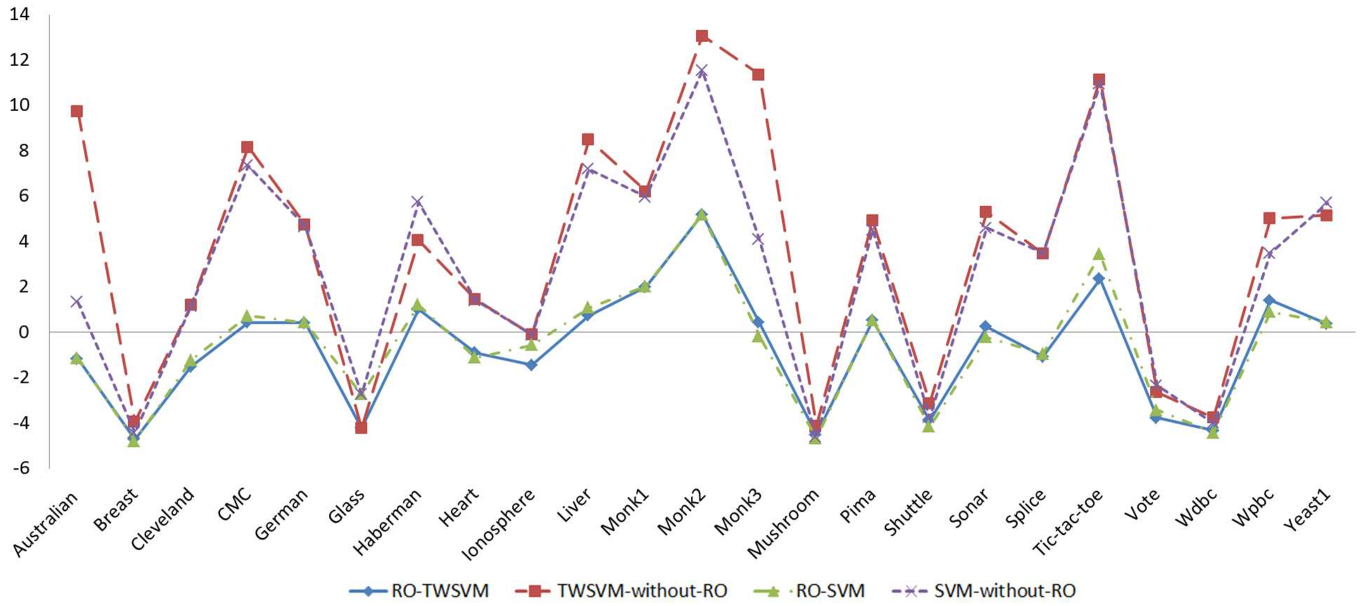
Table 9 and Table 10 record the comparison results between RO-TWSVM and RO-SVM. From these tables we observe for the data sets German Credit and Cleveland, our proposed RO-TWSVM outperforms RO-SVM for C1 through C4 and for both kernels. For the Pima data set, RO-TWSVM outperforms RO-SVM dominantly using a linear kernel and achieves a comparable performance using an RBF kernel. For the Breast Cancer data set, RO-SVM produces a lower classification cost for all the cost models for both kernels. However, we notice that for the Breast Cancer data set, TWSVM and SVM both achieve a high classification rate (above 95%) which shows this data set is quite separable. In fact, for a separable data set, the reject option setting may not be necessary because standard classifiers without RO could already achieve high classification accuracy. In summary, for data sets which are not quite separable, based on the Wilcoxon rank sum test, our proposed RO-TWSVM outperforms RO-SVM in two out of four data sets (German and Cleveland) while the two methods are comparable for one data set (Pima). However, for the data set (Breast Cancer) which is relatively separable by one hyperplane, RO-SVM outperforms the proposed RO-TWSVM.

5.2. Friedman and Nemenyi post-hoc test

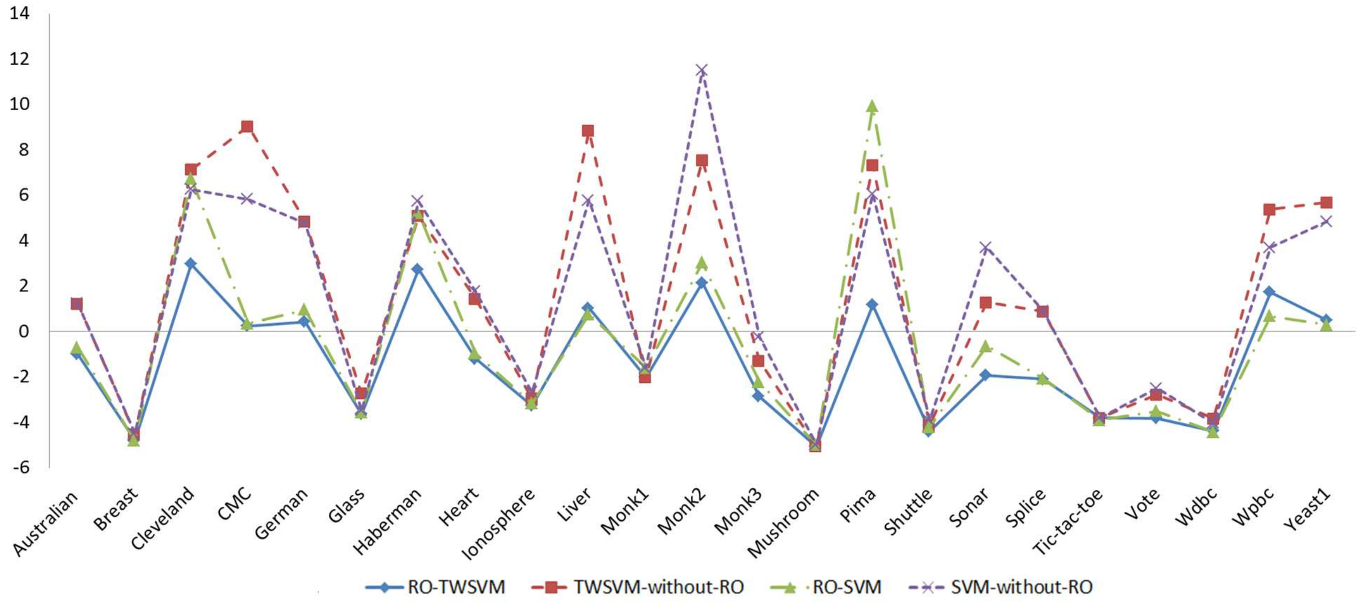
In this subsection, we compare the proposed RO-TWSVM with 3 competing methods, namely TWSVM-without-RO, RO-SVM [14] and SVM-without-RO (i.e., SVM without a reject option), on 23 standard binary classification data sets listed in Table 11 [28, 29, 30]. The adopted cost models follow those listed in Table 6. For each data set and cost model (CM1 through CM4), 10 expected costs for each method are calculated using a linear and an RBF kernel, respectively. To get a clear picture of the performance with respect to each data set, the average values of the expected costs are plotted in Fig. 9(a) and Fig. 9(b) for classifiers with linear and RBF kernel, respectively. It is observed that, on average, RO-TWSVM can produce a lower expected cost than both TWSVM-without-RO and SVM-without-RO. This observation further verifies the conclusion we obtain in Section 5.1 that embedding a reject region can enhance the robustness of the corresponding classifiers. Comparison results between RO-TWSVM and RO-SVM are observed to be dependent on the adopted kernels. When the linear kernel is adopted, RO-TWSVM and RO-SVM achieve a comparable performance in view of average costs (see Fig. 9(a)). In contrary, RO-TWSVM can produce a lower classification cost than RO-SVM when the RBF kernel is adopted (see Fig. 9(b)). From these figures we conclude that, on average, RO-TWSVM outperforms RO-SVM based on the reduction of the classification cost.

Since the average cost is a general criterion to measure the performance of the methods, we conduct a Friedman test together with a Nemenyi test to verify the statistical significance of experimental conclusions. The Friedman test is a non-parametric method to compare multiple methods over multiple data sets based on their average ranks [31]. The null hypothesis of Friedman test is that all 4 methods are equivalent based on their average ranks. In our experiment, the method with the lowest classification cost is ranked as 1, the second lowest is ranked as 2, and so on. If the methods produce the same cost, an average rank is assigned to each of them. The confidence level of the Friedman test is set as 95% where the null hypothesis can be rejected if the calculated p value less than 0.05. A post-hoc analysis called Nemenyi test [31] is subsequently conducted to evaluate whether the differences in the average ranks are statistically significant among the compared methods. For the Nemenyi test, the expected costs of any two methods are significantly different if the corresponding average ranks differ by at least the Critical Difference (CD) [31]. Otherwise, the two methods have no significance difference in performance based on their classification costs. The graphical representations of Nemenyi test results are shown in Fig. 10 and Fig. 11, where the methods that are not significantly different in performance are connected. In the following, we show the results of two scenarios which consist of all methods adopting a linear kernel and an RBF kernel, respectively. In these figures, the numbers in the brackets are the specific average ranks of the corresponding methods.

From Fig. 10 and Fig. 11, it is firstly observed that both RO-TWSVM and RO-SVM have significantly higher average ranks (i.e., with lower rank value) than TWSVM-without-



(a)



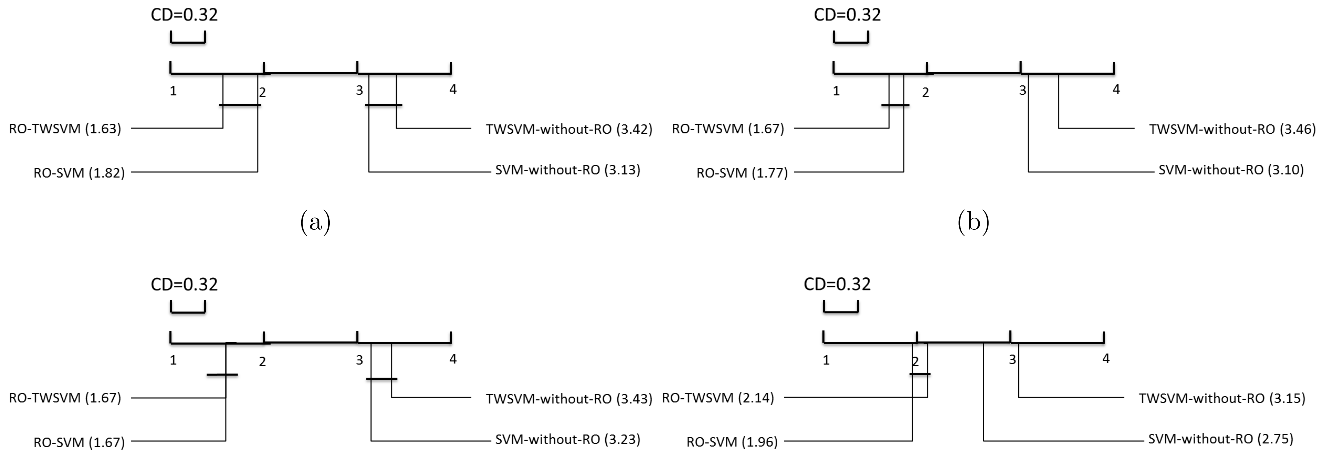
(b)

Fig. 9. The average cost for each data set : (a) linear kernel , (b) RBF kernel.

RO and SVM-without-RO, respectively. These results demonstrate that incorporating a reject option can significantly improve the performance of the original classifiers in terms of reducing classification costs. Secondly, the Nemenyi test results between RO-TWSVM and RO-SVM are observed to be different for different kernels. Using the linear kernel, there is no significant difference between the classification costs of RO-TWSVM and RO-SVM. In contrast, in the case of using RBF kernel, RO-TWSVM shows statistically better performance than RO-SVM by means of the Nemenyi test. Therefore, we can conclude that, for these data sets, RO-TWSVM

Table 11. Summary of experimental data sets for Friedman and Nemenyi tests.

	Name	Number of Samples	%Positive	%Negative
1	Australian Credit	690	44.49%	55.51%
2	Breast Cancer Wisconsin	683	34.99%	65.01%
3	Cleveland	297	27.95%	72.05%
4	CMC	1437	57.30%	42.70%
5	German Credit	1000	30%	70%
6	Glass	214	23.83%	76.71%
7	Haberman	306	26.47%	73.53%
8	Heart Statlog	270	44.44%	53.87%
9	Ionosphere	351	64.01%	35.09%
10	Liver	345	57.97%	42.03%
11	Monk1	124	50%	50%
12	Monk2	169	37.87%	62.13%
13	Monk3	122	49.18%	50.82%
14	Mushroom	5644	61.80%	38.20%
15	Pima	768	34.90%	65.10%
16	shuttle	279	52.33%	47.67%
17	Sonar	208	53.37%	46.63%
18	Splice	1000	51.70%	48.30%
19	Tic-tac-toe	958	34.66%	65.34%
20	Vote	435	61.38%	38.62%
21	wdbc	569	62.74%	37.26%
22	Wpbc	194	23.71%	76.29%
23	Yeast1	1484	28.91%	71.09%



(a)

(b)

(c)

(d)

Fig. 10. Comparison results of four methods using linear kernel through Nemenyi test: (a) CM1, (b) CM2, (c) CM3 and (d) CM4.

achieves a comparable performance with RO-SVM when using the linear kernel. Whereas using the RBF kernel, RO-TWSVM significantly outperforms RO-SVM in view of classification costs.

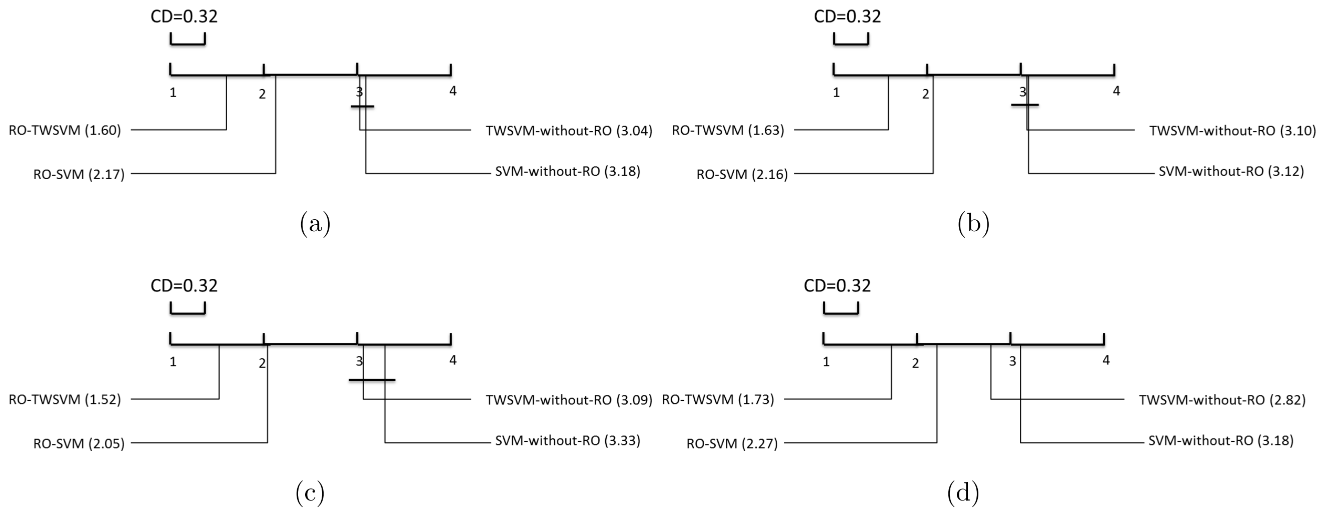


Fig. 11. Comparison results of 4 methods using RBF kernel through Nemenyi test: (a) CM1, (b) CM2, (c) CM3 and (d) CM4.

Remark. According to [31], the Wilcoxon and Friedman test are non-parametric methods suitable for statistical evaluation of classifiers. These two tests are particularly applicable for our experiments since the expected costs for different data sets are comparable (in another word, commensurability [31]). Also, they do not assume normal distributions or homogeneity of variance for the compared random variables (in our case, the expected costs) [31]. These relaxed assumptions ensure the generality of the tests. Hence, it is appropriate that we conduct the Wilcoxon rank sum test instead of the paired t-test [5] when comparing two methods. As for comparing multiple methods over multiple data sets, the Friedman and Nemenyi tests are recommended [31]. In our experiments, we attempt to include all the available binary classification data sets from the related papers [28, 29, 30] to reduce as much as possible the impact of data set dependency. To the best of our knowledge, this is the first work which applies the Friedman and Nemenyi tests over such a broad number of data sets in the field of classification with a reject option.

In summary, we have conducted extensive experiments on multiple real-world data sets for the proposed RO-TWSVM method in comparison with several related methods:

1. The proposed RO-TWSVM significantly outperforms TWSVM-without-RO in terms of classification costs for all the data sets based on the statistical tests in both Section 5.1 and 5.2.
2. In Section 5.1, through Wilcoxon rank sum test on 4 data sets from [14], the performance comparison between RO-TWSVM and RO-SVM [14] is data set dependent. For the data sets which can be easily discriminated by one hyperplane, the two methods achieve comparable performance. Whereas, RO-TWSVM significantly outperforms RO-SVM for data sets which are difficult to be discriminated by one hyperplane.
3. In Section 5.2, four methods, namely, the proposed RO-TWSVM, the TWSVM-without-

RO, the RO-SVM and the SVM-without-RO, are compared on 23 real-world data sets. The comparison is based on average costs for each data set and Friedman test together with Nemenyi test. As for the comparison between RO-TWSVM and RO-SVM, while the two methods achieve comparable performance using a linear kernel, RO-TWSVM significantly outperforms RO-SVM when using an RBF kernel.

6. Conclusion

In this paper, we have proposed to embed TWSVM with a reject option (RO-TWSVM) through determining a pair of operating points on the ROC curve. The method is adaptable to changes of cost settings which is a desirable property in real-world applications. Also, the proposed RO-TWSVM has an advantage over RO-SVM [14] for certain data sets like "Cross Planes" due to the flexibility offered by the nonparallel hyperplanes of TWSVM. The experimental results on multiple real-world data sets showed significant performance improvement of the proposed RO-TWSVM over TWSVM-without-RO in view of classification costs. As for the comparison between RO-TWSVM and RO-SVM, while the two methods achieve comparable performance using a linear kernel, RO-TWSVM significantly outperforms RO-SVM when using an RBF kernel. The RO-TWSVM is formulated for binary classification since the standard ROC curve is based on dichotomizers. Therefore, an interesting future work is to extend this method to solve multi-category classification problems.

References

[1] M. Golfarelli, D. Maio, D. Malton, On the error-reject trade-off in biometric verification systems, IEEE Transactions on Pattern Analysis and Machine Intelligence 19 (7) (1997) 786–796.

[2] Z. Zidelmal, A. Amirou, A. Belouchrani, Heartbeat classification using support vector machines (SVMs) with an embedded reject option, International Journal of Pattern Recognition and Artificial Intelligence 26 (01) (2012) 1250001.

[3] S. Mukherjee, P. Tamayo, D. Slonim, A. Verri, T. Golub, J. Mesirov, T. Poggio, Support vector machine classification of microarray data.

[4] R. Zhang, D. N. Metaxas, RO-SVM: Support vector machine with reject option for image categorization., in: BMVC, Citeseer, 2006, pp. 1209–1218.

[5] T. Pietraszek, On the use of ROC analysis for the optimization of abstaining classifiers, Machine Learning 68 (2) (2007) 137–169.

[6] R. El-Yaniv, Y. Wiener, On the foundations of noise-free selective classification, The Journal of Machine Learning Research 11 (2010) 1605–1641.

[7] C. K. Chow, An optimum character recognition system using decision functions, IRE Transactions on Electronic Computers (4) (1957) 247–254.

[8] C. K. Chow, On optimum recognition error and reject tradeoff, IEEE Transactions on Information Theory 16 (1) (1970) 41–46.

[9] B. E. Boser, I. M. Guyon, V. N. Vapnik, A training algorithm for optimal margin classifiers, in: Proceedings of the fifth annual workshop on Computational learning theory, ACM, 1992, pp. 144–152.

[10] V. Vapnik, The nature of statistical learning theory, Springer Science & Business Media, 2013.

[11] Y. Grandvalet, A. Rakotomamonjy, J. Keshet, S. Canu, Support vector machines with a reject option, in: Advances in neural information processing systems, 2009, pp. 537–544.

[12] M. Wegkamp, M. Yuan, et al., Support vector machines with a reject option, Bernoulli 17 (4) (2011) 1368–1385.

[13] G. Fumera, F. Roli, Support vector machines with embedded reject option, in: Pattern Recognition with Support Vector Machines, Springer, 2002, pp. 68–82.

[14] F. Tortorella, Reducing the classification cost of support vector classifiers through an ROC-based reject rule, Pattern Analysis and Applications 7 (2) (2004) 128–143.

[15] F. Tortorella, A ROC-based reject rule for dichotomizers, Pattern Recognition Letters 26 (2) (2005) 167–180.

[16] C. M. Santos-Pereira, A. M. Pires, On optimal reject rules and ROC curves, Pattern recognition letters 26 (7) (2005) 943–952.

[17] S. Bernard, C. Chatelain, S. Adam, R. Sabourin, The multiclass ROC front method for cost-sensitive classification, Pattern Recognition 52 (2016) 46–60.

[18] I. Pillai, G. Fumera, F. Roli, Multi-label classification with a reject option, Pattern Recognition 46 (8) (2013) 2256–2266.

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

[20] W.-J. Chen, Y.-H. Shao, C.-N. Li, N.-Y. Deng, MLTSVM: A novel twin support vector machine to multi-label learning, Pattern Recognition 52 (2016) 61–74.

[21] D. Tomar, S. Agarwal, A comparison on multi-class classification methods based on least squares twin support vector machine, Knowledge-Based Systems 81 (2015) 131–147.

[22] O. L. Mangasarian, E. W. Wild, Multisurface proximal support vector machine classification via generalized eigenvalues, IEEE Transactions on Pattern Analysis and Machine Intelligence 28 (1) (2006) 69–74.

[23] S. Boyd, L. Vandenberghe, Convex optimization, Cambridge University Press, 2004.

[24] R. Khemchandani, S. Chandra, et al., Optimal kernel selection in twin support vector machines, Optimization Letters 3 (1) (2009) 77–88.

[25] M. A. Kumar, M. Gopal, Application of smoothing technique on twin support vector machines, Pattern Recognition Letters 29 (13) (2008) 1842–1848.

[26] Y.-H. Shao, C.-H. Zhang, X.-B. Wang, N.-Y. Deng, Improvements on twin support vector machines, IEEE Transactions on Neural Networks 22 (6) (2011) 962–968.

[27] C.-C. Chang, C.-J. Lin, LIBSVM: A library for support vector machines, ACM Transactions on Intelligent Systems and Technology 2 (2011) 27:1–27:27, software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`.

[28] K.-A. Toh, H.-L. Eng, Between classification-error approximation and weighted least-squares learning, Pattern Analysis and Machine Intelligence, IEEE Transactions on 30 (4) (2008) 658–669.

[29] K.-A. Toh, G.-C. Tan, Exploiting the relationships among several binary classifiers via data transformation, Pattern Recognition 47 (3) (2014) 1509–1522.

[30] L. Sun, K.-A. Toh, Z. Lin, A center sliding bayesian binary classifier adopting orthogonal polynomials, Pattern Recognition 48 (6) (2015) 2013–2028.

[31] J. Demšar, Statistical comparisons of classifiers over multiple data sets, The Journal of Machine Learning Research 7 (2006) 1–30.