Twin Pinball Loss Support Vector Hyper-sphere Classifier for Pattern Recognition

Rongfen Gong[1, 2], Chengdong Wu[1], Maoxiang Chu[2], Huanqing Wang[3]

1. College of Information Science and Engineering, Northeastern University, Shenyang 110819, China
E-mail: chu52@sina.com

2. School of Electronic and Information Engineering, University of Science and Technology Liaoning, Anshan 114051, China
E-mail: chu52_2004@163.com

3. Department of Mathematics, Bohai University, Jinzhou 150001, China
E-mail: ndwhq@163.com

Abstract: Motivated by twin support vector hyper-sphere (TSVH) and support vector machine with pinball loss (pin-SVM), this papaer formulated a twin pinball loss support vector hyper-sphere (TPSVH) classifier. TPSVH is obtained by introducing pinball loss with quantile distance into TSVH. It has TSVH-type objective function and similar inequality constraints with pin-SVM. And it is stable and insensitive to noise samples, which makes TPSVH fitter for pattern classification problem. From the classification experiments for synthetic and UCI datasets, it can be clearly seen that TPSVH has better classificaiton accurary and generalization performance compared with other classifiers. In a word, the novel classifier proposed in this paper enjoys excellent performance in stability and insensitivity to noise.

Key Words: TSVH, Pin-SVM, Pinball Loss, TPSVH

1 INTRODUCTION

Support vector machine (SVM) [1] proposed by Vapnik et al. has been used widely in pattern classification and regression analysis [2-4]. By maximizing margin between two classes, SVM can realize structural risk minimization for pattern classification. SVM needs to find an optimal hyper-plane by solving a quadratic programming problem (QPP). For non-linear classification, SVM solves QPP by using dual theory and kernel function.

With the development of technology, a set of novel algorithms are derived from SVM [5-9]. In 2007, Javadeva et al. proposed twin support vector machine (TSVM) [10] based on generalized eigenvalue proximal SVM (GEPSVM) [6]. TSVM searches for two nonparallel optimal hyper-planes and ensures each hyper-plane to be close to one class and far from the other one. TSVM is fit for crossed datasets. It solves two smaller QPPs, which improves classification speed. Based on TSVM, a set of improved algorithms are derived [11-13]. In 2014, Peng et al. proposed twin support vector hyper-sphere (TSVH) [14] based on TSVM and support vector data description (SVDD) [5]. TSVH also needs to solve two QPPs, and each QPP has TSVM-type formula. However, it is different from TSVM in searching for a pair of SVDD-type hyper-spheres. In TSVH, each hyper-sphere covers one class of samples and keeps away from the other class of samples as much as possible. So, TSVH can maximize not only inter-class margin but also intra-class clustering degree, which obtains better classification result.

This work is supported by National Nature Science Foundation under Grant 61273078 and University of Science and Technology Liaoning Foundation under Grant 2015RC06 and 2014QN05.

Noise has been studied for a long time in pattern classification problem. Weights were employed to eliminate the impact of noise data in reference [15]. The total margin was used in reference [16] to construct SVM and get the robust estimation. Reference [17] introduced fuzzy set into SVM to obtain less sensitive results for noise. In 2014, Huang et al. proposed SVM with pinball loss (pin-SVM) [18]. In Reference [18], authors consider that idea must make SVM sensitive to those two types of noise: label noise and feature noise [19]. The basic idea of SVM is to maximize the margin between the closest samples. However those closest samples between two classes may be noise, especially feature noise. Reference [18] considers Pin-SVM changes that basic idea into maximizing the quantile distance with pinball loss [20-22]. It adjusts the number of the closest samples with parameter to reduce noise sensitivity.

In this paper, twin pinball loss support vector hyper-sphere (TPSVH) is proposed based on TSVH and pin-SVM. It has TSVH-type formula and searches for a pair of hyper-spheres. TPSVH uses pinball loss to replace hinge loss of TSVH, which makes it similar with pin-SVM in constraints. On one hand, TPSVH can maximize inter-class margin and intra-class clustering degree. On the other hand, TPSVH is not sensitive to the noise around the decision boundary. These advantages improve classification performance. Experiments for synthetic and UCI datasets also prove that TPSVH has better classification accuracy and stability compared with SVM, TSVH and pin-SVM.

2 BACKGROUND

We consider a binary classification problem with dataset $\boldsymbol{T}=\{\boldsymbol{X}_l,\boldsymbol{y}_l\}_{l=1}^m$, where $\boldsymbol{X}_l\in\Re^{d\times 1}$ and $\boldsymbol{y}_l\in\{+1,-1\}$. Sample matrixes of class +1 and −1 in $\boldsymbol{T}$ are defined as

$\boldsymbol{A}=[\boldsymbol{A}_1\cdots\boldsymbol{A}_i\cdots\boldsymbol{A}_{m^+}]^T$ and $\boldsymbol{B}=[\boldsymbol{B}_1\cdots\boldsymbol{B}_j\cdots\boldsymbol{B}_{m^-}]^T$ respectively. Where, $m=m^++m^-$ and $\boldsymbol{A}_i,\boldsymbol{B}_j\in\{\boldsymbol{X}_l\}_{l=1}^m$. Then, binary TSVH and binary pin-SVM classifiers will be described as follows.

2.1 TSVH

TSVH is derived from SVDD and contains the idea of TSVM. TSVH classifier constructs two hyper-spheres instead of hyper-planes. For binary classification problem, those two hyper-spheres are just like the following:

$$\begin{aligned}&\Omega^+:\|\psi(\boldsymbol{x})-\boldsymbol{C}^+\|^2\le(R^+)^2,\\&\Omega^-:\|\psi(\boldsymbol{x})-\boldsymbol{C}^-\|^2\le(R^-)^2,\end{aligned}\tag{1}$$

where $\boldsymbol{C}^+$ and $\boldsymbol{C}^-$ are the centers of Ω^+ and Ω^- respectively. R^+ and R^- are the radiuses of Ω^+ and Ω^- respectively. $\psi(\cdot)$ is mapping function of feature space. $K(\boldsymbol{x}_1,\boldsymbol{x}_2)=\psi(\boldsymbol{x}_1)^T\psi(\boldsymbol{x}_2)$ may be used as a kernel function when mapping. For matrixes $\boldsymbol{A}$ and $\boldsymbol{B}$, QPPs of TSVH can be constructed as follows:

$$\begin{aligned}\min_{\boldsymbol{C}^+,R^+}\ &\frac{1}{2}\sum_{i=1}^{m^+}\|\psi(\boldsymbol{A}_i)-\boldsymbol{C}^+\|^2-v^+(R^+)^2+p^-\sum_{j=1}^{m^-}\zeta_j\\ \text{s.t.}\ &\|\psi(\boldsymbol{B}_j)-\boldsymbol{C}^+\|^2\ge(R^+)^2-\zeta_j,\\&(R^+)^2\ge0,\ \zeta_j\ge0,\ \ j=1,2,\cdots,m^-,\end{aligned}\tag{2}$$

$$\begin{aligned}\min_{\boldsymbol{C}^-,R^-}\ &\frac{1}{2}\sum_{j=1}^{m^-}\|\psi(\boldsymbol{B}_j)-\boldsymbol{C}^-\|^2-v^-(R^-)^2+p^+\sum_{i=1}^{m^+}\xi_i\\ \text{s.t.}\ &\|\psi(\boldsymbol{A}_i)-\boldsymbol{C}^-\|^2\ge(R^-)^2-\xi_i,\\&(R^-)^2\ge0,\ \xi_i\ge0,\ \ i=1,2,\cdots,m^+.\end{aligned}\tag{3}$$

The functions of v^+, v^-, p^+ and p^- are similar. They are all penalty parameters. In the objective function of QPP (2), the minimization of the first term represents that the samples of class +1 are as close to the center of the hyper-sphere as possible. The minimization of the second term represents that the radius of the hyper-sphere is as large as possible. The minimization of the third term with inequality constraint represents that samples of class −1 can avoid being covered by hyper-sphere as much as possible. The terms of QPP (3) are similar with those of QPP (2) respectively. TSVH can maximize intra-class clustering degree. So, for many practical pattern classification problems, TSVH is more suitable than TSVM.

2.2 Pin-SVM

In most versions of SVM classifiers, hinge loss function is widely used. However, Pin-SVM uses pinball loss to replace hinge loss. Pin-SVM can be described with the following QPP:

$$\begin{aligned}\min_{\boldsymbol{w},b}\ &\frac{1}{2}\boldsymbol{w}^T\boldsymbol{w}+p\sum_{l=1}^{m}\zeta_l\\ \text{s.t.}\ &\boldsymbol{y}_l(\boldsymbol{w}^T\psi(\boldsymbol{x}_l)+b)\ge1-\zeta_l,\\&\boldsymbol{y}_l(\boldsymbol{w}^T\psi(\boldsymbol{x}_l)+b)\le1+\frac{1}{\tau}\zeta_l,\ \ l=1,2,\cdots,m,\end{aligned}\tag{4}$$

where τ is the parameter of pinball loss function. If $\boldsymbol{u}=\boldsymbol{y}_l(\boldsymbol{w}^T\psi(\boldsymbol{x}_l)+b)$, pinball loss function adopted by Pin-SVM can be expressed as

$$L_\tau=\begin{cases}1-u, & u\le1,\\-\tau(1-u), & u>1.\end{cases}\tag{5}$$

Specially, when $\tau=0$, pinball loss function can be expressed as

$$L_h=L_{\tau=0}=\max\{0,1-u\},\ u\in\Re.\tag{6}$$

The pinball loss function $L_{\tau=0}$ is just the hinge loss function L_h, and the second constraint of QPP (4) becomes $\zeta_l\ge0$. In the meanwhile, Pin-SVM reduces to standard SVM. Pin-SVM uses parameter τ to reduce noise sensitivity. In order to explain that clearly, three sets are first defined as follows:

$$\begin{aligned}\boldsymbol{T}_q&=\{l\mid\boldsymbol{y}_l(\boldsymbol{w}^T\psi(\boldsymbol{x}_l)+b)<1\},\\\boldsymbol{T}_g&=\{l\mid\boldsymbol{y}_l(\boldsymbol{w}^T\psi(\boldsymbol{x}_l)+b)>1\},\\\boldsymbol{T}_e&=\{l\mid\boldsymbol{y}_l(\boldsymbol{w}^T\psi(\boldsymbol{x}_l)+b)=1\}.\end{aligned}\tag{7}$$

The basic idea of SVM is to maximize the distance between two bounding hyper-planes: $\boldsymbol{w}^T\psi(\boldsymbol{x}_l)+b=1$ and $\boldsymbol{w}^T\psi(\boldsymbol{x}_l)+b=-1$. Hyper-planes $\boldsymbol{w}^T\psi(\boldsymbol{x}_l)+b=\pm1$ can divide samples into $\boldsymbol{T}_q$, $\boldsymbol{T}_g$ and $\boldsymbol{T}_e$. It is well known that there should be few or none samples in $\boldsymbol{T}_q$ for SVM. If the samples in $\boldsymbol{T}_q$ are noise, classification result will be greatly affected. Pin-SVM uses parameter τ to control the position of the bounding hyper-planes, which changes the number of samples in $\boldsymbol{T}_q$ and $\boldsymbol{T}_g$. Pin-SVM reduces the adverse effect of noise by adjusting the number of the samples in $\boldsymbol{T}_q$.

3 TWIN PINBALL LOSS SUPPORT VECTOR HYPER-SPHERE

3.1 Binary TPSVH

It can be seen from QPPs (2) and (3) that binary TSVH classifier includes quadratic loss and hinge loss functions. If u^+ and u^- are defined, hinge loss functions in QPPs (2) and (3) can be expressed as:

$$\begin{aligned}L_h^+&=\max\{0,(R^+)^2(1-\boldsymbol{u}^+)\},\\L_h^-&=\max\{0,(R^-)^2(1-\boldsymbol{u}^-)\},\end{aligned}\tag{8}$$

$$\boldsymbol{u}^+=\frac{\|\psi(\boldsymbol{B}_j)-\boldsymbol{C}^+\|^2}{(R^+)^2},\ \boldsymbol{u}^-=\frac{\|\psi(\boldsymbol{A}_i)-\boldsymbol{C}^-\|^2}{(R^-)^2}.\tag{9}$$

And they are shown in Fig. 1(a). It has been known from the above section that TSVH classifier with hinge loss is sensitive to noise. TSVH is improved as TPSVH after pinball loss function is introduced. Just like pin-SVM, TPSVH is not sensitive to noise samples. Firstly, pinball loss function used in TPSVH is defined as L_τ^+ and L_τ^- shown in Fig. 1(b).

$$\begin{aligned}L_\tau^+&=\begin{cases}(R^+)^2(1-u^+), & u^+\le1,\\-\tau(R^+)^2(1-u^+), & u^+>1,\end{cases}\\L_\tau^-&=\begin{cases}(R^-)^2(1-u^-), & u^-\le1,\\-\tau(R^-)^2(1-u^-), & u^->1.\end{cases}\end{aligned}\tag{10}$$

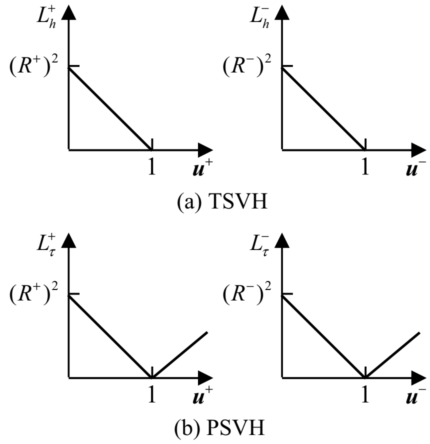

Fig. 1. (a) Hinge loss, (b) Pinball loss.

If L_h^+ and L_h^- of TSVH are replaced with L_τ^+ and L_τ^- respectively, QPPs of TPSVH can be obtained as follows:

$$\begin{aligned}&\min_{C^+,R^+} \frac{1}{2}\sum_{i=1}^{m^+}\|\psi(\boldsymbol{A}_i)-\boldsymbol{C}^+\|^2 - p_r^+(R^+)^2 + p_e^-\sum_{j=1}^{m^-}\zeta_j \\ &\text{s.t. } \|\psi(\boldsymbol{B}_j)-\boldsymbol{C}^+\|^2 \ge (R^+)^2-\zeta_j, \\ &\quad\ \|\psi(\boldsymbol{B}_j)-\boldsymbol{C}^+\|^2 \le (R^+)^2+\frac{1}{\tau}\zeta_j,\ j=1,2,\cdots,m^-,\end{aligned} \tag{11}$$

$$\begin{aligned}&\min_{C^-,R^-} \frac{1}{2}\sum_{j=1}^{m^-}\|\psi(\boldsymbol{B}_j)-\boldsymbol{C}^-\|^2 - p_r^-(R^-)^2 + p_e^+\sum_{i=1}^{m^+}\xi_i \\ &\text{s.t. } \|\psi(\boldsymbol{A}_i)-\boldsymbol{C}^-\|^2 \ge (R^-)^2-\xi_i, \\ &\quad\ \|\psi(\boldsymbol{A}_i)-\boldsymbol{C}^-\|^2 \le (R^-)^2+\frac{1}{\tau}\xi_i,\ i=1,2,\cdots,m^+,\end{aligned} \tag{12}$$

where p_r^+, p_r^-, p_e^+ and p_e^- are all penalty factors. The optimal classification hyper-spheres Ω^+ and Ω^- of TPSVH are shown in formula (1). It can be seen from the objective function of QPP (11) that the first term is quadratic loss function. It shows that the sum of the squared distances between samples of class $+1$ and the center of classification hyper-sphere is minimized. In other words, the classification hyper-sphere covers the samples of class $+1$ as much as possible. The second term maximizes the radius of Ω^+, which ensures structural risk minimization. The third term with inequality constraint means to minimize pinball loss. On one hand, pinball loss makes the classifier less sensitive to samples of class -1 in the hyper-sphere. On the other hand, minimization of pinball loss ensures the classification error bound of pinball loss preserves the same as that of hinge loss. Similar conclusions can be obtained from objective function and constraint of QPP (12).

If $\tau = 0$, pinball loss function reduces to hinge loss. Then the second constraints in QPP (11) and (12) become $\zeta_j \ge 0$ and $\xi_i \ge 0$ respectively. And TPSVH reduces to TSVH. Reference [18] discusses pinball loss can reduce the adverse effect of noise samples around the decision boundary. TPSVH with pinball loss function can reduce noise sensitivity by adjusting τ. Then, take the classification hyper-sphere of class $+1$ as an example to illustrate that conclusion. The minimization of the third term and the constraint in QPP (11) ensure less samples of class -1 are covered into Ω^+. Samples of class -1 distributed around the decision boundary of Ω^+ will greatly affect the position of the decision boundary. TPSVH can adjust the number of samples in and out of Ω^+ with parameter τ. More samples of class -1 in Ω^+ reduce the adverse effect which noise samples have on the third term in QPP (11). So, TPSVH is not sensitive to noise samples.

3.2 Solution to TPSVH

In order to solve QPPs of TPSVH classifier, operators of $\boldsymbol{\alpha}_j^- \ge 0$ and $\boldsymbol{\beta}_j^- \ge 0$ are introduced. Lagrangian function corresponding to QPP (11) is given as follows:

$$\begin{aligned}L = &\frac{1}{2}\sum_{i=1}^{m^+}\|\psi(\boldsymbol{A}_i)-\boldsymbol{C}^+\|^2 - p_r^+(R^+)^2 + p_e^-\sum_{j=1}^{m^-}\zeta_j - \\ &\sum_{j=1}^{m^-}\boldsymbol{\alpha}_j^-\left(\|\psi(\boldsymbol{B}_j)-\boldsymbol{C}^+\|^2-(R^+)^2+\zeta_j\right)+ \\ &\sum_{j=1}^{m^-}\boldsymbol{\beta}_j^-\left(\|\psi(\boldsymbol{B}_j)-\boldsymbol{C}^+\|^2-(R^+)^2-\frac{1}{\tau}\zeta_j\right).\end{aligned} \tag{13}$$

Karush-Kuhn-Tucker conditions are used to obtain the following results:

$$\begin{aligned}\frac{\partial L}{\partial \boldsymbol{C}^+} = &-\sum_{i=1}^{m^+}\left(\psi(\boldsymbol{A}_i)-\boldsymbol{C}^+\right)+2\sum_{j=1}^{m^-}\boldsymbol{\alpha}_j^-\left(\psi(\boldsymbol{B}_j)-\boldsymbol{C}^+\right) \\ &-2\sum_{j=1}^{m^-}\boldsymbol{\beta}_j^-\left(\psi(\boldsymbol{B}_j)-\boldsymbol{C}^+\right)=0,\end{aligned} \tag{14}$$

$$\frac{\partial L}{\partial (R^+)^2}L = -p_r^+ + \sum_{j=1}^{m^-}\boldsymbol{\alpha}_j^- - \sum_{j=1}^{m^-}\boldsymbol{\beta}_j^- = 0, \tag{15}$$

$$\frac{\partial L}{\partial \zeta_j} = p_e^- - \boldsymbol{\alpha}_j^- - \frac{1}{\tau}\boldsymbol{\beta}_j^- = 0, \tag{16}$$

$$\boldsymbol{\alpha}_j^-\left(\|\psi(\boldsymbol{B}_j)-\boldsymbol{C}^+\|^2-(R^+)^2+\zeta_j\right)=0, \tag{17}$$

$$\boldsymbol{\beta}_j^-\left(\|\psi(\boldsymbol{B}_j)-\boldsymbol{C}^+\|^2-(R^+)^2-\frac{1}{\tau}\zeta_j\right)=0. \tag{18}$$

According to (14), the center $\boldsymbol{C}^+$ of Ω^+ can be calculated with the following formula:

$$\boldsymbol{C}^+ = \frac{\sum_{i=1}^{m^+}\psi(\boldsymbol{A}_i)-2\sum_{j=1}^{m^-}\left(\boldsymbol{\alpha}_j^- - \boldsymbol{\beta}_j^-\right)\psi(\boldsymbol{B}_j)}{m^+ - 2\sum_{j=1}^{m^-}\left(\boldsymbol{\alpha}_j^- - \boldsymbol{\beta}_j^-\right)}. \tag{19}$$

If (15), (16) and (19) are substituted into (13), and constant terms are discarded, the dual QPP of (11) can be obtained:

$$\begin{aligned}\max_{\lambda^-} &-\frac{2}{m^+-2p_r^+}\sum_{j1=1}^{m^-}\sum_{j2=1}^{m^-}\boldsymbol{\lambda}_{j1}^-\boldsymbol{\lambda}_{j2}^-K(\boldsymbol{B}_{j1},\boldsymbol{B}_{j2})+ \\ &\sum_{j=1}^{m^-}\boldsymbol{\lambda}_j^-\left(\frac{2}{m^+-2p_r^+}\sum_{i=1}^{m^+}K(\boldsymbol{B}_j,\boldsymbol{A}_i)-K(\boldsymbol{B}_j,\boldsymbol{B}_j)\right) \\ \text{s.t. } &\sum_{j=1}^{m^-}\boldsymbol{\lambda}_j^- = p_r^+, \\ &-\tau p_e^- \le \boldsymbol{\lambda}_j^- \le p_e^-,\ j=1,2,\cdots,m^-,\end{aligned} \tag{20}$$

where $\boldsymbol{\lambda}_j^- \ge 0$. Define $\boldsymbol{e}^-$ as an ones column vector with dimension m^- and $\boldsymbol{\lambda}^- = [\boldsymbol{\lambda}_1^-\ \boldsymbol{\lambda}_2^- \cdots \boldsymbol{\lambda}_{m^-}^-]^T$. According to the deduction, QPP (20) with matrix form can be obtained:

$$\begin{aligned}&\min_{\boldsymbol{\lambda}^-} (\boldsymbol{\lambda}^-)^T \boldsymbol{H}\boldsymbol{\lambda}^- \\ &\text{s.t.} \quad (\boldsymbol{e}^-)^T \boldsymbol{\lambda}^- = p_r^+, \\ &\quad -\tau p_e^- \boldsymbol{e}^- \le \boldsymbol{\lambda}^- \le p_e^- \boldsymbol{e}^-,\end{aligned} \tag{21}$$

where $\boldsymbol{H} \in \Re^{m^- \times m^-}$ can be expressed as $\boldsymbol{H} = (h_{j1,j2})$. Then, $h_{j1,j2}$ can be calculated with the following formula:

$$\begin{aligned}h_{j1,j2} = &\frac{m^+ - 2p_r^+}{4p_r^+}\left(K(\boldsymbol{B}_{j1}, \boldsymbol{B}_{j1}) + K(\boldsymbol{B}_{j2}, \boldsymbol{B}_{j2})\right) + \\ &K(\boldsymbol{B}_{j1}, \boldsymbol{B}_{j2}) - \frac{1}{2p_r^+}\sum_{i=1}^{m^+}\left(K(\boldsymbol{A}_i, \boldsymbol{B}_{j1}) + K(\boldsymbol{A}_i, \boldsymbol{B}_{j2})\right).\end{aligned} \tag{22}$$

Define $\boldsymbol{\alpha}^- = [\boldsymbol{\alpha}_1^- \; \boldsymbol{\alpha}_2^- \cdots \boldsymbol{\alpha}_{m^-}^-]^T$ and $\boldsymbol{\beta}^- = [\boldsymbol{\beta}_1^- \; \boldsymbol{\beta}_2^- \cdots \boldsymbol{\beta}_{m^-}^-]^T$. The relationship among $\boldsymbol{\lambda}^-$, $\boldsymbol{\alpha}_j^-$ and $\boldsymbol{\beta}_j^-$ are just like the following:

$$\begin{aligned}&(1+\frac{1}{\tau})\boldsymbol{\beta}^- = p_e^- \boldsymbol{e}^- - \boldsymbol{\lambda}^-, \\ &\boldsymbol{\alpha}^- - \boldsymbol{\beta}^- = \boldsymbol{\lambda}^-.\end{aligned} \tag{23}$$

If $-\tau p_e^- < \boldsymbol{\lambda}_j^- < p_e^-$, then $\boldsymbol{\alpha}_j^- \neq 0$ and $\boldsymbol{\beta}_j^- \neq 0$. Define set

$$S^- = \{ j \mid \boldsymbol{\alpha}_j^- \neq 0, \boldsymbol{\beta}_j^- \neq 0 \text{ and } j = 1, 2, \cdots, m^- \}. \tag{24}$$

According to (17) and (18), $(R^+)^2 = \| \psi(\boldsymbol{B}_j) - \boldsymbol{C}^+ \|^2$ can be obtained, where $j \in S^-$. For accurate calculation, the formula of $(R^+)^2$ can be determined as the follows:

$$(R^+)^2 = \frac{1}{|S^-|}\sum_{j \in S^-} \| \psi(\boldsymbol{B}_j) - \boldsymbol{C}^+ \|^2, \tag{25}$$

where $|S^-|$ is the number of the items in S^-.

Define $\boldsymbol{\alpha}^+ = [\boldsymbol{\alpha}_1^+ \; \boldsymbol{\alpha}_2^+ \cdots \boldsymbol{\alpha}_{m^+}^+]^T$, $\boldsymbol{\beta}^+ = [\boldsymbol{\beta}_1^+ \; \boldsymbol{\beta}_2^+ \cdots \boldsymbol{\beta}_{m^+}^+]^T$ and $\boldsymbol{\lambda}^+ = [\boldsymbol{\lambda}_1^+ \; \boldsymbol{\lambda}_2^+ \cdots \boldsymbol{\lambda}_{m^+}^+]^T$. And define $\boldsymbol{e}^+$ as an ones column vector with dimension m^+. Dual QPP of (12) with matrix form can be similarly obtained:

$$\begin{aligned}&\min_{\boldsymbol{\lambda}^+} (\boldsymbol{\lambda}^+)^T \boldsymbol{G}\boldsymbol{\lambda}^+ \\ &\text{s.t.} \quad (\boldsymbol{e}^+)^T \boldsymbol{\lambda}^+ = p_r^-, \\ &\quad -\tau p_e^+ \boldsymbol{e}^+ \le \boldsymbol{\lambda}^+ \le p_e^+ \boldsymbol{e}^+,\end{aligned} \tag{26}$$

where $\boldsymbol{G} \in \Re^{m^+ \times m^+}$ can be expressed as $\boldsymbol{G} = (g_{i1,i2})$. Then, $g_{i1,i2}$ can be calculated with the following formula:

$$\begin{aligned}g_{i1,i2} = &K(\boldsymbol{A}_{i1}, \boldsymbol{A}_{i2}) + \frac{m^- - 2p_r^-}{4p_r^-}\left(K(\boldsymbol{A}_{i1}, \boldsymbol{A}_{i1}) + K(\boldsymbol{A}_{i2}, \boldsymbol{A}_{i2})\right) - \\ &\frac{1}{2p_r^-}\sum_{j=1}^{m^-}\left(K(\boldsymbol{B}_j, \boldsymbol{A}_{i1}) + K(\boldsymbol{B}_j, \boldsymbol{A}_{i2})\right).\end{aligned} \tag{27}$$

Similarly, relationship among $\boldsymbol{\alpha}^+$, $\boldsymbol{\beta}^+$ and $\boldsymbol{\lambda}^+$ are just like the following:

$$\begin{aligned}&(1+\frac{1}{\tau})\boldsymbol{\beta}^+ = p_e^+ \boldsymbol{e}^+ - \boldsymbol{\lambda}^+, \\ &\boldsymbol{\alpha}^+ - \boldsymbol{\beta}^+ = \boldsymbol{\lambda}^+.\end{aligned} \tag{28}$$

On the other hand, the center $\boldsymbol{C}^-$ of classification hyper-sphere Ω^- can be calculated with the following formula:

$$\boldsymbol{C}^- = \frac{\sum_{j=1}^{m^-}\psi(\boldsymbol{B}_j) - 2\sum_{i=1}^{m^+}\left(\boldsymbol{\alpha}_i^+ - \boldsymbol{\beta}_i^+\right)\psi(\boldsymbol{A}_i)}{m^- - 2\sum_{i=1}^{m^+}\left(\boldsymbol{\alpha}_i^+ - \boldsymbol{\beta}_i^+\right)}. \tag{29}$$

Define set

$$S^+ = \{ i \mid \boldsymbol{\alpha}_i^+ \neq 0, \boldsymbol{\beta}_i^+ \neq 0 \text{ and } i = 1, 2, \cdots, m^+ \}, \tag{30}$$

then the formula of $(R^-)^2$ can be determined as the follows:

$$(R^-)^2 = \frac{1}{|S^+|}\sum_{j \in S^+} \| \psi(\boldsymbol{A}_i) - \boldsymbol{C}^- \|^2. \tag{31}$$

The following formula can determine a new sample $\boldsymbol{x}$ belongs to class $+1$ or -1:

$$f(\boldsymbol{x}) = \arg\min\left\{\frac{\| \psi(\boldsymbol{x}) - \boldsymbol{C}^+ \|^2}{(R^+)^2}, \frac{\| \psi(\boldsymbol{x}) - \boldsymbol{C}^- \|^2}{(R^-)^2}\right\}. \tag{32}$$

4 NUMERICAL EXPERIMENTS

In order to testify the efficiency of the novel algorithm proposed in this paper, SVM, TSVH, pin-SVM and TPSVH are used to do classification experiments. For fairness, all classifiers only use a quadratic programming solver embedded in Maltab. All experiments are implemented in Matlab 7.11 on Windows 7 running on a PC with Intel(R) Core(TM) I5 CPU (3.2GHZ) and 4GB RAM.

4.1 Synthetic Datasets

TPSVH proposed in this paper is used to handle noise samples in the dataset. So, two 2-dimension synthetic datasets with Gaussian distribution are created according to the method in reference [18]. There are 275 samples of one class in Data 1, and the mean μ_1 is $[0.5, -2.5]^T$ and covariance matrix Σ_1 is $\text{diag}(0.2, 2.5)$. There are 275 samples of the other class in Data 1, and the mean μ_2 is $[-0.5, 2.5]^T$ and covariance matrix Σ_2 is $\text{diag}(0.2, 2.5)$. Data 2 can be built by introducing 5 percent of noise samples into Data 1. The labels of these noise samples are selected from $\{+1, -1\}$ with equal probability. And they satisfy Gaussian distribution with

$$\mu_n = [0, 0]^T \text{ and } \Sigma_n = \begin{bmatrix} 2.5 & -0.2 \\ -0.2 & 2.5 \end{bmatrix}. \tag{33}$$

Firstly, in order to highlight the function of parameter τ, the other parameters with the same type are equal for all classifiers. Penalty parameter C of SVM, v^+, v^-, p^+ and p^- of TSVH, p of pin-SVM, and p_r^+, p_r^-, p_e^+ and p_e^- of TPSVH are all set to 1. RBF for K is adopted in all classifiers and kernel radius is set to 10. Before classification experiment, we repeat the creating and training process ten times for Data 1 and 2.

Then, SVM, TSVH, pin-SVM and TPSVH classifiers are compared to prove the improvement brought by pinball loss. The mean of accuracy for all classifiers are shown in Table 1 and 2. For Data 1, TPSVH as well as the other classifiers has satisfying results, which shows TPSVH has stable classification performance. For Data 2, the accuracy of

TPSVH is higher than that of TSVH and SVM, which proves that pinball loss can improve the classification accuracy. And the above results also proved that TPSVH can tolerate noise samples. The accuracy of TPSVH is higher than that of Pin-SVM, which shows the combination of pinball loss and TSVH can improve the performance of classifier. Moreover, for TPSVH and Pin-SVM, classification accuracy changes with τ. We can get the best classification accuracy by adjusting τ.

Table 1. Classification accuracy for Data 1

τ	TPSVH	Pin-SVM	TSVH	SVM
1	98.16	98.04	97.91	98.01
0.5	98.18	98.09		
0.2	98.26	98.13		
0.1	98.09	98.06		

Table 2. Classification accuracy for Data 2

τ	TPSVH	Pin-SVM	TSVH	SVM
1	97.20	96.97	96.54	96.35
0.5	97.27	97.09		
0.2	97.16	97.01		
0.1	97.09	96.91		

4.2 UCI Datasets

In order to illustrate the performance of TPSVH, all datasets used in the experiments are real world data downloaded from UCI Repository of Machine Learning Dataset. Moreover, in order to highlight the ability which TPSVH has in dealing with noise samples, experiments are done on UCI datasets with noise. According to the method in reference [18], 5 percent of samples randomly selected from every dataset are corrupted in label and feature. 80 percent of samples in UCI datasets are randomly selected as training datasets, and the others are testing datasets. Testing method is ten-fold cross validation. RBF for K is adopted for all classifiers. And τ is selected from {1, 0.5, 0.2, 0.1}, and the others are chosen from $[2^{-7}, 2^{7}]$.

Table 3 shows the mean and standard deviation of accuracy corresponding to different classifiers. All those results are obtained on UCI datasets with noise. It can be seen that TPSVH has satisfying results. The mean of TPSVH is higher than the others, which proves that TPSVH has better classification performance for dataset with noise. The standard deviation of TPSVH is close to that of Pin-SVM, but smaller than that of TSVH and SVM, which means that TPSVH is as stable as Pin-SVM. The mean of TPSVH is higher than that of Pin-SVM, TSVH and SVM, which proves that the combination of pinball loss and TSVH improves classification performance.

Table 4 also shows the mean and standard deviation of accuracy corresponding to different classifiers. However, all those results are obtained on original UCI datasets. It can be seen that TPSVH has satisfying results. Most of the means of TPSVH are higher than the others. And standard deviation of TPSVH is close to that of Pin-SVM. The above results show that TPSVH has satisfying classification performance for original datasets. From the comparison between Table 3 and 4, it can be seen that TPSVH is stable for original datasets, and superior for datasets with noise.

Table 3. Calssification accuracy on testing data for UCI datasets with noise

Dataset	TPSVH	Pin-SVM	TSVH	SVM
Heart	77.92±2.37	77.79±1.94	76.67±2.73	76.60±2.27
Breast	96.08±1.44	95.77±1.02	94.59±1.54	95.35±1.07
Haberman	73.06±2.50	73.07±2.50	71.79±3.01	71.69±2.49
Spect	82.22±3.58	81.82±3.72	81.96±3.56	82.03±4.95
German	74.05±2.52	73.28±2.08	73.20±2.67	73.16±2.41
Magic	81.73±0.88	80.82±0.64	80.22±1.04	80.11±0.87
Pima	77.09±1.92	77.35±1.20	75.82±2.26	75.70±1.88
Transfusion	77.86±1.53	77.37±1.14	77.01±1.79	76.91±1.90

Table 4. Calssification accuracy on testing data for original UCI datasets

Datasets	TPSVH	Pin-SVM	TSVH	SVM
Heart	80.46±1.86	80.22±1.73	80.25±1.95	80.20±1.92
Breast	96.15±0.83	96.21±0.74	96.19±0.98	96.20±0.84
Haberman	73.02±1.10	72.88±1.10	72.31±1.74	72.10±1.53
Spect	85.76±1.47	82.59±1.56	85.98±1.54	86.63±1.76
German	75.51±1.70	74.39±1.56	75.45±1.82	75.13±1.87
Magic	83.24±1.79	83.15±0.72	83.07±1.66	83.23±1.24
Pima	77.32±1.22	76.83±1.13	76.55±1.34	76.42±1.72
Transfusion	78.99±1.52	78.91±1.44	78.35±1.87	78.22±1.60

5 CONCLUSION

In this paper, binary TPSVH classifier is proposed for dataset with noise. TPSVH is obtained by replacing hinge loss with pinball loss. It has two QPPs with TSVH-type, and is similar with pin-SVM in inequality constraints. On one hand, TPSVH as well as TSVH has the merits of maximization of inter-class margin and intra-class clustering degree, which is fitter for pattern classification problem. On the other hand, TPSVH as well as pin-SVM can adjust the number of samples in and out of hyper-sphere with parameter, which reduces the sensitivity to noise samples. And testing experiments are done on synthetic datasets and UCI datasets with noise respectively. Experimental results show that TPSVH can improve the performance of classifier and tolerate noise samples. Moreover, for corrupted and original datasets, TPSVH has the best classification accuracy compared with the other classifiers. All results prove that TPSVH has superior performance, especially for datasets with noise. In the future, we will study how to sparse samples and improve training efficiency with optimization algorithm.

REFERENCES

[1] V. Vapnik, The nature of statistical learning, Springer-Verlag, New York, NY, USA, 1995.

[2] C. J. C. Burges, A tutorial on support vector machines for pattern recognition, Data Mining and Knowledge Discovery, Vol.2, No.2, 121-167, 1998.

[3] P. S. Bradley and O. L. Mangasarian, Massive data discrimination via linear support vector machines, Optimization Methods and Software, Vol.13, No.1, 1-10, 2000.

[4] Y. Bazi, F. Melgani, Semisupervised PSO-SVM regression for biophysical parameter estimation, IEEE Trans. on Geoscience and Remote Sensing, Vol.45, No.6, 1887-1895, 2007.

[5] D. M. J. Tax and R. P. W. Duin, Support vector data description, Machine Learning, Vol.54, No.1, 45-66, 2004.

[6] O. L. Mangasarian and E. W. Wild, Multisurface proximal support vector Classification via Generalized Eigenvalues, IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol.28, No.1, 69-74, 2006.

[7] Z. M. Yang, Y. H. Shao and J. Liang, Unascertained support vector machine, Acta Automatica Sinica, Vol.39, No.6, 896-901, 2013.

[8] Y. J. Tian, Z. Q. Qi, X. C. Ju, Y. Shi and X. H. Liu, Nonparallel support vector machines for pattern classification, IEEE Trans. On Cybernetics, Vol.44, No.7, 1067-1079, 2014.

[9] E. Castillo, D. Peteiro-Barral, B. G. Berdiñas and O. Fontenla-Romero, Distributed one-class support vector machine, International Journal of Neural Systems, Vol.25, No.7, 1550029, 2015.

[10] Javadeva, R. Khemchandani and S. Chandra, Twin support vector machines for pattern classification, IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol.29, No.5, 905-910, 2007.

[11] M. A. Kumar and M. Gopal, Least squares twin support vector machines for pattern classification, Expert Systems with Applications, Vol.36, No.4, 7535-7543.

[12] X. J. Peng, TPMSVM: A novel twin parametric-margin support vector machine for pattern recognition, Pattern Recognition, Vol.44, No.10, 2678–2692, 2011.

[13] M. X. Chu R. F. Gong and A. N. Wang, Strip steel surface defect classification method based on enhanced twin Support vector machine, ISIJ International, Vol. 54, No.1, 119-124, 2014.

[14] X. J. Peng and D. Xu, Twin support vector hypersphere (TSVH) classifier for pattern recognition, Neural Computing & Applications, Vol.24, No.5, 1207-1220, 2014.

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

[16] M. Yoon, Y. Yun and H. Nakayama, A role of total margin in support vector machines, Proceedings of International Joint Conference on Neural Networks, 2049-2053, 2003.

[17] B. Jin, Y. C. Tang and Y. Q. Zhang, Support vector machines with genetic fuzzy feature transformation for biomedical data classification, Information Sciences, Vol.177, No.2, 476-489, 2007.

[18] X. L. Huang, L. Shi and J. A. K. Suykens, Support vector machine classifier with pinball loss, IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol.36, No.5, 984-997, 2014.

[19] J. Bi and T. Zhang, Support vector classification with input data uncertainty, Proceedings of NIPS, 161-168, 2004.

[20] R. Koenker, Quantile Regression, Cambridge University Press, Cambridge, U.K., 2005.

[21] I. Steinwart and A. Christmann, How SVMs can estimate quantiles and the median, Proceedings of NIPS, 305-312, 2007.

[22] I. Steinwart and A. Christmann, Estimating conditional quantiles with the help of the pinball loss, Bernoulli, Vol.17, No.1, 211–225, 2011.