Extension of pin-SVM to Negative τ

Xiaolin Huang, *Member, IEEE*, Lei Shi, and Johan A. K. Suykens, *Senior Member, IEEE*

***Abstract*—Applying the pinball loss in a support vector machine (SVM) classifier results in pin-SVM. The pinball loss is characterized by a parameter τ. When τ equals zero, the pinball loss reduces to the hinge loss, which is the classical loss applied in SVMs. If a positive τ is used, the classifier trained by pin-SVM is related to a quantile distance and hence is insensitive to feature noise. In this paper, we will further show that the non-negative constraint on τ is not necessary. The pin-SVM with a negative τ is proposed and evaluated on real-life data sets. An extreme choice is $\tau = -1$, which has a simplified solution and can give a good classifier in many applications. The analysis and the performance show that it is good to consider both positive and negative τ values in pin-SVM.**

***Index Terms*—support vector machine, pinball loss, classification**

I. Introduction

The pinball loss is defined as

$$L_\tau(u) = \begin{cases} u, & u \geq 0, \\ -\tau u, & u < 0, \end{cases}$$

where τ is the absolute value of the slope on $u < 0$. The pinball loss is a generalization of the ℓ_1 loss ($\tau = 1$) and the hinge loss ($\tau = 0$). In the classical support vector machine (SVM, [1] [2]), one minimizes the sum of the regularization term and the hinge loss, i.e.,

$$\min_{w,b} \quad \frac{1}{2}\|w\|_2^2 + C\sum_{i=1}^{m} L_{\tau=0}\left(1 - y_i(w^T\phi(x_i) + b)\right), \qquad (1)$$

where $x_i \in \mathbb{R}^n, y_i \in \{-1, +1\}$ are the training data, $\phi(x)$ is the feature map, and $C > 0$ is the trade-off parameter. This method is called C-SVM, which is basically to maximize the margin by minimizing $\|w\|_2^2$. In C-SVM, the margin is related to the closest distance between two classes, since the hinge loss is minimized. For a positive τ value, the pinball loss $L_\tau(u)$ has been well studied in quantile regression, see, e.g., [3] and [4]. From the link between the pinball loss and the quantile value, C-SVM (1) has been extended to the following support vector machine with the pinball loss (*pin-SVM*, [5])

$$\min_{w,b} \quad \frac{1}{2}\|w\|_2^2 + C\sum_{i=1}^{m} L_\tau\left(1 - y_i(w^T\phi(x_i) + b)\right). \qquad (2)$$

When a positive τ is used, the margin considered in pin-SVM corresponds to the quantile distance. Compared with the closest distance, the quantile distance is less sensitive to the noise on features.

The hinge loss is a particular case of the pinball loss for $\tau = 0$. Hence, pin-SVM can be regarded as an extension to C-SVM. Introducing flexibility on τ can improve the classification performance of C-SVM, as analyzed in [5]. We found that different τ values are suitable for different data structures. In some applications, C-SVM performs better, i.e., the best choice of τ is zero. A natural question is that when pin-SVM with $\tau = 0$ performs better than any positive value, is it possible to use a negative τ? and if so, does a negative τ lead to an even better result?

In Fig.1, we plot the pinball loss for different τ values. Convexity of the pinball loss requires that $\tau \geq -1$. From this point of view, the non-negative condition on τ is indeed not necessary and pin-SVM with a negative τ is worthy studying.

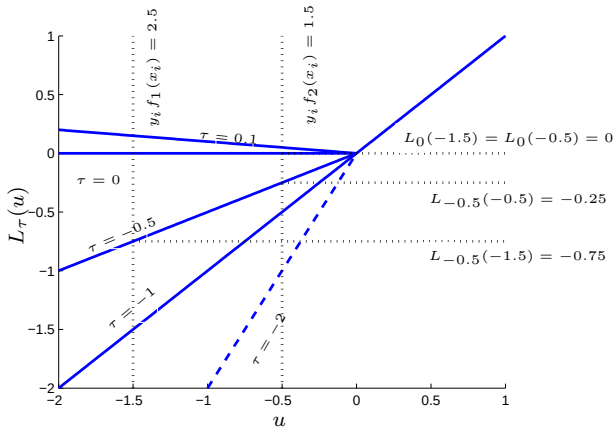

Fig. 1. The plots of the pinball loss for different τ values. The convex ones are displayed by solid lines. When $\tau < -1$, the pinball loss will be non-convex as shown by the dashed line. Consider two functions $f_1(x)$ and $f_2(x)$ and assume that $y_i f_1(x_i) = 2.5$, $y_i f_2(x_i) = 1.5$. The pinball loss with $\tau = 0$ does not distinguish the two functions. Since $y_i f_1(x_i) > y_i f_2(x_i)$, $L_\tau(u)$ with a negative τ gives more gain to $f_1(x)$, which illustrates some rationale of the pinball loss with a negative τ.

Manuscript received 2014;

This work was supported in part by the scholarship of the Flemish Government; Research Council KUL: GOA/11/05 Ambiorics, GOA/10/09 MaNet, CoE EF/05/006 Optimization in Engineering (OPTEC), IOF-SCORES4CHEM, several PhD/postdoc & fellow grants; Flemish Government: FWO: PhD/postdoc grants, projects: G0226.06 (cooperative systems and optimization), G.0302.07 (SVM/Kernel), G.0320.08 (convex MPC), G.0558.08 (Robust MHE), G.0557.08 (Glycemia2), G.0588.09 (Brain-machine) research communities (WOG: ICCoS, ANMMM, MLDM); G.0377.09 (Mechatronics MPC), G.0377.12 (Structured models), IWT: PhD Grants, Eureka-Flite+, SBO LeCoPro, SBO Climaqs, SBO POM, O&O-Dsquare; Belgian Federal Science Policy Office: IUAP P6/04 (DYSCO, Dynamical systems, control and optimization, 2007-2011); IBBT; EU: ERNSI; ERC AdG A-DATADRIVE-B, FP7-HD-MPC (INFSO-ICT-223854), COST intelliCIS, FP7-EMBOCON (ICT-248940); Contract Research: AMINAL; Other: Helmholtz: viCERP, ACCM, Bauknecht, Hoerbiger. L. Shi is also supported by the National Natural Science Foundation of China (11201079). Johan Suykens is a professor at KU Leuven, Belgium.

X. Huang and J. A. K. Suykens are with the Department of Electrical Engineering ESAT-STADIUS, KU Leuven, B-3001 Leuven, Belgium. (e-mails: huangxl06@mails.tsinghua.edu.cn, johan.suykens@esat.kuleuven.be). L. Shi is with School of Mathematical Sciences, Fudan University, Shanghai, 200433, P.R. China. (e-mail: leishi@fudan.edu.cn)

Fig.1 also gives an intuitive explanation of a negative τ. Suppose that there are two functions $f_1(x)$ and $f_2(x)$. At the sampling point (x_i, y_i), the two functions both give correct labels but have different values, e.g., $y_i f_1(x_i) = 2.5, y_i f_2(x_i) = 1.5$. Then it is reasonable to regard $f_1(x)$ better than $f_2(x)$. In this point of view, it makes sense to consider a negative τ in the pinball loss. A negative value of $L_\tau(u)$ corresponds to a gain, instead of a loss. Besides this explanation, we will analyze other properties of the pin-SVM with negative τ values, and then evaluate it on real-life data sets in this paper.

The rest of this paper is organized as follows: in Section II, the pin-SVM with a negative τ is proposed and investigated. Section III discusses the properties for different choices of τ. Then pin-SVM is evaluated by numerical experiments in Section IV. Section V ends the paper with conclusions.

II. Support Vector Machine with the Pinball Loss

A. Pin-SVM formulation

When the pinball loss is applied in classification, the corresponding support vector machine in the primal space is given by (2). The dual problem has been discussed in [5]. In this subsection, we will revisit the dual problem and investigate the role of τ. First, (2) can be formulated as the following constrained quadratic programming problem

$$\begin{aligned} \min_{w,b,\xi} \quad & \frac{1}{2}w^T w + \sum_{i=1}^{m} C_i \xi_i \\ \text{s.t.} \quad & y_i \left[w^T \phi(x_i) + b\right] \geq 1 - \xi_i, i = 1, 2, \ldots, m, \\ & y_i \left[w^T \phi(x_i) + b\right] \leq 1 + \frac{1}{\tau}\xi_i, i = 1, 2, \ldots, m, \end{aligned} \tag{3}$$

where C_i could be different. A typical setting, which is simple and suitable for unbalanced training problems, is

$$\begin{aligned} C_i &= C_0, && \forall i : y_i = 1, \\ C_i &= \tfrac{\#j:y_j=-1}{\#j:y_j=1} C_0, && \forall i : y_i = -1, \end{aligned} \tag{4}$$

where $C_0 > 0$ is a constant defined by users. We introduce the Lagrange multipliers $\alpha_i, \beta_i \geq 0$ corresponding to the constraints in (3). These dual variables meet the following complementary slackness condition,

$$\begin{aligned} &\alpha_i \left(1 - \xi_i - y_i \left[w^T \phi(x_i) + b\right]\right) = 0, i = 1, \ldots, m, \\ &\beta_i \left(y_i \left[w^T \phi(x_i) + b\right] - 1/\tau - 1\right) = 0, i = 1, \ldots, m. \end{aligned} \tag{5}$$

Then we get the following dual problem,

$$\begin{aligned} \min_{\alpha,\beta} \quad & \frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m}(\alpha_i - \beta_i) y_i \mathcal{K}_{ij} y_j (\alpha_j - \beta_j) - \sum_{i=1}^{m}(\alpha_i - \beta_i) \\ \text{s.t.} \quad & \sum_{i=1}^{m} y_i (\alpha_i - \beta_i) = 0 \\ & \alpha_i + \frac{1}{\tau}\beta_i = C_i, i = 1, 2, \ldots, m, \\ & \alpha_i \geq 0, \beta_i \geq 0, i = 1, 2, \ldots, m. \end{aligned}$$

Introduce $\lambda_i = \alpha_i - \beta_i$ and eliminate the equality constraint $\alpha_i + \frac{1}{\tau}\beta_i = C_i$. Then the dual problem of pin-SVM is formulated as

$$\begin{aligned} \min_{\lambda} \quad & \frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m}\lambda_i y_i \mathcal{K}_{ij} y_j \lambda_j - \sum_{i=1}^{m}\lambda_i \\ \text{s.t.} \quad & \sum_{i=1}^{m} y_i \lambda_i = 0, \\ & -\tau C_i \leq \lambda_i \leq C_i, i = 1, 2, \ldots, m, \end{aligned} \tag{6}$$

where $\mathcal{K}$ corresponds to a positive definite kernel with $\mathcal{K}_{ij} = \mathcal{K}(x_i, x_j) = \phi(x_i)^T \phi(x_j)$.

Solving (6) results in the optimal solution. Then the obtained function can be represented as

$$f(x) = \sum_{i=1}^{m} y_i \lambda_i \mathcal{K}(x, x_i) + b,$$

where b is computed according to the complementary slackness condition (5), i.e., $y_i f(x_i) = 1, \forall i : \alpha_i \neq 0 \ \& \ \beta_i \neq 0$. Since $\lambda_i = \alpha_i - \beta_i$ and $\alpha_i \beta_i = 0$, we can calculate the bias term by

$$\sum_{j=1}^{m} y_j \lambda_j \mathcal{K}(x_i, x_j) + b = y_i, \forall i : -\tau C_i < \lambda_i < C_i.$$

In the primal space, τ describes the slope of the pinball loss $L_\tau(u)$, as displayed in Fig.1. In the dual problem (6), the objective is a quadratic function independent of τ and the feasible set is $-\tau C_i \leq \lambda_i \leq C_i$, of which the size is controlled by τ. In Fig.2, the dotted line stands for the contour map of a quadratic function. The feasible set of C-SVM with $C_1 = C_2 = 1.5$ is displayed by the solid lines. From this figure, we again find that the requirement $\tau = 0$ is not necessary. A positive or negative τ value can be used. For example, the dashed lines illustrate the feasible set when $\tau = -0.33$. With $C_i \geq 0$ and $\tau_i \geq -1$, the feasible set $-\tau C_i \leq \lambda_i \leq C_i$ is not empty. C_i determines the location of a constraint box and τ_i gives its width. In a more general case, both C_i and τ_i could be different for each dual variable λ_i. As an example, one box with $C_1 = 1.25, \tau_1 = -0.2$ and $C_2 = 1.75, \tau_2 = 0.8$ is shown by dot-dashed lines. For simplicity, we in this paper do not tune parameters for each data point but use a single τ value and set C_i as (4).

B. Pinball loss with negative τ

From both the primal and the dual problem of pin-SVM, one can see that neither the requirement $\tau = 0$ (C-SVM) nor $\tau \geq 0$ (pin-SVM) is necessary. We only need $\tau \geq -1$. Otherwise, the primal problem (2) will be non-convex and the dual problem (6) will be infeasible. The properties of pin-SVM with $\tau = 0$ have been well studied. The pin-SVM with a positive τ value has been discussed in [5]. Now we investigate the case $-1 \leq \tau < 0$. Intuitively, $L_\tau(u)$ with $-1 \leq \tau < 0$ can be interpreted as: if an observation (x_i, y_i) is classified correctly and $y_i f(x_i) > 1$, then some gain is obtained by $L_\tau(u)$ instead of the hinge loss.

To further discuss its property, we assume that the training data are independently drawn from a probability measure ρ defined on $X \times Y$, where $X \subseteq \mathbb{R}^n$ is the input space and

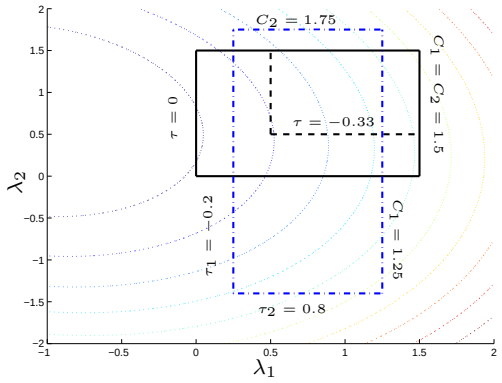

Fig. 2. The feasible sets for different τ and C values. The dotted lines describe the contour map of a quadratic function. The solid lines display the feasible set for C-SVM with $C_1 = C_2 = 1.5$. C determines the location of the feasible set and τ gives the width. The requirement of τ being zero or positive is not necessary. Moreover, C_i and τ_i can be different for each sampling. In this figure, the box constraints for $C_1 = C_2 = 1.5, \tau_1 = \tau_2 = -0.33$ and $C_1 = 1.25, C_2 = 1.75, \tau_1 = -0.2, \tau_2 = 0.8$ are displayed by dashed and dot-dashed lines, respectively.

$Y = \{-1, +1\}$ represents the class label. For any loss L, the expected *L-risk* of a measurable function $f : X \to \mathbb{R}$ is defined as follows,

$$\mathcal{R}_{L,\rho}(f) = \int_{X\times Y} L(1 - yf(x))d\rho.$$

Minimizing the expected risk over all measurable functions results in function $f_{L,\rho}$, which is defined as follows,

$$f_{L,\rho}(x) = \arg\min_{t\in\mathbb{R}} \int_Y L\left(1 - y(x)t\right) d\rho(y|x), \ \forall x \in X.$$

When $L_\tau(u)$ is applied, $f_{L_\tau,\rho}(x)$ with a nonnegative τ is a Bayes rule, which outputs $+1$ when $\mathrm{Prob}(y = +1|x) > \mathrm{Prob}(y = -1|x)$ and outputs -1 when $\mathrm{Prob}(y = +1|x) < \mathrm{Prob}(y = -1|x)$. This property has been proved in [5]. When a negative τ is used, though minimizing $\mathcal{R}_{L,\rho}(f)$ does not lead to a Bayes classifier, L_τ with $\tau \geq -1$ is still a calibrated loss, i.e., $f_{L_\tau,\rho}(x) > 0$, if $\mathrm{Prob}(y = +1|x) > \mathrm{Prob}(y = -1|x)$ and $f_{L_\tau,\rho}(x) < 0$, if $\mathrm{Prob}(y = +1|x) < \mathrm{Prob}(y = -1|x)$. The conclusion and the proof are given below.

Theorem 1: $L_\tau(u)$ is calibrated if $\tau \geq -1$.

Proof: We use $g_1(t)$, $g_2(t)$, and $g_3(t)$ to denote the following functions, respectively,

$$\begin{aligned}
g_1(t) &= \big(-\mathrm{Prob}(y = +1|x) - \tau\mathrm{Prob}(y = -1|x)\big)t \\
&\quad + \mathrm{Prob}(y = +1|x) - \tau\mathrm{Prob}(y = -1|x), \\
g_2(t) &= \big(-\mathrm{Prob}(y = +1|x) + \mathrm{Prob}(y = -1|x)\big)t \\
&\quad + \mathrm{Prob}(y = +1|x) + \mathrm{Prob}(y = -1|x), \\
g_3(t) &= \big(\tau\mathrm{Prob}(y = +1|x) + \mathrm{Prob}(y = -1|x)\big)t \\
&\quad - \tau\mathrm{Prob}(y = +1|x) + \mathrm{Prob}(y = -1|x).
\end{aligned}$$

Then, from the definition of L-risk, we have

$$\begin{aligned}
\mathcal{R}_{L_\tau,\rho}(f) &= \int_Y L_\tau\left(1 - y(x)t\right) d\rho(y|x) \\
&= L_\tau(1-t)\mathrm{Prob}(y = +1|x) \\
&\quad + L_\tau(1+t)\mathrm{Prob}(y = -1|x) \\
&= \begin{cases} g_1(t), & t \leq -1, \\ g_2(t), & -1 < t < 1, \\ g_2(t), & t \geq 1. \end{cases}
\end{aligned}$$

When $\mathrm{Prob}(y = +1|x) > \mathrm{Prob}(y = -1|x)$, the condition $\tau \geq -1$ means that $-\mathrm{Prob}(y = +1|x) - \tau\mathrm{Prob}(y = -1|x) < 0$, hence

$$\min_{t\leq -1} g_1(t) = 2\mathrm{Prob}(y = +1|x),$$

which is achieved at $t = -1$. Meanwhile, $\min_{-1\leq t\leq 1} g_2(t) = 2\mathrm{Prob}(y = -1|x)$ is achieved at $t = 1$ and $g_3(t) \leq 2\mathrm{Prob}(y = -1|x)$ for any $t \geq 1$. Therefore, when $\tau \geq -1$ and $\mathrm{Prob}(y = +1|x) > \mathrm{Prob}(y = -1|x)$, minimizing $\mathcal{R}_{L_\tau,\rho}(f)$ leads to a positive output.

Similar proof shows that, with the condition that $\tau \geq -1$, $f_{L_\tau,\rho}(x) < 0$, if $\mathrm{Prob}(y = +1|x) < \mathrm{Prob}(y = -1|x)$. Therefore, $L_\tau(u)$ with $\tau \geq -1$ is a calibrated loss. ■

Notice that when a negative τ is used, the minimizer of the pinball loss is no longer a Bayes classifier. But still it is meaningful to minimize $L_\tau(u)$ with a negative τ to train a good classifier, since it is calibrated. The motivation of using a negative τ in pin-SVM also can be understood as giving a gain to a point that is correctly classified. In the next section, we will discuss more about the properties of the pinball loss for different τ values. We will provide some hints of choosing a suitable τ value for different problems.

III. Roles of Different τ Values

A. Quantile analysis

When a positive τ is used, the pinball loss is closely related to the quantile value and has been widely applied in quantile regression. In the SVM framework (2), one minimizes $w^T w$ to maximize the distance between the surface $\{x : f(x) = -1\}$ and $\{x : f(x) = +1\}$. In pin-SVM (2), the margin that we are going to maximize is still $2/\|w\|_2$, which measures the distance between $\{x : f(x) = -1\}$ and $\{x : f(x) = +1\}$, but $\{x : f(x) = \pm 1\}$ are located at quantile positions. The quantile level is monotone with respect to τ. For example, if $\tau = 0$, then $\{x : f(x) = \pm 1\}$ are in the extreme positions in the separable cases. When a positive τ value is used, $\{x : f(x) = \pm 1\}$ are related to quantile values, which brings insensitivity to feature noise.

If a negative τ is used, pin-SVM (2) looses the usual quantile meaning and it will locate the surfaces $\{x : f(x) = \pm 1\}$ at a position exceeding the extreme position. Let us consider a simple linear classification problem. In this example, the data of class $+1$ and class -1 come from two normal distributions. The probability density functions are given by the green and red lines in Fig.3(a). The training data are shown by the green stars and the red crosses in Fig.3(b). In this problem, only the horizontal position is considered as a feature. For this problem,

we use pin-SVM (2) to train a classifier, where linear functions are used and C_0 is set to be 100. Different τ values are considered and the corresponding surfaces $\{x : f(x) = \pm 1\}$ are displayed in Fig.3(b), where the solid lines are for $\tau = 0.0$, the dashed lines are for $\tau = 0.1$, and the dotted lines are for $\tau = -0.1$.

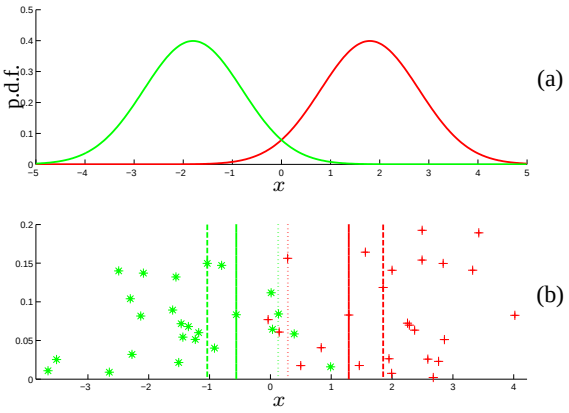

Fig. 3. Sampling points and classification results of pin-SVM. Points in class $+1$ and -1 are drawn from the distribution shown in (a) and are displayed by green stars and red crosses in (b), respectively. The surfaces $\{x : f(x) = \pm 1\}$ for $\tau = 0.1, 0, -0.1$ are illustrated by dashed, solid, and dotted lines.

When a smaller τ value is used, the surfaces are closer to the decision boundary. Generally, the points near the boundary are easily corrupted by noise. Thus, with τ decreasing, the results are more likely be affected by noise. On the other hand, if the data quantity is good, using a large τ value will be too conservative and a smaller τ value may lead to a good result. Another case is that when there are plenty of data, the noise effect may be weakened and then pin-SVM with a negative τ will be a good choice. The above discussions hold as well for nonlinear kernels in the feature space. To illustrate this, we show a nonlinear classification problem by Fig.4, where the same radial basis function (RBF) kernel and the same regularization parameter, but different τ values are used.

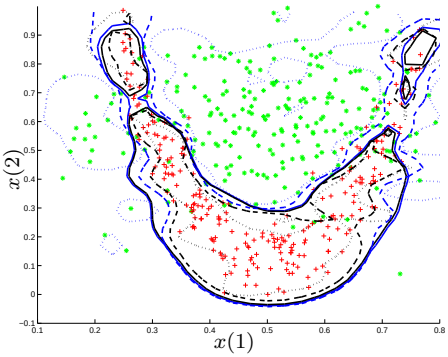

Fig. 4. Sampling points and classification results of pin-SVM with the RBF kernel. Points in class $+1$ and -1 are shown by green stars and red crosses, respectively. The blue and black lines display surfaces $\{x : f(x) = +1\}$ and $\{x : f(x) = -1\}$. The solid lines are for $\tau = -0.5$, dashed lines for $\tau = -0.1$, and the dotted lines for $\tau = 0$.

B. Within-class scatter

Since a quantile distance is considered, the pin-SVM with a positive τ enjoys insensitivity to feature noise, which also can be explained from the viewpoint of Fisher discriminant analysis [6] [7]. Specifically, $L_\tau(u)$ with a positive τ pursues both small within-class scatter and small misclassification error. In regular binary classification problems, the data within one class have centrality. Then it is reasonable to minimize the within-class scatter. However, for some applications, the data come from mixture distributions, for which the within-class scatter is not small.

In this paper, we use a toy example to illustrate this situation. In this example, the features x_i in each class come from a mixture distribution consisting of two Gaussian distributions. The contour maps of the distributions are displayed by the curves in Fig.5(a). We can choose a positive τ, e.g., $\tau = 0.3$, to find a classifier with a small within-class scatter. The obtained classifier is shown by the dotted line, which seems not suitable for this problem. Instead, we should put more emphasis on misclassification error and consider a small τ. The dashed line in Fig.5(a) corresponds to $\tau = -0.3$ and the solid line corresponds to $\tau = -0.08$. The probability density functions of $yf(x)$ are given in Fig.5(b), from which one can observe that $\tau = -0.08$ provides a good classifier but its within-class scatter is not small.

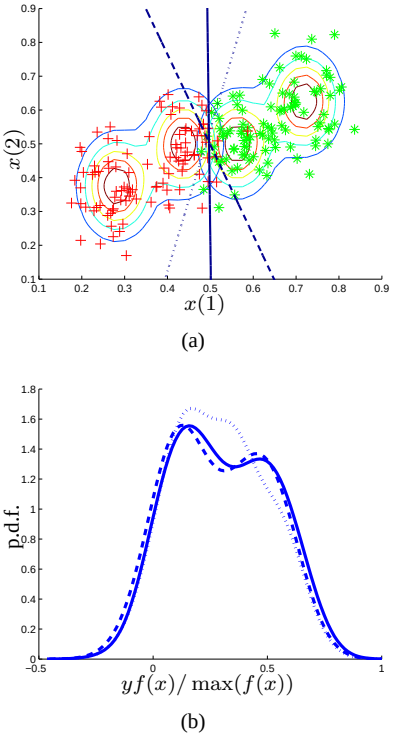

Fig. 5. Classification performance when the data are generated from mixture distributions. (a) The level sets of the feature distributions are shown by curves. The samplings data of two classes are given by green stars and red crosses. The dotted, solid, and dashed lines display the classifier trained by pin-SVM with $\tau = 0.3, -0.08$, and $\tau = -0.3$, respectively. (b) The probability density functions of $yf(x)$ for pin-SVM with $\tau = 0.3$ (dotted line), -0.08 (solid line), and $\tau = -0.3$ (dashed line). Notice that to give a clear comparison, $yf(x)$ is normalized.

A similar phenomenon takes place for the "Spect" data set (downloaded from the UCI Repository of Machine Learning Datasets, [8]). As reported by [5], the classification accuracy of $\tau = 0$ is better than that of a positive τ value. From the probability density function of $y_i f(x_i)$ (see Fig.10 in [5]), we can observe a mixture distribution. Then, one should put more emphasis on misclassification error by using negative τ values. Otherwise, a positive τ could be suitable. We test different τ values and report the classification accuracy on "Spect", "Monk1", "Monk2", and "Monk3" in Fig.6. The basic finding from these examples is that we may need different τ values for different problems.

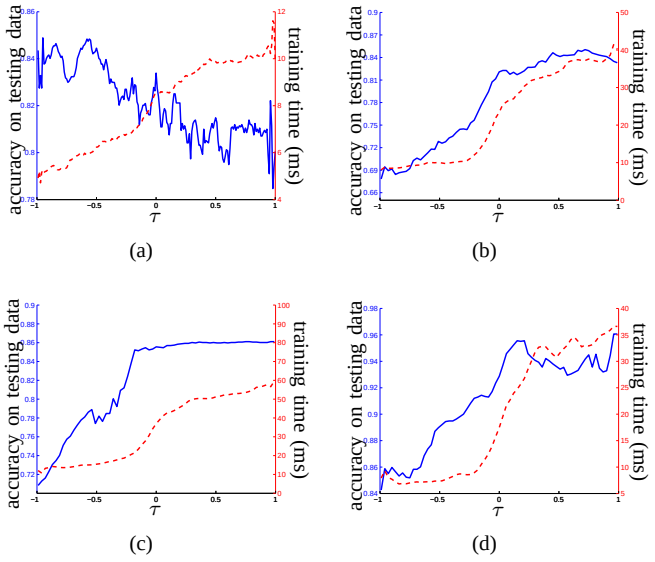

Fig. 6. The accuracy (blue solid lines) and the training time (red dashed lines) for different τ values. The data sets include (a) Spect; (b) Monk1; (c) Monk2; (d) Monk3. The training time and τ value are positive correlated. The best testing accuracy is achieved at different τ values for different sets.

C. Solving method and training time

As discussed above, by selecting a suitable τ, pin-SVM can handle well various problems and different noise levels. In the dual space, pin-SVM can be formulated as (6), which is a quadratic programming problem constrained in $-\tau C_i \leq \lambda_i \leq C_i$ as illustrated in Fig.2. The constraints on the dual variables are not coupled, from which it follows that one can update only a part of the dual variables and keep the remaining unchanged. This method is called a sequential minimal optimization (SMO), which is one of the most popular methods for solving SVMs in the dual space. For C-SVM, the corresponding SMO algorithms can be found in [9]–[12]. For pin-SVM, the SMO algorithm has been established in [13]. The numerical experiments therein shows its effectiveness on solving pin-SVM for different τ values.

Recall (6) for the dual formulation of pin-SVM. One can see that τ controls the size of the feasible set. When the set is large enough, the constraints of (6) are not active at the optimal solution. Then, the optimal solution can be easily calculated from an unconstrained quadratic problem. On the other hand, if we set $\tau = -1$, the feasible set becomes a singleton $\lambda_i = C_i$ and we can obtain the optimal solution without training. Thus, the training time is not strictly monotone with respect to τ.

We consider the training time by the experiments in the last subsection. Pin-SVM is trained by the SMO and the training times for different τ are reported in Fig.6. Generally, pin-SVM with a negative τ is related to less training time.

D. Particular cases $\tau = -1$ and $\tau = 0$

Among all the choices, $\tau = -1$ is a particular case for which the optimal dual variables can be obtained directly. One can verify that when C_i are set as (4), the bias term has no effect on the objective value of the primal problem (2). We simply set b equal to zero. Then the function corresponding to pin-SVM with $\tau = -1$ is

$$f(x) = \sum_{i:y_i=+1} C_i \mathcal{K}(x, x_i) - \sum_{i:y_i=-1} C_i \mathcal{K}(x, x_i).$$

If a linear kernel $\mathcal{K}(x, x_i) = x^T x_i$ is used, the decision rule becomes

$$\mathrm{sgn}(f(x)) = \begin{cases} +1, & \text{if } x^T \bar{x}_+ > x^T \bar{x}_-, \\ -1, & \text{if } x^T \bar{x}_+ < x^T \bar{x}_-, \end{cases}$$

where $\bar{x}_+$ and $\bar{x}_-$ are the mean of $\{x_i,\ i : y_i = +1\}$ and $\{x_i,\ i : y_i = -1\}$, respectively. In other words, we use the angles between x and the centers of the two classes to determine the label. When a nonlinear kernel is used, pin-SVM with $\tau = -1$ is to classify the data according to the angle in the feature space.

Another particular situation is $\tau = 0$, which corresponds to C-SVM. In pin-SVM, the dual variables can be categorized into three types: *lower bounded support vectors* ($\lambda_i = -\tau C$), *free support vectors* ($-\tau C < \lambda_i < C$), and *upper bounded support vectors* ($\lambda_i = C$). When $\tau = 0$, the lower bounded support vectors become zero. This brings sparseness, which is meaningful for reducing the storage space for support vectors, but is not necessary in the viewpoint of accuracy.

IV. Numerical Experiments

In this section, we will illustrate the performance of pin-SVM with different τ values on real-life data sets. The involved data are downloaded from the UCI Repository of Machine Learning Dataset [8] and LIBSVM data sets [12]. We randomly select m observations to train the classifier and use the remaining for testing. All the experiments are done in Matlab 2013a in Core i5-1.80 GHz, 4.0GB RAM.

We will consider $\tau = -1, -0.3, -0.1, 0, 0.1, 0.3, 0.5$. As analyzed before, no optimization problem needs to be solved for $\tau = -1$. For other cases, we will apply the SMO algorithm in [13] to train pin-SVM (6). In this experiment, the RBF kernel is used and C_i is set according to (4). When the data size is less than 10000, the regularization coefficient C_0 and the bandwidth in the RBF kernel σ are tuned by 10-fold cross-validation. Otherwise, we set $C_0 = 1$ and tune σ only. The above training and testing process are repeated 10 times. Then the average classification accuracy on the testing sets and the standard deviation are reported in Table I, where the size of the training sets and average computing time are given as well.

TABLE I
TESTING ACCURACY AND THE AVERAGE TRAINING TIME

Data	m	$\tau=-1.0$	$\tau=-0.3$	$\tau=-0.1$	$\tau=0$	$\tau=0.1$	$\tau=0.3$	$\tau=0.5$
Haber.	150	70.99 ± 4.98	72.68 ± 3.35	73.38 ± 2.78	74.27 ± 2.53	73.63 ± 2.96	72.61 ± 4.33	**74.52** ± 2.63
		—	13.8 ms	18.0 ms	26.5 ms	24.4 ms	24.9 ms	24.6 ms
Statlog	150	81.49 ± 2.76	83.22 ± 2.84	82.90 ± 2.20	82.82 ± 2.05	**83.40** ± 2.01	83.15 ± 1.42	82.32 ± 2.39
		—	10.1 ms	14.61 ms	24.2 ms	27.8 ms	30.2 ms	32.8 ms
Iono.	200	85.07 ± 1.76	92.17 ± 1.95	93.49 ± 1.16	94.01 ± 1.22	**94.08** ± 1.39	93.42 ± 1.16	93.62 ± 1.32
		—	13.6 ms	18.7 ms	32.8 ms	40.6 ms	44.4 ms	47.4 ms
Pima	300	70.23 ± 2.03	**74.26** ± 1.25	74.20 ± 1.78	73.56 ± 1.65	73.45 ± 1.93	73.58 ± 1.68	73.48 ± 1.77
		—	21.3 ms	32.2 ms	55.9 ms	59.9 ms	67.2 ms	73.2 ms
	500	72.12 ± 2.53	73.90 ± 1.65	73.53 ± 2.78	74.14 ± 2.45	**74.51** ± 3.39	73.77 ± 2.54	73.36 ± 3.14
		—	42.2 ms	64.8 ms	111 ms	126 ms	135 ms	144 ms
Breast	300	95.85 ± 0.58	96.97 ± 0.89	**97.30** ± 0.81	96.42 ± 0.94	96.45 ± 0.81	96.27 ± 0.80	96.22 ± 0.85
		—	13.2 ms	16.7 ms	31.4 ms	45.2 ms	48.2 ms	49.7 ms
	500	95.25 ± 0.95	97.10 ± 1.51	**97.20** ± 1.44	95.65 ± 1.42	95.60 ± 1.66	95.30 ± 2.02	95.45 ± 1.69
		—	26.2 ms	31.8 ms	57.4 ms	71.3 ms	73.9 ms	74.0 ms
Trans.	300	70.11 ± 3.46	73.34 ± 2.50	73.32 ± 2.48	73.64 ± 2.42	73.70 ± 2.02	73.74 ± 2.57	**74.15** ± 2.28
		—	25.8 ms	28.7 ms	34.5 ms	28.4 ms	29.0 ms	28.8 ms
	500	74.03 ± 4.50	74.01 ± 2.16	73.94 ± 2.68	75.55 ± 2.83	75.50 ± 2.88	**76.14** ± 3.01	**76.14** ± 3.07
		—	47.3 ms	52.3 ms	57.6 ms	45.1 ms	45.1 ms	45.3 ms
Splice	500	70.44 ± 0.89	84.58 ± 0.60	84.80 ± 0.72	85.72 ± 3.34	86.12 ± 0.70	85.93 ± 1.18	**86.25** ± 1.04
		—	49.9 ms	76.1 ms	102 ms	93.2 ms	98.9 ms	99.3 ms
	1000	72.13 ± 0.74	86.73 ± 0.10	87.96 ± 0.03	**88.20** ± 2.44	87.98 ± 1.79	87.63 ± 1.36	87.78 ± 1.55
		—	120 ms	181 ms	276 ms	237 ms	235 ms	250 ms
Magic	10000	74.98 ± 0.27	79.25 ± 0.27	83.94 ± 0.20	85.01 ± 0.24	**85.15** ± 0.48	84.31 ± 0.44	83.79 ± 0.69
		—	14.6 s	18.4 s	23.4 s	29.0 s	30.1 s	30.3 s
	15000	74.85 ± 0.75	79.28 ± 0.70	84.10 ± 0.63	**85.66** ± 0.62	85.01 ± 0.09	84.49 ± 0.58	83.88 ± 0.90
		—	42.6 s	50.5 s	61.8 s	70.8 s	79.0 s	78.4 s
RNA	10000	86.30 ± 3.10	95.40 ± 0.19	**96.09** ± 0.06	95.76 ± 0.68	93.63 ± 0.43	94.06 ± 0.87	93.61 ± 0.72
		—	11.3 s	12.5 s	15.8 s	14.2 s	15.6 s	15.7 s
	30000	83.82 ± 3.86	95.60 ± 0.06	**96.21** ± 0.03	94.26 ± 1.05	92.26 ± 0.95	92.11 ± 1.11	91.08 ± 0.89
		—	98.0 s	102 s	123 s	114 s	124 s	141 s
IJCNN1	20000	71.87 ± 1.44	92.15 ± 0.50	93.46 ± 0.81	92.62 ± 0.65	**93.75** ± 1.13	92.42 ± 0.91	92.12 ± 1.11
		—	98.8 s	121 s	147 s	212 s	213 s	209 s
	45000	74.21 ± 2.81	92.44 ± 0.22	**94.01** ± 0.06	93.37 ± 0.34	93.60 ± 0.45	93.20 ± 0.26	90.82 ± 0.44
		—	434 s	463 s	568 s	979 s	1002 s	1056 s

According to the reported results, we can conclude that it is not necessary to require $\tau \geq 0$ and a properly selected τ could bring accurate results. The training time on the experiments also coincides with the previous analysis. An interesting observation is that a small τ seems suitable when there are many training points. The reason can be explained from the quantile analysis as discussed in Section III.A. Heuristically, a negative τ may be a good choice when there are many training data. Moreover, a small τ may correspond to a small training time. The extreme choice $\tau = -1$ is also good to consider, taking into consideration that one does not need to solve a quadratic programming.

V. CONCLUSION

Support vector machines with the pinball loss have a tuning parameter τ, which is the slope of the loss function. It controls the feasible set in the dual space. Traditionally, τ is fixed to be zero (corresponding to C-SVM) or a positive τ. However, the nonnegative condition on τ is not necessary and only $\tau \geq -1$ is needed to keep the problem convex. In this paper, we extended pin-SVM to negative τ values and proved that the pinball loss with $\tau \geq -1$ remains calibrated. The properties for different τ values (positive, zero, and negative) have been discussed and the performance has been evaluated on numerical experiments. The results imply that pin-SVM with different τ values is a promising method. It is concluded and proposed to consider both positive and negative values for training classifiers.

REFERENCES

[1] C. Cortes and V. Vapnik, "Support-vector networks", *Machine Learning*, vol. 20, no. 3, pp. 273–297, 1995.

[2] V. Vapnik, *Statistical Learning Theory*. Wiley, New York, 1998.

[3] R. Koenker, *Quantile Regression*. Cambridge University Press, 2005.

[4] I. Steinwart and A. Christmann, "Estimating conditional quantiles with the help of the pinball loss," *Bernoulli*, vol. 17, no. 1, pp. 211–225, 2011.

[5] X. Huang, L. Shi, and J.A.K. Suykens, "Support vector machine classifier with pinball loss," to appear in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, doi:10.1109/TPAMI.2013.178.

[6] C. Bishop, *Neural Networks for Pattern Recognition*, Clarendon Press, 1995.

[7] T. Van Gestel, J.A.K. Suykens, G. Lanckriet, A. Lambrechts, B. De Moor, and J. Vandewalle, "Bayesian framework for least-squares support vector machine classifiers, gaussian processes, and kernel fisher discriminant analysis," *Neural Computation*, vol. 14, no. 5, pp. 1115–1147, 2002.

[8] A. Frank and A. Asuncion, "UCI machine learning repository," 2010. [Online]. Available: http://archive.ics.uci.edu/ml

[9] J. C. Platt, "Fast training of support vector machines using sequential minimal optimization," in *Advances in Kernel Methods – Support Vector Learning*. MIT Press, Cambridge, 1999, pp. 185–208.

[10] R.-E. Fan, P.-H. Chen, and C.-J. Lin, "Working set selection using second order information for training support vector machines," *The Journal of Machine Learning Research*, vol. 6, pp. 1889–1918, 2005.

[11] L. Bottou and C.-J. Lin, "Support vector machine solvers," in *Large Scale Kernel machines*, MIT Press, Cambridge, 2007, pp. 301–320.

[12] C.-C. Chang and C.-J. Lin, "LIBSVM: a library for support vector machines," *ACM Transactions on Intelligent Systems and Technology*, vol. 2, no. 3, p. 27, 2011.

[13] X. Huang, L. Shi, and J.A.K. Suykens, "Sequential minimial optimization for SVM with pinball loss," Internal Report 14-10, ESAT-STADIUS, KU Leuven, 2014.