Hindawi
International Journal of Intelligent Systems
Volume 2023, Article ID 2201330, 22 pages
https://doi.org/10.1155/2023/2201330

WILEY | Hindawi

Research Article

Capped Asymmetric Elastic Net Support Vector Machine for Robust Binary Classification

Kai Qi and Hu Yang

College of Mathematics and Statistics, Chongqing University, Chongqing, China

Correspondence should be addressed to Hu Yang; yh@cqu.edu.cn

Received 27 September 2022; Revised 30 December 2022; Accepted 6 January 2023; Published 21 February 2023

Academic Editor: B. B. Gupta

Copyright © 2023 Kai Qi and Hu Yang. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Recently, there are lots of literature on improving the robustness of SVM by constructing nonconvex functions, but they seldom theoretically study the robust property of the constructed functions. In this paper, based on our recent work, we present a novel capped asymmetric elastic net (CaEN) loss and equip it with the SVM as CaENSVM. We derive the influence function of the estimators of the CaENSVM to theoretically explain the robustness of the proposed method. Our results can be easily extended to other similar nonconvex loss functions. We further show that the influence function of the CaENSVM is bounded, so that the robustness of the CaENSVM can be theoretically explained. Other theoretical analysis demonstrates that the CaENSVM satisfies the Bayes rule and the corresponding generalization error bound based on Rademacher complexity guarantees its good generalization capability. Since CaEN loss is concave, we implement an efficient DC procedure based on the stochastic gradient descent algorithm (Pegasos) to solve the optimization problem. A host of experiments are conducted to verify the effectiveness of our proposed CaENSVM model.

1. Introduction

Support vector machine (SVM), first proposed by Cortes and Vapnik [1], is a powerful binary classification tool and has been widely used in various fields, such as bioinformatics analysis [2, 3], industrial flaw detection [4], and financial forecasting [5]. On the one hand, the SVM can be easily understood in the geometric view, i.e., it aims to seek a single separating hyperplane for classifying datasets. On the other hand, there is a solid statistical theory basis behind to well guarantee the classification performance of the SVM [6–8]. Thus, it has been drawing much attention to study the SVM [9–15]. Although a host of literature demonstrate the advantages of support vector classifiers, there is still a room for improvement.

One drawback is the sensitivity to feature noise or more specifically the instability for resampling. In fact, the SVM can be fit in the regularization framework of loss + penalty by adopting hinge loss, i.e., $l_{\text{hinge}}(u) = \max(u, 0)$. Huang et al. [11] pointed out that the hinge loss-based SVM lacks resistance to feature noise and the final separating hyperplane is severely disturbed by the feature noise around the decision boundary. To tackle this problem, motivated by the quantile in the statistical field, Huang et al. [11] constructed the so-called pinball loss and applied it to the SVM to propose PinSVM. Later, Xu et al. [16] extended this idea to the twin support vector machine, which can simultaneously obtain a pair of nonparallel separating hyperplanes. There are two typical defects of the pinball loss. The first defect is the heavy optimization burden caused by the singularity of the pinball loss function at zero. Huang et al. [17] considered an asymmetric least squared loss, which is also stable for resampling but is smooth everywhere. A similar idea was studied by Liu et al. [18]. Unlike them, Li and Lv [19] utilized Chen–Harker–Kanzow–Smale function to construct a smooth approximation of the pinball loss. The second defect is the lack of sparseness. According to the pinball loss function, those correctly classified training samples still produce losses, which is completely different from the hinge loss and increases the training cost. To enhance the

sparseness, Huang et al. [11] introduced the pinball loss with ϵ-insensitive zone, which can achieve sparsity and maintain the stability simultaneously. Shen et al. [20] truncated the left part of the pinball loss and proposed $\overline{pin}$-SVM, providing a more flexible framework for the tradeoff between sparsity and stability. Yang and Xu [21] applied a safe screening rule for accelerating the PinSVM. More PinSVM-related work can be found in [22–27].

Another drawback is the sensitivity to label noise (outliers). Since $l_{\text{hinge}}(u)$ tends to infinite as $u \longrightarrow \infty$, outliers often produce large losses, indicating that the resulting decision hyperplane is possibly deviated. Wu and Liu [28] suggested a truncated hinge loss, termed as ramp loss, to suppress the influences of outliers. Based on the ramp loss and motivated by the Huberized scheme, Wang et al. [29] proposed a smooth ramp loss function, which is twice differentiable and can be efficiently solved. Liu et al. [30] applied ramp loss to a nonparallel support vector machine. Tang et al. [13] combined the pinball loss with ramp loss to propose the valley loss, which is both stable for resampling and robust to outliers. For more related literature, one can refer to [31–34]. Another different strategy to improve robustness is using correntropy-induced loss (C-loss) [35]. Based on C-loss, Xu et al. [36] proposed the rescaled hinge loss. Due to the properties of an exponential function, the rescaled hinge loss is bounded and the induced support vector classifier is insensitive to label noise. Yang and Dong [37] applied the idea of the pinball loss to C-loss and proposed a new generalized quantile loss, which can be viewed as a rescaled version of the pinball loss. The generalized quantile loss inherits the stability for resampling from the pinball loss and the robustness to outliers from C-loss. Similarly, we recently proposed a joint rescaled asymmetric least squared (RaLS) loss and applied it to a nonparallel support vector machine [38]. The proposed RaLS loss is smooth everywhere and enjoys both stability and robustness.

Inspired by the previous work, we construct a novel capped asymmetric elastic net (CaEN) loss and apply it to the SVM (CaENSVM) in this paper. As a generalization of the pinball loss and ramp loss, the designed CaEN loss is bounded and asymmetric, as well. However, it is more flexible than the previous truncated pinball loss functions. To demonstrate its advantages, we theoretically investigate several properties of the CaEN loss, including noise insensitivity, Bayes rule, and generalization error bound. The main contributions of this work can be summarized as follows:

(i) A novel capped asymmetric elastic net (CaEN) loss is proposed to achieve stability for resampling and robustness to outliers simultaneously. The advantages of the elastic net (EN) loss for the SVM were theoretically discussed in our recent work [3]. The derived VTUB significantly characterizes the advantage of the EN loss. Thus, it is meaningful to improve the performance of the EN loss under the framework of the SVM.

(ii) We derive the influence function (see Theorem 1) to demonstrate the robustness of the CaENSVM. Though there are lots of literature on improving the robustness of the SVM by constructing a nonconvex function, they seldom theoretically show the robust property of the constructed functions. Obviously, theoretical results can ensure the effectiveness of similar works. We use the influence function in the statistics to show the robustness of our constructed CaEN loss. Our results can be easily extended to other similar loss functions, such as ramp-type losses and rescaled-type losses. We further show the influence function of the CaENSVM is bounded, so that the robustness of the CaENSVM can be theoretically explained.

(iii) CaEN loss is proved to be equivalent to the Bayes rule, and the corresponding generalization error bound based on Rademacher complexity well guarantees a good generalization capability.

(iv) CaENSVM is applied to deal with a real problem, i.e., handwritten digit recognition. Experimental results show that CaENSVM is superior to many state-of-the-art methods.

The remainder of this paper is organized as follows: In Section 2, we introduce several related studies. In Section 3, we first formulate the proposed CaENSVM. Then, an efficient DC procedure based on the stochastic gradient descent algorithm is implemented for optimizing the CaENSVM problem. Theoretical analysis on the properties of the CaENSVM, including noise insensitivity, Bayes rule, and generalization error bound, is carefully discussed in Section 4. We conducted lots of experiments in Section 5 to investigate the performance of the CaENSVM. In Section 6, a conclusion summarizes the main contributions and further potential directions.

2. Background

In this section, we review several related works. Considering a binary classification problem with n training samples and p features, let $x_i \in \mathbb{R}^{p\times 1}$ and $y_i \in \{+1, -1\}$ be i-th instance and the corresponding label, respectively. All samples are reorganized as a data matrix $X \in \mathbb{R}^{n\times p}$. Without particular explanations, all vectors are in column form.

2.1. SVM with Elastic Net Loss. Recently, Qi et al. [39] proposed a so-called elastic net (EN) loss given as follows:

$$l_{EN}(u; c_1, c_2) = \begin{cases} \frac{c_1}{2}u^2 + c_2 u, & u \geq 0, \\ 0, & u < 0, \end{cases} \tag{1}$$

where c_1 and c_2 are both two positive tuning parameters. EN loss is a fusion of the standard hinge loss and the squared hinge loss [9]. Figure 1(a) shows the shapes of the EN loss with different sets of parameters.

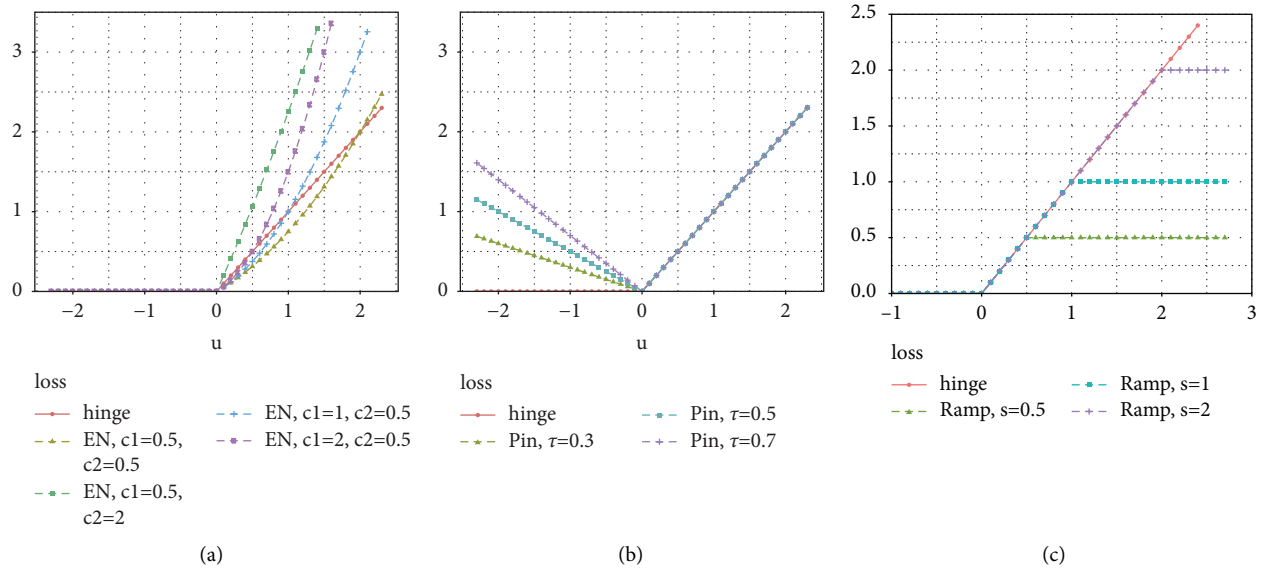

Figure 1: Different types of loss functions: (a) "EN loss" is an elastic net loss, (b) "pin loss" is a pinball loss, and (c) "ramp loss" is a ramp loss.

Based on EN loss, Qi et al. [39] constructed the following elastic net support vector machine (ENSVM):

$$\min_{w,b} \frac{1}{2}\left(\|w\|_2^2 + b^2\right) + \sum_{i=1}^{n} l_{EN}\left(1 - y_i\left(w^T x_i + b\right); c_1, c_2\right), \tag{2}$$

where $w \in \mathbb{R}^{p\times 1}$ and $b \in \mathbb{R}$ are the normal vector and the intercept of the separating hyperplane, respectively. After obtaining w and b from (2), the decision function of a new sample x_{new} is $f(x_{\text{new}}) = \text{sign}(w^T x_{\text{new}} + b)$, where $\text{sign}(\cdot)$ is a sign function, which maps a real number to its sign and zero to zero.

We can equivalently transform (2) into the following constrained optimization problem:

$$\begin{aligned} &\min_{w,b} \frac{1}{2}\left(\|w\|_2^2 + b^2\right) + \frac{c_1}{2}\xi^T\xi + c_2 e^T\xi, \\ &\text{s.t. } D(Xw + eb) \geq e - \xi, \xi \geq \mathbf{0}, \end{aligned} \tag{3}$$

where $D = \text{diag}(y_1, \cdots, y_n) \in \mathbb{R}^{n\times n}$ is a diagonal matrix, $e \in \mathbb{R}^{n\times 1}$ is filled with ones, and $\xi = (\xi_1, \cdots, \xi_n)^T \in \mathbb{R}^n$ is a slack variable. For ENSVM (3), one can clearly see that the ENSVM resembles the standard SVM [1] with $c_1 = 0$ and reduces to the Lagrangian SVM [9] with $c_2 = 0$. Thus, the ENSVM is more flexible. Moreover, in our recent work [3], we derived the so-called VTUB for the SVM with the EN loss to demonstrate its unique advantages. Thus, it is meaningful to improve the performance of the EN loss under the framework of the SVM.

2.2. SVM with Pinball Loss. Huang et al. [10] proved that the support vector classifiers with hinge-type losses, including ENSVM, are sensitive to feature noise, or specifically, are unstable for resampling. Motivated by the quantile in the statistical field, Huang et al. [10] proposed a novel pinball loss defined as

$$l_{\text{Pin}}(u;\tau) = \begin{cases} u, & u \geq 0, \\ -\tau u, & u < 0, \end{cases} \tag{4}$$

where $\tau \in [0, 1]$ controls the level of stability for resampling. Figure 1(b) illustrates the shapes of the pin loss with different values of τ. As the figure shows, unlike the hinge-type losses, l_{Pin} also produces losses for $u < 0$, which can benefit the classifier for balancing the disturbance of the feature noise around the decision boundary (see subsection 3.3 in the study by Huang et al. [10] for details).

Following the method of formulating the hinge loss-based SVM, Huang et al. [10] constructed the pinball loss SVM (PinSVM) as

$$\min_{w,b} \frac{1}{2}\|w\|_2^2 + c\sum_{i=1}^{n} l_{\text{Pin}}\left(1 - y_i\left(w^T x_i + b\right); \tau\right), \tag{5}$$

where $c \geq 0$ is a tuning parameter. We can also equivalently rewrite (5) as

$$\begin{aligned} &\min_{w,b} \frac{1}{2}\left(\|w\|_2^2 + b^2\right) + ce^T\xi, \\ &\text{s.t. } D(Xw + eb) \geq e - \xi, \\ &D(Xw + eb) \leq e + \frac{1}{\tau}\xi. \end{aligned} \tag{6}$$

Note that if $\tau = 0$, the second constraint of problem (6) turns to $\xi \geq \mathbf{0}$, and PinSVM reduces to the standard SVM with hinge loss. In other words, the PinSVM can be viewed as a generalization of the standard SVM.

2.3. SVM with Ramp Loss. Since the values of losses produced by the hinge-type loss [1, 9, 39] and the pinball-type loss [11, 17, 18] functions tend to infinite when $u \longrightarrow \infty$, the support vector classifiers induced by these losses are sensitive to label noise (outliers). To reduce the influence of label

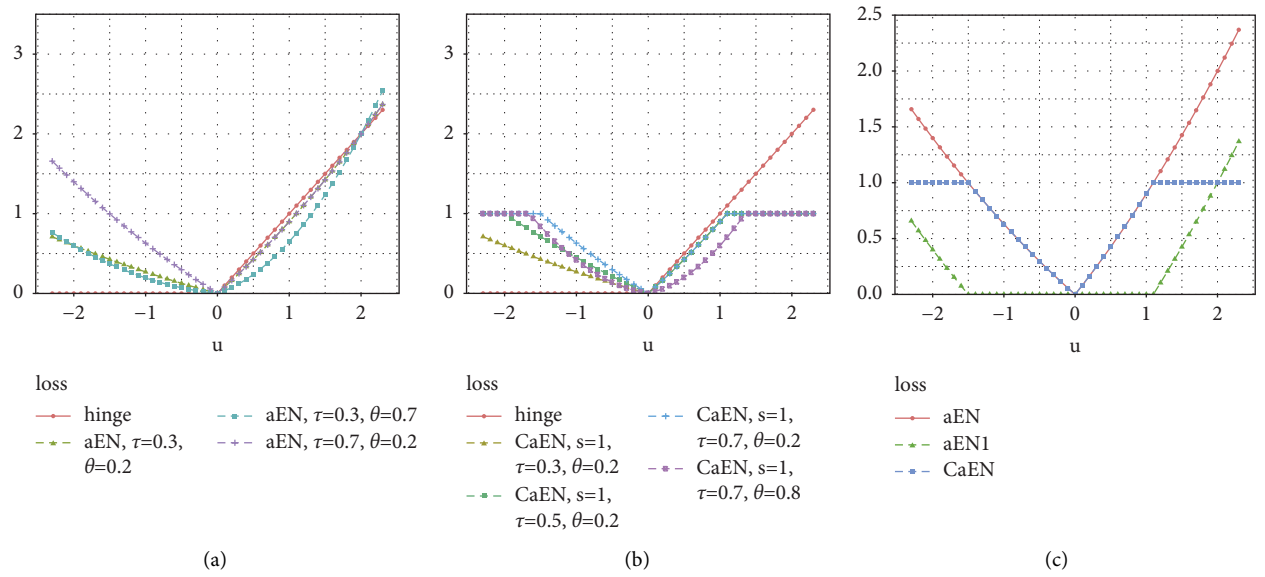

FIGURE 2: (a) "aEN loss" is an asymmetric elastic net loss. (b) "CaEN loss" is a capped asymmetric elastic net loss. (c) The convex decomposition of CaEN loss, where $s = 1, \tau = 0.7$, and $\theta = 0.2$.

noise, Wu and Liu [28] truncated the hinge loss and proposed the so-called ramp loss, which is defined as

$$l_{\text{Ramp}}(u; s) = \begin{cases} 0, & u < 0, \\ u, & 0 \le u \le s, \\ s, & u > s, \end{cases} \tag{7}$$

where $s > 0$ controls the truncation level. Figure 1(c) shows different shapes of the ramp loss. As compared with the hinge-type losses, ramp loss is upper bounded, such that it can weaken the disturbance of label noise.

Applying ramp loss to the SVM, we can obtain RampSVM, i.e.,

$$\min_{w,b} \frac{1}{2}\|w\|_2^2 + c\sum_{i=1}^{n} l_{\text{Ramp}}\left(1 - y_i\left(w^T x_i + b\right); s\right), \tag{8}$$

where $c > 0$ is a tuning parameter.

3. Capped Asymmetric Elastic Net Loss-Based SVM

3.1. The CaENSVM Model. In our recent work [3], we derived the so-called VTUB to demonstrate the unique advantages of the EN loss under the framework of the SVM. Thus, it is meaningful to improve the performance of the EN loss. Motivated by the pinball loss, we designed the following asymmetric elastic net (aEN) loss:

$$l_{aEN}(u; \tau, \theta) = \begin{cases} \dfrac{\theta}{2}u^2 + (1-\theta)u, & u \ge 0, \\ \tau\left(\dfrac{\theta}{2}u^2 - (1-\theta)u\right), & u < 0, \end{cases} \tag{9}$$

where $\theta \in [0, 1]$ corresponds to a tradeoff between L_1 norm and L_2 norm, $\tau \in [0, 1]$ is a tuning parameter controlling the bias of the penalization for positive and negative losses. Figure 2(a) illustrates different shapes of aEN losses with different sets of tuning parameters. According to definition (9), l_{aEN} can be regarded as a generalization of elastic net loss, pinball loss, and asymmetric least squared loss functions [17], since l_{aEN} reduces to l_{EN} for $\tau = 0$, l_{aEN} becomes l_{Pin} for $\theta = 0$ and l_{aEN} is equivalent to l_{aLS} for $\theta = 1$.

According to (9), l_{aEN} goes to infinity along with $u \longrightarrow \infty$, so the proposed aEN loss is also sensitive to outliers (label noise). To improve its robustness against outliers, we further use capped trick to propose a capped asymmetric elastic net (CaEN) loss function, which is defined as

$$l_{CaEN}(u; \tau, \theta, s) = \min\left(l_{aEN}(u), s\right), \tag{10}$$

where $s > 0$ is a thresholding parameter. Figure 2(b) depicts the shapes of the CaEN loss with different groups of parameters. As shown in Figure 2(b) , the proposed CaEN loss function is exactly upper bounded and concave. The truncation of the left part of the CaEN loss can increase the sparseness [13, 20], while the boundness of the right part of the CaEN loss can enhance its robustness against label noise. Detailed discussion on the properties of CaENSVM is provided in Section 4. However, the concavity of the CaEN loss often involves high optimization costs.

Combining CaEN loss with the standard SVM, we propose a novel robust CaEN loss-based SVM (termed as CaENSVM), which is formulated as

$$\min_{w} \frac{1}{2}\|w\|_2^2 + \frac{c}{n}\sum_{i=1}^{n} l_{CaEN}\left(1 - y_i w^T x_i; \tau, \theta, s\right), \tag{11}$$

where $c > 0$ is the tuning parameter. We remove the intercept term b in (11) for simplicity, which can be achieved before training by centering features just like [10, 13]. After obtaining w from (11), the decision function of the linear CaENSVM for a new sample x_{new} is $f(x_{\text{new}}) = \text{sign}(w^T x_{\text{new}})$.

Require: $k = 0$; $\Theta^{(0)}$, the initial value of Θ.
Ensure: optimal solution of (15).
(1) **repeat**
(2) $\Theta^{(k+1)} = \operatorname{argmin}_{\Theta} g(\Theta) - h'(\Theta^{(k)})^T(\Theta - \Theta^{(k)})$, h' is the derivative of h with respect to Θ.
(3) **until** convergence.
(4) **return** $\Theta^{(k+1)}$.

Algorithm 1: A general framework of the DC algorithm for (15).

For nonlinear CaENSVM, we consider the following kernel-generated separating hyperplane [40, 41]:

$$w^T\phi(x) = 0, \tag{12}$$

where $\phi(x)$ maps x to a high-dimensional Hilbert space. In application, we often utilize kernel trick, i.e., $\phi(x) = K(X, x)$, where $K(\cdot,\cdot)$ is a kernel function and $K(X), (x) = (K)(x_1, (x), \cdots, K(x_n, (x)^T$. Then, by replacing $w^T x_i$ with $w^T K(X, x_i)$ in (11), we can obtain the nonlinear CaENSVM model. For determining the class of a new sample x_{new}, we only need to replace x_{new} with $K(X, x_{\text{new}})$ in the linear decision function.

3.2. DC Algorithm for CaENSVM. The truncation for aEN loss results in a nonconvex loss, indicating that solving CaENSVM (11) involves nonconvex minimization, which is often difficult. Note that, though CaEN loss is concave, we can decompose l_{CaEN} into the difference of two convex functions, i.e.,

$$l_{CaEN}(u; \tau, \theta, s) = l_{aEN}(u; \tau, \theta) - l_{aEN1}(u; \tau, \theta, s), \tag{13}$$

where l_{aEN1} corresponds to the so-called aEN1 loss, which is given as follows:

$$l_{aEN1}(u; \tau, \theta, s) = \max\left(l_{aEN}(u; \tau, \theta) - s, 0\right). \tag{14}$$

Figure 2(c) depicts the above convex decomposition of the CaEN loss. As the figure shows, both aEN and aEN1 losses are exactly convex. By calculating the difference of aEN and aEN1 losses, we can finally obtain the nonconvex CaEN loss. Using this property of the CaEN loss, we apply the DC (difference of convex functions) algorithm [42] to optimize problem (11).

Considering the following optimization problem:

$$\min_{\Theta} f(\Theta) = g(\Theta) - h(\Theta), \Theta \in \mathbb{R}^m, \tag{15}$$

where g and h are both convex functions on $\mathbb{R}^m$. To solve problem (15), the DC algorithm turns to minimize a sequence of convex subproblems. A general framework of the DC algorithm for (15) is illustrated as Algorithm 1.

Recalling (11), based on the decomposition (13), we can reformulate CaENSVM optimization problem as

$$w^* = \operatorname*{argmin}_{w} \underbrace{\frac{1}{2}\|w\|_2^2 + \frac{c}{n}\sum_{i=1}^{n} l_{aEN}\left(1 - y_i w^T x_i; \tau, \theta\right)}_{g} \underbrace{- \frac{c}{n}\sum_{i=1}^{n} l_{aEN1}\left(1 - y_i w^T x_i; \tau, \theta, s\right)}_{h}, \tag{16}$$

where g and h are both convex functions on $\mathbb{R}^p$.

According to the DC algorithm, we have to calculate the derivative of h with respect to w. Since l_{aEN1} in (16) has sharp points, h is also nondifferentiable. Thus, we utilize the subgradient instead of the derivative. For a set of tuning parameters (τ, θ, s), the subgradient of l_{aEN1} with respect to w is given as follows:

$$\nabla l_{aEN1}(w)_i = \begin{cases} \tau\left(\theta x_i x_i^T w + (1 - 2\theta) y_i x_i\right), & 1 - y_i w^T x_i < u_1, \\ \mathbf{0}, & u_1 < 1 - y_i w^T x_i < u_2, \\ \theta x_i x_i^T w - y_i x_i, & 1 - y_i w^T x_i > u_2, \end{cases} \tag{17}$$

where u_1 and $u_2\,(u_1 < 0 < u_2)$ are two sharp points of both $l_{aEN1}(u)$ and $l_{CaEN}(u)$ except for zero, which can be easily calculated from (9) and (14) and are given as follows:

Require: T_1, T_2, eps, $w^{(0)}$, $\{(x_i^T, y_i)\}_{i=1}^n$, c, τ, θ, and s.
Ensure: optimal solution of (11).
Set $t = 0$, $v^{(0)} = w^{(0)}$.
while $(t \le T_2)$ **do**
  Choose $A_t \subset \{1, 2, \cdots, n\}$, where $|A_t| = m$, uniformly at random.
  Compute $\nabla F(v^{(t)})$ by (21) and (22).
  Set $\eta_t = c/t$.
  Set $v^{(t+1)} \leftarrow v^{(t)} - \eta_t \nabla F(v^{(t)})$.
end while
Set $w^{(1)} = v^{(t+1)}$.
Set $k = 0$.
while $(k \le T_1$ & $\|w^{(k+1)} - w^{(k)}\| \ge \text{eps})$ **do**
  Set $k \leftarrow k + 1$.
  Set $t = 0$, $v^{(0)} = w^{(k)}$.
  while $(t \le T_2)$ **do**
    Choose $A_t \subset \{1, 2, \cdots, n\}$, where $|A_t| = m$, uniformly at random.
    Compute $\nabla F(v^{(t)})$ by (21) and (22).
    Set $\eta_t = c/t$.
    Set $v^{(t+1)} \leftarrow v^{(t)} - \eta_t \nabla F(v^{(t)})$.
  end while
  Set $w^{(k+1)} = v^{(t+1)}$.
end while
return $w^{(k+1)}$.

Algorithm 2: Pegasos-based DC procedure for CaENSVM.

$$u_1 = \frac{1 - \theta - \sqrt{(1-\theta)^2 + 2\theta s/\tau}}{\theta}, u_2 = \frac{\theta - 1 + \sqrt{(1-\theta)^2 + 2\theta s}}{\theta}. \tag{18}$$

Therefore, by Algorithm 1 and given $w^{(k)}$, the main optimized subproblem is

$$\begin{aligned} w^{(k+1)} &= \underset{w}{\operatorname{argmin}} \frac{1}{2}\|w\|_2^2 + \frac{c}{n}\sum_{i=1}^{n} l_{aEN}\left(1 - y_i w^T x_i; \tau, \theta\right) - \frac{c}{n}\sum_{i=1}^{n} \nabla l_{aEN1}\left(w^{(k)}\right)_i^T \left(w - w^{(k)}\right) \\ &= \underset{w}{\operatorname{argmin}} \frac{1}{2}\|w\|_2^2 + \frac{c}{n}\sum_{i=1}^{n} l_{aEN}\left(1 - y_i w^T x_i; \tau, \theta\right) - \frac{c}{n}\sum_{i=1}^{n} \nabla l_{aEN1}\left(w^{(k)}\right)_i^T w. \end{aligned} \tag{19}$$

For the sake of scalability and efficiency, we apply a stochastic gradient descent algorithm, i.e., Pegasos [43], to solve problem (19). Let $A_t \subset \{1, 2, \cdots, n\}$ and $|A_t| = m$ be a subset of k samples, randomly chosen from the whole dataset for the t-th iteration during optimizing a problem (19). Thus, we consider the following approximate objective function:

$$F(v; A_t) = \frac{1}{2}\|v\|_2^2 + \frac{c}{m}\sum_{i \in A_t} l_{aEN}\left(1 - y_i v^T x_i; \tau, \theta\right) - \frac{c}{m}\sum_{i \in A_t} \nabla l_{aEN1}\left(w^{(k)}\right)_i^T v, v \in \mathbb{R}^p. \tag{20}$$

Then, the subgradient of $F(v; A_t)$ with respect to v at $v^{(t)}$ is given as follows:

$$\nabla F\left(v^{(t)}\right) = v^{(t)} + \frac{c}{m}\sum_{i \in A_t} \delta\left(v^{(t)}\right)_i - \frac{c}{m}\sum_{i \in A_t} \nabla l_{aEN1}\left(w^{(k)}\right)_i, \tag{21}$$

where $v^{(t)}$ is the optimal value at the t-th iteration and

$$\delta\left(v^{(t)}\right)_i = \begin{cases} \tau\left(\theta x_i x_i^T v^{(t)} + (1-2\theta) y_i x_i\right), & 1 - y_i x_i^T v^{(t)} < 0, \\ \theta x_i x_i^T v^{(t)} - y_i x_i, & 1 - y_i x_i^T v^{(t)} > 0. \end{cases} \tag{22}$$

In line with Pegasos, the update can be written as

$$v^{(t+1)} \leftarrow v^{(t)} - \eta_t \nabla F\left(v^{(t)}\right), \tag{23}$$

where $\eta_t = c/t$ is the step size.

Finally, based on the aforementioned results about the DC algorithm and Pegasos, we design a Pegasos-based DC procedure to solve the CaENSVM optimization problem (11), which is shown in Algorithm 2. Note that if we substitute the training sample matrix X with the kernelized form $X_K = K(X, X^T) = (K(X, x_1), \cdots, K(X, x_n))$, we can directly apply the implemented Algorithm 2 to solve nonlinear CaENSVM.

4. Properties of CaENSVM

In this section, we theoretically investigate several properties of the proposed CaENSVM, including noise insensitivity, Bayes rule, and generalization error bound. Since the following analysis involves the statistical distribution of the training dataset, we first make several notations and assumptions. Supposing the training samples $\{(x_i^T, y_i)\}_{i=1}^n$ are independently drawn from a probability measure ρ on $\mathcal{X} \times \mathcal{Y}$, where $\mathcal{Y} \in \{+1, -1\}$. Let $\mathrm{Prob}(\cdot)$ and $\mathrm{Prob}(\cdot|\cdot)$ be the probability and the conditional probability, respectively.

4.1. Noise Insensitivity. By the construction of the CaEN loss, it inherits the robustness to label noise (outliers) from ramp loss and the resampling stability to feature noise from the pinball loss. Therefore, we focus on the noise insensitivity of CaENSVM from two aspects: the robustness to label noise and the resampling stability to feature noise.

4.1.1. Robustness to Label Noise. For the robustness to label noise, we show this property throughout proving the boundness of the influence function, which was first introduced by Hampel [44]. The influence function aims to measure the stability of estimators against an infinitesimal contamination. The influence function of a robust estimator should be bounded [44, 45]. Before giving the main result, we have to make the following assumption for the distribution of the training dataset, which is common in statistical analysis.

Assumption 1. The random variable $x \in \mathcal{X}$ has finite second moment.

We denote by $(x_0^T, y_0)^T$ a sample point with mass probability distribution Δ_{x_0,y_0}. Given the distribution $\mathcal{F}$ of $(x^T, y)^T$ in $\mathbb{R}^{p+1}$, let the mixed distribution of $\mathcal{F}$ and Δ_{x_0,y_0} be $\mathcal{F}_\epsilon = (1-\epsilon)\mathcal{F} + \epsilon\Delta_{x_0,y_0}$, where $\epsilon \in (0,1)$ is the proportion parameter. Fixing τ, θ and s, let

$$\begin{cases} w_0^* = \underset{w}{\operatorname{argmin}}\left\{c \int l_{CaEN}\left(1 - y w^T x\right) \mathrm{d}\mathcal{F} + \frac{1}{2}\|w\|_2^2\right\}, \\ w_\epsilon^* = \underset{w}{\operatorname{argmin}}\left\{c \int l_{CaEN}\left(1 - y w^T x\right) \mathrm{d}\mathcal{F}_\epsilon + \frac{1}{2}\|w\|_2^2\right\}. \end{cases} \tag{24}$$

Then, the influence function at a sample point $(x_0^T, y_0)^T$ is defined as

$$\mathrm{IF}\left(x_0, y_0; w_0^*\right) = \lim_{\epsilon \longrightarrow 0^+} \frac{w_\epsilon^* - w_0^*}{\epsilon}, \tag{25}$$

provided that the limit exists.

Theorem 1. *(influence function). For linear CaENSVM (11) with τ, θ and s fixed, the influence function $\mathrm{IF}(x_0, y_0; w_0^*)$ at a sample point $(x_0^T, y_0)^T$ is given by*

$$IF\left(x_0, y_0; w_0^*\right) = W_0^{-1}\left(\nabla l_{CaEN}\left(1 - y_0 x_0^T w_0^*\right) y_0 x_0 - \frac{1}{c} w_0^* + \gamma_0\right), \tag{26}$$

where $W_0 = 1/cI + \int x x^T \nabla^2 l_{CaEN}(1 - y x^T w_0^*) \mathrm{d}\mathcal{F}$, and

$$\nabla^2 l_{CaEN}(u) = \begin{cases} 0, & u < u_1, \\ \tau\theta, & u_1 < u < 0, \\ \theta, & 0 < u < u_2, \\ 0, & u > u_2, \end{cases} \tag{27}$$

and

$$\gamma_0 = \int y x \frac{\partial}{\partial \epsilon}\left(\zeta_1(\epsilon, x, y) + \zeta_2(\epsilon, x, y) + \zeta_3(\epsilon, x, y)\right) \mathrm{d}\mathcal{F}\Big|_{\epsilon=0}, \tag{28}$$

where $\zeta_1(\epsilon, x, y) \in [\tau(\theta u_2 + (1-\theta)), 0]$, $\zeta_2(\epsilon, x, y) \in [-\tau(1-\theta), (1-\theta)]$, and $\zeta_3(\epsilon, x, y) \in [0, \theta u_2 + (1-\theta)]$.

Proof. According to KKT conditions, w_ϵ^* must satisfy

$$-c \int \nabla l_{CaEN}\left(1 - y x^T w_\epsilon^*\right) y x \mathrm{d}\mathcal{F}_\epsilon + w_\epsilon^* = \mathbf{0}. \tag{29}$$

Since $\mathcal{F}_\epsilon = (1-\epsilon)\mathcal{F} + \epsilon\Delta_{x_0,y_0}$, equation (29) can be rewritten as

$$\frac{1}{c} w_\epsilon^* = (1-\epsilon) \int \nabla l_{CaEN}\left(1 - y x^T w_\epsilon^*\right) y x \mathrm{d}\mathcal{F} + \epsilon \nabla l_{CaEN}\left(1 - y_0 x_0^T w_\epsilon^*\right) y_0 x_0. \tag{30}$$

Differentiating with respect to ϵ in both sides of (30) and letting $\epsilon \longrightarrow 0$, we have

$$\frac{1}{c}\frac{\partial w_\epsilon^*}{\partial \epsilon}\Big|_{\epsilon=0} = -\int \nabla l_{CaEN}\left(1 - yx^T w_0^*\right) yx \mathrm{d}\mathscr{F} - \int xx^T \nabla^2 l_{CaEN}\left(1 - yx^T w_0^*\right) \mathrm{d}\mathscr{F} \cdot \frac{\partial w_\epsilon^*}{\partial \epsilon}\Big|_{\epsilon=0} + \gamma_0 + \nabla l_{CaEN}\left(1 - y_0 x_0^T w_0^*\right) y_0 x_0, \tag{31}$$

where

$$\gamma_0 = \int yx \frac{\partial}{\partial \epsilon}\left(\zeta_1(\epsilon, x, y) + \zeta_2(\epsilon, x, y) + \zeta_3(\epsilon, x, y)\right) \mathrm{d}\mathscr{F}\Big|_{\epsilon=0}, \tag{32}$$

where $\zeta_1(\epsilon, x, y) \in [\tau(\theta u_2 + (1-\theta)), 0]$, $\zeta_2(\epsilon, x, y) \in [-\tau(1-\theta), (1-\theta)]$, and $\zeta_3(\epsilon, x, y) \in [0, \theta u_2 + (1-\theta)]$ are from the results of the optimality condition (36).

Combining (30) and (31), we can obtain that

$$\left(\frac{1}{c} I + \int xx^T \nabla^2 l_{CaEN}\left(1 - yx^T w_0^*\right) I\mathscr{F}\right) \mathrm{IF}(x_0, y_0; w_0^*) = \nabla l_{CaEN}\left(1 - y_0 x_0^T w_0^*\right) y_0 x_0 - \frac{1}{c} w_0^* + \gamma_0, \tag{33}$$

where I is an identity matrix with a proper size. Let $W_0 = \int xx^T \nabla^2 l_{CaEN}(1 - yx^T w_0^*) \mathrm{d}\mathscr{F}$. Since $\nabla^2 l_{CaEN}(u)$ is nonnegative by (27), W_0 can be always invertible. Therefore, we finally obtain

$$\mathrm{IF}(x_0, y_0; w_0^*) = W_0^{-1}\left(\nabla l_{CaEN}\left(1 - y_0 x_0^T w_0^*\right) y_0 x_0 - \frac{1}{c} w_0^* + \gamma_0\right). \tag{34}$$

The proof is completed. □

Corollary 1. *The influence function* $\mathrm{IF}(x_0, y_0; w_0^*)$ *is bounded, i.e., the CaENSVM is robust to label noise.*

Proof. First, since $\zeta_i(\epsilon, x, y), i = 1, 2, 3$ are bounded and continuous with respect to ϵ in closed intervals, their corresponding derivatives with respect to ϵ are also bounded. Then, by Assumption 1 and Theorem 1, we have

$$\left\|\mathrm{IF}(x_0, y_0; w_0^*)\right\| \le \lambda_{\min}(W_0)\left((\theta u_2 + (1-\theta))\|x_0\| + \frac{1}{c}\|w_0^*\| + \|\gamma_0\|\right) < \infty. \tag{35}$$

For $u_1 \le 1 - y_0 x_0^T w_0^* \le u_2$, where $\lambda_{\min}(\cdot)$ is the smallest eigenvalue of a matrix. Otherwise, by (36), $\nabla l_{CaEN}(1 - y_0 x_0^T w_0^*) = 0$, the boundness of $\mathrm{IF}(x_0, y_0; w_0^*)$ also holds. □

Remark 1. According to Corollary 1, the derivative of loss, i.e., $\nabla l_{\mathrm{CaEN}}(u)$, significantly relates to the characteristics of the influence function. In fact, because $\nabla l_{\mathrm{CaEN}}(u)$ is bounded, we can easily deduce the bounds of the influence function. For those convex losses, such as elastic net loss and pinball loss, their derivatives are boundless, which means the corresponding estimators are not robust. In other words, Corollary 1 reveals the reason of the robustness of the CaEN loss, or those concave losses.

4.1.2. Resampling Stability to Feature Noise. For the resampling stability to feature noise, we demonstrate this property in line with Huang et al. [11]. Recalling the CaEN loss (10), the subgradient of l_{CaEN} with respect to u is given by

$$\nabla l_{CaEN}(u) = \begin{cases} 0, & u < u_1, \\ [\tau(\theta u_1 - (1-\theta)), 0], & u = u_1, \\ \tau(\theta u - (1-\theta)), & u_1 < u < 0, \\ [-\tau(1-\theta), (1-\theta)], & u = 0, \\ \theta u + (1-\theta), & 0 < u < u_2, \\ [0, \theta u_2 + (1-\theta)], & u = u_2, \\ 0, & u > u_2. \end{cases} \tag{36}$$

Then, by KKT (Karush–Kuhn–Tucker) conditions, the solution of CaENSVM satisfies

$$\mathbf{0} \in \frac{n}{c} w - \sum_{i=1}^{n} \nabla l_{CaEN}\left(1 - y_i w^T x_i\right) y_i x_i, \tag{37}$$

where $\mathbf{0}$ is a proper length vector with all components equal to zero. For given w, the training sample index set can be partitioned into the following seven sets:

$$
\begin{aligned}
S_0^w &= \left\{i:\ 1 - y_i w^T x_i < u_1\right\},\\
S_1^w &= \left\{i:\ 1 - y_i w^T x_i = u_1\right\},\\
S_2^w &= \left\{i:\ u_1 < 1 - y_i w^T x_i < 0\right\},\\
S_3^w &= \left\{i:\ 1 - y_i w^T x_i = 0\right\},\\
S_4^w &= \left\{i:\ 0 < 1 - y_i w^T x_i < u_2\right\},\\
S_5^w &= \left\{i:\ 1 - y_i w^T x_i = u_2\right\},\\
S_6^w &= \left\{i:\ 1 - y_i w^T x_i > u_2\right\}.
\end{aligned}
\tag{38}
$$

Using the notations (38), the optimal conditions (37) can be equivalently rewritten as

$$
\begin{aligned}
&\frac{n}{c}w - \sum_{i\in S_1^w} \nabla l_{CaEN}\left(1 - y_i w^T x_i\right)\Big) y_i x_i - \sum_{i\in S_2^w} \nabla l_{CaEN}\left(1 - y_i w^T x_i\right)\Big) y_i x_i - \sum_{i\in S_3^w} \nabla l_{CaEN}\left(1 - y_i w^T x_i\right)\Big) y_i x_i\\
&\quad - \sum_{i\in S_4^w} \nabla l_{CaEN}\left(1 - y_i w^T x_i\right)\Big) y_i x_i - \sum_{i\in S_5^w} \nabla l_{CaEN}\left(1 - y_i w^T x_i\right)\Big) y_i x_i = \mathbf{0}.
\end{aligned}
\tag{39}
$$

Since S_1^w, S_3^w, and S_5^w are based on equalities, it is reasonable to see that S_1^w, S_3^w, and S_5^w have much smaller sizes than S_2^w or S_4^w. Thus, the contributions of S_1^w, S_3^w, and S_5^w to (39) are considerably weak. In other words, we can roughly determine w by S_2^w and S_4^w, i.e.,

$$
\frac{n}{c}w + \tau \sum_{i\in S_2^w} \Big((1-\theta) - \theta\Big(1 - y_i w^T x_i\Big)\Big) y_i x_i - \sum_{i\in S_4^w} \Big(\theta\Big(1 - y_i w^T x_i\Big) + (1-\theta)\Big) y_i x_i \cong \mathbf{0}. \tag{40}
$$

With parameters τ and θ properly selected, equation (40) indicates that τ controls the sensitivity of the CaENSVM to feature noise. In fact, by (38), $((1-\theta) - \theta(1 - y_i w^T x_i))$ and $(\theta(1 - y_i w^T x_i) + (1-\theta))$ are both positive, which means that a large τ (close to 1) can well balance the size of S_2^w and S_4^w for zero mean feature noise. Therefore, the effect of zero mean feature noise is weakened, and the final separating hyperplane of the CaENSVM is stable for resampling. Along with τ decreasing (close to 0), by (38), the final separating hyperplane is gradually dominated by the instances in S_4^w. As a result, the classification results are significantly disturbed by the zero mean feature noise around the decision boundary.

4.2. Bayes Rule. Let $P(x) = \text{Prob}(\mathcal{Y} = 1|\mathcal{X} = x)$ be the conditional probability of the positive class given $\mathcal{X} = x$. Lin [46] claimed that $\text{sign}(P(x) - 1/2)$ is the decision-theoretic optimal classification rule with the smallest generalization error, which is the so-called Bayes rule. We can also equivalently define Bayes rule as

$$
f_C(x) = \begin{cases} +1, & \text{Prob}(y = +1|x) \geq \text{Prob}(y = -1|x);\\ -1, & \text{Prob}(y = +1|x) < \text{Prob}(y = -1|x). \end{cases}
\tag{41}
$$

For any loss function $l(\cdot)$, we define the expected risk of a classifier $f: \mathcal{X} \longmapsto \mathcal{Y}$ as

$$
R_{l,\rho}(f) = \int_{\mathcal{X}\times\mathcal{Y}} l(1 - yf(x))\mathrm{d}\rho. \tag{42}
$$

By minimizing the expected risk over all measurable functions, we can obtain $f_{l,\rho}(x)$ as

$$
f_{l,\rho}(x) = \underset{\mu}{\text{argmin}} \int_{\mathcal{Y}} l(1 - y\mu)\mathrm{d}\rho(y|x), \forall x \in \mathcal{X}, \tag{43}
$$

where $\rho(y|x)$ is the conditional distribution of y given x. Note that $\rho(y|x)$ is a binary distribution, corresponding to $\text{Prob}(y = +1|x)$ and $\text{Prob}(y = -1|x)$.

Huang et al. [11] proved that the pinball loss can lead to the Bayes classifier. In the following, we demonstrate that the Bayes rule also holds for our proposed capped asymmetric elastic net loss function.

Theorem 2. *The decision function $f_{l_{\text{CaEN}},\rho}$ obtained by minimizing l_{CaEN}-based expected risk over all measurable functions $f: \mathcal{X} \longrightarrow \mathcal{Y}$ is equivalent to the Bayes rule, i.e., $f_{l_{\text{CaEN}},\rho}(x) = f_C(x), \forall x \in \mathcal{X}$.*

Proof. By simple calculating, we can obtain

$$\int_{\mathscr{Y}} l_{CaEN}(1-y\mu)\mathrm{d}\rho(y|x) = l_{CaEN}(1-\mu)\mathrm{Prob}(y=+1|x) + l_{CaEN}(1+\mu)\mathrm{Prob}(y=-1|x). \tag{44}$$

According to (10), it follows that

$$l_{CaEN}(1-\mu) = \begin{cases} s, & \mu < 1-u_2, \\ \dfrac{\theta}{2}(1-\mu)^2 + (1-\theta)(1-\mu), & 1-u_2 \le \mu < 1, \\ \tau\left(\dfrac{\theta}{2}(1-\mu)^2 - (1-\theta)(1-\mu)\right), & 1 \le \mu < 1-u_1, \\ s, & \mu \ge 1-u_1, \end{cases} \tag{45}$$

and

$$l_{CaEN}(1+\mu) = \begin{cases} s, & \mu \le u_1 - 1, \\ \tau\left(\dfrac{\theta}{2}(1+\mu)^2 - (1-\theta)(1+\mu)\right), & u_1 - 1 < \mu \le -1, \\ \dfrac{\theta}{2}(1+\mu)^2 + (1-\theta)(1+\mu), & -1 < \mu \le u_2 - 1, \\ s, & \mu > u_2 - 1. \end{cases} \tag{46}$$

Let $P(+1) = \mathrm{Prob}(y=+1|x)$ and $P(-1) = \mathrm{Prob}(y=-1|x)$, respectively. We denote by $r_1(\theta,\mu) = \theta/2(1-\mu)^2 + (1-\theta)(1-\mu)$, $r_2(\theta,\mu) = \tau(\theta/2(1-\mu)^2 - (1-\theta)(1-\mu))$, $r_3(\theta,\mu) = \theta/2(1+\mu)^2 + (1-\theta)(1+\mu)$, and $r_4(\theta,\mu) = \tau(\theta/2(1+\mu)^2 - (1-\theta)(1-\mu))$ for convenience, respectively. We found that the expected risk relates to the value of u_2. Thus, based on (45) and (46) and the continuity of l_{CaEN}, we discuss all cases as follows:

For the case of $0 < u_2 < 1$, we have

$$\int_{\mathscr{Y}} l_{CaEN}(1-y\mu)\mathrm{d}\rho(y|x) = \begin{cases} sP(+1) + sP(-1), & \mu \le u_1 - 1, \\ sP(+1) + r_4(\theta,\mu)P(-1), & u_1 - 1 < \mu \le -1, \\ sP(+1) + r_3(\theta,\mu)P(-1), & -1 < \mu \le u_2 - 1, \\ sP(+1) + sP(-1), & u_2 - 1 < \mu \le 1 - u_2, \\ r_1(\theta,\mu)P(+1) + sP(-1), & 1 - u_2 < \mu \le 1, \\ r_2(\theta,\mu)P(+1) + sP(-1), & 1 < \mu \le 1 - u_1, \\ sP(+1) + sP(-1), & \mu > 1 - u_1. \end{cases} \tag{47}$$

If $u_2 = 1$, $u_2 - 1 = 1 - u_2$ and the interval $[u_2 - 1, 1 - u_2]$ disappears. One can easily verify that the minimal value is not affected. Supposing that $P(+1) > P(-1)$, the minimal value of μ is $+1$ according to the increase-decrease characteristics of l_{CaEN} with respect to μ in each interval. Similarly, the minimal value of μ is -1 if $P(+1) < P(-1)$ or the minimal value of μ is $+1$ or -1 if $P(+1) = P(-1)$. Consequently, we have $f_{l_{CaEN}}(x) = f_C(x)$ when $0 < u_2 \le 1$.

For the case of $1 < u_2 < 2$, we have

$$\int_{\mathscr{Y}} l_{CaEN}(1 - y\mu)\mathrm{d}\rho(y \mid x) = \begin{cases} sP(+1) + sP(-1), & \mu \le u_1 - 1, \\ sP(+1) + r_4(\theta, \mu)P(-1), & u_1 - 1 < \mu \le -1, \\ sP(+1) + r_3(\theta, \mu)P(-1), & -1 < \mu \le 1 - u_2, \\ r_1(\theta, \mu)P(+1) + r_3(\theta, \mu)P(-1), & 1 - u_2 < \mu \le u_2 - 1, \\ r_1(\theta, \mu)P(+1) + sP(-1), & u_2 - 1 < \mu \le 1, \\ r_2(\theta, \mu)P(+1) + sP(-1), & 1 < \mu \le 1 - u_1, \\ sP(+1) + sP(-1), & \mu > 1 - u_1. \end{cases} \tag{48}$$

If $u_2 = 2$, we have $1 - u_2 = -1$ and $u_2 - 1 = 1$. Then, the intervals $[-1, 1 - u_2]$, and $[1, u_2 - 1]$ both disappear, which has no effects on the minimum value. After simple calculations, one can find that the minimal value is the same as $0 < u_2 \le 1$. Thus, we have $f_{l_{CaEN}}(x) = f_C(x)$ when $1 < u_2 \le 2$.

For the case of $2 < u_2 < 2 - u_1$, we have

$$\int_{\mathscr{Y}} l_{CaEN}(1 - y\mu)\mathrm{d}\rho(y \mid x) = \begin{cases} sP(+1) + sP(-1), & \mu \le u_1 - 1, \\ sP(+1) + r_4(\theta, \mu)P(-1), & u_1 - 1 < \mu \le 1 - u_2, \\ r_1(\theta, \mu)P(+1) + r_4(\theta, \mu)P(-1), & 1 - u_2 < \mu \le -1, \\ r_1(\theta, \mu)P(+1) + r_3(\theta, \mu)P(-1), & -1 < \mu \le +1, \\ r_2(\theta, \mu)P(+1) + r_3(\theta, \mu)P(-1), & 1 < \mu \le u_2 - 1, \\ r_2(\theta, \mu)P(+1) + sP(-1), & u_2 - 1 < \mu \le 1 - u_1, \\ sP(+1) + sP(-1), & \mu > 1 - u_1. \end{cases} \tag{49}$$

If $u_2 = 2 - u_1$, $u_1 - 1 = 1 - u_2$, and $u_2 - 1 = 1 - u_1$, the interval $[u_1 - 1, 1 - u_2]$, and $[u_2 - 1, 1 - u_1]$ disappear, which has no effects on the minimum value; the minimal value is the same as $0 < u_2 \le 1$, i.e., we have $f_{l_{CaEN}}(x) = f_C(x)$ when $2 < u_2 \le 2 - u_1$.

For the case of $u_2 > 2 - u_1$, we have

$$\int_{\mathscr{Y}} l_{CaEN}(1 - y\mu)\mathrm{d}\rho(y \mid x) = \begin{cases} sP(+1) + sP(-1), & \mu \le 1 - u_2, \\ r_1(\theta, \mu)P(+1) + sP(-1), & 1 - u_2 < \mu \le u_1 - 1, \\ r_1(\theta, \mu)P(+1) + r_4(\theta, \mu)P(-1), & u_1 - 1 < \mu \le -1, \\ r_1(\theta, \mu)P(+1) + r_3(\theta, \mu)P(-1), & -1 < \mu \le +1, \\ r_2(\theta, \mu)P(+1) + r_3(\theta, \mu)P(-1), & 1 < \mu \le 1 - u_1, \\ sP(+1) + r_3(\theta, \mu)P(-1), & 1 - u_1 < \mu \le u_2 - 1, \\ sP(+1) + sP(-1), & \mu > u_2 - 1. \end{cases} \tag{50}$$

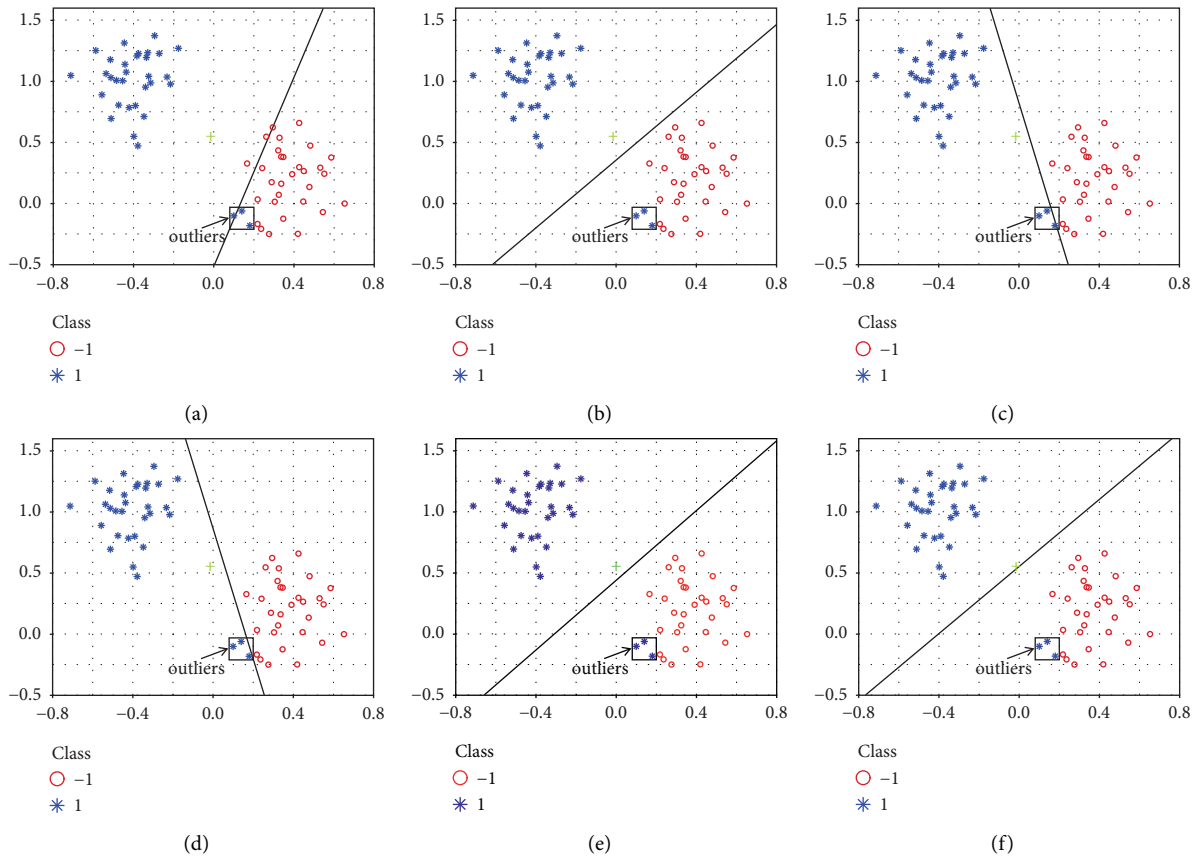

Figure 3: The separating hyperplanes (black solid lines) obtained by ENSVM, PinSVM, RampSVM, Rhinge-SVM, Valley-SVM, and CaENSVM, respectively. The green "+" notation is the midpoint between the centers of two classes of samples. (a) ENSVM. (b) PinSVM. (c) RampSVM. (d) Rhinge-SVM. (e) Valley-SVM. (f) CaENSVM.

One can easily obtain the same minimal value as $0 < u_2 \leq 1$, i.e., we have $f_{l_{CaEN}}(x) = f_C(x)$ when $u_2 > 2 - u_1$. With the abovementioned results, minimizing l_{CaEN}-based expected risk over all measurable functions can lead to the Bayes rule. Thus, Theorem 2 is proved. □

4.3. Generalization Error Bound. We have proved that the proposed CaENSVM is equivalent to the Bayes rule, which is also called classification-calibrated in the study of Bartlett et al. [7], indicating that the CaENSVM enjoys many good properties. Here, we further give the generalization error bound of the CaENSVM based on the empirical Rademacher complexity [6], where the empirical Rademacher complexity is defined as follows:

Definition 1. Supposing that $x_1, \cdots, x_n$ are independently selected from $\mathcal{X}$ with a probability distribution ν, and F is a real-valued function class mapping from $\mathcal{X}$ to $\mathbb{R}$, the empirical Rademacher complexity of F is a random variable defined as

$$\widehat{R}_n(F) = \mathbb{E}_\sigma\left[\sup_{f\in F}\left|\frac{2}{n}\sum_{i=1}^{n}\sigma_i f(x_i)\right| : x_1, \cdots, x_n\right], \tag{51}$$

where $\sigma = (\sigma_1, \ldots, \sigma_n)^T$ is independently uniform $\{\pm 1\}$-valued (Rademacher) random variables and $\mathbb{E}_\sigma[\cdot]$ means the expectation over σ. The Rademacher complexity of F is

$$R_n(F) = \mathbb{E}_\nu\left(\widehat{R}_n(F)\right), \tag{52}$$

where $\mathbb{E}_\nu[\cdot]$ means the expectation over ν.

According to (51), $\widehat{R}_n(F)$ can be viewed as a correlation between $\mathbf{f} = (f(x_1), \cdots, f(x_n))^T$ and σ for given $x_1, \cdots, x_n$. Thus, the higher $\widehat{R}_n(F)$ is, the more complex F is. By (52), $R_n(F)$ depicts the average complexity of F based on ν instead of a set of particular samples.

In line with the abovementioned definitions and lemmas in the study of Bartlett and Mendelson [6], we provide the following theorem to yield the generalization error bound of the CaENSVM.

Theorem 3. *Fix $\zeta \in (0, 1)$ and $B \in \mathbb{R}^+$ and consider the binary classification problem on $\{(x_i^T, y_i)\}_{i=1}^n$ drawn independently from a probability distribution $\mathcal{F}$. Let $F = \{f | f: x \mapsto w^T\phi(x), \|w\| \leq B\}$ and $G = \{g | g: (y, f(x)) \mapsto -yf(x), f \in F\}$ be function classes, respectively. If the CaEN loss (10) with $s \geq 1/2$ and the optimal*

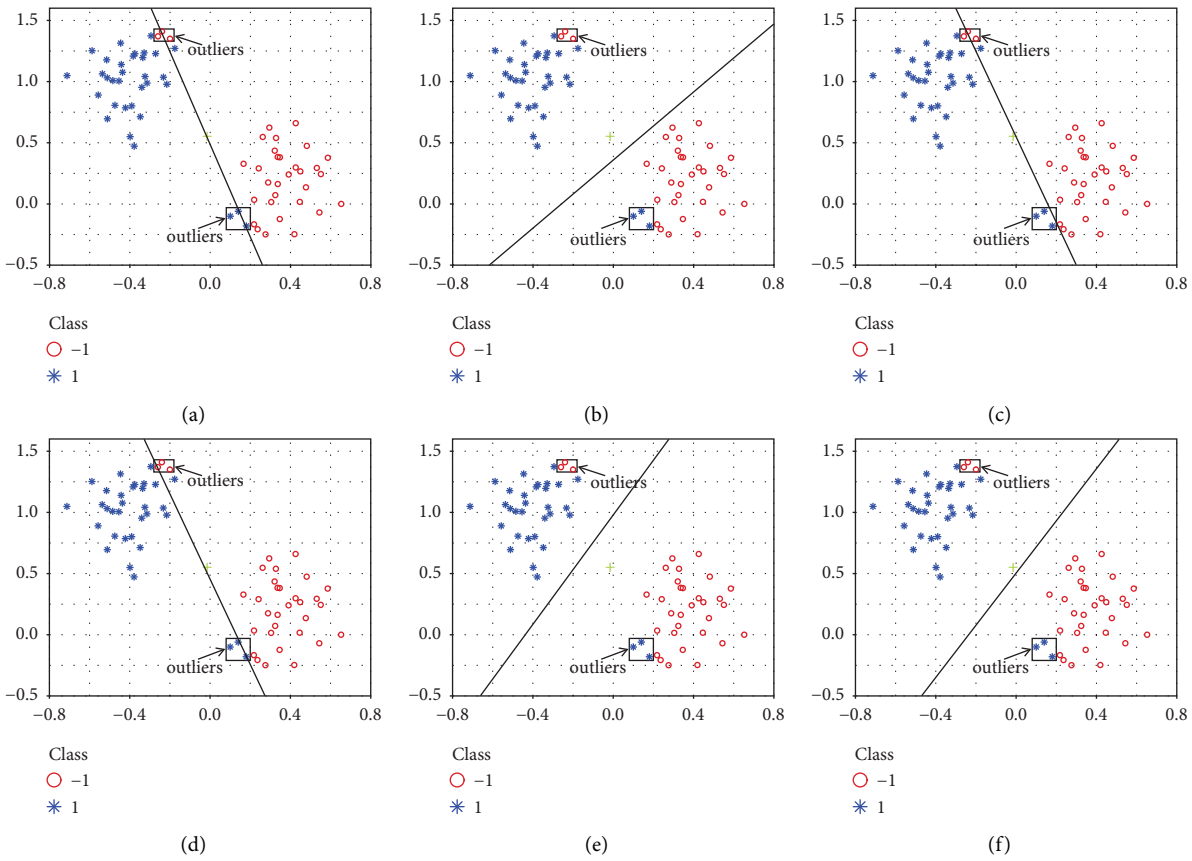

FIGURE 4: The separating hyperplanes (black solid lines) obtained by ENSVM, PinSVM, RampSVM, Rhinge-SVM, Valley-SVM, and CaENSVM, respectively. The green "+" notation is the midpoint between the centers of two classes of samples. (a) ENSVM. (b) PinSVM. (c) RampSVM. (d) Rhinge-SVM. (e) Valley-SVM. (f) CaENSVM.

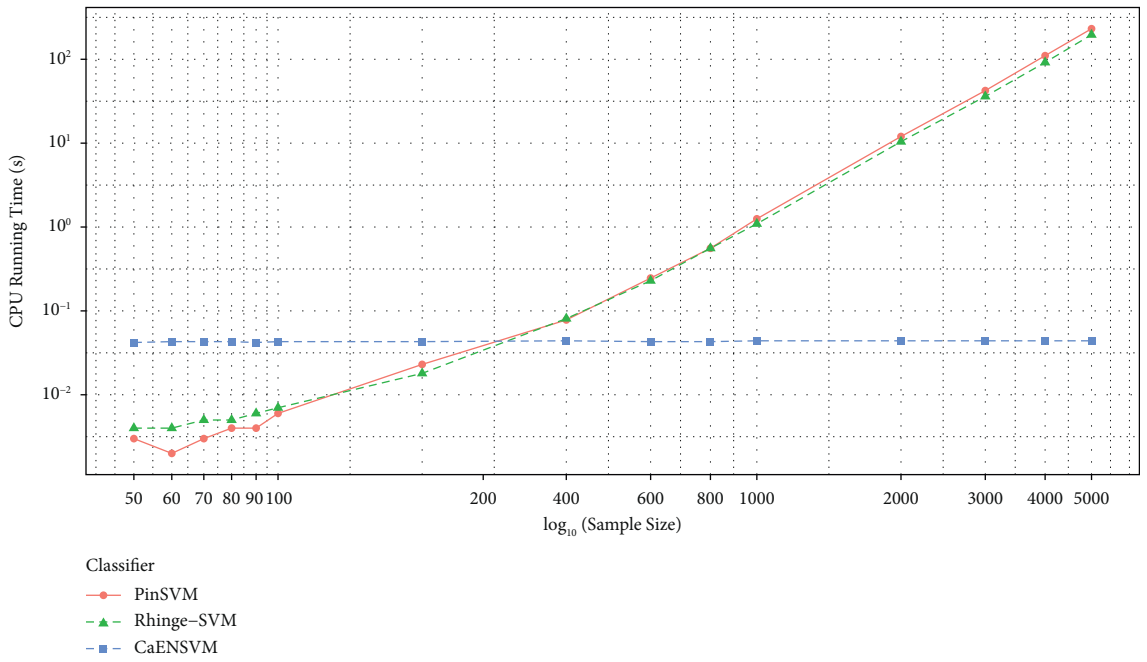

FIGURE 5: The one-run CPU time (in seconds) of PinSVM, Rhinge-SVM, and CaENSVM with linear kernel. x-coordinate is the $\log_{10}$(sample size), while y-coordinate is the training time (seconds).

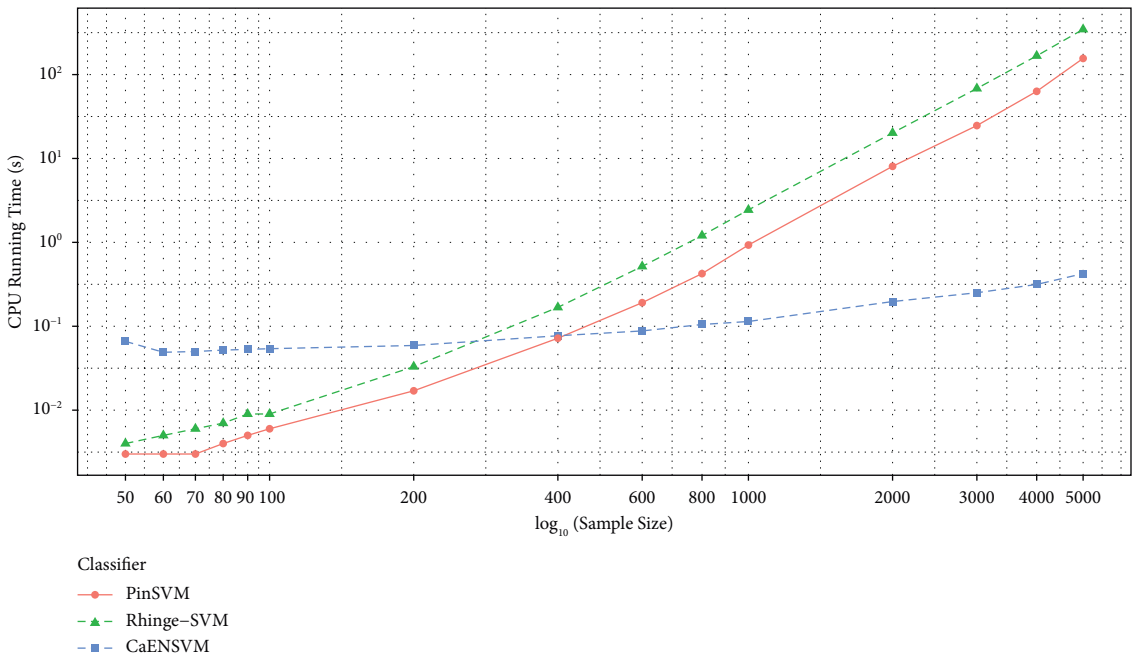

Figure 6: The one-run CPU time (in seconds) of PinSVM, Rhinge-SVM, and CaENSVM with Gaussian kernel. x-coordinate is the $\log_{10}$ (sample size), while y-coordinate is the training time (seconds).

Table 1: The information of UCI datasets.

ID	Dataset	#Obs	#Fea
1	Absenteeism	740	19
2	Autism	702	15
3	baDS	1372	5
4	Banknote	1371	4
5	Forestfire	121	10
6	ILPD	579	10
7	Knowledge	251	5
8	Messidor	1151	19
9	Popfailure	540	15
10	Seed	210	8
11	wdbc	569	34
12	Wine	130	13
13	Amphibians	189	21
14	Raisin	900	7
15	Tripadvisor	980	9

w_0^* satisfies $\|w_0^*\| \le B$, then with probability at least $1-\zeta$, the prediction function $f(x)$ satisfies $f \in F$ and

$$\text{Prob}\left(yf(x) \le 0\right) \le \frac{2}{n}\sum_{i=1}^{n} l_{CaEN}\left(1 - y_i f(x_i)\right) + \frac{8B\left(\theta u_2 + (1-\theta)\right)}{n}\sqrt{\sum_{i=1}^{n} K(x_i, x_i)} + \sqrt{\frac{8\ln(2/\zeta)}{n}}. \tag{53}$$

Proof. define the Heaviside function $\Xi(\cdot)$ as

$$\Xi(u) = \begin{cases} 1, & u \ge 0, \\ 0, & \text{otherwise.} \end{cases} \tag{54}$$

Then, we can easily obtain

$$\text{Prob}\left(yf(x) \le 0\right) = \mathbb{E}_\rho\left[\Xi\left(-yf(x)\right)\right]. \tag{55}$$

By defining $\Psi(u) = 2l_{CaEN}(1+u;\tau,\theta,s=1/2)$, we can easily verify that $u_2 \le 1$ and $\Psi(u) \in [0,1]$ dominates the function $\Xi(\cdot)$ on the support of ρ, we can obtain the

Table 2: The mean accuracy (Acc.) and standard deviation (sd) with linear kernel for UCI datasets.

	ENSVM Acc. ± sd	PinSVM Acc. ± sd	RampSVM Acc. ± sd	Rhinge-SVM Acc. ± sd	Valley-SVM Acc. ± sd	CaENSVM Acc. ± sd
(a) 0% *label noise*						
Absenteeism	0.946 ± 0.014	0.946 ± 0.018	0.946 ± 0.012	0.946 ± 0.051	0.954 ± 0.012	**0.973 ± 0.013**
Autism	0.836 ± 0.023	0.734 ± 0.024	0.734 ± 0.025	0.729 ± 0.083	0.810 ± 0.067	**0.993 ± 0.013**
baDS	0.968 ± 0.012	0.912 ± 0.029	0.982 ± 0.015	0.972 ± 0.015	0.977 ± 0.011	**0.987 ± 0.007**
Banknote	0.983 ± 0.013	0.936 ± 0.014	0.983 ± 0.011	0.975 ± 0.018	0.974 ± 0.006	**0.992 ± 0.005**
Forestfire	0.870 ± 0.082	0.652 ± 0.015	0.652 ± 0.009	0.809 ± 0.073	0.835 ± 0.057	**0.922 ± 0.020**
ILPD	0.649 ± 0.016	0.287 ± 0.025	0.287 ± 0.013	0.381 ± 0.687	0.447 ± 0.220	**0.716 ± 0.012**
Knowledge	0.931 ± 0.040	**0.967 ± 0.031**	0.959 ± 0.032	0.939 ± 0.043	0.718 ± 0.152	**0.967 ± 0.031**
Messidor	0.678 ± 0.044	0.481 ± 0.025	0.542 ± 0.119	**0.687 ± 0.037**	0.675 ± 0.023	0.642 ± 0.073
Popfailure	0.735 ± 0.364	0.084 ± 0.013	0.916 ± 0.009	0.490 ± 0.380	0.084 ± 0.012	**0.929 ± 0.025**
Seed	0.907 ± 0.041	0.500 ± 0.022	0.500 ± 0.010	0.800 ± 0.041	0.686 ± 0.150	**0.929 ± 0.025**
wdbc	0.894 ± 0.027	0.372 ± 0.036	0.372 ± 0.024	0.857 ± 0.052	**0.897 ± 0.018**	0.715 ± 0.178
Wine	0.888 ± 0.052	0.560 ± 0.011	0.560 ± 0.012	0.872 ± 0.052	0.904 ± 0.061	**0.912 ± 0.052**
Amphibians	0.671 ± 0.044	0.692 ± 0.082	0.681 ± 0.088	**0.698 ± 0.055**	0.687 ± 0.045	**0.698 ± 0.070**
Raisin	0.864 ± 0.030	0.864 ± 0.038	0.863 ± 0.034	0.864 ± 0.031	0.834 ± 0.041	**0.872 ± 0.013**
Tripadvisor	0.733 ± 0.012	0.732 ± 0.005	0.735 ± 0.009	0.731 ± 0.008	0.740 ± 0.009	**0.741 ± 0.008**
(b) 15% *label noise*						
Absenteeism	0.935 ± 0.024	0.946 ± 0.012	0.946 ± 0.008	0.795 ± 0.331	0.946 ± 0.020	**0.948 ± 0.010**
Autism	0.806 ± 0.035	0.734 ± 0.023	0.734 ± 0.002	0.794 ± 0.068	0.796 ± 0.088	**0.813 ± 0.028**
baDS	0.973 ± 0.017	0.915 ± 0.047	0.973 ± 0.012	0.974 ± 0.007	0.968 ± 0.012	**0.977 ± 0.010**
Banknote	0.969 ± 0.011	0.905 ± 0.051	**0.979 ± 0.014**	0.975 ± 0.012	0.972 ± 0.011	**0.979 ± 0.011**
Forestfire	0.848 ± 0.095	0.652 ± 0.015	0.652 ± 0.023	0.739 ± 0.141	0.800 ± 0.140	**0.852 ± 0.050**
ILPD	0.402 ± 0.158	0.287 ± 0.008	0.287 ± 0.025	0.640 ± 0.175	0.593 ± 0.182	**0.654 ± 0.161**
Knowledge	0.947 ± 0.031	0.972 ± 0.023	0.972 ± 0.023	0.976 ± 0.027	0.690 ± 0.114	**0.979 ± 0.029**
Messidor	0.585 ± 0.096	0.470 ± 0.022	0.548 ± 0.128	0.578 ± 0.026	0.635 ± 0.052	**0.636 ± 0.033**
Popfailure	0.677 ± 0.332	0.084 ± 0.000	**0.905 ± 0.022**	0.084 ± 0.006	0.084 ± 0.023	0.802 ± 0.099
Seed	**0.900 ± 0.059**	0.500 ± 0.025	0.500 ± 0.019	0.807 ± 0.057	0.857 ± 0.124	0.824 ± 0.160
wdbc	0.810 ± 0.038	0.372 ± 0.031	0.372 ± 0.015	0.780 ± 0.042	0.832 ± 0.045	**0.864 ± 0.039**
Wine	0.872 ± 0.018	0.560 ± 0.010	0.560 ± 0.032	0.768 ± 0.107	0.888 ± 0.044	**0.896 ± 0.036**
Amphibians	0.660 ± 0.024	0.665 ± 0.053	0.681 ± 0.048	0.665 ± 0.031	0.638 ± 0.065	**0.692 ± 0.053**
Raisin	0.860 ± 0.031	0.855 ± 0.025	0.870 ± 0.026	0.862 ± 0.026	0.856 ± 0.019	**0.875 ± 0.018**
Tripadvisor	0.723 ± 0.014	0.729 ± 0.002	0.728 ± 0.005	**0.730 ± 0.004**	0.710 ± 0.018	0.724 ± 0.004
(c) 25% *label noise*						
Absenteeism	0.916 ± 0.067	0.940 ± 0.023	0.940 ± 0.008	0.772 ± 0.360	0.945 ± 0.003	**0.946 ± 0.008**
Autism	0.746 ± 0.050	0.734 ± 0.010	0.763 ± 0.064	**0.816 ± 0.092**	0.779 ± 0.136	0.780 ± 0.035
baDS	0.972 ± 0.014	0.887 ± 0.098	0.977 ± 0.008	0.966 ± 0.017	0.965 ± 0.020	**0.978 ± 0.013**
Banknote	0.972 ± 0.013	0.966 ± 0.021	0.980 ± 0.014	0.976 ± 0.015	0.978 ± 0.011	**0.982 ± 0.011**
Forestfire	0.791 ± 0.048	0.652 ± 0.025	0.652 ± 0.012	0.661 ± 0.142	0.783 ± 0.102	**0.800 ± 0.079**
ILPD	0.421 ± 0.186	0.287 ± 0.011	0.346 ± 0.132	0.515 ± 0.179	**0.657 ± 0.044**	0.546 ± 0.125
Knowledge	0.939 ± 0.038	0.963 ± 0.030	0.967 ± 0.023	0.967 ± 0.023	0.657 ± 0.106	**0.972 ± 0.030**
Messidor	**0.686 ± 0.032**	0.470 ± 0.023	0.564 ± 0.088	0.610 ± 0.057	0.660 ± 0.048	0.595 ± 0.101
Popfailure	0.673 ± 0.038	0.084 ± 0.034	**0.914 ± 0.020**	0.084 ± 0.025	0.090 ± 0.013	0.693 ± 0.031
Seed	0.907 ± 0.048	0.500 ± 0.012	0.500 ± 0.023	0.910 ± 0.100	0.600 ± 0.096	**0.914 ± 0.074**
wdbc	0.871 ± 0.039	0.372 ± 0.015	0.372 ± 0.041	0.731 ± 0.082	0.892 ± 0.039	**0.894 ± 0.047**
Wine	0.852 ± 0.066	0.560 ± 0.020	0.560 ± 0.026	0.664 ± 0.061	0.856 ± 0.108	**0.860 ± 0.321**
Amphibians	0.649 ± 0.085	**0.725 ± 0.048**	0.676 ± 0.054	0.687 ± 0.059	0.665 ± 0.056	0.698 ± 0.052
Raisin	0.856 ± 0.025	0.851 ± 0.011	0.863 ± 0.014	0.863 ± 0.014	0.846 ± 0.031	**0.867 ± 0.006**
Tripadvisor	0.723 ± 0.031	0.731 ± 0.007	0.736 ± 0.015	0.736 ± 0.013	0.732 ± 0.008	**0.737 ± 0.011**

The bold is the best one.

following inequality by Theorem 8 in the study of Bartlett and Mendelson [6] as

$$\mathbb{E}_\rho[\Xi(g(y,f(x)))] \le \mathbb{E}_\rho[\Psi(g(y,f(x)))] \le \widehat{\mathbb{E}}_n[\Psi(g(y,f(x)))] + \widehat{R}_n(\widetilde{\Psi}G) + \sqrt{\frac{8\ln(2/\zeta)}{n}}, \tag{56}$$

Table 3: The mean accuracy (Acc.) and standard deviation (sd) with Gaussian kernel for UCI datasets.

	ENSVM Acc. ± sd	PinSVM Acc. ± sd	RampSVM Acc. ± sd	Rhinge-SVM Acc. ± sd	Valley-SVM Acc. ± sd	CaENSVM Acc. ± sd
(a) 0% *label noise*						
Absenteeism	0.946 ± 0.053	**0.955 ± 0.003**	0.947 ± 0.010	0.946 ± 0.120	0.946 ± 0.022	**0.955 ± 0.003**
Autism	0.921 ± 0.106	0.978 ± 0.012	0.997 ± 0.004	0.978 ± 0.089	0.903 ± 0.081	**0.999 ± 0.003**
baDS	0.987 ± 0.010	**1.000 ± 0.000**	**1.000 ± 0.000**	**1.000 ± 0.000**	0.944 ± 0.05	**1.000 ± 0.000**
Banknote	0.980 ± 0.010	0.996 ± 0.008	0.993 ± 0.010	0.944 ± 0.018	0.957 ± 0.024	**0.998 ± 0.005**
Forestfire	0.861 ± 0.084	0.913 ± 0.062	0.870 ± 0.044	0.635 ± 0.146	0.722 ± 0.090	**0.931 ± 0.039**
ILPD	0.711 ± 0.013	0.716 ± 0.022	0.621 ± 0.187	0.714 ± 0.222	0.374 ± 0.195	**0.727 ± 0.018**
Knowledge	0.882 ± 0.070	0.951 ± 0.023	0.963 ± 0.027	0.926 ± 0.148	0.718 ± 0.125	**0.968 ± 0.028**
Messidor	0.585 ± 0.033	**0.649 ± 0.038**	0.603 ± 0.017	0.529 ± 0.043	0.497 ± 0.040	0.619 ± 0.054
Popfailure	0.563 ± 0.439	0.744 ± 0.369	**0.905 ± 0.018**	0.497 ± 0.291	0.084 ± 0.034	0.903 ± 0.022
Seed	0.936 ± 0.046	**0.971 ± 0.030**	0.936 ± 0.046	0.949 ± 0.036	0.943 ± 0.054	**0.971 ± 0.030**
wdbc	0.727 ± 0.079	**0.917 ± 0.018**	0.680 ± 0.040	0.907 ± 0.114	0.372 ± 0.023	0.908 ± 0.015
Wine	0.728 ± 0.107	**0.896 ± 0.046**	0.648 ± 0.066	0.864 ± 0.046	0.568 ± 0.018	0.888 ± 0.033
Amphibians	0.633 ± 0.093	**0.789 ± 0.035**	0.698 ± 0.067	0.730 ± 0.051	0.568 ± 0.000	0.692 ± 0.059
Raisin	0.865 ± 0.023	0.860 ± 0.028	0.852 ± 0.016	0.860 ± 0.016	0.828 ± 0.043	**0.868 ± 0.012**
Tripadvisor	0.725 ± 0.007	0.733 ± 0.016	0.722 ± 0.014	**0.742 ± 0.009**	0.728 ± 0.000	0.734 ± 0.014
(b) 15% *label noise*						
Absenteeism	0.946 ± 0.013	0.946 ± 0.014	0.947 ± 0.003	0.943 ± 0.015	0.946 ± 0.008	**0.970 ± 0.053**
Autism	0.873 ± 0.127	0.951 ± 0.025	0.996 ± 0.004	0.964 ± 0.008	0.846 ± 0.105	**0.997 ± 0.065**
baDS	0.945 ± 0.032	0.946 ± 0.004	0.941 ± 0.013	0.949 ± 0.003	0.940 ± 0.035	**0.951 ± 0.055**
Banknote	0.986 ± 0.010	0.993 ± 0.007	**1.000 ± 0.000**	0.966 ± 0.008	0.879 ± 0.148	**1.000 ± 0.000**
Forestfire	0.818 ± 0.095	**0.913 ± 0.069**	0.905 ± 0.078	0.910 ± 0.072	0.739 ± 0.087	0.912 ± 0.082
ILPD	0.556 ± 0.231	**0.717 ± 0.026**	0.696 ± 0.049	0.701 ± 0.029	0.287 ± 0.021	0.698 ± 0.076
Knowledge	0.857 ± 0.097	0.929 ± 0.029	0.917 ± 0.031	0.921 ± 0.034	0.710 ± 0.113	**0.937 ± 0.024**
Messidor	0.586 ± 0.031	0.629 ± 0.039	0.625 ± 0.028	**0.651 ± 0.020**	0.478 ± 0.017	0.634 ± 0.062
Popfailure	0.222 ± 0.309	0.731 ± 0.364	0.725 ± 0.360	0.392 ± 0.423	0.084 ± 0.023	**0.735 ± 0.192**
Seed	**0.936 ± 0.039**	0.914 ± 0.074	**0.936 ± 0.039**	**0.936 ± 0.030**	0.836 ± 0.188	0.929 ± 0.061
wdbc	0.703 ± 0.079	0.436 ± 0.053	0.626 ± 0.143	0.864 ± 0.042	0.372 ± 0.023	**0.890 ± 0.066**
Wine	0.648 ± 0.087	0.812 ± 0.072	0.648 ± 0.087	0.820 ± 0.063	0.560 ± 0.010	**0.844 ± 0.128**
Amphibians	0.611 ± 0.053	0.660 ± 0.062	0.649 ± 0.099	0.708 ± 0.084	0.579 ± 0.024	**0.714 ± 0.056**
Raisin	0.693 ± 0.177	0.860 ± 0.021	0.864 ± 0.020	0.855 ± 0.033	0.840 ± 0.014	**0.867 ± 0.026**
Tripadvisor	0.726 ± 0.011	0.718 ± 0.019	0.691 ± 0.039	0.726 ± 0.022	0.656 ± 0.102	**0.728 ± 0.010**
(c) 25% *label noise*						
Absenteeism	0.945 ± 0.003	0.936 ± 0.013	0.943 ± 0.008	0.945 ± 0.008	**0.946 ± 0.023**	**0.946 ± 0.015**
Autism	0.883 ± 0.085	0.967 ± 0.014	0.969 ± 0.006	0.973 ± 0.021	0.796 ± 0.085	**0.974 ± 0.012**
baDS	0.933 ± 0.065	0.972 ± 0.004	0.971 ± 0.010	0.975 ± 0.006	0.956 ± 0.022	**0.978 ± 0.082**
Banknote	0.987 ± 0.004	0.990 ± 0.008	0.990 ± 0.002	**0.995 ± 0.006**	0.953 ± 0.016	0.993 ± 0.042
Forestfire	0.809 ± 0.109	**0.870 ± 0.044**	0.852 ± 0.039	0.852 ± 0.058	0.817 ± 0.108	0.867 ± 0.021
ILPD	0.602 ± 0.178	0.697 ± 0.007	0.689 ± 0.015	0.709 ± 0.048	0.287 ± 0.021	**0.712 ± 0.091**
Knowledge	0.841 ± 0.060	0.972 ± 0.023	0.943 ± 0.034	**0.980 ± 0.014**	0.669 ± 0.170	0.814 ± 0.038
Messidor	0.537 ± 0.053	0.617 ± 0.027	0.583 ± 0.040	0.604 ± 0.027	0.502 ± 0.043	**0.628 ± 0.057**
Popfailure	0.366 ± 0.386	0.249 ± 0.368	0.733 ± 0.363	0.411 ± 0.448	0.084 ± 0.041	**0.784 ± 0.098**
Seed	0.914 ± 0.065	**0.936 ± 0.064**	0.929 ± 0.067	0.929 ± 0.044	0.914 ± 0.054	0.871 ± 0.054
wdbc	0.681 ± 0.054	**0.864 ± 0.066**	0.492 ± 0.171	0.844 ± 0.055	0.372 ± 0.023	0.774 ± 0.032
Wine	0.672 ± 0.072	0.716 ± 0.046	0.672 ± 0.052	0.732 ± 0.100	0.560 ± 0.008	**0.748 ± 0.059**
Amphibians	0.606 ± 0.122	0.714 ± 0.056	**0.735 ± 0.048**	0.714 ± 0.041	0.563 ± 0.012	0.714 ± 0.073
Raisin	0.819 ± 0.048	0.852 ± 0.021	0.853 ± 0.017	0.858 ± 0.015	0.747 ± 0.148	**0.860 ± 0.023**
Tripadvisor	0.673 ± 0.121	0.702 ± 0.041	0.701 ± 0.057	0.708 ± 0.022	0.721 ± 0.008	**0.722 ± 0.008**

The bold is the best one.

where $\widehat{\mathbb{E}}_n[\Psi(g(y, f(x)))] = (1/n)\sum_{i=1}^{n}\Psi(g(f(x_i), y_i))$ and $\widetilde{\Psi}G = \{(x, y)\mapsto\Psi(g(y, f(x))) - \Psi(g(y, 0)): f \in F\}$.

According to the CaEN loss (10) with the optimal w_0^*, we have

$$\widehat{\mathbb{E}}_n[\Psi(g(y, f(x)))] = \frac{1}{n}\sum_{i=1}^{n}\Psi(g(y_i, f(x_i))) = \frac{2}{n}\sum_{i=1}^{n} l_{CaEN}(1 - y_i f(x_i)). \tag{57}$$

Table 4: Average ranks of each SVM with respect to different kernels and ratios of label noise for UCI datasets.

	ENSVM	PinSVM	RampSVM	Rhinge-SVM	Valley-SVM	CaENSVM
(a) Linear kernel						
Avg. rank 0%	3.200	4.833	4.267	3.733	3.500	1.467
Avg. rank 15%	3.567	4.933	3.833	3.433	3.800	1.433
Avg. rank 25%	3.867	5.067	3.733	3.400	3.467	1.467
(b) Gaussian kernel						
Avg. rank 0%	4.367	2.100	3.600	3.700	5.533	1.700
Avg. rank 15%	4.333	3.067	3.400	2.900	5.800	1.500
Avg. rank 25%	4.833	3.033	3.533	2.367	5.200	2.033

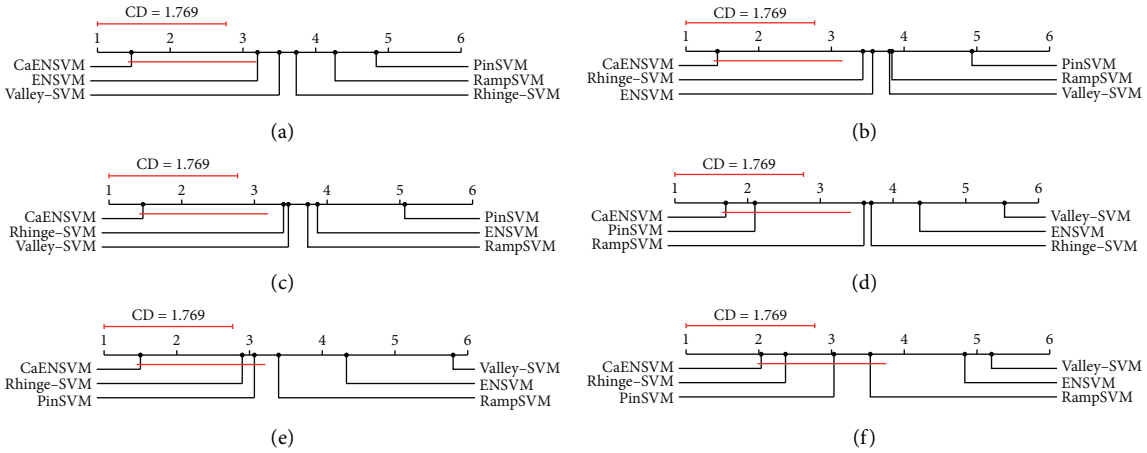

Figure 7: Comparison by the Nemenyi test under 0.1 significant level. (a)–(c) correspond to the linear cases with 0%, 15%, and 25% label noises, while (d)–(f) correspond to the nonlinear cases with 0%, 15%, and 25% label noises, respectively. (a) Case 1. (b) Case 2. (c) Case 3. (d) Case 4. (e) Case 5. (f) Case 6.

By Theorem 14 in the study of Bartlett and Mendelson [6], and since Lipschitz function $\Psi(u)$ pertains to Lipschitz constant $\theta u_2 + (1-\theta)$, we have

$$\widehat{R}_n\left(\widetilde{\Psi}^{\circ}G\right) \le 4\left(\theta u_2 + (1-\theta)\right)\widehat{R}_n(G). \tag{58}$$

According to the definition (51), we can deduce that

$$\begin{aligned}\widehat{R}_n(G) &= \mathbb{E}_\sigma\left[\sup_{g\in G}\left|\frac{2}{n}\sum_{i=1}^{n}\sigma_i g(y_i, f(x_i))\right| : x_1, \cdots, x_n\right] = \mathbb{E}_\sigma\left[\sup_{f\in F}\left|\frac{2}{n}\sum_{i=1}^{n}(-\sigma_i y_i) f(x_i)\right)\right| : x_1, \cdots, x_n\right] \\ &= \mathbb{E}_\sigma\left[\sup_{f\in F}\left|\frac{2}{n}\sum_{i=1}^{n}\sigma_i f(x_i)\right)\right| : x_1, \cdots, x_n\right] = \widehat{R}_n(F).\end{aligned} \tag{59}$$

Hence, by Lemma 22 in the study of Bartlett and Mendelson [6], $\widehat{R}_n(F)$ satisfies

$$\widehat{R}_n(F) \le \frac{2B}{n}\sqrt{\sum_{i=1}^{n}K(x_i, x_i)}. \tag{60}$$

Finally, combining (56)–(58) and (60), we can reach the result of the theorem. □

Remark 2. From Theorem 3, if the number of training samples $n \longrightarrow +\infty$, we have $\text{Prob}(yf(x) \le 0) \longrightarrow 0$. Therefore, the generalization capability of the proposed CaENSVM can be theoretically guaranteed.

Remark 3. According to the proof of Theorem 3, s = 1/2 can lead to a tighter generalization error bound. However, experimental results indicate that larger s may produce a more satisfactory classifier.

5. Numerical Studies

In this section, we conduct a host of experiments to investigate the performance of the proposed CaENSVM on both synthetic and benchmark datasets. For fair assessment, we compare with several famous or recent related SVMs, including ENSVM [39], PinSVM [11], RampSVM [28], Rhinge-SVM [36], and Valley-SVM [13]. Note that Tang et al. [13] first proposed the valley loss and applied it to the

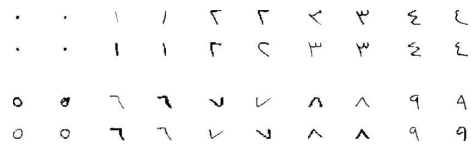

Figure 8: Selected images from the PMU-UD dataset.

RSSVM for robust multiclass classification, we here combine the valley loss with the standard SVM to construct a novel robust Valley-SVM for binary classification. All experiments are carried out in R 4.0.5 on Ubuntu 18 running on a PC with system configuration AMD R9 5900x CPU (3.70 GHz) with 16 GB of RAM.

For PinSVM and CaENSVM, we set $\tau \in \{0.3, 0.5, 0.7\}$. For RampSVM, Valley-SVM, and CaENSVM, we optimize s in $\{1, 1.5, 2\}$. Parameter η of Rhinge-SVM is tuned in $\{0.2, 0.5, 1, 2\}$ like [36] and parameter θ of CaENSVM is tuned in $\{0.1, 0.5, 0.9\}$, respectively. For nonlinear cases, we consider Gaussian kernel, i.e., $K(x_1, x_2) = \exp(-\gamma\|x_1 - x_2\|)$, where $\gamma > 0$ is the kernel parameter. If not otherwise specified, all remaining parameters are optimized in $\{2^{-8}, 2^{-7}, \cdots, 2^7, 2^8\}$, including kernel parameter. We use the five-fold cross-validation strategy to search for the optimal parameters. Note that, for the implemented Pegasos-based DC algorithm, eps is fixed to 10^{-3}, $T_1 = 10$, and $T_2 = 500$ through our experiments. According to the numerical studies, this setting can often lead to a satisfactory result.

5.1. Synthetic Datasets. We generate a two-dimensional synthetic dataset to test the robustness of CaENSVM. The training dataset consists of 60 equal positive and negative samples. The positive samples are independently drawn from a two-dimensional normal distribution with the mean vector $\mu_+ = (-0.4, 1.0)^T$ and the covariance matrix $V = \mathrm{diag}(0.02, 0.06)$. The negative samples are independently drawn from a similar normal distribution with the mean vector $\mu_- = (0.4, 0.2)^T$ and the same covariance matrix.

Case 1. In this case, we only add three extra outliers for positive samples. Figure 3 shows the training samples and the separating lines (black solid lines) obtained by six SVMs. Note that the green "+" notation in each figure is the midpoint between the centers of two classes of samples. Since the distributions behind two classes of training samples only differ from the location of centers, it is reasonable for the obtained separating line crossing the midpoint, i.e., the green "+" notation in each figure. In other words, we can measure the level of disturbance caused by outliers by comparing the relative location between the obtained separating lines and the corresponding midpoints.

According to Figure 3, our proposed CaENSVM is more robust to outliers and produces the most satisfactory classifier. Due to the infinity of EN losses, ENSVM is easily attracted by outliers. As Figure 3(a) shows, ENSVM distinguishes two classes of samples worst, and the obtained separating line is clearly away from the midpoint, indicating its high sensitivity to outliers. Though pinball loss is unbounded, the correctly classified samples also produce losses, which behave like a balance and can reduce the attraction of outliers. From Figure 3(b), PinSVM performs little better than the ENSVM. Note that the separating line induced by the PinSVM is also close to outliers, which means PinSVM is still sensitive to outliers. ramp, Rhinge, and valley losses are all concave and bounded; they can limit the influence of outliers and contribute to robust classifiers. The performances of RampSVM and Rhinge-SVM closely resemble each other. However, both provide totally different shapes of classifiers compared with others. According to Figures 3(c) and 3(d), the corresponding classifiers indicate their overfitness as well as the sensitivity to outliers. Valley-SVM and our proposed CaENSVM share analogical performance. According to Figures 3(e) and 3(f) and based on the midpoint, our CaENSVM seems slightly better than Valley-SVM.

Table 5: The information of the PMU-UD dataset.

ID	Dataset	#Obs	#Fea
1	2-vs-3	937	9600
2	2-vs-4	1003	9600
3	2-vs-6	1007	9600
4	2-vs-7	1019	9600
5	2-vs-8	1003	9600
6	3-vs-4	986	9600
7	6-vs-7	1072	9600
8	6-vs-8	1056	9600
9	7-vs-8	1068	9600

Case 2. In this case, we add three extra outliers for both positive and negative samples. Figure 4 shows the training samples and the final decision lines (black solid lines) of six SVMs. The midpoint between the centers of two classes of samples is also marked in each figure to help measure the robustness of each classifier.

According to Figure 4, our proposed CaENSVM still performs best compared with other five SVMs. ENSVM, RampSVM, and Rhinge-SVM present similar performances. They are all severely attracted by outliers and provide unsatisfactory decision lines. Though the decision line obtained by PinSVM is also deviated by outliers, it seems clearly better than those given by the ENSVM, RampSVM, and Rhinge-SVM. In our opinion, since pinball loss produces losses for corrected classified training samples, it can balance and reduce the influence of outliers to some extent. Valley-SVM performs competitively like PinSVM, but the decision line of Valley-SVM is also drawn by outliers. In comparison with other five SVMs, our proposed CaENSVM shows the best robustness against outliers as before.

Case 3. In this case, we investigate the time cost of the implemented Pegasos-based DC procedure for the proposed CaENSVM. Specifically, we turn to generate from 50 to 5000 equal positive and negative training samples. For a fair assessment, we compare the PinSVM solved by clipDCD [47] and Rhinge-SVM solved by clipDCD-based half-

Table 6: The mean accuracy (Acc.) and standard deviation (sd) with linear kernel for the PMU-UD dataset.

	ENSVM Acc. ± sd	PinSVM Acc. ± sd	RampSVM Acc. ± sd	Rhinge-SVM Acc. ± sd	Valley-SVM Acc. ± sd	CaENSVM Acc. ± sd
2-vs-3	0.942 ± 0.011	0.922 ± 0.006	0.942 ± 0.004	0.975 ± 0.010	0.949 ± 0.054	**0.979 ± 0.017**
2-vs-4	0.947 ± 0.026	0.925 ± 0.015	0.925 ± 0.020	0.948 ± 0.020	**0.951 ± 0.022**	**0.951 ± 0.014**
2-vs-6	0.999 ± 0.002	0.997 ± 0.012	0.997 ± 0.014	0.990 ± 0.004	0.999 ± 0.002	**1.000 ± 0.000**
2-vs-7	**0.999 ± 0.002**	0.992 ± 0.009	0.993 ± 0.003	0.993 ± 0.006	**0.999 ± 0.002**	**0.999 ± 0.002**
2-vs-8	0.994 ± 0.003	0.995 ± 0.001	0.992 ± 0.001	0.991 ± 0.007	0.997 ± 0.003	**0.998 ± 0.003**
3-vs-4	0.981 ± 0.002	0.933 ± 0.002	0.953 ± 0.005	0.971 ± 0.008	**0.983 ± 0.012**	0.976 ± 0.011
5-vs-9	**0.994 ± 0.002**	0.980 ± 0.004	0.981 ± 0.006	**0.988 ± 0.011**	0.994 ± 0.002	0.988 ± 0.007
6-vs-7	0.962 ± 0.012	0.970 ± 0.023	0.965 ± 0.020	**0.985 ± 0.012**	0.960 ± 0.056	0.973 ± 0.043
6-vs-8	0.996 ± 0.006	0.988 ± 0.010	0.989 ± 0.013	0.980 ± 0.016	0.994 ± 0.008	**0.998 ± 0.002**
7-vs-8	0.973 ± 0.006	0.983 ± 0.008	0.970 ± 0.121	0.960 ± 0.010	0.971 ± 0.011	**0.986 ± 0.004**

The bold is the best one.

quadratic optimization algorithm [36]. All tuning parameters are fixed to 0.5 for simplicity. Figures 5 and 6 show the one-run CPU time of PinSVM, Rhinge-SVM, and CaENSVM with linear and Guassian kernels, respectively.

According to Figures 5 and 6, our implemented Pegasos-based DC procedure obviously runs the fastest in comparison with PinSVM and Rhinge-SVM. For the linear kernel, PinSVM and Rhinge-SVM consume time similarly, though Rhinge-SVM needs to iterate a clipDCD chunk many times. It may be that the adaptive weighting scheme and sparsity of Rhinge-SVM can help reduce the training cost. Our CaENSVM runs fast, and the time cost is free of the sample size, indicating the scalability of Pegasos [43]. For the Guassian kernel, Rhinge-SVM is clearly more time-consuming than PinSVM due to the outer iteration of the half-quadratic procedure. The training time of CaENSVM also depends on the sample size, but it is still efficient. According to the experimental results, our proposed CaENSVM can be easily applied in solving large-scale data classification problems.

5.2. UCI Datasets

5.2.1. Experimental Settings and Results. We select twelve UCI datasets to further demonstrate the advantages of our proposed CaENSVM. The detailed information of the chosen datasets is listed in Table 1. To investigate the label noise insensitivity, we artificially add 15% and 25% label noises to the raw datasets, i.e., we randomly select 15% and 25% training samples and exchange their labels. The experimental results with linear and Gaussian kernels based on five-fold cross-validation criterion are shown in Tables 2 and 3, respectively.

From Table 2 our proposed CaENSVM with linear kernel outperforms others in most cases, according to the average prediction accuracies. For the case without label noise, our CaENSVM performs slightly better than the ENSVM, followed by Valley-SVM. Rhinge-SVM and RampSVM have competitive performances, while PinSVM seems to be the worst. Along with the ratio of label noise increasing, the performances of all SVMs seem roughly reduced. Specifically, the prediction accuracy of the ENSVM is most affected by label noise, since the elastic net loss lacks robustness. Though PinSVM performs little stably due to the asymmetric pinball loss, its prediction accuracy is often the most unsatisfactory and competitive with the ENSVM for the case with moderate and high label noise. Due to the robustness of ramp, Rhinge, and valley losses, RampSVM and Rhinge-SVM together with Valley-SVM present more clearly high prediction accuracies for the case with label noise in comparison with ENSVM and PinSVM. Because our designed CaEN loss enjoys both outlier insensitivity and resampling stability, the CaENSVM always achieves the highest average prediction accuracies for the cases with label noise. Therefore, the advantage of CaENSVM becomes more concrete, indicating its good robustness to label noise.

From Table 3, our proposed CaENSVM with Gaussian kernel slightly performs better than others according to the average prediction accuracies. For the raw datasets, the CaENSVM gives higher average prediction accuracies in more than half of the datasets. The PinSVM turns to perform well, only behind the CaENSVM. RampSVM also has a good performance, but it is clearly worse than PinSVM. The remaining three SVMs are competitive with each other. For the datasets with moderate and high label noise, the CaENSVM still has a weak superiority, followed by Rhinge-SVM and PinSVM. ENSVM and Valley-SVM seem to be the worst in most datasets. Compared with the predicting performance in the linear cases, the superiority of our proposed CaENSVM is not overwhelming. In our opinion, this situation is possibly caused by the Pegasos procedure, since the efficiency of Pegasos is reduced in the nonlinear cases [43]. We will consider designing efficient algorithms for nonlinear cases in future.

5.2.2. Comparisons by Statistical Test. In this part, we conduct a statistical test on the experimental results of UCI datasets to further demonstrate the advantage of the CaENSVM. Specifically, we utilize the famous Friedman test with the corresponding post hoc test [48] to study whether there is a statistically significant difference among CaENSVM and other five compared SVMs. Firstly, we calculate the average ranks of each method with respect to different kernels and ratios of label noise in Tables 2 and 3. The results are presented in Table 4.

Secondly, we need to obtain the Friedman statistic based on Table 4. Let $\mathscr{D}_n$ and $\mathscr{C}_k$ be the total number of compared datasets and classifiers, respectively. By Tables 2 and 3, we

Table 7: The mean accuracy (Acc.) and standard deviation (sd) with Gaussian kernel for the PMU-UD dataset.

	ENSVM Acc. ± sd	PinSVM Acc. ± sd	RampSVM Acc. ± sd	Rhinge-SVM Acc. ± sd	Valley-SVM Acc. ± sd	CaENSVM Acc. ± sd
2-vs-3	0.954 ± 0.024	0.990 ± 0.010	0.987 ± 0.008	**0.992 ± 0.010**	0.952 ± 0.039	0.979 ± 0.018
2-vs-4	0.914 ± 0.029	0.994 ± 0.007	0.972 ± 0.014	0.995 ± 0.005	0.955 ± 0.023	**0.996 ± 0.002**
2-vs-6	0.997 ± 0.004	0.999 ± 0.002	**1.000 ± 0.000**	0.999 ± 0.002	0.962 ± 0.029	0.967 ± 0.030
2-vs-7	**1.000 ± 0.000**	**1.000 ± 0.000**	**1.000 ± 0.000**	**1.000 ± 0.000**	0.995 ± 0.004	**1.000 ± 0.000**
2-vs-8	0.988 ± 0.012	**1.000 ± 0.000**	0.998 ± 0.003	**1.000 ± 0.000**	0.806 ± 0.257	**1.000 ± 0.000**
5-vs-9	0.987 ± 0.011	**0.998 ± 0.003**	0.993 ± 0.005	**0.998 ± 0.003**	0.947 ± 0.029	0.949 ± 0.062
6-vs-7	0.962 ± 0.036	0.997 ± 0.003	0.993 ± 0.005	0.997 ± 0.003	0.971 ± 0.025	**1.000 ± 0.000**
6-vs-8	0.899 ± 0.198	0.995 ± 0.004	0.994 ± 0.006	0.939 ± 0.095	0.996 ± 0.004	**0.998 ± 0.003**
7-vs-8	0.966 ± 0.019	0.981 ± 0.006	0.981 ± 0.008	**0.992 ± 0.004**	0.966 ± 0.020	0.985 ± 0.010

The bold is the best one.

have $\mathscr{D}_n = 15$ and $\mathscr{C}_k = 6$ for each type of kernel and ratio of label noise. Then, we consider the following F statistic:

$$F_F = \frac{(\mathscr{D}_n - 1)\chi_F^2}{\mathscr{D}_n(\mathscr{C}_k - 1) - \chi_F^2}, \tag{61}$$

where χ_F^2 is the raw Friedman statistic defined as

$$\chi_F^2 = \frac{12\mathscr{D}_n}{\mathscr{C}_k(\mathscr{C}_k + 1)}\left(\sum_{i=1}^{\mathscr{C}_k}\mathscr{R}_i^2 - \frac{\mathscr{C}_k(\mathscr{C}_k + 1)^2}{4}\right), \tag{62}$$

where $\mathscr{R}_i$ is the average rank of the i-th classifier for each type of kernel and ratio of label noise. The obtained F_F statistic obeys F distribution with $(\mathscr{C}_k - 1)$ and $(\mathscr{C}_k - 1)(\mathscr{D}_n - 1)$ degrees of freedom. When the significant level is set to 0.1, we have $F_{0.1}(5, 70) = 1.93$. According to Table 4, we obtain that the values of F_F are 8.57, 8.34, and 8.87 with respect to different ratios of label noise for linear cases, 19.27, 21.21, and 12.69 with respect to different ratios of label noise for nonlinear cases. All values of F_F statistic are larger than the critical value 1.93, which means that there is indeed a statistical difference among those compared six SVMs.

Thirdly, we apply the Nemenyi test for post hoc test to further distinguish the detailed differences of six classifiers. The critical domain (CD) of the difference of the average ranks of two SVMs given by Nemenyi is defined as

$$CD = q_{0.1}\sqrt{\frac{\mathscr{C}_k(\mathscr{C}_k + 1)}{6\mathscr{D}_n}} = 1.769, \tag{63}$$

where $q_{0.1} = 2.589$. If the absolute difference of two SVMs is larger than CD, it means that they perform statistically differently. Otherwise, they have no statistical differences with each other. Figure 7 shows the comparison of the average ranks of each SVM for different type of kernels and ratios of label noise.

According to Figure 7, there is no significant difference between CaENSVM and ENSVM with linear kernel. However, by Figures 7(a)–7(c), the difference between CaENSVM and ENSVM becomes larger as the ratio of label noise increases, which means the proposed CaENSVM is still better than others. Rhinge-SVM, RampSVM, and PinSVM always have no significant differences for linear cases, they all are significantly worse than CaENSVM. Valley-SVM shows good robustness for a high ratio of label noise, but there is still a gap between it and CaENSVM. For the cases with Gaussian kernel, our CaENSVM always presents slightly better performances than others, though the differences among it and Rhinge-SVM and PinSVM are not significant. The performances of Valley-SVM and ENSVM are significantly worse than those of other four methods, especially for high ratio of label noise. In short, our CaENSVM enjoys good performance in the statistical viewpoint.

5.3. Handwritten Digit Recognition. In this subsection, we apply CaENSVM to solve a real problem, i.e., handwritten digit recognition. The test dataset is the PMU-UD dataset from the study of Alghazo et al. [49], containing handwritten Urdu/Arabic numerals from 0 to 9. Each handwritten number is standardized as a 120×80 image. Figure 8 shows four selected images for every handwritten number. We choose the datasets corresponding to two different handwritten numbers each time to conduct binary classification. The information of all considered datasets is listed in Table 5.

For a fair evaluation, we also compare our CaENSVM with ENSVM, PinSVM, RampSVM, Rhinge-SVM, and Valley-SVM. The average prediction accuracies with linear and Gaussian kernels based on the five-fold cross-validation criterion are presented in Tables 6 and 7, respectively. According to Tables 6 and 7, our CaENSVM achieves the highest prediction accuracies in more than half of the cases, indicating its excellent performance in handwritten digit recognition problems.

6. Conclusion

In this paper, we have proposed a novel robust support vector classifier (CaENSVM) with a capped elastic net loss function. Theoretical analysis is conducted to thoroughly demonstrate the properties of CaENSVM, including noise insensitivity, Bayes rule, and the generalization error bound based on the Rademacher complexity. It is worth noting that we use the influence function to explain well the robustness of CaENSVM. Though the constructed CaEN loss is nonconvex, the implemented Pegasos-based DC algorithm can efficiently solve the CaENSVM optimization problem. The results of numerical studies indicate the following: (1) our CaENSVM is robust to outliers and performs better than many similar state-of-the-art SVMs according to prediction accuracy. The superiority of the

CaENSVM is also supported by the statistical test. (2) The performance of the CaENSVM for nonlinear cases is not overwhelming like the performance for linear cases in comparison with other methods, indicating that there may be a room for improving the efficiency of the algorithm. In fact, though Pegasos algorithm is scalable, the performance with nonlinear kernel is little unsatisfactory. Further work will focus on designing a more stable and efficient algorithm to achieve higher prediction accuracy, especially for nonlinear cases. Note that the *R* code of the proposed CaENSVM is developed by the authors and it is available at https://github.com/jiandan94/CaENSVM.

Data Availability

Data are available from the authors upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by the fund from the National Natural Science Foundation of China 11671059.

References

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

[2] C. Acevedo, J. Gómez, and C. Rojas, "Academic stress detection on university students during COVID-19 outbreak by using an electronic nose and the galvanic skin response," *Biomedical Signal Processing and Control*, vol. 68, Article ID 102756, 2021.

[3] K. Qi and H. Yang, "Elastic net nonparallel hyperplane support vector machine and its geometrical rationality," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 33, no. 12, pp. 7199–7209, 2022.

[4] D. Wang, X. Zhang, H. Chen, Y. Zhou, and F. Cheng, "A sintering state recognition framework to integrate prior knowledge and hidden information considering class imbalance," *IEEE Transactions on Industrial Electronics*, vol. 68, no. 8, pp. 7400–7411, 2021.

[5] A. Sadeghi, A. Daneshvar, and M. Madanchi Zaj, "Combined ensemble multi-class SVM and fuzzy NSGA-II for trend forecasting and trading in Forex markets," *Expert Systems with Applications*, vol. 185, Article ID 115566, 2021.

[6] P. Bartlett and S. Mendelson, "Rademacher and Gaussian complexities: risk bounds and structural results," *Journal of Machine Learning Research*, vol. 3, pp. 463–482, 2002.

[7] P. L. Bartlett, M. I. Jordan, and J. D. McAuliffe, "Convexity, classification, and risk bounds," *Journal of the American Statistical Association*, vol. 101, pp. 138–156, 2006.

[8] G. Blanchard, O. Bousquet, and P. Massart, "Statistical performance of support vector machines," *Annals of Statistics*, vol. 36, no. 2, pp. 489–531, 2008.

[9] O. Mangasarian and D. Musicant, "Lagrangian support vector machines," *Journal of Machine Learning Research*, vol. 1, pp. 161–177, 2001.

[10] X. Huang, L. Shi, and J. Suykens, "Ramp loss linear programming support vector machine," *Journal of Machine Learning Research*, vol. 15, no. 1, pp. 2185–2211, 2014.

[11] X. Huang, L. Shi, and J. A. K. Suykens, "Support vector machine classifier with pinball loss," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 5, pp. 984–997, 2014.

[12] L. Xu, X. Wang, L. Bai et al., "Probabilistic SVM classifier ensemble selection based on GMDH-type neural network," *Pattern Recognition*, vol. 106, Article ID 107373, 2020.

[13] L. Tang, Y. Tian, W. Li, and P. M. Pardalos, "Valley-loss regular simplex support vector machine for robust multiclass classification," *Knowledge-Based Systems*, vol. 216, Article ID 106801, 2021.

[14] W. Wang and X. Qiao, "Set-Valued support vector machine with bounded error rates," *Journal of the American Statistical Association*, pp. 1–13, 2022.

[15] C. Fu, S. Zhou, J. Zhang, B. Han, Y. Chen, and F. Ye, "Risk-Averse support vector classifier machine via moments penalization," *International Journal of Machine Learning and Cybernetics*, vol. 13, no. 11, pp. 3341–3358, 2022.

[16] Y. Xu, Z. Yang, and X. Pan, "A novel twin support-vector machine with pinball loss," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 28, no. 2, pp. 359–370, 2017.

[17] X. Huang, L. Shi, and J. A. Suykens, "Asymmetric least squares support vector machine classifiers," *Computational Statistics & Data Analysis*, vol. 70, pp. 395–405, 2014.

[18] M. Z. Liu, Y. H. Shao, C. N. Li, and W. J. Chen, "Smooth pinball loss nonparallel support vector machine for robust classification," *Applied Soft Computing*, vol. 98, Article ID 106840, 2021.

[19] K. Li and Z. Lv, "Smooth twin bounded support vector machine with pinball loss," *Applied Intelligence*, vol. 51, no. 8, pp. 5489–5505, 2021.

[20] X. Shen, L. Niu, Z. Qi, and Y. Tian, "Support vector machine classifier with truncated pinball loss," *Pattern Recognition*, vol. 68, pp. 199–210, 2017.

[21] Z. Yang and Y. Xu, "A safe accelerative approach for pinball support vector machine classifier," *Knowledge-Based Systems*, vol. 147, pp. 12–24, 2018.

[22] M. Tanveer, A. Tiwari, R. Choudhary, and S. Jalan, "Sparse pinball twin support vector machines," *Applied Soft Computing*, vol. 78, pp. 164–175, 2019.

[23] S. Sharma, R. Rastogi, and S. Chandra, "Large-scale twin parametric support vector machine using pinball loss function," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 51, no. 2, pp. 987–1003, 2021.

[24] H. Wang and Y. Xu, "Sparse elastic net multi-label rank support vector machine with pinball loss and its applications," *Applied Soft Computing*, vol. 104, Article ID 107232, 2021.

[25] D. Gupta, B. B. Hazarika, and M. Berlin, "Robust regularized extreme learning machine with asymmetric Huber loss function," *Neural Computing & Applications*, vol. 32, no. 16, Article ID 12971, 2020.

[26] S. C. Prasad, P. Anagha, and S. Balasundaram, "Robust pinball twin bounded support vector machine for data classification," *Neural Processing Letters*, 2022.

[27] D. Gupta and U. Gupta, "On robust asymmetric Lagrangian ν-twin support vector regression using pinball loss function," *Applied Soft Computing*, vol. 102, Article ID 107099, 2021.

[28] Y. Wu and Y. Liu, "Robust truncated hinge loss support vector machines," *Journal of the American Statistical Association*, vol. 102, no. 479, pp. 974–983, 2007.

[29] L. Wang, H. Jia, and J. Li, "Training robust support vector machine with smooth Ramp loss in the primal space," *Neurocomputing*, vol. 71, no. 13-15, pp. 3020–3025, 2008.

[30] D. Liu, Y. Shi, and Y. Tian, "Ramp loss nonparallel support vector machine for pattern classification," *Knowledge-Based Systems*, vol. 85, pp. 224–233, 2015.
[31] S. M. Hosseini Bamakan, H. Wang, and Y. Shi, "Ramp loss K-Support Vector Classification-Regression; a robust and sparse multi-class approach to the intrusion detection problem," *Knowledge-Based Systems*, vol. 126, pp. 113–126, 2017.
[32] C. Wang, Q. Ye, P. Luo, N. Ye, and L. Fu, "Robust capped L1-norm twin support vector machine," *Neural Networks*, vol. 114, pp. 47–59, 2019.
[33] C. Yuan and L. Yang, "Capped L2,p-norm metric based robust least squares twin support vector machine for pattern classification," *Neural Networks*, vol. 142, pp. 457–478, 2021.
[34] P. Borah and D. Gupta, "Robust twin bounded support vector machines for outliers and imbalanced data," *Applied Intelligence*, vol. 51, no. 8, pp. 5314–5343, 2021.
[35] G. Xu, B. G. Hu, and J. C. Principe, "Robust C-loss kernel classifiers," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, no. 3, pp. 510–522, 2018.
[36] G. Xu, Z. Cao, B. G. Hu, and J. C. Principe, "Robust support vector machines based on the rescaled hinge loss function," *Pattern Recognition*, vol. 63, pp. 139–148, 2017.
[37] L. Yang and H. Dong, "Robust support vector machine with generalized quantile loss for classification and regression," *Applied Soft Computing*, vol. 81, Article ID 105483, 2019.
[38] K. Qi and H. Yang, "Joint rescaled asymmetric least squared nonparallel support vector machine with a stochastic quasi-Newton based algorithm," *Applied Intelligence*, vol. 52, no. 12, Article ID 14387, 2022.
[39] K. Qi, H. Yang, Q. Hu, and D. Yang, "A new adaptive weighted imbalanced data classifier via improved support vector machines with high-dimension nature," *Knowledge-Based Systems*, vol. 185, Article ID 104933, 2019.
[40] K. R. Jayadeva, R. Khemchandani, and S. Chandra, "Twin support vector machines for pattern classification," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 5, pp. 905–910, 2007.
[41] J. Ma, L. Yang, and Q. Sun, "Adaptive robust learning framework for twin support vector machine classification," *Knowledge-Based Systems*, vol. 211, Article ID 106536, 2021.
[42] L. T. H. An and P. D. Tao, "The DC (difference of convex functions) programming and DCA revisited with DC models of real world nonconvex optimization problems," *Annals of Operations Research*, vol. 133, no. 1-4, pp. 23–46, 2005.
[43] S. Shalev-Shwartz, Y. Singer, N. Srebro, and A. Cotter, "Pegasos: primal estimated sub-gradient solver for svm," *Mathematical Programming*, vol. 127, no. 1, pp. 3–30, 2011.
[44] F. Hampel, *Contributions to the theory of robust estimation*, PhD thesis, University of California, Berkeley, CA, USA, 1968.
[45] X. Wang, Y. Jiang, M. Huang, and H. Zhang, "Robust variable selection with exponential squared loss," *Journal of the American Statistical Association*, vol. 108, no. 502, pp. 632–643, 2013.
[46] Y. Lin, "A note on margin-based loss functions in classification," *Statistics & Probability Letters*, vol. 68, no. 1, pp. 73–82, 2004.
[47] X. Peng, D. Chen, and L. Kong, "A clipping dual coordinate descent algorithm for solving support vector machines," *Knowledge-Based Systems*, vol. 71, pp. 266–278, 2014.
[48] J. Demšar, "Statistical comparisons of classifiers over multiple data sets," *Journal of Machine Learning Research*, vol. 7, pp. 1–30, 2006.
[49] J. M. Alghazo, G. Latif, L. Alzubaidi, and A. Elhassan, "Multi-language handwritten digits recognition based on novel structural features," *Journal of Imaging Science and Technology*, vol. 63, no. 2, Article ID 20502, 2019.