Dual Balanced Large Marginal Machine for Imbalanced Data Classification

Denghao Dong[a], Shuangyi Fan[a], Junlin Chen[a], Fudi Wang[a], Libo Zhang[a,*]

[a] *College of Artificial Intelligence, Southwest University, Chongqing, 400715, P.R. China*

Abstract

As a modification of Support Vector Machine (SVM), Large Margin Distribution Machine (LDM) enhances generalization performance by integrating the marginal distribution of samples into the objective function. Nevertheless, the current LDM models (LDMs) still suffer from limitations in imbalanced classification tasks, i.e., the classification hyperplane tends to be skewed towards the minority class. Moreover, the inherited hinge loss function from SVM results in inferior noise immunity and stability. To overcome these limitations, we propose a novel dual balanced (DB) distance function to replace the traditional distance metric and introduce the unified pinball loss (UP) as a replacement for the inherited hinge loss, which significantly enhances the model's generalization capacity in imbalanced classification tasks. Specifically, we combine the imbalance level and bias to construct a pair of composite distances for the majority and minority classes, respectively. The dual balanced distance contributes to reducing the slackness of the minority class and the influence of the majority class, leading to a balanced classification hyperplane. Additionally, the introduction of the unified pinball loss function improves the model's robustness and noise resistance. We have evaluated our proposed model in the fields of meteorology, finance, healthcare, and engineering, demonstrating its superior performance.

Keywords: Imbalanced Data, Large Marginal Distribution Machine, Composite Distance, Pinball Loss.

1. Introduction

Support Vector Machine (SVM) is a widely used machine learning classification method introduced by Vapnik et al. [1]. It has been widely utilized in classification [2, 3, 4, 5, 6], face detection [7, 8], and other fields [9, 10, 11, 12]. To minimize the structural risk, SVM maximizes the minimum interval between two classes, i.e., maximizes the distance between the support vector and the classification hyperplane. However, according to the marginal theory, marginal distribution, rather than the minimum interval, is more important for the generalization performance [13]. Such

*Corresponding author: Libo Zhang

Email addresses: dhdong1129@163.com (Denghao Dong), ShuangyiFan111@163.com (Shuangyi Fan), jlchen08@163.com (Junlin Chen), blue16@email.swu.edu.cn (Fudi Wang), lbzhang@swu.edu.cn (Libo Zhang)

a conjecture has been theoretically investigated [14, 15], and the assertion is finally proved by Gao and Zhou [15]. Therefore, based on this theory, Zhang et al. proposed the Large Margin Distribution Machine (LDM), which achieves better generalization performance [16]. LDM considers the overall distribution of samples, not just support vectors [17, 18]. Specifically, it introduces two statistics, margin mean and margin variance, into the objective function, and optimizes the interval distribution while ensuring the minimum structural risk.

However, the introduction of the second-order statistic in LDM also poses some problems. Since the LDM needs to maximize the mean margin, it leads to a hyperplane skewed toward the minority classes, which generates low generalization performance for the samples of fewer classes. Therefore, the classification task for imbalanced datasets becomes less accurate and performance degrades. Moreover, since LDM is an improvement on the SVM, it is a direct addition of two statistics and does not modify the loss function, it still uses the hinge loss function. It is found that the hinge loss function may make the model sensitive to noise [19, 20]. Specifically, the hinge loss is only related to the shortest distance between sets, which leads to sensitivity to noise and instability to resampling [21]. There are various improvements to the LDM model, such as Fuzzy LDM, Cost-Sensitive LDM (CS-LDM) [22] and Unconstrained LDM (ULDM) [23] etc. Despite the promising results of these modifications, the existing LDM model does not achieve the expected results in terms of targeting the noise introduced by imbalanced datasets [24, 25]. Although the influence of noise can be weakened by introducing the fuzzy set theory and giving samples different confidence levels [26, 27, 28, 29, 30], these methods fail to keep the elegant formulation of the classic SVM and LDM [19].

To address the fundamental problems of LDM, structural changes are required. The introduction of margin mean in LDM reduces its generalization performance for minority classes. Therefore, it is necessary to revise the margin definition to give more weight to minority class samples. To tackle this problem, the contribution of minority class samples should be improved by changing the definition of the margin. Cheng et al. [22] constructed a distance coefficient based on the number of imbalances, which achieves a better result. However, it failed to consider both the degree of data imbalance and bias and the adjustment was not satisfactory. In addition, CSLDM does not provide additional treatment for the large class samples and thus needs to be improved. Moreover, to enhance the noise immunity of the imbalance classification task, the loss function also needs to be improved. The pinball loss is related to quantiles, which is well studied in regression [31, 32, 33]. By replacing the hinge loss, Huang et al. first introduce pinball loss into SVM (PinSVM), which achieves better anti-noise performance, as it constructs penalty terms for all samples, not just the nearest sample points in traditional SVM[19, 17]. In 2021, Anand et al. propose a Unified Pinball (UP) loss function, which is theoretically and empirically better than pinball loss, particularly for noise-corrupted data [34].

Motivated by the above work, we reworked the definition of the interval, introduced the composite distance, and improved the structure of the LDM by proposing the LDM with UP loss function (DBUPLDM). Specifically, DBUPLDM sets two balancing factors to increase the importance of the minority class while weakening the dominance of the majority class. We set an adaptive balance function to increase the distance of the small class samples and decrease the distance of the large class samples, and set a higher penalty term for the small class samples to suppress their slackness. Moreover, the proposed DBUPLDM includes the traditional LDM as a special case. The main contributions of this paper are summarized below:

- The proposed DBUPLDM has an adaptive balanced function, which can effectively solve the problem of the imbalanced dataset.
- To achieve better noise immunity, the quantile distance is introduced to replace the shortest distance between sample sets in traditional LDM.
- The results of various experiments indicate that the proposed models are superior to the existing models in terms of classification accuracy, stability, and anti-noise performance.

The rest of this paper is organized as follows: Section 2 briefly reviews the traditional LDM models, as well as the UP loss function. Section 3 proposes the Dual Balanced method and DBUPLDM model. Section 4 discusses the main findings of our experimental results on benchmark datasets. Finally, Section 5 summarizes this paper and also presents future developments.

2. Related work

In this section, the traditional LDM with hinge loss and the UP loss are briefly introduced.

2.1. Traditional LDM with hinge loss

LDM introduces the margin mean and variance of the distance from all sample points to their respective hyperplanes based on SVM, so it has better generalization ability due to taking into account the distribution information of all sample points. The margin mean and variance are given by

$$\gamma_1 = \frac{1}{l}\sum_{i=1}^{l} y_i\left(w^T x_i + b\right),$$

$$and \tag{1}$$

$$\gamma_2 = \sum_{i=1}^{l}\sum_{j=1}^{l}\left(y_i\left(w^T x_i + b\right) - y_j\left(w^T x_j + b\right)\right)^2.$$

The hard-margin LDM model can be written as

$$\begin{aligned} &\min_{w} \frac{1}{2}\|w\|^2 - \lambda_1\gamma_1 + \lambda_2\gamma_2 \\ &\text{s.t. } y_i(w^T x_i + b) \geq 1, i = 1, \ldots, l, \end{aligned} \tag{2}$$

where λ_1 and λ_2 are the parameters for trading off the margin variance, the margin mean and the model complexity.

In real-world scenarios, most data cannot be directly separated linearly. For the non-separable cases, similar to soft-margin SVM, the soft-margin LDM is defined as

$$
\begin{aligned}
&\min_{w,\xi} \frac{1}{2}\|w\|^2 - \lambda_1\gamma_1 + \lambda_2\gamma_2 + C_0\sum_{i=1}^{l} L_{\text{hinge}}\ \left(1 - y_i\left(w^T x_i + b\right)\right) \\
&\text{s.t. } L_{\text{hinge}}\ \left(1 - y_i\left(w^T x_i + b\right)\right) \geq 1 - y_i(w^T x_i + b) \\
&\qquad L_{\text{hinge}}\ \left(1 - y_i\left(w^T x_i + b\right)\right) \geq 0, i = 1, \ldots, l,
\end{aligned}
\tag{3}
$$

where C_0 is penalty factor and $L_{\text{hinge}}\ \left(1 - y_i\left(w^T x_i + b\right)\right)$ is the hinge loss, which is defined as

$$
L_{\text{hinge}}\ (u) = \max\{0, u\}, \forall u \in \mathbb{R}. \tag{4}
$$

Using the hinge loss function L_{hinge}, a similar reasoning can convert the soft-margin LDM (i.e. Eq. (3)) into an unconstrained problem

$$
\min_{w,\xi} \frac{1}{2}\|w\|^2 - \lambda_1\gamma_1 + \lambda_2\gamma_2 + C_0\sum_{i=1}^{l} L_{\text{hinge}}\ \left(1 - y_i\left(w^T x_i + b\right)\right). \tag{5}
$$

As can be seen from the Eq. (5), LDM inherits the hinge loss, which leads to the question that LDM is sensitive to noise and unstable to resampling.

2.2. The UP loss function

Generally, the Pinball loss function can be written as

$$
L_\tau(u) = \max\{u, -\tau u\}, \tag{6}
$$

where $\tau \in [-1, 1]$ is the parameter that determines the quantile level. Note that the hinge and l_1 loss functions are recovered when $\tau = 0$ and 1, respectively. According to Eq. (4) and (6), the relationship between UP and hinge losses is depicted clearly in Fig. 1.

According to Anand et al., when τ is positive or negative, the classifier model is inconsistent [34]. Therefore, we refer to the pinball loss that applies to both positive and negative values as UP loss. Denote $u = 1 - y\left(w^T x + b\right)$, and it represents the distance from the sample point to the hyperplane of this class for SVM. As can be seen from Fig. 1, when $u > 0$, it means that the sample points appear between the positive and negative hyperplanes. In this case, both hinge loss and UP loss will punish these samples in unit rate. When $u < 0$, it means that the sample points are correctly classified and appear outside the hyperplane. In this case, the penalty given to these samples by hinge loss is 0, but UP loss will also give these samples a penalty at a relatively low rate τ. The above difference shows that hinge loss only focuses on these samples that appear between the positive and negative hyperplanes, while the UP loss takes into account all sample points and is less sensitive to noise.

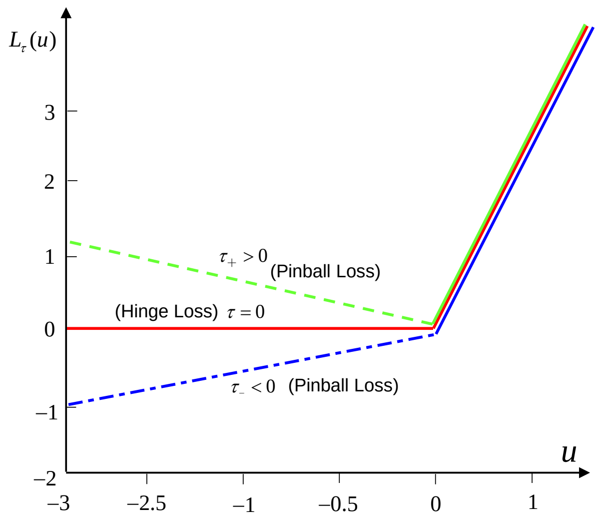


Figure 1: Comparison of loss functions $L_{hinge}(u)$ and $L_\tau(u)$.

Theory and experiments prove that, compared with hinge loss, UP loss can bring better classification performance to SVM. By replacing hinge loss with UP loss, we can get the linearized UPSVM model as follows

$$\min_{w,b} \frac{1}{2}\|w\|^2 + C_0 \sum_{i=1}^{l} L_\tau \left(1 - y_i \left(w^T x_i + b\right)\right), \tag{7}$$

which is equivalent to

$$\min_{w,b} \frac{1}{2}\|w\|^2 + C_0 \sum_{i=1}^{l} \left(\max\left\{1 - y_i \left(w^T x_i + b\right), -\tau \left(1 - y_i \left(w^T x_i + b\right)\right)\right\}\right). \tag{8}$$

In Eq. (8), Pin-SVM and traditional SVM can be recovered by letting $\tau \geq 0$ and $\tau = 0$, respectively.

3. DBUPLDM

When dealing with imbalanced datasets, models are often affected by this imbalance to varying degrees. Since the LDM needs to maximize the mean margin, the classification hyperplane is skewed toward a class with a small size, which reduces the model's ability to generalize to samples from imbalanced datasets. Firstly, a new dual balanced factor function is introduced in subsection 3.1. Based on the dual balance function and UP loss function, we propose the DBUPLDM model in subsection 3.2. In subsection 3.3, we present the explicit solution to DBUPLDM and provide the derivation in detail.

3.1. Dual balance

This subsection proposes a novel balance factor to measure the degree of imbalance, from which the composite distance is constructed.

Definition 1. *The penalty function of DBUPLDM is defined as:*

$$C_i{}^* = \begin{cases} C_0, x_i \in \mathit{Small\ Category} \\ kC_0, x_i \in \mathit{Large\ Category} \end{cases}, k = \min(\psi^-, \psi^+)/\max(\psi^-, \psi^+). \tag{9}$$

From Definition 1, it can be seen that k is used to denote the proportion of imbalanced categories and can measure the magnitude of this bias. k is always a value less than one, thus reducing the penalty of large category samples in slack and increasing the penalty factor of small category samples. Therefore, the relaxation of the small category samples is suppressed and their importance is enhanced.

Definition 2. *Define the balance factor as:*

$$\theta_i = 1 \pm (\frac{1}{k})^\rho \frac{|\psi^- - \psi^+|}{\psi^- + \psi^+}, \tag{10}$$

where ψ represents the number of samples, $k = \min(\psi^-, \psi^+)/\max(\psi^-, \psi^+)$, and ρ is an adjustable parameter.

Definition 2 presents the balance factor function. For imbalanced datasets, the classification hyperplane will be biased towards small class samples, which reduces the generalization performance. The balance factor function is used to adjust for this undesirable effect. For large class samples, the balance factor function generates an indicator variable greater than zero and less than one. On the other hand, for small class samples, it generates a variable greater than 1. For different situations, the balance factor function can be expanded as:

$$\theta_i = \begin{cases} 1 + y_i(\psi^-/\psi^+)^\rho(\psi^- - \psi^+)/(\psi^+ + \psi^-), 0 < \psi^+ < \psi^- \\ 1 - y_i(\psi^+/\psi^-)^\rho(\psi^+ - \psi^-)/(\psi^+ + \psi^-), 0 < \psi^- < \psi^+ \end{cases}. \tag{11}$$

Definition 3. *The composite distance is defined as:*

$$\gamma^* = \theta_{\mathrm{i}} y_i \omega^T \phi(x_i) \tag{12}$$

Considering that θ_i is a segmented function, bringing it into Eq. (12), the form of the composite distance can be written as:

$$\gamma^* = \theta_{\mathrm{i}} y_i \omega^T \phi(x_i) = (1 \pm \Delta) y_i \omega^T \phi(x_i) = \gamma \pm \Delta y_i \omega^T \phi(x_i), \Delta = (\frac{1}{k})^\rho \frac{|\psi^- - \psi^+|}{\psi^- + \psi^+} \tag{13}$$

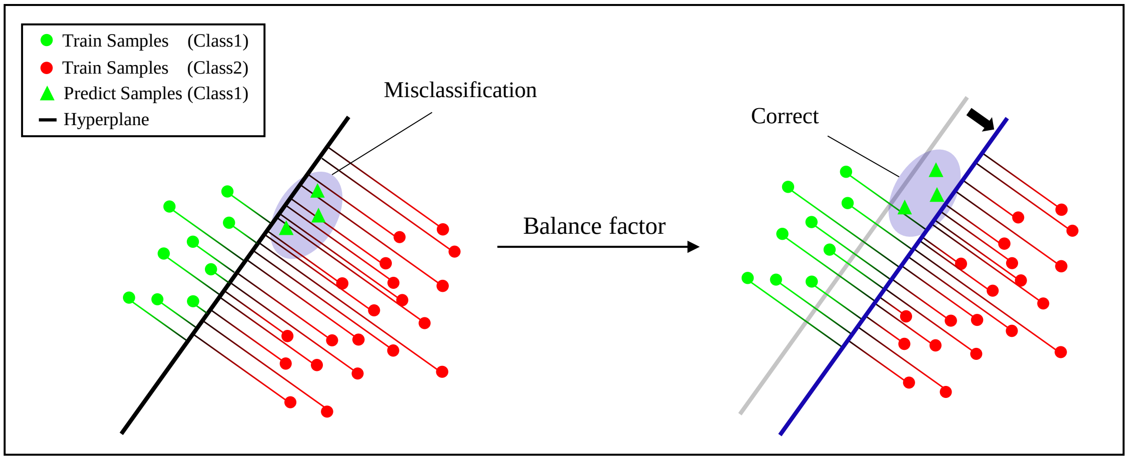


Figure 2: Demonstration of the effect of balance factors. It can be clearly seen that the classification hyperplane will no longer tilt towards minority classes, and the number of misclassified classes in minority classes will be greatly reduced.

After replacing the original distance with the composite distance, the margin mean is transformed into:

$$\gamma_1{}^* = \sum_{i=1}^{m} \theta_i y_i \omega^T \phi(x_i) = \sum_{i=1}^{m} (1 \pm \Delta) y_i \omega^T \phi(x_i) = \gamma \pm \sum_{i=1}^{m} \Delta y_i \omega^T \phi(x_i) \tag{14}$$

According to Eq. (1), the matrix form of $\gamma_1{}^*$ can be written as:

$$\gamma_1{}^* = \frac{1}{m} (X\theta Y)^T \omega \tag{15}$$

where $\theta = diag[\theta_1, \theta_2, ..., \theta_m]$. The effect of the balance factor can be represented by Fig. 2.

By constructing the composite distance through the balance factor function, the distance of the small class samples will become larger, while the distance of the large class samples will become smaller subsequently. In this way, the classification hyperplane can be in a balanced state during the training process, and will not be affected by the imbalanced data on both sides.

3.2. Model construction

As we mentioned earlier, LDM considers the margin distribution of all samples and has better generalization ability, i.e., it also inherits the hinge loss of SVM. LDM with hinge loss only requires a certain penalty to points located between positive and negative hyperplanes, and the penalty parameters of other points are 0, which causes LDM to be sensitive to noise and unstable resampling. Therefore, we replaced traditional hinge loss with UP loss, providing a more subtle penalty mechanism and improving robustness.

To make improvements to the low generalization performance of the model in handling imbalanced data, we combine the idea of dual balance with UP loss and propose a novel dual balanced large margin classifier (called DBUPLDM), which is formulated as

$$\min_{w,b} \frac{1}{2}\|w\|^2 - \lambda_1 {\gamma_1}^* + \lambda_2\gamma_2 + \sum_{i=1}^{l} \theta_i P_\tau \left(1 - y_i \left(w^T x_i + b\right)\right), \tag{16}$$

where P_τ is a cost-sensitive UP loss function. As a cost-sensitive quantile distance, $P_\tau(u_i)$ is expressed as

$$P_\tau(u_i) = {C_i}^* L_\tau(u_i) = \max\{u_i {C_i}^*, -\tau u_i {C_i}^*\}, \tag{17}$$

where $\tau \in [-1, 1]$ is the parameter that determines the quantile level. Introducing the slack variable $\xi_i = L_\tau(1 - y_i(w^T x_i + b)$, the equivalent form of Eq. (16) can be rewritten as

$$\begin{aligned} &\min_{w,b,\xi} \frac{1}{2}\|w\|^2 - \lambda_1 {\gamma_1}^* + \lambda_2\gamma_2 + \sum_{i=1}^{l} \rho_i\theta_i\xi_i \\ &\text{s.t.} \quad \xi_i \geq 1 - y_i \left(w^T x_i + b\right), \\ &\qquad\quad \xi_i \geq -\tau \left(1 - y_i \left(w^T x_i + b\right)\right), i = 1, \ldots, l. \end{aligned} \tag{18}$$

By utilizing a kernel function, the input data is mapped to the Hilbert space $\mathbb{H}$ through a typical nonlinear mapping. Assuming that the data point x_i is mapped from $\mathbb{R}^n$ to $\mathbb{R}^e (e > n)$, the relationship can be expressed as $\phi : X \to \mathbb{H}$. Letting $D = y_i \left(w^T \phi(x_i) + b\right)$, we can get the objective function for the nonlinear case of the proposed DBUPLDM model

$$\begin{aligned} &\min_{w,b,\xi} \frac{1}{2}\|w\|^2 - \lambda_1 {\gamma_1}^* + \lambda_2\gamma_2 + \sum_{i=1}^{l} \rho_i\theta_i\xi_i \\ &\text{s.t. } \xi_i \geq 1 - D_i, \\ &\qquad \xi_i \geq -\tau \left(1 - D_i\right), i = 1, \ldots, l, \end{aligned} \tag{19}$$

where $\gamma_1 = \frac{1}{l}\sum_{i=1}^{l} D_i$ and $\gamma_2 = \sum_{i=1}^{l}\sum_{j=1}^{l} \left(D_i - D_j\right)^2$.

Remark 1. *The performance of DBUPLDM will be affected by the value of τ and the balance factor. The proposed DBUPLDM model is a general version of LDM and SVM models.*

- *when $\tau = 0$, the proposed DBUPLDM is degenerated into a dual balanced LDM model with hinge loss.*
- *When $\theta_i = 1$ (i=1, 2, ..., l) and $\tau \geq 0$, the proposed DBUPLDM is degenerated into a LDM model with pinball loss, which can be called Pin-LDM.*
- *When $\lambda_1 = \lambda_2 = 0$ and $\theta_i = 1$, the proposed DBUPLDM is degenerated into UPSVM [34].*
- *When $\lambda_1 = \lambda_2 = 0$, and $\tau \geq 0$, the proposed model is degenerated into Pin-SVM [19, 17].*

3.3. QPP for solving DBUPLDM problem

Introducing Lagrange multipliers α_i and β_i, the Lagrangian function associated with Eq. (19) is

$$\begin{aligned}
&\mathrm{L}(w,b,\xi;\alpha,\beta)=\frac{1}{2}\|w\|^2-\frac{\lambda_1}{l}\sum_{i=1}^{l}D_i\\
&+\lambda_2\sum_{i=1}^{l}\sum_{j=1}^{l}\left(D_i-D_j\right)^2+\sum_{i=1}^{l}\rho_i\theta_i\xi_i\\
&-\sum_{i=1}^{l}\alpha_i\left(D_i+\xi_i-1\right)-\sum_{i=1}^{l}\beta_i\left(\tau\left(1-D_i\right)+\xi_i\right).
\end{aligned}\tag{20}$$

Since it is very difficult to directly handle the Lagrangian function in Eq. (20), we below convert it to matrix form.

Theorem 1. *The optimal solution w^* for Eq. (19) admits a representation of the form:*

$$\boldsymbol{w}^*=\sum_{i=1}^{l}\delta_i\phi\left(\boldsymbol{x}_i\right)=\Phi\boldsymbol{\delta},\tag{21}$$

where $\boldsymbol{\delta}=\left[\delta_1,\ldots,\delta_l\right]^T$ is a vector by stacking the specific coefficients and $\Phi=\left[\phi_1,\ldots,\phi_l\right]^T$.

Proof. The proof of this theorem refers to [35]. □

By applying Theorem 1, one can establish the following equalities

$$\begin{aligned}
&\sum_{i=1}^{l}D_i\theta_{\mathrm{i}}=l\gamma_1^*,\\
&\gamma_1^*=\frac{1}{l}\sum_{i=1}^{l}D_i\theta_{\mathrm{i}}=\frac{1}{l}(\theta Y)^TK\delta,\\
&\gamma_2=\sum_{i=1}^{l}\sum_{j=1}^{l}\left(D_i-D_j\right)^2\\
&\quad=\frac{2\delta^TKK^T\delta}{l}-\frac{2\delta^TKYY^TK\delta}{l^2},
\end{aligned}\tag{22}$$

where the kernel function $K=\Phi^T\Phi$, and Y is the label vector of the sample. The Gaussian kernel function can be expressed as

$$K\left(x_i,x_j\right)=\exp\left(\frac{-\left\|x_i-x_j\right\|_2}{2\sigma^2}\right),\tag{23}$$

where σ is known as the width parameter, see [36]. For the linear kernel, $K=x_i^Tx_j$.

Substituting Eq. (21) and Eq. (22) into Eq. (19) gives

$$\begin{aligned}
&\min_{\delta,\xi} \frac{1}{2}\delta^T Q\delta + E^T\delta + \sum_{i=1}^{l}\rho_i\theta_i\xi_i \\
&\text{s.t. } \ \xi_i \geq 1 - D_i, \\
&\qquad \ \xi_i \geq -\tau\left(1 - D_i\right), i = 1, \ldots, l,
\end{aligned} \tag{24}$$

where

$$\begin{aligned}
&Q = 4\lambda_2\left(lK^TK - (KY)(KY)^T + 4\lambda_2 l^2 K\right)/l^2, \\
&E = -\lambda_1 K\theta Y/l.
\end{aligned} \tag{25}$$

Theorem 2. *The dual problem of the DBUPLDM optimization problem (Eq. (24)) admits a representation of the form:*

$$\begin{aligned}
&\min_{\boldsymbol{\alpha},\hat{\boldsymbol{\beta}}} \frac{1}{2}(\boldsymbol{\alpha} - v\hat{\boldsymbol{\beta}})^T A(\boldsymbol{\alpha} - v\hat{\boldsymbol{\beta}}) + \left(\frac{\lambda_1 A\theta e - le}{l}\right)^T(\boldsymbol{\alpha} - v\hat{\boldsymbol{\beta}}) \\
&s.t. \qquad \alpha_i + \frac{\hat{\beta}_i}{|\tau|} = \rho_i, \\
&\qquad\qquad \alpha_i \geq 0, \hat{\beta}_i \geq 0, i = 1, \ldots, l,
\end{aligned} \tag{26}$$

where $\boldsymbol{\alpha} = [\alpha_1, \alpha_2, ..., \alpha_l]^T$ *and* $\hat{\boldsymbol{\beta}} = [\hat{\beta}_1, \hat{\beta}_2, ..., \hat{\beta}_l]^T$*,* v *is an indicator function.*

Proof. Combine Eq. (22), the Lagrangian function of Eq. (24) is

$$\begin{aligned}
&L(\delta, \xi; \alpha, \beta) = \frac{1}{2}\delta^T Q\delta + E^T\delta + \sum_{i=1}^{l}\rho_i\theta_i\xi_i \\
&- \sum_{i=1}^{l}\alpha_i(\xi_i + y_i\delta^T K_i - 1) - \sum_{i=1}^{l}\beta_i(\xi_i + \tau(1 - y_i\delta^T K_i)).
\end{aligned} \tag{27}$$

According to the Karush–Kuhn–Tucker (KKT) optimization condition, the following equations can be obtained

$$\begin{aligned}
&\frac{\partial L}{\partial \delta} = Q\delta + E - \sum_{i=1}^{l}\alpha_i y_i K_i + \tau\sum_{i=1}^{l}\beta_i y_i K_i = 0, \\
&\frac{\partial L}{\partial \xi_i} = \rho_i\theta_i - \alpha_i - \beta_i = 0, \forall i = 1, 2, \ldots, l.
\end{aligned} \tag{28}$$

Therefore, the dual problem of the original Eq. (24) is

$$\begin{aligned}
&\min_{\boldsymbol{\alpha},\boldsymbol{\beta}} \frac{1}{2}(\boldsymbol{\alpha} - \tau\hat{\boldsymbol{\beta}})^T A(\boldsymbol{\alpha} - \tau\hat{\boldsymbol{\beta}}) + \left(\frac{\lambda_1 A\theta e - le}{l}\right)^T(\boldsymbol{\alpha} - \tau\hat{\boldsymbol{\beta}}) \\
&\text{s.t.} \qquad \alpha_i + \beta_i = \rho_i, \\
&\qquad\qquad \alpha_i \geq 0, \beta_i \geq 0, i = 1, \ldots, l,
\end{aligned} \tag{29}$$

where

$$
\begin{aligned}
&A = [\mathrm{diag}(Y)]KQ^{-1}K[\mathrm{diag}(Y)], \\
&e = [1, \ldots, 1]_{1\times m}.
\end{aligned} \tag{30}
$$

It can be seen from the Eq. (29) that when τ is negative, the final optimization problem of the model is different from that when τ is positive. This proves that Huang et al.'s method of extending τ to negative values [17] is wrong.

We define the signum function v_u by

$$
v_u = \begin{cases} 1 & \text{if } u \geq 0, \\ -1 & \text{otherwise,} \end{cases} \tag{31}
$$

and let $\hat{\beta} = |\tau|\beta$, then the dual Eq. (29) can be given by

$$
\begin{aligned}
\min_{\boldsymbol{\alpha},\hat{\boldsymbol{\beta}}} \ & \frac{1}{2}(\boldsymbol{\alpha} - v_\tau\hat{\boldsymbol{\beta}})^T A(\boldsymbol{\alpha} - v_\tau\hat{\boldsymbol{\beta}}) + \left(\frac{\lambda_1 A\theta e - le}{l}\right)^T (\boldsymbol{\alpha} - v_\tau\hat{\boldsymbol{\beta}}_{\boldsymbol{j}}) \\
\text{s.t.} \quad & \alpha_i + \frac{\hat{\beta}_i}{|\tau|} = \rho_i, \\
& \alpha_i \geq 0, \hat{\beta}_i \geq 0, i = 1, \ldots, l.
\end{aligned} \tag{32}
$$

□

Again, we observe the equivalence between the traditional LDM and the DBUPLDM with $\tau = 0$: when τ is small enough, $\frac{\hat{\beta}}{|\tau|}$ can provide any positive value, thus the corresponding constraint is satisfied if and only if $-v\,|\tau|\,\rho_i\theta_i \leq \alpha_i \leq \rho_i\theta_i$. Hence, Eq. (32) reduces to the dual formulation of the hinge loss LDM

$$
\begin{aligned}
&\min_{\boldsymbol{\alpha}} \frac{1}{2}\boldsymbol{\alpha}^T A\boldsymbol{\alpha} + \left(\frac{\lambda_1 A\theta e - le}{l}\right)^T \boldsymbol{\alpha} \\
&\text{s.t.} \qquad -v\,|\tau|\,\rho_i\theta_i \leq \alpha_i \leq \rho_i\theta_i, i = 1, \ldots, l.
\end{aligned} \tag{33}
$$

After obtaining the solution (α^*, β^*) of the dual problem (Eq. (32)), δ can be given by

$$
\begin{aligned}
\delta &= Q^{-1}\left\{K[diag(Y)](\alpha^* - v_\tau\hat{\beta}^*) - E\right\} \\
&= Q^{-1}\left\{\frac{\lambda_1}{l}K[diag(Y)]e + K[diag(Y)](\alpha^* - v_\tau\hat{\beta}^*)\right\} \\
&= Q^{-1}K[diag(Y)](\frac{\lambda_1\theta e}{m} + \alpha^* - v_\tau\hat{\beta}^*).
\end{aligned} \tag{34}
$$

For a given test point $x_z \in \mathbb{R}^n$, the decision function is obtained as

$$f(x_z) = \mathrm{sgn}(w^T\phi(x_z)) = \mathrm{sgn}(\sum_{i=1}^{l} \delta_i K(x_i, x_z)). \tag{35}$$

Once the decision function of DBUPLDM is obtained, the categorization task can be performed by Eq. (35) for a new test sample.

4. Experiments

Table 1: A collection of dataset attributes

Category	Dataset	#Feature	#Majority	#Minority	#imbalance ratio(Maj/Min)
Meteorological	Australian	14	383	307	1.25
	Ionosphere	34	225	126	1.79
	Oil_Spill	49	896	41	21.85
	Sonar	61	111	97	1.14
Financial	German	25	700	300	2.33
	Bank	14	1018	205	4.97
	Plrx	12	130	52	2.50
Medical	Herberman	3	225	81	2.78
	Pima_Indian	8	500	268	1.87
	WDBC	29	357	212	1.68
	Thyroid_Diff	5	275	108	2.55
	Diabetes	8	500	268	1.87
	Pharyngitis	18	603	73	8.26
	Blood_Transfusion_Service	4	570	178	3.20
	Fertility	9	88	12	7.33
Engineering	Echo	7	88	43	2.05
	Phoneme	5	767	109	7.04
	Statlog	13	150	120	1.25
	Ecoil	8	184	143	1.29
Others	Monk1	6	278	278	1.00
	Monk2	6	395	204	1.94
	Monk3	6	288	266	1.08
	Website_Phishing	9	1250	103	12.14
	Votes	16	269	166	1.62

In this section, we present experimental results to evaluate the performance of the proposed DBUPLDM models. First, in Section 4.2 we conduct synthetic experiments to verify the reliability and validity of our model. Next, in Section 4.3 we evaluate the performance of our models on 17 binary datasets from the University of California-Irvine (UCI) machine learning data repository [37], which allows us to effectively and reasonably assess the models' performance. Additionally, we add Gaussian white noise to the UCI datasets in Section 4.4 to assess the noise immunity of DBUPLDM. Moreover, to further validate the superiority of our models, we compare different metrics on some additional UCI datasets with higher imbalance ratios in Section 4.5. Finally, we perform a parametric sensitivity analysis on the models in Section 4.6. The imbalance ratio of each UCI dataset used in our experiments is summarized in Table 1.

Table 2: Parameter settings for synthetic datasets with Gaussian noise.

Type	Parameter	Setting
Normal Samples	Mean of Class 1	[0.5 , -3.5]
	Mean of Class 2	[-0.5 , 3.5]
	Covariance of the Categories	[0.2 , 0 ; 0 , 0.3]
	Number of Majority Class	**100,150,200,250**
	Number of Minority Class	**50**
Noise	Mean of Class 1 and Class 2	[0.0 , 0.0]
	Covariance of the Categories	[1 , -0.8 ; -0.8 , 1]
	Number of Class1 or Class 2	5,10,15,20,25,30

4.1. Parameter settings

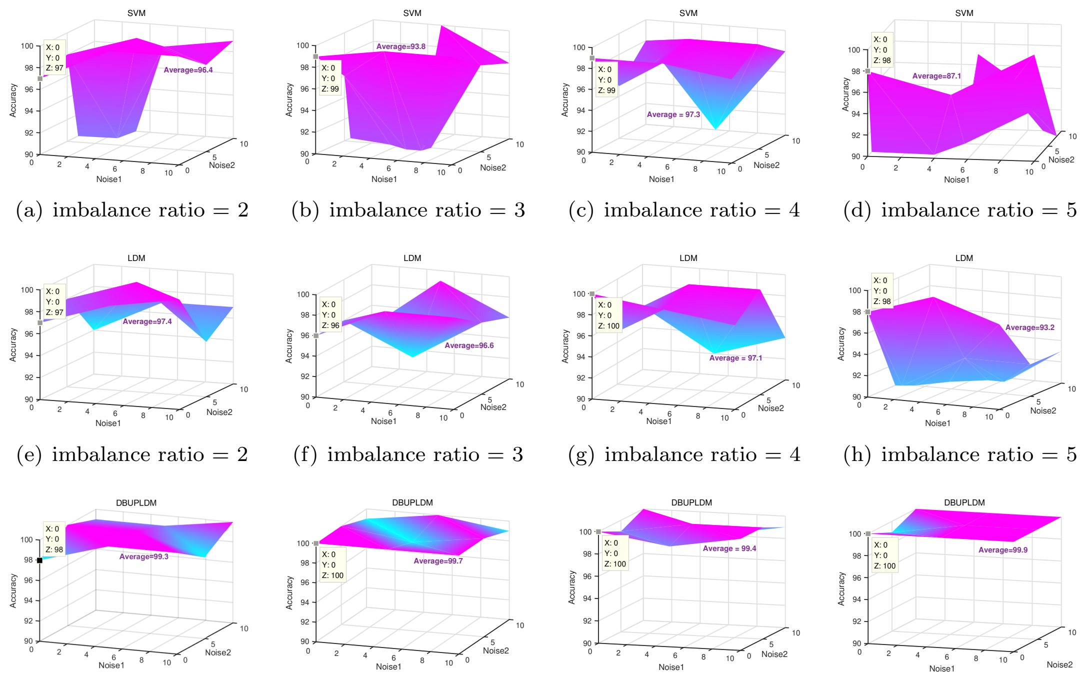
Data points in the datasets are normalized to $[0, 1]$ to avoid the dominance of input features with greater numerical values over other smaller values. The traversal range of parameters in the experiment is $\eta \in [\{2^{-6}, 2^{-5}, 2^{-4}, 2^{-3}, 2^{-2}, 2^{-1}, 2^{0}, 2^{1},\ 2^{2}, 2^{3}\}]$, σ (parameter of Gaussian kernel) $\in \{2^{-7}, 2^{-6}, ..., 2^{7}\}$, τ ∈{-1,-0.99,-0.98,...,0,0.01,0.02,...,1}, $\lambda_1 \in \{2^{-7}, 2^{-6}, ..., 2^{1}\}$, λ_2 traverses the same range as λ_1. The accuracy of each algorithm depends on the optimal combination of these parameters. All experiments in this paper are performed on a computer with 8 $\times$ 4.00 GHz CPU and 32 GB of memory using MATLAB R2018a.

4.2. Experiments on artificial datasets

To verify the performance of DBUPLDM for the imbalance classification task, we conducted experiments using datasets with varying levels of imbalance and added different types of noise. Each sample or noise point x_i follows a normal distribution, such that: $x_i \sim \mathcal{N}(\mu, \Sigma)$, where μ and Σ represent the mean and covariance of the two dimensions of the sample, respectively. The details of our experimental setup are summarized in Table 2.

The imbalance ratio changes as the number of majority class samples is varied. To verify the anti-noise performance of DBUPLDM, the amount of noise in each category is controlled from 0 to 10, varying the degree of imbalance, and the experimental results are shown in Fig. 3.

Fig. 3 compares the noise immunity performance of DBUPLDM with that of LDM and SVM on an imbalanced classification task. The three algorithms are evaluated using varying levels of positive class and negative class noise, and an imbalance ratio of 2, 3, 4, or 5. As shown in Fig. 3, DBUPLDM consistently achieves the highest accuracy when no noise is added, regardless of the imbalance ratio (as indicated in the figure caption). When a certain amount of noise is introduced, the classification accuracy curves for SVM and LDM fluctuate significantly, indicating their high sensitivity to noise. Compared to SVM, LDM has better noise immunity due to its consideration of marginal theory. However, the accuracy surface of DBUPLDM remains smoother while still



(a) imbalance ratio = 2 (b) imbalance ratio = 3 (c) imbalance ratio = 4 (d) imbalance ratio = 5

(e) imbalance ratio = 2 (f) imbalance ratio = 3 (g) imbalance ratio = 4 (h) imbalance ratio = 5

(i) imbalance ratio = 2 (j) imbalance ratio = 3 (k) imbalance ratio = 4 (l) imbalance ratio = 5

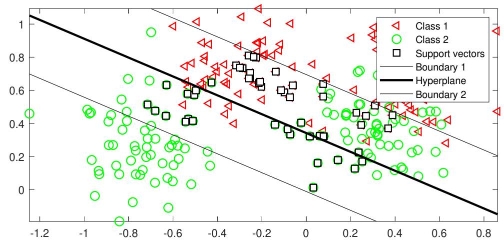
Figure 3: Noise immunity performance of SVM, LDM, and DBUPLDM under the different imbalance ratios.

achieving higher accuracy. Furthermore, as the imbalance ratio increases, the accuracy surfaces of SVM and LDM become more volatile due to the statistical characteristics of the data. On the other hand, DBUPLDM consistently maintains the highest average accuracy and stability despite varying degrees of imbalance. These experimental results effectively demonstrate the superior noise immunity and generalization performance of DBUPLDM.

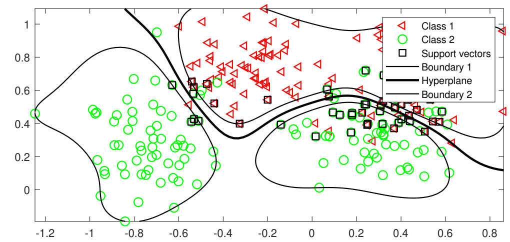
To provide a more intuitive understanding of how DBUPLDM handles boundary points with varying τ values, we conduct experiments on the artificially synthesized two-dimensional dataset Ripley. Fig. 4 presents the boundary information obtained from these experiments.

As illustrated by Fig. 4, in the case of a linear kernel, when $\tau = -0.3$, points located outside of the positive and negative hyperplanes receive a negative penalty, resulting in a decrease in the margin between the hyperplanes. Similarly, for a Gaussian kernel, when $\tau = 0.3$, the margin between hyperplanes increases. The different τ values correspond to different quantile levels, demonstrating the flexibility of DBUPLDM.

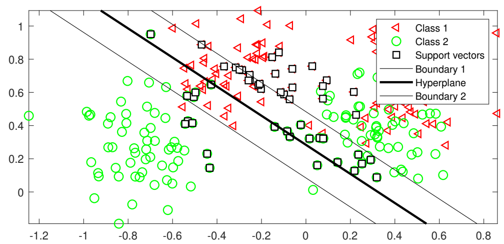
When $\tau = 0$, DBUPLDM degenerates into DBLDM with hinge loss, as shown in (a) and (b) of Fig. 4. By comparing the accuracy rates, it is clear that DBUPLDM achieves higher accuracy with different τ values, whether using a linear kernel or a Gaussian kernel. These experiments demonstrate that DBUPLDM has better flexibility and is less affected by boundary noise compared



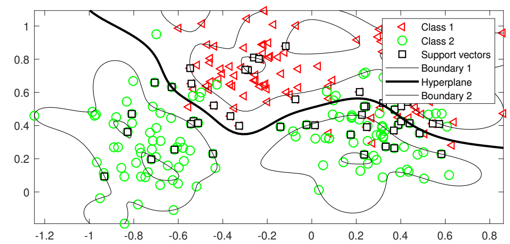
(a) $\tau = 0$ with linear kernel (Accuracy=71.9%)



(b) $\tau = 0$ with RBF kernel (Accuracy=90.7%)



(c) $\tau = -0.3$ with linear kernel (Accuracy=72.0%)



(d) $\tau = 0.3$ with RBF kernel (Accuracy=90.9%)

Figure 4: Comparison of classification accuracy corresponding to different τ values and kernel functions.

to traditional LDM.

4.3. Experiments on UCI datasets

In this subsection, we compare DBUPLDM with four other benchmark algorithms. We select 17 UCI datasets with varying features and imbalance ratios and add Gaussian white noise to these datasets to investigate the anti-noise performance of DBUPLDM. For "Monk1", "Monk2", "Monk3", and "Spect" data, UCI provides training and test sets. For other datasets, we divide each of them into the training set and test set in a certain proportion.

As shown in Table 1, there is data imbalance in almost all training sets. To fully validate the performance of our model on imbalanced datasets, we adjust this dataset imbalance by setting different parameters η: setting η to 1, 1/2, 1/4, 1/8, 1/16, 1/32, and 1/64, which means the ratio of the minority class to the majority class. The SVM, PinSVM, UPSVM, and LDM algorithms are tested ten times each for all datasets, and the average accuracy is selected as the final comparison criterion. The experimental results are presented in Tables 3 and 4.

Table 3 and 4 demonstrate that DBUPLDM has the best overall performance on all 17 datasets. DBUPLDM consistently maintains the highest average accuracy across all datasets regardless of the adjustment parameter η. Specifically, the accuracy of DBUPLDM tends to increase and then decrease as η decreases, with the best performance found at $\eta = 0.125$. Compared to SVM and some improved versions of SVM (i.e., PinSVM and UPSVM), LDM's accuracy is higher because it takes into account the marginal distribution theory and adds a second-order statistic to the model.

Table 3: Comparison of the accuracy of the SVM, PinSVM, UPSVM, LDM, and DBUPLDM on the UCI datasets

	Dataset	SVM		LDM		PinSVM			UPSVM			**DBUPLDM**		
		Accuracy	C	Accuracy	C	Accuracy	C	τ	Accuracy	C	τ	Accuracy	C	τ
$\eta = 1$	Monk1	64.35	0.06	67.59	0.06	65.28	0.03	0.20	66.44	4.00	-0.40	**68.29**	1.00	0.70
	Monk2	67.13	0.02	67.13/	0.02	67.13	0.02	-0.90	67.13	0.02	-0.40	**67.82**	0.50	0.50
	Monk3	81.02	0.06	84.03	0.13	84.26	0.02	1.00	86.34	0.25	-0.40	**88.89**	0.50	-0.30
	Pima-Indian	76.42	0.02	77.36	0.02	77.36	1.00	-1.00	76.42	0.02	-0.30	78.30	2.00	0.50
	WDBC	86.67	0.06	85.00	0.06	86.67	0.06	0.00	86.67	0.06	0.00	**87.50**	0.06	0.20
	Echo	**94.04**	2.00	93.38	8.00	**94.04**	2.00	0.00	**94.04**	2.00	0.00	92.05	0.02	-0.90
	Ionosphere	67.09	0.02	**79.91**	2.00	71.15	0.03	-1.00	71.15	8.00	-0.70	79.06	8.00	-0.30
	Heberman	79.29	0.02	**98.82**	0.06	95.86	0.02	0.10	95.86	0.02	0.10	**98.82**	0.50	-0.20
	Statlog	70.59	0.02	88.24	0.03	70.59	0.02	0.00	90.20	0.25	-0.20	**92.16**	0.25	-0.50
	Germans	32.80	0.02	68.00	0.02	67.20	0.25	-1.00	67.20	8.00	-0.80	**76.20**	2.00	-0.10
	Australian	84.48	0.03	86.55	0.03	84.48	0.03	0.00	**87.24**	0.03	-0.30	**87.24**	0.03	-0.90
	Votes	85.11	0.02	95.74	1.00	94.04	0.02	-0.90	91.49	0.02	-0.10	95.32	0.50	0.00
	Diabetes	67.91	0.02	81.72	1.00	67.91	0.02	0.00	72.76	8.00	-0.80	81.72	2.00	-0.50
	Fertility	**94.00**	0.02	**94.00**	0.02	**94.00**	0.02	0.00	**94.00**	0.02	0.00	**94.00**	0.02	-1.00
	Sonar	69.44	0.13	74.07	0.06	72.22	0.02	-1.00	74.07	0.03	-0.30	**76.85**	0.50	0.40
	Ecoil	86.61	0.25	**94.49**	0.13	**94.49**	0.06	0.90	**94.49**	0.25	-0.10	**94.49**	0.02	-1.00
	Plrx	67.07	0.02	67.07	0.02	67.07	0.02	0.00	67.07	0.02	-1.00	**70.73**	2.00	0.40
	Ave.	74.94		83.50		79.63			81.33			**84.08**		
$\eta = 0.5$	Monk1	62.73	0.13	67.59	0.06	65.28	0.03	0.20	66.44	4.00	-0.40	**68.29**	1.00	0.70
	Monk2	67.13	0.02	67.13	0.02	67.13	0.02	-0.90	67.13	0.02	-0.40	**67.59**	0.50	0.90
	Monk3	81.02	0.06	84.03	0.13	84.26	0.02	1.00	86.34	0.25	-0.40	**88.89**	0.50	-0.30
	Pima-Indian	76.42	0.02	77.36	0.02	78.30	0.25	-1.00	76.42	0.02	-0.30	**79.25**	4.00	0.30
	WDBC	86.67	0.06	85.00	0.06	86.67	0.06	0.00	86.67	0.06	0.00	**87.50**	0.06	0.20
	Echo	**94.04**	2.00	93.38	8.00	**94.04**	2.00	0.00	**94.04**	2.00	0.00	92.72	2.00	1.00
	Ionosphere	67.09	0.02	**79.91**	2.00	67.95	0.06	-1.00	71.15	8.00	-0.70	78.42	8.00	0.20
	Heberman	79.29	0.02	**98.82**	0.06	95.86	0.02	0.10	95.86	0.02	0.10	**98.82**	0.50	-0.20
	Statlog	70.59	0.02	88.24	0.03	76.47	4.00	-1.00	90.20	0.25	-0.20	**92.16**	0.13	-0.50
	Germans	32.80	0.02	68.00	0.02	67.20	0.06	-1.00	67.20	2.00	-0.80	**77.40**	8.00	-0.60
	Australian	84.48	0.03	86.55	0.03	84.48	0.03	0.00	**87.24**	0.03	-0.30	**87.24**	0.03	-0.90
	Votes	85.11	0.02	**95.74**	1.00	94.04	0.02	-0.90	91.49	0.02	-0.10	94.89	0.25	-0.50
	Diabetes	67.91	0.02	81.72	1.00	72.76	0.25	-1.00	72.76	8.00	-0.80	81.72	0.25	-0.60
	Fertility	**94.00**	0.02	**94.00**	0.02	**94.00**	0.02	0.00	**94.00**	0.02	0.00	**94.00**	0.02	-1.00
	Sonar	69.44	0.13	74.07	0.06	71.30	0.06	0.10	74.07	0.03	-0.30	**76.85**	0.50	0.40
	Ecoil	86.61	0.25	**94.49**	0.13	**94.49**	0.06	0.90	**94.49**	0.25	-0.10	**94.49**	0.02	-1.00
	Plrx	67.07	0.02	67.07	0.02	**68.29**	0.02	-1.00	67.07	0.02	-1.00	**68.29**	8.00	-0.20
	Ave.	74.85		82.54		80.15			81.33			**84.03**		
$\eta = 0.25$	Monk1	62.73	0.13	67.59	0.06	65.28	0.03	0.20	66.44	4.00	-0.40	**68.29**	1.00	0.70
	Monk2	67.13	0.02	67.13	0.02	67.13	0.02	-0.90	67.13	0.02	-0.40	**68.29**	1.00	0.10
	Monk3	81.02	0.06	84.03	0.13	84.26	0.02	1.00	86.34	0.25	-0.40	**88.89**	0.50	-0.30
	Pima-Indian	76.42	0.02	77.36	0.02	78.30	0.25	-1.00	76.42	0.02	-0.30	**79.25**	8.00	0.10
	WDBC	86.67	0.06	85.00	0.06	86.67	0.06	0.00	86.67	0.06	0.00	**87.50**	0.06	0.20
	Echo	**94.04**	2.00	93.38	8.00	**94.04**	2.00	0.00	**94.04**	2.00	0.00	92.72	8.00	-0.30
	Ionosphere	67.09	0.02	**79.91**	2.00	67.95	0.06	-1.00	71.15	8.00	-0.70	79.06	2.00	0.40
	Heberman	79.29	0.02	**98.82**	0.06	95.86	0.02	0.10	95.86	0.02	0.10	**98.82**	0.50	-0.20
	Statlog	70.59	0.02	88.24	0.03	76.47	4.00	-1.00	90.20	0.25	-0.20	**92.16**	0.25	-0.50
	Germans	32.80	0.02	68.00	0.02	67.20	0.06	-1.00	67.20	2.00	-0.80	**76.80**	4.00	0.40
	Australian	84.48	0.03	86.55	0.03	84.48	0.03	0.00	**87.24**	0.03	-0.30	**87.24**	0.03	-0.90
	Votes	85.11	0.02	95.74	1.00	94.04	0.02	-0.90	91.49	0.02	-0.10	**96.17**	1.00	0.70
	Diabetes	67.91	0.02	81.72	1.00	72.76	0.25	-1.00	72.76	8.00	-0.80	82.09	0.25	-0.60
	Fertility	**94.00**	0.02	**94.00**	0.02	**94.00**	0.02	0.00	**94.00**	0.02	0.00	**94.00**	0.02	-1.00
	Sonar	69.44	0.13	74.07	0.06	71.30	0.06	0.10	74.07	0.03	-0.30	**76.85**	0.50	0.40
	Ecoil	86.61	0.25	**94.49**	0.13	**94.49**	0.06	0.90	**94.49**	0.25	-0.10	**94.49**	0.02	-1.00
	Plrx	67.07	0.02	67.07	0.02	68.29	0.02	-1.00	67.07	0.02	-1.00	**69.51**	8.00	-0.10
	Ave.	74.85		82.54		80.15			81.33			**84.24**		

(1) The best accuracy among all methods is bolded (similarly hereinafter);
(2) **Ave.** represents the average value of the metrics (similarly hereinafter).
(3) In SVM and LDM, $\tau = 0$, as they both utilize hinge loss function (similarly hereinafter).

Table 4: Comparison of the accuracy of the SVM, PinSVM, UPSVM, LDM, and DBUPLDM on the UCI datasets

	Dataset	SVM		LDM		PinSVM			UPSVM			**DBUPLDM**		
		Accuracy	C	Accuracy	C	Accuracy	C	τ	Accuracy	C	τ	Accuracy	C	τ
	Monk1	62.73	0.13	67.59	0.06	65.28	0.03	0.20	66.44	4.00	-0.40	**68.29**	1.00	0.70
	Monk2	67.13	0.02	67.13	0.02	67.13	0.02	-0.90	67.13	0.02	-0.40	**67.82**	8.00	-0.20
	Monk3	81.02	0.06	84.03	0.13	84.26	0.02	1.00	86.34	0.25	-0.40	**88.89**	0.50	-0.30
	Pima-Indian	76.42	0.02	77.36	0.02	78.30	0.25	-1.00	76.42	0.02	-0.30	**79.25**	8.00	0.20
	WDBC	86.67	0.06	85.00	0.06	86.67	0.06	0.00	86.67	0.06	0.00	**87.50**	0.06	0.20
	Echo	**94.04**	2.00	93.38	8.00	**94.04**	2.00	0.00	**94.04**	2.00	0.00	92.72	4.00	0.70
	Ionosphere	67.09	0.02	**79.91**	2.00	67.95	0.06	-1.00	71.15	8.00	-0.70	78.85	2.00	0.40
	Heberman	79.29	0.02	**98.82**	0.06	95.86	0.02	0.10	95.86	0.02	0.10	**98.82**	0.50	-0.20
	Statlog	70.59	0.02	88.24	0.03	76.47	4.00	-1.00	90.20	0.25	-0.20	**92.16**	0.25	-0.50
$\eta = 0.125$	Germans	32.80	0.02	68.00	0.02	67.20	0.06	-1.00	67.20	2.00	-0.80	**76.40**	8.00	-0.50
	Australian	84.48	0.03	86.55	0.03	84.48	0.03	0.00	**87.24**	0.03	-0.30	**87.24**	0.03	-0.90
	Votes	85.11	0.02	**95.74**	1.00	94.04	0.02	-0.90	91.49	0.02	-0.10	95.32	8.00	0.30
	Diabetes	67.91	0.02	81.72	1.00	72.76	0.25	-1.00	72.76	8.00	-0.80	**82.09**	0.25	-0.60
	Fertility	94.00	0.02	94.00	0.02	94.00	0.02	0.00	94.00	0.02	0.00	**96.00**	2.00	0.60
	Sonar	69.44	0.13	74.07	0.06	71.30	0.06	0.10	74.07	0.03	-0.30	**76.85**	0.50	0.40
	Ecoil	86.61	0.25	**94.49**	0.13	**94.49**	0.06	0.90	**94.49**	0.25	-0.10	**94.49**	0.02	-1.00
	Plrx	67.07	0.02	67.07	0.02	68.29	0.02	-1.00	67.07	0.02	-1.00	**69.51**	8.00	0.00
	Ave.	74.85		82.54		80.15			81.33			**84.25**		
	Monk1	62.73	0.13	67.59	0.06	65.28	0.03	0.20	66.44	4.00	-0.40	**68.29**	1.00	0.70
	Monk2	67.13	0.02	67.13	0.02	67.13	0.02	-0.90	67.13	0.02	-0.40	**67.59**	0.25	-0.60
	Monk3	81.02	0.06	84.03	0.13	84.26	0.02	1.00	86.34	0.25	-0.40	**88.89**	0.50	-0.30
	Pima-Indian	76.42	0.02	77.36	0.02	78.30	0.25	-1.00	76.42	0.02	-0.30	**79.25**	8.00	0.20
	WDBC	86.67	0.06	85.00	0.06	86.67	0.06	0.00	86.67	0.06	0.00	**87.50**	0.06	0.20
	Echo	**94.04**	2.00	93.38	8.00	**94.04**	2.00	0.00	**94.04**	2.00	0.00	92.72	4.00	0.70
	Ionosphere	67.09	0.02	**79.91**	2.00	67.95	0.06	-1.00	71.15	8.00	-0.70	78.21	2.00	-0.70
	Heberman	79.29	0.02	**98.82**	0.06	95.86	0.02	0.10	95.86	0.02	0.10	**98.82**	0.50	-0.20
	Statlog	70.59	0.02	88.24	0.03	76.47	4.00	-1.00	90.20	0.25	-0.20	**92.16**	0.25	-0.50
$\eta = 0.0625$	Germans	32.80	0.02	68.00	0.02	67.20	0.06	-1.00	67.20	2.00	-0.80	**76.20**	4.00	0.20
	Australian	84.48	0.03	86.55	0.03	84.48	0.03	0.00	**87.24**	0.03	-0.30	**87.24**	0.03	-0.90
	Votes	85.11	0.02	**95.74**	1.00	94.04	0.02	-0.90	91.49	0.02	-0.10	**95.74**	8.00	1.00
	Diabetes	67.91	0.02	81.72	1.00	72.76	0.25	-1.00	72.76	8.00	-0.80	**82.09**	0.25	-0.60
	Fertility	94.00	0.02	94.00	0.02	94.00	0.02	0.00	94.00	0.02	0.00	**96.00**	0.50	0.90
	Sonar	69.44	0.13	74.07	0.06	71.30	0.06	0.10	74.07	0.03	-0.30	**76.85**	0.50	0.40
	Ecoil	86.61	0.25	**94.49**	0.13	**94.49**	0.06	0.90	**94.49**	0.25	-0.10	**94.49**	0.02	-1.00
	Plrx	67.07	0.02	67.07	0.02	68.29	0.02	-1.00	67.07	0.02	-1.00	**69.51**	8.00	0.00
	Ave.	74.85		82.54		80.15			81.33			**84.21**		
	Monk1	62.73	0.13	67.59	0.06	65.28	0.03	0.20	66.44	4.00	-0.40	**68.29**	1.00	0.70
	Monk2	67.13	0.02	67.13	0.02	67.13	0.02	-0.90	67.13	0.02	-0.40	**67.59**	0.25	-0.60
	Monk3	81.02	0.06	84.03	0.13	84.26	0.02	1.00	86.34	0.25	-0.40	**88.89**	0.50	-0.30
	Pima-Indian	76.42	0.02	77.36	0.02	**78.30**	0.25	-1.00	76.42	0.02	-0.30	**78.30**	4.00	0.20
	WDBC	86.67	0.06	85.00	0.06	86.67	0.06	0.00	86.67	0.06	0.00	**87.50**	0.06	0.20
	Echo	**94.04**	2.00	93.38	8.00	**94.04**	2.00	0.00	**94.04**	2.00	0.00	92.05	0.06	-1.00
	Ionosphere	67.09	0.02	**79.91**	2.00	67.95	0.06	-1.00	71.15	8.00	-0.70	78.21	2.00	-0.70
	Heberman	79.29	0.02	**98.82**	0.06	95.86	0.02	0.10	95.86	0.02	0.10	**98.82**	0.50	-0.20
	Statlog	70.59	0.02	88.24	0.03	76.47	4.00	-1.00	90.20	0.25	-0.20	**92.16**	0.25	-0.50
$\eta = 0.03125$	Germans	32.80	0.02	68.00	0.02	67.20	0.06	-1.00	67.20	2.00	-0.80	**76.80**	8.00	-0.50
	Australian	84.48	0.03	86.55	0.03	84.48	0.03	0.00	**87.24**	0.03	-0.30	**87.24**	0.03	-0.90
	Votes	85.11	0.02	**95.74**	1.00	94.04	0.02	-0.90	91.49	0.02	-0.10	95.32	8.00	0.30
	Diabetes	67.91	0.02	81.72	1.00	72.76	0.25	-1.00	72.76	8.00	-0.80	**82.09**	0.25	-0.60
	Fertility	**94.00**	0.02	**94.00**	0.02	**94.00**	0.02	0.00	**94.00**	0.02	0.00	**94.00**	0.02	-1.00
	Sonar	69.44	0.13	74.07	0.06	71.30	0.06	0.10	74.07	0.03	-0.30	**76.85**	0.50	0.40
	Ecoil	86.61	0.25	**94.49**	0.13	**94.49**	0.06	0.90	**94.49**	0.25	-0.10	**94.49**	0.02	-1.00
	Plrx	67.07	0.02	67.07	0.02	68.29	0.02	-1.00	67.07	0.02	-1.00	**69.51**	8.00	0.00
	Ave.	74.85		82.54		80.15			81.33			**84.01**		
	Total Average	74.86		82.68		80.06			81.33			**84.14**		

However, due to its use of hinge loss function, LDM is not suitable for imbalanced classification tasks, resulting in lower accuracy than that of DBUPLDM. Interestingly, algorithms with pinball loss functions (PinSVM, UPSVM, and DBUPLDM) exhibit better performance when τ takes negative values.

To provide a more visual representation of the excellent performance of DBUPLDM, we have summarized the information from Table 3 and 4 in Table 5.

Table 5 presents the number of accuracy wins, draws, and losses of each algorithm concerning the 17 datasets. The results are denoted as "WIN", "DRAW", and "LOSS" to indicate each algorithm's performance when compared to others. DBUPLDM achieves the best performance in this analysis. Since 17 datasets were selected and each algorithm needs to be compared with the remaining 4 algorithms, the sum of "WIN", "DRAW", and "LOSS" for each algorithm is $17\times4=68$. Therefore, the total sum across the six parameters η is $68\times6=408$. The average win rate of DBUPLDM is 79% (324/408) with a loss rate of only 8% (35/408). Specifically for each of the η-parameters, the performance of DBUPLDM maintains a win rate of higher than 75% (51/68) and a loss rate of always less than 10%. These results demonstrate the consistently outstanding performance of our proposed DBUPLDM.

Table 5: The number of wins, draws and losses for SVM, LDM, PinSVM, UPSVM and DBUPLDM

	η	SVM	UPSVM	PinSVM	**DBUPLDM**	LDM
WIN(Aggregated 17 datasets)	1.00	3(68)	23(68)	14(68)	**52(68)**•	31(68)
	0.50	3(68)	23(68)	21(68)	**51(68)**•	31(68)
	0.25	3(68)	36(68)	21(68)	**54(68)**•	29(68)
	0.13	3(68)	22(68)	21(68)	**57(68)**•	31(68)
	0.06	3(68)	23(68)	20(68)	**57(68)**•	30(68)
	0.03	4(68)	24(68)	21(68)	**53(68)**•	31(68)
	ALL	19	151	118	**324**•	183
DRAW(Aggregated17 datasets)	1.00	18(68)	23(68)	24(68)	9(68)	14(68)
	0.50	15(68)	21(68)	19(68)	10(68)	13(68)
	0.25	15(68)	22(68)	18(68)	9(68)	11(68)
	0.13	14(68)	21(68)	17(68)	5(68)	13(68)
	0.06	14(68)	21(68)	17(68)	6(68)	14(68)
	0.03	15(68)	21(68)	19(68)	10(68)	14(68)
	ALL	91	129	114	49	79
LOSS(Aggregated17 datasets)	1.00	47(68)	22(68)	30(68)	**7(68)**◦	23(68)
	0.50	50(68)	24(68)	28(68)	**7(68)**◦	24(68)
	0.25	50(68)	10(68)	29(68)	**5(68)**◦	28(68)
	0.13	51(68)	25(68)	30(68)	**6(68)**◦	24(68)
	0.06	51(68)	24(68)	31(68)	**5(68)**◦	24(68)
	0.03	49(68)	23(68)	28(68)	**5(68)**◦	23(68)
	ALL	298	128	176	**35**◦	146

In order to verify whether the superior performance of DBUPLDM stems from our proposed dual balanced method and unified pinball loss function, an ablation experiment is conducted on the basis of the above experiments. The parameter ρ is set to 0.125, and the experimental results are presented in Table 6 and illustrated in Fig. 5.

Table 6: Minimum distance with second order statistics

Method	Hinge loss	Unified Pinball loss
No Balanced Factors	82.54 (LDM)	83.54 (UPLDM)
Balanced Factors	83.02 (DBLDM)	**84.25 (DBUPLDM)**

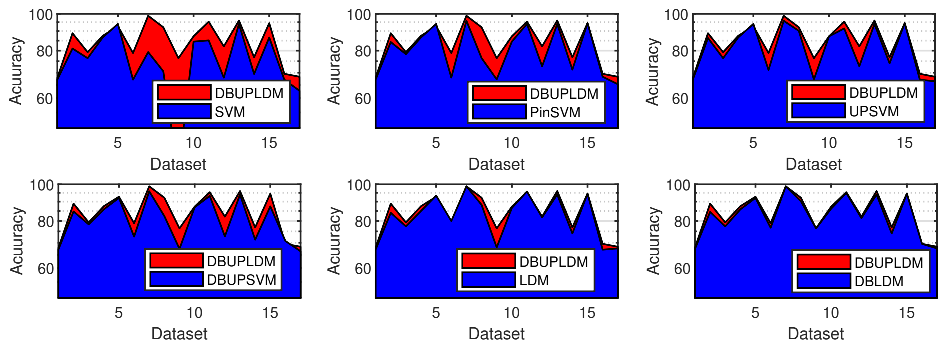


Figure 5: The performance comparison of SVM, LDM, PinSVM, UPSVM, and DBUPLDM in 17 dataset ($\rho = 0.125$).

The comparison metric for the ablation experiments is formulated as the average performance of the algorithms on the 17 datasets, measured by their average accuracy. As can be seen from Table 6 of the ablation experiment results, the base model without any treatment (LDM) has the worst performance, while adding the balanced factors (DBLDM) or modifying the loss function (UPLDM) leads to an improvement in classification accuracy by 0.48% and 1%, respectively. Moreover, when both the balanced factors and modified loss function are added, accuracy is improved by 1.7%.

Fig. 5 illustrates the accuracy improvement of DBUPLDM compared to other algorithms on the 17 datasets. SVM has the most significant improvement, followed by PinSVM, UPSVM, DBUPSVM, LDM, and DBLDM. These results demonstrate that the proposed balanced factors and unified pinball loss function play a crucial role in improving the performance of the proposed algorithm. Furthermore, the overall accuracy of LDM and DBUPLDM outperforms that of SVM and UPSVM, respectively. This observation highlights the effectiveness of considering the overall distribution of samples to improve algorithm accuracy.

In conclusion, LDM outperforms SVM, and the UP loss leads to better generalization performance than hinge loss. The proposed DBUPLDM has better classification performance than the other algorithms, due to the incorporation of dual balanced factors and the unified pinball loss function.

4.4. Experiments on UCI datasets with feature noise

To gain insights from the perspective of noise immunity, we add Gaussian white noise to the 17 datasets to properly distort the original data distribution. We set the mean value to 0 and the noise ratio to $\varepsilon = 0.05$ and 0.1, respectively.

Table 7: Comparison of SVM, PinSVM, UPSVM, LDM, and DBUPLDM on noise-added UCI datasets

	Dataset	SVM		LDM		PinSVM			UPSVM			**DBUPLDM**		
		Accuracy	C	Accuracy	C	Accuracy	C	tau	Accuracy	C	tau	Accuracy	C	tau
$\varepsilon = 0.05$	Monk1	57.87	0.25	61.11	4.00	60.65	0.03	1.00	**63.43**	4.00	-0.50	61.34	2.00	-0.10
	Monk2	**67.13**	0.02	**67.13**	0.02	**67.13**	0.13	-0.90	**67.13**	0.02	-0.40	**67.13**	0.02	-1.00
	Monk3	72.45	0.50	72.45	0.50	**73.38**	0.25	0.60	**73.38**	0.25	0.60	71.99	4.00	-0.60
	Pima-Indian	76.42	0.02	77.36	0.02	76.42	0.25	-0.90	76.42	0.02	-0.20	**80.19**	8.00	0.20
	WDBC	84.17	0.06	**85.00**	0.03	84.17	0.06	0.00	84.17	0.06	0.00	**85.00**	0.13	-0.70
	Echo	90.73	0.02	90.73	0.02	90.73	0.02	0.00	90.73	0.02	-1.00	**91.39**	8.00	0.10
	Ionosphere	67.09	0.02	**78.85**	4.00	67.74	0.25	-1.00	69.23	2.00	-0.70	77.99	4.00	0.40
	Heberman	91.72	0.02	**96.45**	8.00	92.90	0.13	-1.00	95.86	0.25	-0.90	**96.45**	0.50	-0.10
	Statlog	76.47	0.02	92.16	0.02	76.47	0.02	0.00	82.35	0.25	-0.20	**96.08**	1.00	0.00
	Germans	32.80	0.02	67.40	0.02	67.20	0.25	-0.90	59.20	4.00	-0.90	**73.60**	4.00	0.40
	Australian	55.86	0.02	74.48	1.00	69.66	8.00	-1.00	**76.21**	8.00	-0.80	75.52	0.25	0.30
	Votes	80.00	0.02	80.43	0.50	80.85	0.02	-1.00	**83.83**	0.02	-0.80	**83.83**	8.00	0.70
	Diabetes	67.91	0.02	80.22	2.00	72.39	0.03	-1.00	72.76	4.00	-0.80	**80.60**	0.03	-0.40
	Fertility	**94.00**	0.02	**94.00**	0.02	**94.00**	0.02	0.00	**94.00**	0.02	-1.00	**94.00**	0.02	-1.00
	Sonar	58.33	0.25	62.04	8.00	59.26	0.25	-0.40	62.04	0.50	-0.20	**64.81**	4.00	-0.30
	Ecoil	51.18	0.02	54.33	0.03	59.84	0.06	-1.00	**62.20**	0.02	-0.70	61.42	8.00	0.30
	Plrx	67.07	0.02	67.07	0.02	67.07	0.02	0.00	68.29	0.06	-1.00	**70.73**	8.00	0.30
	Ave.	70.07		76.54		74.11			75.37			**78.36**		
$\varepsilon = 0.1$	Monk1	57.64	0.25	61.11	4.00	60.42	0.02	0.50	**63.43**	4.00	-0.50	61.34	2.00	-0.10
	Monk2	**67.13**	0.02	**67.13**	0.02	**67.13**	0.13	-0.90	**67.13**	0.02	-0.50	**67.13**	0.02	-1.00
	Monk3	71.76	0.25	72.22	0.50	**73.38**	0.50	0.60	**73.38**	0.50	0.60	71.99	4.00	-0.60
	Pima-Indian	76.42	0.02	77.36	0.02	76.42	0.25	-0.90	76.42	0.02	-0.20	**80.19**	8.00	0.20
	WDBC	84.17	0.06	**85.00**	0.03	84.17	0.06	0.00	84.17	0.06	0.00	**85.00**	0.13	-0.70
	Echo	90.73	0.02	90.73	0.02	90.73	0.02	0.00	**91.39**	0.03	-1.00	**91.39**	8.00	0.10
	Ionosphere	67.09	0.02	**78.85**	8.00	69.87	4.00	-1.00	69.02	2.00	-0.70	77.78	4.00	0.40
	Heberman	91.72	0.02	**96.45**	8.00	95.27	0.03	-1.00	95.86	0.25	-0.90	**96.45**	0.50	-0.10
	Statlog	76.47	0.02	94.12	0.25	76.47	0.02	0.00	82.35	0.25	-0.20	**96.08**	1.00	0.00
	Germans	32.80	0.02	67.40	0.02	67.20	0.25	-0.90	59.20	2.00	-0.90	**73.60**	4.00	0.40
	Australian	55.86	0.02	74.48	1.00	65.86	0.03	-1.00	**75.86**	8.00	-0.80	75.52	0.25	0.30
	Votes	82.13	0.02	80.85	0.25	82.13	0.02	0.00	**84.26**	0.02	-0.80	83.83	8.00	0.70
	Diabetes	67.91	0.03	80.22	2.00	69.03	0.06	-1.00	72.76	4.00	-0.80	**80.60**	0.03	-0.40
	Fertility	**94.00**	0.02	**94.00**	0.02	**94.00**	0.02	0.00	**94.00**	0.02	-1.00	**94.00**	0.02	-1.00
	Sonar	59.26	0.25	62.04	8.00	59.26	0.02	-1.00	62.04	0.25	-1.00	**64.81**	2.00	-0.30
	Ecoil	51.18	0.02	54.33	0.03	**62.20**	8.00	-1.00	**62.20**	0.02	-0.70	61.42	8.00	0.30
	Plrx	67.07	0.02	67.07	0.02	67.07	0.02	0.00	68.29	0.06	-0.30	**70.73**	8.00	0.30
	Ave.	70.20		76.67		74.15			75.40			**78.34**		

Table 7 presents the experimental results. Compared to SVM, PinSVM, UPSVM, LDM, and DBUPLDM algorithms show better noise-handling capabilities on most datasets. Additionally, LDMs exhibit better classification accuracy than SVMs due to the second-order statistics introduced by the model. However, DBUPLDM consistently outperformed LDM across all 17 datasets, regardless of the noise ratio. By setting s=0.05, the average accuracy of DBUPLDM significantly improved compared to other algorithms across all 17 datasets. Specifically, DBUPLDM outperformed SVM by 8.9%, PinSVM by 4.25%, UPSVM by 2.09%, and LDM by 1.82%.

4.5. Experiments on UCI datasets with high imbalance ratio

To further validate the superiority of our model, we select multiple datasets with higher imbalance ratio, including Bank, Thyroid_Diff, Phonemes, Website Phishing, Oil_Spill, Pharyngitis, and Blood_Transfusion_Services, as shown in Table 1. For these datasets, we use various machine learning algorithms, including SVM, LDM, PSVM, UPSVM, CSLDM, and our proposed DBUPLDM. We separately calculate the comprehensive performance of these algorithms on performance metrics such as AUC, F-measure, and G-means.

The experimental results are detailed in Table 8. In terms of AUC, DBUPLDM achieves the highest score or is close to the highest score in all datasets. Specifically, in the Bank, Thyroid Diff, Phoneme, and Website Phishing datasets, DBUPLDM's performance is second only to the optimal algorithm, while the other datasets achieve the best performance. This indicates that DBUPLDM has outstanding advantages in overall discrimination ability. In terms of F-measure performance, DBUPLDM also demonstrates significant advantages. Taking bank as an example, the F-measure of DBUPLDM reached 33.06%, far exceeding other algorithms. This shows that DBUPLDM achieves a good balance between accuracy and recall in handling positive categories. In terms of G-means, DBUPLDM performe well in Thyroid Diff and Pharyngitis. G-means emphasizes the superiority of the algorithm in balancing positive and negative samples, while DBUPLDM performs well in this regard. The average results also indicate that DBUPLDM exhibits the most favorable overall performance evaluation. Overall, DBUPLDM achieves significant advantages in various performance indicators. Compared to traditional machine learning algorithms, DBUPLDM has achieved optimal or near optimal performance on multiple datasets, indicating its wide applicability and excellent performance in various classification tasks.

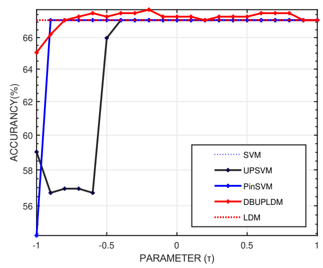
4.6. Parameter sensitivity and stability

To further demonstrate the performance of DBUPLDM, we analyze the parameter τ. As previously mentioned, τ can reduce the algorithm's sensitivity to the classification boundary according to the division distance, so it obtains higher classification accuracy. However, in DBUPLDM, the participation of dual balanced factors and second-order statistics γ_1 and γ_2 enhances the algorithm's robustness, making it less susceptible to huge fluctuations resulting from a single parameter τ change.

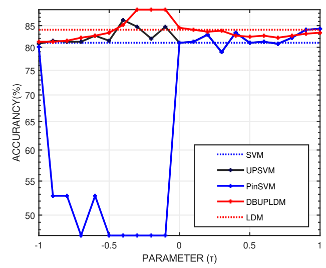
We compare the performance of PinSVM, UPSVM, and DBUPLDM (with parameter τ) to verify the advantages of dual balanced factors and second-order statistics. The τ value domain is

Table 8: Performance metrics for various algorithms on imbalanced datasets.

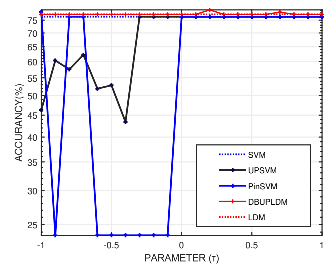
Dataset	Metrics	SVM	LDM	PSVM	UPSVM	CSLDM	DBUPLDM
Bank	AUC	52.79	**53.37**	52.79	52.79	48.13	52.79
	F-measure	0.00	5.45	0.00	28.74	18.37	**33.06**
	G-means	0.00	17.07	0.00	0.00	**38.75**	0.00
Thyroid_Diff	AUC	49.91	48.03	48.03	48.03	**57.23**	49.22
	F-measure	44.16	0.00	47.67	46.51	40.34	**48.70**
	G-means	27.54	0.00	0.00	0.00	**55.81**	50.38
Phoneme	AUC	**51.38**	51.33	51.33	51.33	43.13	51.33
	F-measure	22.04	0.00	22.04	**22.51**	14.04	22.00
	G-means	5.11	0.00	0.00	0.00	**40.16**	0.00
Website_Phishing	AUC	51.84	51.19	51.84	51.84	**52.90**	51.19
	F-measure	3.77	0.00	14.72	18.37	12.80	**19.05**
	G-means	13.99	0.00	13.99	13.99	**37.46**	0.00
Oil_Spill	AUC	**42.49**	**42.49**	**42.49**	**42.49**	42.06	**42.49**
	F-measure	0.00	0.00	0.00	**15.00**	0.00	10.94
	G-means	0.00	0.00	0.00	0.00	0.00	0.00
Pharyngitis	AUC	53.92	53.92	53.92	53.92	49.94	**56.03**
	F-measure	94.36	94.36	94.36	94.36	56.56	**94.51**
	G-means	0.00	0.00	0.00	0.00	**48.03**	32.44
Blood_Transfusion_Service	AUC	**49.01**	**49.01**	**49.01**	**49.01**	45.31	**49.01**
	F-measure	38.44	0.00	38.44	38.70	26.98	**39.34**
	G-means	0.00	0.00	0.00	0.00	**46.36**	0.00
Ave.	AUC	50.19	49.91	49.92	49.92	48.39	**50.29**
	F-measure	28.97	14.26	31.03	37.74	24.16	**38.23**
	G-means	6.66	2.44	2.00	2.00	**38.08**	11.83



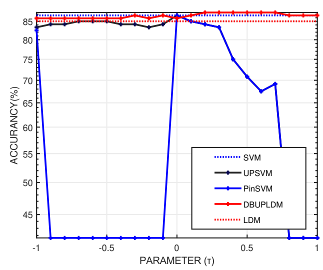
(a) Monk1 dataset



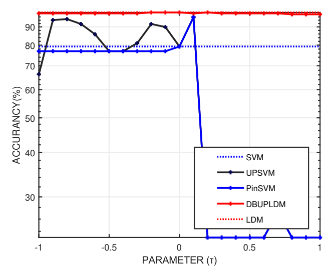
(b) Monk 2 dataset



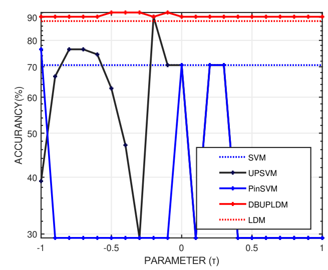
(c) Monk 3 dataset



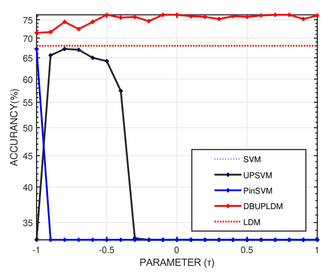
(d) Echo dataset



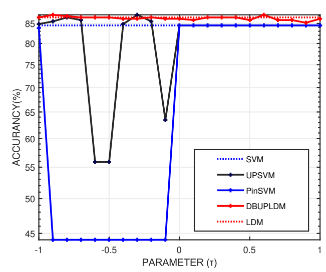
(e) Germans dataset



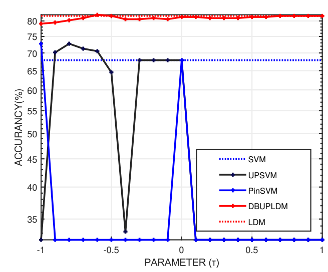
(f) BUPA dataset



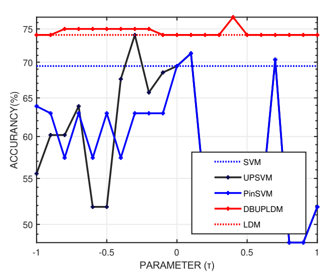
(g) Daibetes dataset



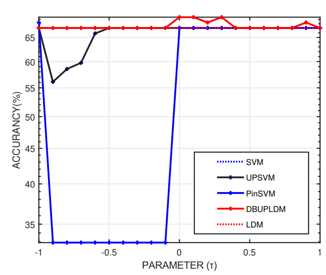
(h) Sonar dataset



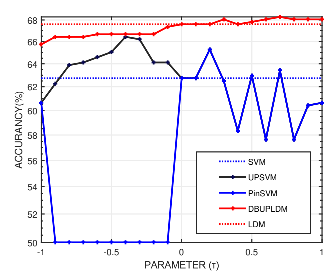
(i) Prlx dataset



(j) Prlx dataset



(k) Prlx dataset



(l) Prlx dataset

Figure 6: The parameter sensitivity performance of SVM, LDM, Pin-SVM, UPSVM, and DBUPLDM in different datasets.)

set to [-1, 1], with a step length of 0.1. Fig. 6 shows the performance of these algorithms on the datasets.

As seen from Fig. 6, DBUPLDM achieves almost the best accuracy on all datasets. Moreover, the accuracy of PinSVM and UPSVM fluctuates significantly with changes in τ values, while DBUPLDM exhibits more stable changes, highlighting its superior stability. Further analysis reveals that DBUPLDM had the best classification accuracy at most τ values, except for some special values.

4.7. Overall summary

Summarizing the analysis of all experimental results, we have the following observations.

- DBUPLDM demonstrates excellent generalization performance on artificial datasets, consistently exhibiting superior accuracy and stability across different balanced factors and artificial noise levels.
- DBUPLDM achieves the highest classification accuracy on UCI datasets with various noise levels, indicating its robustness against noise.
- DBUPLDM achieves optimal or near optimal results in various performance indicators, demonstrating its outstanding performance.
- Through adjusting the parameter τ, we discover that DBUPLDM has superior robustness and model stability, and is an excellent robust algorithm.

5. Conclusion

In this paper, we propose a dual balanced large margin classifier with unified pinball loss (DBUPLDM). Specifically, DBUPLDM changes the definition of the mean margin and constructs a new dual balance factor function to handle the imbalanced data classification task. In addition, DBUPLDM replaces the hinge loss function with quantile distance-dependent UP loss to improve the noise insensitivity of the model. We compare the proposed model with algorithms such as SVM, PinSVM, UPSVM, CSLDM, and LDM, and design a series of experiments to verify the high generalization performance and stability of the model. The experimental results of each experiment show that DBUPLDM is an excellent algorithm that can solve the problems caused by imbalanced datasets and noise more effectively. Future work will investigate the combination of fuzzy strategies and classification tasks.

CRediT authorship

Denghao Dong: Writing - original draft, Methodology, Software, Writing - review editing. **Shuangyi Fan:** Software, Formal analysis. **Junlin Chen:** Writing – review & editing. **Fudi Wang:** Software – review & editing. **Libo Zhang:** Validation, Writing - review editing, Funding acquisition.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

This work is supported by the National Natural Science Foundation of China (No. 62106205), the Natural Science Foundation of Chongqing (No. cstc2021jcyj-msxmX0824), the Youth project of science and technology research program of Chongqing Education Commission of China (No. KJQN202100207).

References

[1] Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine learning*, 20(3):273–297, 1995.

[2] Salim Rezvani and Xizhao Wang. Class imbalance learning using fuzzy art and intuitionistic fuzzy twin support vector machines. *Information Sciences*, 578:659–682, 2021.

[3] Georgios N Kouziokas. Svm kernel based on particle swarm optimized vector and bayesian optimized svm in atmospheric particulate matter forecasting. *Applied Soft Computing*, 93:106410, 2020.

[4] Fabrice Rossi and Nathalie Villa. Support vector machine for functional data classification. *Neurocomputing*, 69(7-9):730–742, 2006.

[5] Salim Rezvani and Xizhao Wang. Intuitionistic fuzzy twin support vector machines for imbalanced data. *Neurocomputing*, 507:16–25, 2022.

[6] Guolin Yu, Jun Ma, and Chenzhen Xie. Hessian scatter regularized twin support vector machine for semi-supervised classification. *Engineering Applications of Artificial Intelligence*, 119:105751, 2023.

[7] Peichung Shih and Chengjun Liu. Face detection using discriminating feature analysis and support vector machine. *Pattern Recognition*, 39(2):260–276, 2006.

[8] Arbia Soula, Khaoula Tbarki, Riadh Ksantini, Salma Ben Said, and Zied Lachiri. A novel incremental kernel nonparametric svm model (ikn-svm) for data classification: An application to face detection. *Engineering Applications of Artificial Intelligence*, 89:103468, 2020.

[9] Wencheng Huang, Hongyi Liu, Yue Zhang, Rongwei Mi, Chuangui Tong, Wei Xiao, and Bin Shuai. Railway dangerous goods transportation system risk identification: comparisons among svm, pso-svm, ga-svm and gs-svm. *Applied Soft Computing*, 109:107541, 2021.

[10] Dandi Yang, Nan Hou, Jingyi Lu, and Daan Ji. Novel leakage detection by ensemble 1dcnn-vapso-svm in oil and gas pipeline systems. *Applied Soft Computing*, 115:108212, 2022.

[11] Fatemeh Akhoni Pourhosseini, Kumars Ebrahimi, and Mohammad Hosein Omid. Prediction of total dissolved solids, based on optimization of new hybrid svm models. *Engineering Applications of Artificial Intelligence*, 126:106780, 2023.

[12] S. Lyaqini, A. Hadri, A. Ellahyani, and M. Nachaoui. Primal dual algorithm for solving the nonsmooth twin svm. *Engineering Applications of Artificial Intelligence*, 128:107567, 2024.

[13] Lev Reyzin and Robert E Schapire. How boosting the margin can also boost classifier complexity. In *Proceedings of the 23rd international conference on Machine learning*, pages 753–760, 2006.

[14] Liwei Wang, Masashi Sugiyama, Zhaoxiang Jing, Cheng Yang, Zhi-Hua Zhou, and Jufu Feng. A refined margin analysis for boosting algorithms via equilibrium margin. *The Journal of Machine Learning Research*, 12:1835–1863, 2011.

[15] Wei Gao and Zhi-Hua Zhou. On the doubt about margin explanation of boosting. *Artificial Intelligence*, 203:1–18, 2013.

[16] Teng Zhang and Zhi-Hua Zhou. Large margin distribution machine. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 313–322, 2014.

[17] Xiaolin Huang, Lei Shi, and Johan AK Suykens. Solution path for pin-svm classifiers with positive and negative τ values. *IEEE transactions on neural networks and learning systems*, 28(7):1584–1593, 2016.

[18] Libo Zhang, Qiang Jin, Shuangyi Fan, and Dun Liu. A novel dual-center based intuitionistic fuzzy twin bounded large margin distribution machines. *IEEE Transactions on Fuzzy Systems*, 2023.

[19] Xiaolin Huang, Lei Shi, and Johan AK Suykens. Support vector machine classifier with pinball loss. *IEEE transactions on pattern analysis and machine intelligence*, 36(5):984–997, 2013.

[20] Libo Zhang, Denghao Dong, Lianyi Luo, and Dun Liu. A novel fuzzy large margin distribution machine with unified pinball loss. *IEEE Transactions on Fuzzy Systems*, pages 1–13, 2023.

[21] M Tanveer, Tarun Gupta, Miten Shah, and Bharat Richhariya. Sparse twin support vector clustering using pinball loss. *IEEE Journal of Biomedical and Health Informatics*, 25(10):3776–3783, 2021.

[22] Fanyong Cheng, Jing Zhang, and Cuihong Wen. Cost-sensitive large margin distribution machine for classification of imbalanced data. *Pattern Recognition Letters*, 80:107–112, 2016.

[23] Shigeo Abe. Unconstrained large margin distribution machines. *Pattern Recognition Letters*, 98:96–102, 2017.

[24] Wenxin Zhu, Yunyan Song, and Yingyuan Xiao. Support vector machine classifier with huberized pinball loss. *Engineering Applications of Artificial Intelligence*, 91:103635, 2020.

[25] Jinrui Yang, Shuangyi Fan, Li Liu, and Libo Zhang. Optimal margin distribution matrix machine. *Expert Systems with Applications*, page 122497, 2023.

[26] Chun-Fu Lin and Sheng-De Wang. Fuzzy support vector machines. *IEEE transactions on neural networks*, 13(2):464–471, 2002.

[27] I Guyon, N Matic, and V Vapnik. Discovering informative patterns and data cleaning, advances in knowledge discovery and data mining ed by um fayyad, g. piatelsky-shapiro, p. smyth and r, 1996.

[28] Qing Song, Wenjie Hu, and Wenfang Xie. Robust support vector machine with bullet hole image classification. *IEEE transactions on systems, man, and cybernetics, part C (applications and reviews)*, 32(4):440–448, 2002.

[29] Wenjie J Hu and Qing Song. An accelerated decomposition algorithm for robust support vector machines. *IEEE Transactions on Circuits and Systems II: Express Briefs*, 51(5):234–240, 2004.

[30] Baihua Chen, Yuling Fan, Weiyao Lan, Jinghua Liu, Chao Cao, and Yunlong Gao. Fuzzy support vector machine with graph for classifying imbalanced datasets. *Neurocomputing*, 514:296–312, 2022.

[31] Andreas Christmann and Ingo Steinwart. How svms can estimate quantiles and the median. *Advances in neural information processing systems*, 20, 2007.

[32] Ingo Steinwart and Andreas Christmann. Estimating conditional quantiles with the help of the pinball loss. *Bernoulli*, 17(1):211–225, 2011.

[33] Roger Koenker and Kevin F Hallock. Quantile regression. *Journal of economic perspectives*, 15(4):143–156, 2001.

[34] Pritam Anand, Reshma Rastogi, and Suresh Chandra. Improvement over pinball loss support vector machine. *arXiv preprint arXiv:2106.01109*, 2021.

[35] A Smola. Support vector machines, regularization, optimization, and beyond. *Learning with Kernels*, 2002.

[36] Bernhard Schölkopf and AJ Smola. Learning with kernels mit press. *Cambridge, MA*, 2002.

[37] Arthur Asuncion and David Newman. Uci machine learning repository, 2007.