A Novel Fuzzy Large Margin Distribution Machine with Unified Pinball Loss

Libo Zhang, *Member, IEEE*, Denghao Dong, *Student Member, IEEE*, Lianyi Luo and Dun Liu*, *Member, IEEE*

***Abstract*—Based on the support vector machine (SVM), the Large Margin Distribution Machine (LDM) improves the generalization performance by incorporating the marginal distribution theory. Nevertheless, the current LDM models (LDMs) still exhibit limitations when it comes to handling noisy data, such as: i) LDMs fails to effectively discern the samples being noise and consequently falls short in robust defenses. ii) The hinge loss of LDMs is predicated upon the minimal inter-category separation, rendering the corresponding classifier highly susceptible to perturbations induced by noise. To address these limitations, we leverage the fuzzy set theory and pinball loss function, and propose a novel Fuzzy Large Margin Distribution Machine with Unified Pinball Loss (FUPLDM), which is performed as: i) An innovative fuzzy membership function is developed, utilizing two distinct types of feature centers and their associations with the samples. The function assigns a probability to each sample, indicating the likelihood of it being classified as noise. As a result, the model gains the remarkable ability to accurately identify and distinguish noise from other data. ii) A unified pinball (UP) loss is utilized to replace the hinge loss function inherited from SVM. The pinball loss function is based on interquartile distance, which is less affected by noise and can well improve the noise immunity of the classifier at the boundary. Therefore, FUPLDM has superior noise recognition capabilities and substantial noise resistance against its detrimental effects. Furthermore, We also analyzed the properties of FUPLDM, including noise insensitivity, intra-class distance, inter-class scatter, and misclassification error. At last, we conduct a series of comparative experiments on artificial synthetic datasets, UCI benchmark datasets, and noise-added UCI datasets, which demonstrate the effectiveness and superiority of FUPLDM.**

***Index Terms*—Fuzzy Membership, Pinball Loss, Noise immunity, Large Margin Distribution Machine**

I. Introduction

SUPPORT Vector Machine (SVM) is a prevalent classification algorithm, which has induced a wide concern for decades [1]–[3]. Its basic idea is to minimize classification errors and maximize the margin between two classes of samples by minimizing the regularization term [4]–[6]. However, based on the Structural Risk Minimization (SRM) theory, marginal distribution is more important than minimum margin in optimizing generalization performance [7], [8]. In 2014, Zhang et al. [9] proposed a Large Margin Distributor (LDM), which is an improvement of SVM. By adding two second-order statistics, i.e., the margin mean and variance of the marginal distribution, all the sample points can affect the classification hyperplane, not just support vectors [10]. Therefore, LDM achieves better generalization performance than SVM. Due to its unique advantages, there are already many variants of LDM (LDMs), such as Cost-Sensitive Large margin Distribution Machine (CS-LDM) [11], Unconstrained Large Margin Distribution Machines (ULDM) [12], Twin Large Margin Distribution Machine (TLDM) [13] and so on. Despite these improvements, they still have not achieved the desired results in terms of noise resistance.

The most fundamental reason is that LDM inherits the structural deficiencies of SVM. Specifically, hinge loss is only related to the shortest distance between two sets, which has been verified to be the reason why conventional SVMs are sensitive to noise and unstable for re-sampling [14], [15]. In 2014, Huang et al. introduced pinball loss into SVM (Pin-SVM), which substantially revised the structure of SVM [16]. Later, Pin-SVM has been experimentally verified to achieve better performance than conventional SVMs [17]–[19]. Anand found that there was major difficulty in Pin-SVM model when the hyperparameter τ takes a negative value, and established a different optimization problem for $\tau \in [-1,0)$ [20]. Due to the inherited hinge loss from SVM, LDMs definitely suffer from the noise-sensitive defect at the boundary. Inspired by their works, the noise susceptibility of LDM can also be weakened by utilizing the pinball loss.

Describing the credibility of samples is a crucial way to further enhance the anti-noise performance of classifiers [21]–[23]. Due to the noise and uncertainty in actual scenes, different samples may have different confidence levels [24]–[26]. Fuzzy set theory has been widely adopted to deal with ambiguity and uncertainty in different domains [27]–[30]. Specifically, the fuzzy membership degree indicates how much the element belongs to one specific class, which can be used to characterize the confidence level of a sample [31]–[33]. Sample's reliability is related to its distance to the class center [34]. For example, as widely recognized in machine learning, the outliers are less credible than samples near the class center. By utilizing a distance-based fuzzy membership function, Fuzzy C-Means (FCM) has achieved competitive results in many tasks, including clustering, decision making, image scanning and complex network construction [35]–[38]. However, as a soft clustering algorithm, FCM cannot exploit the label information in supervised tasks. Therefore, further

*Corresponding author.

Libo Zhang, Denghao Dong, Lianyi Luo are with the School of Artificial Intelligence, Southwest University, Chongqing, 400715, China; (e-mail: lbzhang@swu.edu.cn; dhdong1129@163.com; ly_luo2001@163.com) Dun Liu is with the School of Economics and Management, Southwest Jiaotong University, Chengdu, 610031, China. (e-mail: liudun@swjtu.edu.cn)

This work is supported by the National Nature Science Foundation of China (Nos. 62106205 and 62276217), 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), the Science Fund for Distinguished Young Scholars of Sichuan Province (No. 2022JDJQ0034).

modifications are required for LDM.

Therefore, we propose a novel Fuzzy LDM model with Unified Pinball loss (FUPLDM), which essentially improve the overall noise resistance of the classifier. Firstly, by replacing the hinge loss, the Unified Pinball (UP) loss is utilized to measure the quantile distance between two categories of sample sets. Secondly, to weaken the impact of unbalanced data sets problem, a balance weight factor is introduced to modify the object function. Thirdly, according to the characteristics of the sample points, we propose a novel Both class Centers-based Fuzzy Membership (2-FM) function to evaluate the confidence of each sample. Specifically, 2-FM is based on the idea of FCM but with some improvements. Considering that the label information of the sample set is known, we compute the two feature centers separately, which somewhat avoids the local minima and the poor robustness that exist in FCM. After deriving the model solution, we investigate its properties, including noise insensitivity, intra-class scattering, and inter-class distance. Finally, we construct a series of numerical experiments to verify the superiority of FUPLDM in terms of classification accuracy, stability and anti-noise performance. The architectural schematic of FUPLDM is shown as Fig. 1.

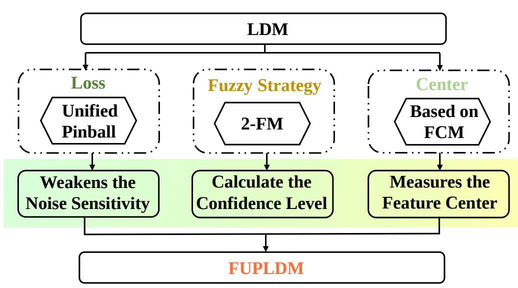


Fig. 1. Architectural schematic of FUPLDM.

The main contributions of this paper are summarized below:

- The proposed FUPLDM greatly improves the noise resistance, robustness and classification performance of LDM, which are verified theoretically and experimentally.
- By utilizing the fuzzy set theory, the novel 2-FM method reasonably characterizes the confidence level of each sample based on its distance to the class center.
- To the best of the author's knowledge, this the first work to introduce the pinball loss to LDM. Specifically, the quantile distance is introduced to replace the shortest distance in existing LDMs, which essentially contributes to overcome the inherent noise susceptibility.

The rest of this paper is organized as follows. Section II briefly describes SVM, Pin-SVM and LDM. A novel FUPLDM classification model and its solution are presented in Section III. Section IV analyzes some properties of FUPLDM. Section V gives the experimental performance of six algorithms on synthetic datasets, UCI datasets with noises, respectively. Finally, conclusions are drawn and future related research work is envisioned in Section VI.

II. PRELIMINARIES

Consider a binary classification problem, the sample set $U = (x_1, y_1) \ldots (x_m, y_m)$ is given, where $x_i \in \mathbb{R}^n$ and $y_i \in \{+1, -1\}$. Let $X = [x_1, \ldots x_m]^T$, $Y = [y_1, \ldots y_m]^T$, $P = \{i \mid y_i = 1\}$ and $N = \{i \mid y_i = -1\}$. Denote $\mathscr{H}$ as a hyperplane given by $\omega^T x + b = 0$ with $\omega \in \mathbb{R}^n$, and $b \in \mathbb{R}$. P and N are separable by $\mathscr{H}$ for $i = 1, \ldots, m$, we can get

$$\begin{cases} \omega^T x_i + b > 0, \forall i \in P, \\ \omega^T x_i + b < 0, \forall i \in N. \end{cases} \tag{1}$$

Let $\Theta_i = y_i \left(\omega^T x_i + b\right)$, which evaluates the distance between point x_i and the hyperplane $\mathscr{H}$. The distance of samples in each class to the corresponding hyperplane $\mathscr{H}$ is defined as

$$t_P(\omega, b) = \min_{i \in P} \Theta_i \ ; \quad t_N(\omega, b) = \min_{i \in N} \Theta_i. \tag{2}$$

The classification hyperplane is obtained by

$$\max_{\|\omega\|=1, b} \left\{ t_P(\omega, b) + t_N(\omega, b) \right\}. \tag{3}$$

The points that satisfy Eq. (3) are called support vectors.

A. Support Vector Machine With Hinge Loss

The hinge loss function is defined as follows:

$$L_{\text{hinge}}(u) = \begin{cases} u & u \geq 0 \\ 0 & u < 0 \end{cases}, \ \forall u \in \mathbb{R}. \tag{4}$$

The traditional SVM formulation with hinge loss can be derived from the following formula:

$$\min_{\omega, b} \frac{1}{2} \|\omega\|^2 + C \sum_{i=1}^{m} L_{\text{hinge}}\left(1 - \Theta_i\right), \tag{5}$$

where $C \geq 0$ is a tunable hyper-parameter, which determines the degree of the overlap in the model with a soft margin. Additionally, the traditional SVM (5) is further equivalently transformed into:

$$\begin{aligned} &\min_{\omega, b, \xi} \ \frac{1}{2} \|\omega\|^2 + C \sum_{i=1}^{m} \xi_i, \\ &s.t. \ \Theta_i \geq 1 - \xi_i, \quad \xi_i \geq 0, \quad i = 1, 2, \ldots, m, \end{aligned} \tag{6}$$

where ξ is called the slack variable. As can be seen from the above formulas, SVM maximizes the minimum distance of the sample points to the hyperplane, i.e., the distance from the support vectors to the hyperplane.

B. Support Vector Machine With Pinball Loss

Since the hinge loss uses the shortest distance between support vectors, traditional SVMs are susceptible to noise points, especially when they are around the classification hyperplane. The pinball loss function can improve this boundary noise sensitivity, which is defined as follows:

$$L_\tau(u) = \begin{cases} u & u \geq 0 \\ -\tau u & u < 0 \end{cases}, \tag{7}$$

where $\tau \in [-1, 0) \cup (0, 1]$ is a hyper-parameter that can control quantiles. According to [16], the Pin-SVM classifier by utilizing the quantile distance to measure margins with the pinball loss function is presented as

$$\min_{\omega,b}\ \frac{1}{2}\|\omega\|^2+C\sum_{i=1}^{m}L_{\tau_+}(1-\Theta_i). \tag{8}$$

The constrained optimization form of Pin-SVM is depicted as follows.

$$\begin{aligned}&\min_{\omega,b,\xi}\ \frac{1}{2}\|\omega\|^2+C\sum_{i=1}^{m}\xi_i,\\ &s.t.\quad 1-\xi_i\le\Theta_i\le 1+\frac{1}{\tau_+}\xi_i,\quad i=1,2,\ldots,m,\end{aligned} \tag{9}$$

where τ_+ denotes $\tau\in(0,1]$.

C. Traditional Large Margin Distribution Machine

According to the SRM theory, the marginal distribution is a better indicator than the minimum margin [12]. The two statistics, margin mean and margin variance, describe the characteristics of the marginal distribution of the data points:

$$\bar{\gamma}=\frac{1}{m}\sum_{i=1}^{m}\Theta_i\ ;\quad \widehat{\gamma}=\sum_{i=1}^{m}\sum_{j=1}^{m}(\Theta_i-\Theta_j)^2, \tag{10}$$

where $\bar{\gamma}$ represents the margin mean and $\widehat{\gamma}$ is the margin variance. So the LDM is presented as

$$\begin{aligned}&\min_{\omega,\xi}\quad \tfrac{1}{2}\omega^T\omega+\lambda_1\hat{\gamma}-\lambda_2\bar{\gamma}+C\sum_{i=1}^{m}\xi_i,\\ &\text{s.t.}\quad \Theta_i\ge 1-\xi_i,\ \xi_i\ge 0,\ i=1,\ldots,m.\end{aligned} \tag{11}$$

Especially, when $\lambda_1=\lambda_2=0$, LDM degenerates into an ordinary SVM. Moreover, the Lagrangian function can deal with this kind of convex optimization problem well.

III. Fuzzy Large Distribution Machine with Unified Pinball Constraints

In this section, we propose a novel FUPLDM model, which overcomes the noise susceptibility and resampling instability of existing LDMs.

A. The Unified Piball Loss

Some related papers claimed that the value of τ does not affect the basic model, and by default the value domain of τ is independent of the constraints of the model [17]. However, Anand found that the Pin-SVM model does not apply when $\tau\in[-1,0]$, and established a different optimization problem for $\tau\in[-1,0)$ [20]. By establishing the relationship between the negative and positive values of τ, an unified pinball (UP) loss for optimization problem (9) can be obtained.

Theorem 1. *The constraints of problem (9) is different when* $\tau\in[-1,0)$ *and* $\tau\in(0,1]$.

Proof. When $\tau\in(0,1\,]$, $-\tau u$ in Eq. (7) is computed as

$$-\tau u\in\left\{\begin{array}{l}[-u,0],\ u\ge 0\Rightarrow u>-\tau u\\ [0,-u],\ u<0\Rightarrow u<-\tau u\end{array}\right.. \tag{12}$$

When $\tau\in[-1,0)$, $-\tau u$ can be calculated as

$$-\tau u\in\left\{\begin{array}{l}[0,\ u],\ u\ge 0\ \Rightarrow u>-\tau u\\ [-u,0],\ u<0\Rightarrow u<-\tau u\end{array}\right.. \tag{13}$$

Based on Eq.s (12) and (13), Eq. (7) can be reformulated as

$$L_\tau(u)=\max(-\tau u,u)\ ,\tau\in[-1,0)\cup(0,1]. \tag{14}$$

Let $\xi_i=L_\tau(1-\Theta_i)=\max(1-\Theta_i,-\tau(1-\Theta_i))$. Based on original model (8), the constrained form of Pin-SVM is put as

$$\begin{aligned}&\min_{\omega,b,\xi}\ \frac{1}{2}\|\omega\|^2+C\sum_{i=1}^{m}\xi_i,\\ &s.t.\ \xi_i\ge 1-\Theta_i,\\ &\qquad \xi_i\ge-\tau(1-\Theta_i),\quad i=1,2,\ldots,m.\end{aligned} \tag{15}$$

When τ is negative, the second constraint can be rewritten as

$$\Theta_i\ge 1+\frac{1}{\tau}\xi_i,\ i=1,2,\ldots,m, \tag{16}$$

which is the correct form of the constraint by the UP loss.

The constraint in model (9) can be formulated as follows:

$$\Theta_i\le 1+\frac{\xi_i}{\tau_-}\Leftrightarrow(\Theta_i-1)\tau_-\ge\xi_i. \tag{17}$$

The result follows. □

B. Both class Centers-based Fuzzy Membership (2-FM)

Based on different considerations, fuzzy membership function can be calculated in different ways. In general, the farther the sample is from the class center, the lower the credibility. For example, as widely recognized in machine learning, the outliers are less reliable than samples near the class center. Hence, the distance-based membership functions have achieved much attention. Among them, FCM exhibits competitive performance. However, FCM has intrinsic defects for supervised problems, as it is an unsupervised method and label information cannot be exploited by it. Therefore, the Both class Centers-based Fuzzy Membership (2-FM) function is proposed in this subsection.

FCM integrates fuzzy set theory in the process of finding the cluster center, which launches a good basis for the subsequent calculation of fuzzy membership. Based on the idea of FCM, we set up a cost function to determine the respective centers of the two types of samples, which is expressed as follows:

$$J_{fm}(\mu,f)=\sum_{i=1}^{m}f_i^{\,l}\,\|x_i-\mu\|^2, \tag{18}$$

$$\mu=\frac{\sum_{i=1}^{m}f_i^{\,l}x_i}{\sum_{i=1}^{m}f_i^{\,l}},f_i=\frac{\left(1/\|x_i-\mu_j\|^2\right)^{1/(l-1)}}{\sum_{k=1}^{2}\left(1/\|x_i-\mu_k\|^2\right)^{1/(l-1)}}, \tag{19}$$

where m and μ are the number and center of samples in Class 1 or Class 2, respectively. The parameter l denotes the fuzzy constant, and is not subject to optimization. $f\in[0,1]$ is the membership matrix. The algorithm converges when f and μ are stable. After the cluster center of each sample (μ^+ and μ^-) is obtained according to FCM, the fuzzy membership degree of each sample is constructed based on the distance, which can be expressed

$$s_i = \begin{cases} \dfrac{\|x_i{}^+ - \mu^+\|^{-2}}{\|x_i{}^+ - \mu^+\|^{-2} + \|x_i{}^+ - \mu^-\|^{-2}}, & i \in P \\ \dfrac{\|x_i{}^- - \mu^-\|^{-2}}{\|x_i{}^- - \mu^+\|^{-2} + \|x_i{}^- - \mu^-\|^{-2}}, & i \in N \end{cases}. \tag{20}$$

From the Eq. (20), we can see that s_i is always a value between [0, 1].In what follows, the numerical example is provided to account for the relevant definitions of Eqs. (18)-(20).

Example 1. *In 2FM, the feature center is composed of the sample's features and their corresponding affiliation with the center. Conventionally, the sample center is calculated as the mean of the sample's features. However, in constructing the sample center within the 2FM framework, special attention is paid to outliers that deviate significantly from the center. During the calculation process, these outliers are assigned a smaller weight, thus mitigating their interference on the final result. A visual depiction of this concept is presented in Fig. 2. The figure illustrates that the feature centers computed by 2FM are located at a greater distance from the boundary. In contrast, when adopting traditional methods, the presence of outliers (as depicted by the boxed samples in the figure) tends to influence the centers towards the classification line, resulting in undesired one.*

To emphasize the advantages of 2FM and other membership functions, we elucidate it by conducting a comparison experiments. The experimental results are represented by Fig. 3. As observed in Fig. 3, the 2-FM function is superior at identifying noise when compared to the traditional fuzzy membership function. This can be attributed to its ability to calculate feature centers by taking into account the affiliation of each sample to the center, rather than solely relying on their mean values. Moreover, the 2-FM function computes centers for each class separately and the final membership of each sample is linked to both centers. As indicated by Eq. (20), the 2-FM function not only brings the samples closer to the center of their respective class, but also distances them from the other class, which is a more rational framework.

Furthermore, a closer examination of the classification surface and noise in Fig. 3 reveals that 2-FM effectively mitigates the tilting of the hyperplane towards noise by accurately identifying noisy samples. In contrast, the traditional fuzzy membership function is less effective at identifying noise and thus tends to tilt the hyperplane towards it, resulting in lower predictive power compared to 2-FM.

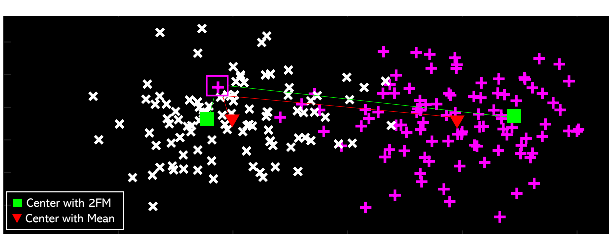
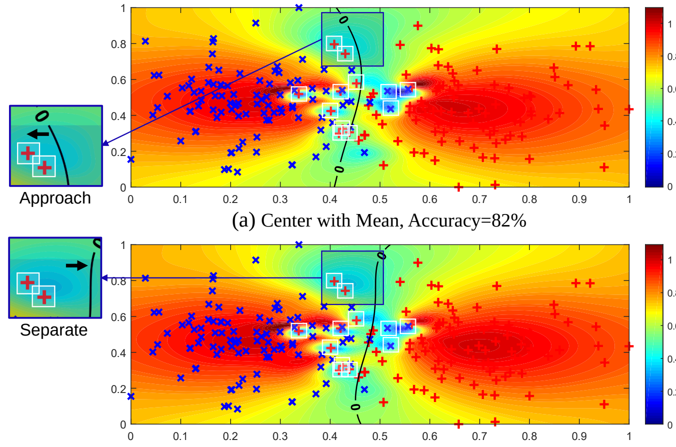


Fig. 2. Illustration of the property of the 2-FM fuzzy function.



(a) Center with Mean, Accuracy=82%

(b) Center with 2-FM, Accuracy=85%

Fig. 3. Illustration of 2-FM fuzzy function under artificially constructed Gaussian distributed data. The manually added noise is depicted by the white box. Subfigure (a) displays the conventional fuzzy membership obtained by averaging the sample features, while subfigure (b) illustrates the fuzzy membership obtained using the 2-FM function. When compared to the results in (a), the fuzzy membership depicted in (b) is more effective at detecting noise (indicated by deeper shadows around the noise) and better at mitigating its influence on the hyperplane.

C. Original optimization Problem of FUPLDM

Given the labeled sample set U, the sample set with the corresponding membership degree is denoted as $\widetilde{U}$:

$$\widetilde{U} = \{(x_1, y_1, s_1), (x_2, y_2, s_2), ..., (x_m, y_m, s_m)\}, \tag{21}$$

where (x_m,y_m) denotes a pair of labeled sample points, and s_m refers to the fuzzy membership of the sample to the corresponding class.

Based on the proposed UP constraints and 2-FM function, the unconstrained form of FUPLDM can be presented as:

$$\min_{\omega} \frac{1}{2}\|\omega\|^2 + \lambda_1\hat{\gamma} - \lambda_2\bar{\gamma} + C\sum_{i=m}^{m} s_i L_\tau(1-\Theta_i). \tag{22}$$

Based on the reformulated constraint by the UP loss in Eq. (16), FUPLDM can be expressed by the following formula:

$$\begin{aligned} \min_{\omega,b,\xi} \ & \frac{1}{2}\|\omega\|^2 + \lambda_1\hat{\gamma} - \lambda_2\bar{\gamma} + C\sum_{i=1}^{m} s_i\xi_i, \\ s.t. \ \ & \Theta_i \geq 1 - \xi_i, \\ & v_\tau\Theta_i \leq v_\tau(1 + \frac{1}{\tau}\xi_i), \ i = 1, 2, \ldots, m, \end{aligned} \tag{23}$$

where v_τ is a symbolic function whose value is 1 when $\tau > 0$, and -1 otherwise. In order to solve the problem of unbalanced data sets, we set different weights for the penalty factor C so that the minority is penalized more, thus increasing its importance, which is as follows:

$$C_i = \begin{cases} C, \ i \in P \\ \dfrac{|P|}{|N|} C, \ i \in N \end{cases}, \tag{24}$$

where C is a defined hyperparameter, $|P|$ and $|N|$ represent the number of positive and negative samples, respectively. The use of weights $\frac{|P|}{|N|}$ enables a higher penalty to be imposed on the samples from the smaller class, thereby prioritizing their importance. So the constrained optimization form of FUPLDM is:

$$\begin{aligned} \min_{\omega,b,\xi} \quad & \frac{1}{2}\|\omega\|^2+\lambda_1\hat{\gamma}-\lambda_2\bar{\gamma}+\sum_{i=1}^{m}C_i s_i \xi_i, \\ s.t. \quad & \Theta_i \geq 1-\xi_i, \\ & v_\tau \Theta_i \leq v_\tau(1+\frac{1}{\tau}\xi_i),\ i=1,2,\dots,m. \end{aligned} \tag{25}$$

Note that, when $\tau \in [-1,0)\cup(0,1]$, the constraints in model (25) hold. As τ tends to 0, the term $\frac{1}{\tau}\xi_i$ tends to positive infinity. At this point, since $v_\tau\Theta_i \leq +\infty$ is a constant relationship, this constraint will lose its binding effect. Hence, the Pinball loss function is converted to the hinge loss function, the second constraint in model (25) is invalid.

Remark 1. *FUPLDM model is a general model, and we have the following observations.*

- *If $\tau=0$, $C_i=C$, model (25) degenerates into fuzzy LDM with hinge loss (F-LDM). In the proposed model, the quantile distance is introduced to replace the shortest distance in existing LDMs, which is the first time that LDM is combined with pinaball loss.*
- *If $s=e$, $C_i=C$, $\tau=0$, model (25) degenerates into traditional LDM. The proposed model introduces the 2-FM strategy and UP loss function, which enhances the classification and anti-noise performance.*
- *If $s=e$, $C_i=C$, $\lambda_1,\lambda_2=0$, the FUPLDM model degenerates into Pin-SVM. FUPLDM modifies the structure of Pin-SVM by unifying the loss function, adding a balance factor and optimizing the margin distribution, which reduces the structural risk and improves the generalization performance.*
- *By combining LDM with fuzzy set theory and the UP loss, our FUPLDM achieves competitive noise resistance and generalization performance, as verified theoretically and experimentally respectively in Sections IV and V.*

D. Kernel Formulation and Dual Problem of FUPLDM

Similar to the conventional SVM algorithm, FUPLDM can also handle linear and nonlinear data. By utilizing a kernel function, the input data is mapped to the Hilbert space $\mathbf{H_0}$ through a typical nonlinear mapping. Assuming that the data point x_i is mapped from $\mathbb{R}^n$ to $\mathbb{R}^l (l>n)$, the relationship can be expressed as $\phi: X \rightarrow \mathbf{H_0}$.

According to model (25), the original quadratic programming problem (QPP) of the FUPLDM is

$$\begin{aligned} \min_{\omega,b,\xi} \quad & \frac{1}{2}\|\omega\|^2+\lambda_1\hat{\gamma}^{(\mathbf{H_0})}-\lambda_2\bar{\gamma}^{(\mathbf{H_0})}+\sum_{i=1}^{m}C s_i \xi_i, \\ s.t. \quad & \Theta_i^{\mathbf{H_0}} \geq 1-\xi_i, \\ & v_\tau \Theta_i^{\mathbf{H_0}} \leq v_\tau(1+\frac{1}{\tau}\xi_i),\ i=1,2,\dots,m, \end{aligned} \tag{26}$$

where $\Theta_i^{\mathbf{H_0}}=y_i(\omega^T\phi(x_i)+b)$. We consider the model when $v_\tau=1$, and then perform an algebraic transformation to obtain the final complete solution. Hence, the Lagrangian function of model (26) is

$$\begin{aligned} \mathscr{L}(\omega,b,\xi;\alpha,\beta) &= \frac{1}{2}\|\omega\|^2-\frac{\lambda_2}{m}\sum_{i=1}^{m}\Theta_i^{\mathbf{H_0}} \\ &+\lambda_1\sum_{i=1}^{m}\sum_{j=1}^{m}\left(\Theta_i^{\mathbf{H_0}}-\Theta_j^{\mathbf{H_0}}\right)^2+\sum_{i=1}^{m}C_i s_i\xi_i \\ &-\sum_{i=1}^{m}\alpha_i\left(\Theta_i^{\mathbf{H_0}}-1+\xi_i\right)-\sum_{i=1}^{m}\beta_i\left(\tau\left(1-\Theta_i^{\mathbf{H_0}}\right)+\xi_i\right). \end{aligned} \tag{27}$$

Since it is difficult to directly derive the derivative of model (27), we can convert it into a matrix form according to the representation theorem in [39].

Theorem 2. *The optimal solution ω^* for model (26) admits a representation of the form:*

$$\omega^*=\sum_{i=1}^{m}\delta_i\phi(x_i)=\Phi\delta, \tag{28}$$

where $\delta=[\delta_1,\dots,\delta_m]^T$ are the specific coefficients and $\Phi=[\phi_1,\dots,\phi_m]^T$.

Proof. The proof is similar to the process in [9]. To avoid repetition, detailed proofs are not listed here. □

According to Theorem 2, $\Theta_i^{\mathbf{H_0}}$, $\bar{\gamma}^{(\mathbf{H_0})}$ and $\hat{\gamma}^{(\mathbf{H_0})}$ can be transformed into the following formula:

$$\begin{aligned} &\sum_{i=1}^{m}\Theta_i^{\mathbf{H_0}}=m\bar{\gamma}^{(\mathbf{H_0})}, \\ &\bar{\gamma}^{(\mathbf{H_0})}=\frac{1}{m}\sum_{i=1}^{m}\Theta_i^{\mathbf{H_0}}=\frac{1}{m}Y^TK\delta, \\ &\hat{\gamma}^{(\mathbf{H_0})}=\sum_{i=1}^{m}\sum_{j=1}^{m}\left(\Theta_i^{\mathbf{H_0}}-\Theta_j^{\mathbf{H_0}}\right)^2 \\ &\qquad=\frac{2\delta^TKK^T\delta}{m}-\frac{2\delta^TKYY^TK\delta}{m^2}, \end{aligned} \tag{29}$$

where $K=\Phi^T\Phi$. In this process, the role of parameter b is omitted, as it has no substantial effect on the magnitude of the margin variance and margin mean [40]. Then, we substitute the Eq. (29) into model (26), the matrix form of FUPLDM is

$$\begin{aligned} \min_{\delta,\xi} \quad & \frac{1}{2}\delta^TQ\delta+E^T\delta+\sum_{i=1}^{m}C_i s_i\xi_i, \\ s.t. \quad & \xi_i \geq 1-\Theta_i^{\mathbf{H_0}}, \\ & \xi_i \geq -\tau(1-\Theta_i^{\mathbf{H_0}}), \qquad (\tau\in(0,1]) \end{aligned} \tag{30}$$

where $Q=4\lambda_1(mK^TK-(KY)(KY)^T+4\lambda_1m^2K)/m^2$ and $E=-\lambda_2KY/m$.

Theorem 3. *The initial form of the QPP of FUPLDM can be calculated as*

$$\begin{aligned} \min_{\alpha,\beta} \quad & \frac{1}{2}(\alpha-\tau\beta)^TA(\alpha-\tau\beta)+(\frac{\lambda_2Ae-me}{m})^T(\alpha-\tau\beta), \\ s.t. \quad & \mathrm{C}_i s_i-\alpha_i-\beta_i=0, \\ & \alpha_i\geq 0,\ \beta_i\geq 0,\ i=1,\dots,m. \qquad (\tau\in(0,1]) \end{aligned} \tag{31}$$

(Refer to supporting document for detailed proof)

Theorem 4. *The final dual QPP of FUPLDM can be written as:*

$$\begin{aligned} \min_{\eta} \quad & \frac{1}{2}\eta^T A\eta + (\frac{\lambda_2 Ae - me}{m})^T\eta, \qquad (\tau \in [-1,0)\cup(0,1]) \\ s.t. \quad & -v|\tau|C_i s_i \le \eta_i \le C_i s_i, \; i=1,...,m. \end{aligned} \tag{32}$$

(Refer to supporting document for detailed proof)

See the supporting documents for the proof section. Once QPP in model (32) is solved, we can find a decision function of FUPLDM to deal with data classification problems. According to the KKT conditions Eq. (33):

$$\begin{aligned} &\frac{\partial\mathscr{L}}{\partial\delta} = Q\delta + E - \sum_{i=1}^{m}\alpha_i y_i K_i + \tau\sum_{i=1}^{m}\beta_i y_i K_i = 0, \\ &\frac{\partial\mathscr{L}}{\partial b} = \sum_{i=1}^{m} y_i(\alpha_i - \tau\beta_i) = 0, \\ &\frac{\partial\mathscr{L}}{\partial\xi_i} = C_i s_i - \alpha_i - \beta_i = 0, \; \forall \text{ i} = 1,2,...,\text{m}. \end{aligned} \tag{33}$$

the coefficient δ can be reasoned as

$$\begin{aligned} \delta &= Q^{-1}\{K[diag(Y)](\alpha - v\mathbf{B}) - E\} \\ &= Q^{-1}\left\{\frac{\lambda_2}{m}K[diag(Y)]e + K[diag(Y)]\eta\right\} \\ &= Q^{-1}K[diag(Y)](\frac{\lambda_2 e + m\eta}{m}). \end{aligned} \tag{34}$$

Hence, a new test instance $\mathbf{x_z}$ can be classified by the following decision function:

$$f(\mathbf{x_z}) = \text{sgn}(\omega^T\phi(\mathbf{x_z})) = \text{sgn}(\sum_{i=1}^{m}\delta_i k(x_i,\mathbf{x_z})). \tag{35}$$

To provide a more lucid presentation of the algorithmic logic behind FUPLDM, we elucidate it using Algorithm 1.

In traditional SVM, the computational complexity involves converting the original optimization problem into a dual form, which has a complexity of $O(m^2)$. Solving the dual problem then has a computational complexity of $O(m^3)$ [41]. On the other hand, LDM requires additional time for computing the first and second-order statistics [9]. Specifically, it also takes $O(m^3)$ to transform the original problem into the dual form, and the complexity of solving the dual problem remains the same as SVM. The complexity of FUPLDM is slightly larger than LDM, as it needs extra time to compute the fuzzy membership. Since FUPLDM draws on the idea of FCM in the calculation of feature centers, the time complexity of this process is consistent with $O(mdt)$, where t and d represents the number of iterations and features. Therefore, the time complexity of FUPLDM is $O(max\{mdt,m^3\})$. Typically, the values of d and t are smaller than m, resulting in a computational complexity of $O(m^3)$ for FUPLDM, which is as same as LDM.

IV. Properties of FUPLDM

In this Section, a comprehensive analysis on the properties of the FUPLDM is performed from the perspective of noise insensitivity, within-class scatter, and intra-class distance.

Algorithm 1: FUPLDM

Input : samples matrix X, label matrix Y, kernel matrix K.
Output: predict label of data $\mathbf{x_z}$.

$\mu^+, \mu^- \leftarrow$ Eq. (19).
Obtain the score value s_i by utilizing Eq.(20), $i \in \mathbf{P},\mathbf{N}$.
while Parameters are not fully traversed **do**
  Update $C_i(i$ = 1,2,...,m) and $\lambda_i(i$ = 1,2).
  Calculate $Q = 4\lambda_1(mK^TK - (KY)(KY)^T + 4\lambda_1 m^2 K)/m^2$.
  $A = [diag(Y)]KQ^{-1}K[diag(Y)]$.
  repeat
    Update variables η in Eq.(32) using the QPP toolkit.
  until convergenc
  $\delta \leftarrow \eta$ by Eq.(34).
  Calculate the label value by Eq. (35) and accuracy.
end while
Acquire the best parameter combination and receive the optimal model.
Calculate the label value of the test sample $\mathbf{x_z}$ by Eq.(35).

A. Noise Insensitivity

Since FUPLDM uses the UP loss function, it can weaken the sensitivity to noise around the decision boundary.

The segmented function $sgn_\tau(u)$ is defined as

$$\text{sgn}_\tau(u) = \begin{cases} 1 & u > 0; \\ [-\tau,1] & u = 0; \\ -\tau & u < 0, \end{cases} \tag{36}$$

where $sgn_\tau(u)$ is the subgradient of UP loss $L_\tau(u)$. Without loss of generality, we focus on a linear classifier. Denote $\mathbf{0}$ as an all-zero vector, then the optimal condition of optimization problem (22) can be written as

$$\begin{aligned} \mathbf{0} \in \omega + 2\lambda_1 &\sum_{i,j=1}^{m}(y_i x_i - y_j x_j)(\Theta_i - \Theta_j) \\ &- \frac{\lambda_2}{m}\sum_{i=1}^{m} y_i x_i - C\sum_{i=1}^{m}\text{sgn}_\tau(1-\Theta_i)y_i x_i. \end{aligned} \tag{37}$$

To concretely explain the noise insensitivity of FUPLDM, We divide the index set into three disjoint sets:

$$\begin{aligned} U_{\omega,b}^{+} &= \{i : 1-\Theta > 0\}; \\ U_{\omega,b}^{0} &= \{i : 1-\Theta = 0\}; \\ U_{\omega,b}^{-} &= \{i : 1-\Theta < 0\}, \end{aligned} \tag{38}$$

where $i = 1,...,m$ and $\Theta = y_i(\omega^T x_i + b)$. Using Eq. (38) and $\vartheta_i \in [-\tau,1]$, then Eq. (37) can be rewritten as

$$\begin{aligned} &\omega + 2\lambda_1\sum_{i,j=1}^{m}(y_i x_i - y_j x_j)(\Theta_i - \Theta_j) - \frac{\lambda_2}{m}\sum_{i=1}^{m} y_i x_i \\ &-C(\sum_{i\in U_{\omega,b}^{+}}^{m} y_i x_i - \tau\sum_{i\in U_{\omega,b}^{-}}^{m} y_i x_i + \sum_{i\in U_{\omega,b}^{0}}^{m} y_i x_i \vartheta_i) = 0. \end{aligned} \tag{39}$$

The above conditions indicate that τ controls the number of samples present in the sets $U^+_{\omega,b}$ and $U^-_{\omega,b}$.

Remark 2. *The performance of FUPLDM will be affected by the value of τ.*

- *When $\tau=1$, both sets contain many points, the presence of a few noises has negligible impact on all data points. For instance, if there is only one noise within a sample size of 10,000, the effect is minimal. So the results are less sensitive to zero-mean noise on the features.*
- *As τ decreases, since $\omega+2\lambda_1\sum_{i,j=1}^{m}(y_ix_i-y_jx_j)(\Theta_i-\Theta_j)-\frac{\lambda_2}{m}\sum_{i=1}^{m}y_ix_i$ does not change, assuming its sign is χ, with equation $\sum_{i\in U^+_{\omega,b}}^{m}y_ix_i-\tau\sum_{i\in U^-_{\omega,b}}^{m}y_ix_i+\sum_{i\in U^0_{\omega,b}}^{m}y_ix_i\vartheta_i=\frac{\chi}{C}$, we can find that there are fewer samples in $U^+_{\omega,b}$ and more samples in $U^-_{\omega,b}$. The schematic diagram is shown in Fig. 4.If there is noise in this case, the effect of this noise becomes larger because the sample in $U^+_{\omega,b}$ becomes smaller, thereby reducing the noise insensitivity.*
- *When τ continues to decrease to a negative value, points with $y_if(x_i)$ will obtain a certain gain. If a point is classified correctly, a larger distance to the boundary is still preferred. This is due to the fact that points closer to the boundary are more likely to have an effect on the classification surface. If the points around these boundaries are noisy, then it can be very misleading for model training. Therefore, when $\tau<0$, a certain gain is given to the samples of $U^-_{\omega,b}$ (i.e., the correctly classified samples). Greater gain is obtained when the correctly classified samples are further away from the decision boundary, which ensures a larger average distance. Therefore, the noise immunity of the model can be significantly improved by adjusting the value of τ.*

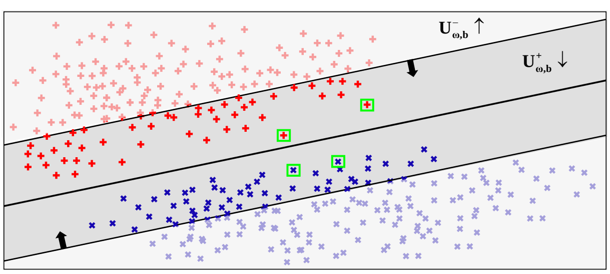


Fig. 4. Noise Insensitivity variation diagram. The shaded area indicates $U^+_{\omega,b}$ and the rest indicates $U^-_{\omega,b}$. The green box is the noise sample.

B. Intra-Class Distance, Inter-Class Scatter, and Misclassification Error

As discussed above, samples in $U^0_{\omega,b}$ can be extended to two hyperplanes $\mathscr{H}^{\mathbf{P}}:\{\omega^Tx+b=1\}$ and $\mathscr{H}^{\mathbf{N}}:\{\omega^Tx+b=-1\}$. We can measure scatter by the sum of the distances between a given point x_{i_0} and other points x_i. Hence, the scatter of $x_i,(i\in P)$ around the point x_{i_0} can be defined as

$$\sum_{i=1}^{m}\left|(\omega^Tx^+_{i_0}+b)-(\omega^Tx^+_i+b)\right|=\sum_{i=1}^{m}\left|\omega^T(x^+_{i_0}-x^+_i)\right|. \quad (40)$$

Since $y_i=\pm1$ and $U^0_{\omega,b,i_0}=\{i:\omega^Tx_{i_0}+b=\pm1\}$, we can get

$$\sum_{i=1}^{m}\left|\omega^T(x^\pm_{i_0}-x^\pm_i)\right|=\sum_{i=1}^{m}\left|1-y^\pm_i(\omega^Tx^\pm_i+b)\right|. \quad (41)$$

After that, we set the misclassification error $C_2L_{hinge}(1-y_if(x))=C_2max\{1-y^\pm_i(\omega^Tx^\pm_i+b),0\}$, and consider the following equation:

$$\begin{aligned}\min_{\omega,b}\ &\frac{1}{2}\|\omega\|^2+C_1\sum_{i=1}^{m}\left|1-y^\pm_i(\omega^Tx^\pm_i+b)\right|\\&+C_2\max\{1-y^\pm_i(\omega^Tx^\pm_i+b),0\}.\end{aligned} \quad (42)$$

In model (42), the first term can be interpreted as maximizing the inter-class distance (minimizing $\|\omega\|^2$ is equivalent to maximizing $1/\|\omega\|^2$). The second and third terms are embodiment of minimizing scattering around the hyperplane and misclassification, respectively. Setting $C=C_1+C_2$ and $\tau=C_1/C$. Add the fuzzy membership s and second-order statistics $\lambda_1\hat{\gamma}$, $\lambda_2\bar{\gamma}$ to the model (42), the model of the proposed FUPLDM can be obtained.

Remark 3. *An excellent classifier should have small intra-class dispersion and small misclassification error [16]. As can be seen from the objective function (42), FUPLDM minimizes the intra-class dispersion $C_1\sum_{i=1}^{m}\left|1-y^\pm_i(\omega^Tx^\pm_i+b)\right|$ and misclassification error $C_2\max\{1-y^\pm_i(\omega^Tx^\pm_i+b),0\}$ while maximizing the inter-class distance $1/\|\omega\|^2$ (i.e. minimizing $\|\omega\|^2$), which satisfies the properties of an ideal classifier. The hinge loss focuses on misclassification, and the absolute loss focuses on intra-class propagation [16], while the UP loss considers both, which is a trade-off consideration.*

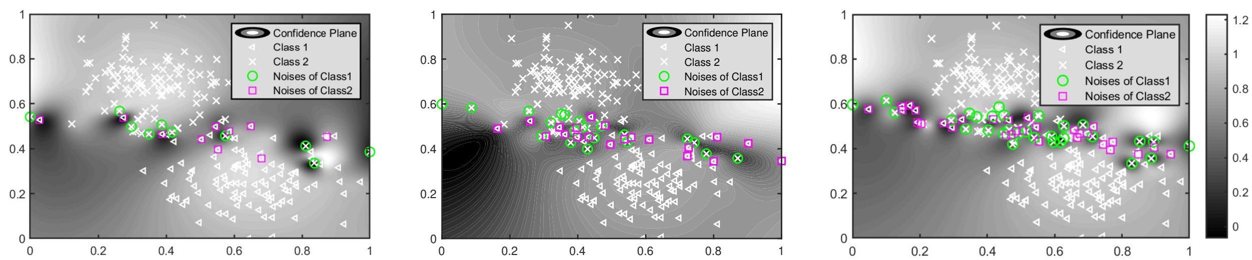
To conclude, the proposed FUPLDM achieves maximization of the intra-class distance while achieving the ideal goal of minimizing within-class scatter and misclassification error.

V. NUMERICAL EXPERIMENTS

To verify the effectiveness of the proposed FUPLDM, we compare FUPLDM with SVM, Pin-SVM, UPSVM, LDM, and F-LDM using an artificial dataset and 16 benchmark datasets with different variance noises. For each dataset, we normalize the data features to [0, 1]. In the selection of the kernel function, a linear kernel and a Gaussian Radial Basis Function (RBF) kernel are adopted, respectively. Some necessary parameters to traverse the range: $C\in\{2^{-6},2^{-5},2^{-4},2^{-3},2^{-2},2^{-1},2^{0},2^{1},2^{2},2^{3}\}$, p (Parameters of RBF) $\in\{2^{-7},2^{-6},...,2^{7}\}$, $\tau\in${-1,-0.99,-0.98,...,0,0.01,0.02,...,1},λ_1, $\lambda_2\in\{2^{-7},2^{-6},...,2^{1}\}$. For 2-FM, the maximum number of iterations is set as 100, the minimum amount of improvement is set as 1e-5. The clustering process stops when the maximum number of iterations is reached, or when the objective function improvement between two consecutive iterations is less than the minimum amount of improvement specified. To evaluate the reliability of the model, we calculated the standard deviation of the accuracy by repeating the experiment ten times with varying proportions of training and prediction sets. The accuracy of each algorithm depends on the optimal combination of these

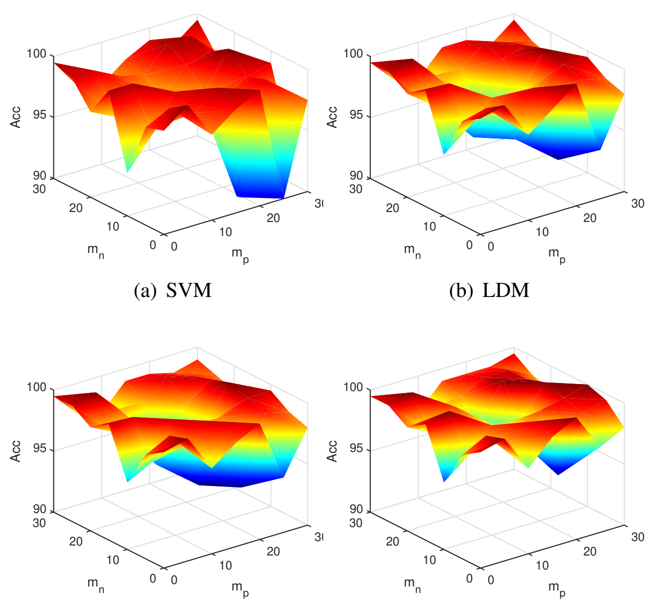
TABLE I
PARAMETER SETTINGS FOR SYNTHETIC DATASETS WITH GAUSSIAN NOISE.

Training Samples					
Positive/Negative Class	Distribution Gaussian distribution I/II	Probability 1	Mean [0.5,-3]/[-0.5,3]	Covariance [0.2, 0; 0, 3]	Number 100
Prediction samples					
Positive/Negative Class	Distribution Gaussian distribution III/IV	Probability 1	Mean [0.5,-2]/[-0.5,3]	Covariance [0.2, 0; 0, 3]	Number 100
Noises					
Positive Class	Distribution Gaussian distributionV/VI	Probability 1	Mean [0,-0.2]/[0,0.2]	Covariance [1, 0.8; 0.8, 1]	Number 0,5,10,15,20,25,30



(a) Dataset with 10 Noise (b) Dataset with 40 Noises (c) Dataset with 60 Noises

Fig. 5. Confidence plane diagram for different noise quantities.



(a) SVM (b) LDM (c) FLDM (d) FUPLDM

Fig. 6. Stability performance of the four algorithms on datasets with different amounts of noise added. (The horizontal and vertical coordinates represent the positive class and negative class noise, respectively, and the vertical coordinate represents the accuracy rate.)

parameters. All experiments in this section are performed on a computer with 8×4.00 GHz CPU and 32GB memory using MATLAB R2018a. For code and data, please visit the website: https://github.com/DDDDDDDH/FUPLDM.git.

TABLE II
DATASET PROPERTY DESCRIPTION.

Category	Datasets	#Samples	#Features
Biomedical	Patients	1631	78
	Heberman	306	3
	Australian	690	14
	Bupa_Liver	345	6
	Breast_Cancer	559	32
	WDBC	569	29
	BUPA	341	6
	Statlog	270	13
Algorithms	Monk 1	556	6
	Monk 2	601	6
	Monk 3	554	6
Text	Votes	435	16
	Creditcart	1565	31
Others	Aps_failure	1707	169
	Wine	178	13
	Echo	327	7
	Pima_Indian	768	8
	Daibetes	768	8
	Fertility	150	9

A. Experiments on an Artificial Dataset

To illustrate the performance of FUPLDM in dealing with noise near the decision boundary, we manually construct a 2-D dataset where samples come from two Gaussian distributions with equal probability: $x_i \ (i \in P) \sim N(\mu_1, \Sigma_1)$, $x_i \ (i \in N) \sim N(\mu_2, \Sigma_2)$ at first. Then, we verify the insensitivity of FUPLDM by adding different numbers of feature noise. The specific settings of training samples, prediction samples and Gaussian noise can be obtained from Table I.

TABLE III
ACCURACY COMPARISON OF THE NINE ALGORITHMS ON THE UCI DATASETS.

Dataset	SVM Accuracy	C	LDM Accuracy	C	FLDM Accuracy	C	Pin-SVM Accuracy	C	UPSVM Accuracy	C	FSVM Accuracy	C	FTSVM Accuracy	C	CDFTSVM Accuracy	C	FUPLDM Accuracy	C
Monk 1	64.35±0.62	0.06	68.06±0.54	0.06	67.59±0.46	0.50	66.67±0.22	0.03	67.13±0.25	0.25	65.05±0.45	2.00	63.89±0.46	0.02	63.66±0.35	2.00	**68.52±0.26**	0.50
Monk 2	67.13±0.14	0.03	67.36±0.15	0.03	67.13±0.16	0.03	67.13±0.15	0.03	67.13±0.21	0.03	67.13±0.24	0.03	46.53±0.11	4.00	67.13±0.28	8.00	**67.59±0.31**	8.00
Monk 3	81.02±0.46	0.06	84.03±0.48	0.13	81.94±0.35	0.03	84.49±0.28	1.00	**88.89±0.18**	0.50	81.02±0.25	0.50	77.08±0.34	0.25	81.48±0.28	0.25	**88.89±0.25**	8.00
Heberman	73.08±1.54	0.03	73.08±1.08	0.03	73.08±1.20	0.03	**73.72±1.00**	0.06	73.08±0.89	0.03	73.08±0.90	0.03	76.28±1.00	0.02	71.15±1.14	2.00	73.08±0.95	0.06
Statlog	86.67±0.67	0.06	85.00±0.56	0.03	85.83±0.58	0.03	86.67±0.46	0.06	86.67±0.26	0.06	85.83±0.62	0.03	80.00±0.75	4.00	**87.50±0.56**	1.00	86.67±0.35	0.25
Pima-Indian	67.09±0.19	0.03	79.91±0.20	2.00	79.70±0.31	4.00	69.87±0.29	1.00	75.21±0.15	0.50	67.09±0.19	4.00	74.79±0.14	0.02	74.79±0.21	0.02	**80.34±0.23**	1.00
Echo	70.59±0.44	0.03	88.24±0.40	0.25	90.20±0.32	0.25	84.31±0.46	0.25	90.20±0.22	0.03	70.59±0.41	0.25	88.24±0.26	4.00	88.24±0.27	8.00	**92.16±0.27**	0.25
Australian	84.48±0.29	0.03	86.55±0.54	0.03	85.86±0.45	0.13	84.48±0.13	0.03	**87.24±0.15**	0.03	84.83±0.26	0.13	81.72±0.28	0.50	81.45±0.31	0.06	**87.24±0.23**	0.03
Bupa	63.16±0.37	0.13	70.53±0.28	4.00	71.58±0.30	8.00	63.16±0.35	0.13	63.16±0.32	4.00	63.16±0.45	0.03	63.74±0.59	0.02	64.91±1.02	0.50	**72.63±0.95**	8.00
Daibetes	67.91±2.13	0.03	81.72±2.21	1.00	80.97±1.25	4.00	70.90±0.52	0.06	78.73±0.48	4.00	67.91±1.05	8.00	77.99±1.11	0.02	79.48±1.00	0.50	**83.21/-0.13**	1.00
Fertility	**94.00±1.16**	0.03	**94.00±1.25**	0.03	**94.00±0.80**	0.03	**94.00±0.64**	0.03	**94.00±0.84**	0.03	**94.00±1.20**	0.06	**94.00±1.24**	2.00	**94.00±0.75**	8.00	**94.00±0.85**	0.03
Original	98.43±0.54	0.13	98.43±0.42	0.03	98.43±0.22	0.03	98.43±0.25	0.13	**98.69±0.32**	0.50	98.59±0.64	0.25	98.59±0.84	0.02	98.59±0.84	0.02	**98.69±0.74**	0.25
BUPA	60±0.16	0.06	**75.86±0.25**	2.00	75.17±0.33	4.00	60.00±0.38	0.06	60.69±0.28	8.00	57.31±0.19	0.50	63.74±0.25	0.02	64.91±0.67	0.50	**75.86±0.54**	8.00
Wine	77.27±0.74	1.00	73.86±0.25	1.00	73.86±0.27	2.00	**88.64±0.56**	8.00	86.36±0.40	8.00	80.68±0.50	0.03	40.91±0.63	0.02	42.05±0.45	0.02	**88.64±0.33**	8.00
Votes	60.43±0.45	0.03	**95.74±0.84**	1.00	**95.74±0.65**	0.25	94.04±0.25	0.03	95.32±0.35	0.50	73.62±0.64	0.03	94.47±0.97	0.02	94.47±0.84	0.02	**95.74±0.54**	0.25
WDBC	77.51±0.87	0.03	98.82±0.88	0.06	98.82±0.57	0.50	91.12±0.66	0.50	98.22±0.87	0.50	79.29±0.64	0.50	93.49±0.50	0.02	94.08±0.45	0.03	**100.00±0.55**	2.00
Creditcard	93.13±1.26	0.50	96.56±1.24	4.00	96.20±0.16	8.00	94.36±1.26	0.13	93.74±1.65	0.02	93.74±0.97	0.03	98.05±0.67	0.02	**98.12±0.24**	0.02	96.20±0.16	8.00
Breast	98.12±0.65	1.00	**98.75±0.64**	0.50	98.43±0.22	1.00	98.12±0.41	1.00	98.12±1.24	1.00	96.55±1.02	0.02	88.86±0.71	0.02	93.15±0.65	0.02	98.43±0.15	1.00
Patients	**100±0.00**	0.02	74.25±0.25	0.02	83.51±0.65	0.02	**100±0.00**	0.02	**100±0.00**	2.00	**100±0.00**	0.03	**100±0.00**	0.02	**100±0.00**	0.02	**100±0.00**	0.02
Aps_failure	93.5±0.16	0.02	**98.35±0.24**	0.13	**98.35±0.35**	0.25	93.50±0.41	1.00	95.92±0.65	4.00	96.69±0.48	8.00	94.00±0.47	0.02	95.00±1.24	0.02	**98.35±0.21**	0.25
ACC.Average	78.89		84.45		84.82		83.18		84.93		79.81		79.82		81.71		**87.31**	

* The best result is shown in bold (similarly hereinafter). 'Acc.' denotes the classification accuracy (similarly hereinafter).

After that, we define the confidence plane as the plane where the samples with fuzzy membership degrees lie. Fig. 5 shows the confidence planes with different amounts of feature noise added. We can find that the color around the added noise is darker, which indicates that FUPLDM actively identifies these noise samples and gives them lower membership values. Therefore, FUPLDM has a large positive effect on the noise prevention in the training process of the classifier.

Besides, we compare the performance of SVM, LDM, F-LDM and FUPLDM by adding noise to the artificial dataset. FUPLDM maintains the highest accuracy on all conditions. Moreover, as the number of noise points increases, its accuracy does not drop very much, which is more robust than the other three algorithms.

In order to visually observe the anti-noise performance of FUPLDM, we did a series of experiments by adding different numbers of positive and negative noise points respectively. The experimental results of SVM, LDM, FLDM and FUPLDM are shown in Fig. 6. The x and y axes represent the number of noise samples in the constructed positive and negative class, respectively, and the z axis is the accuracy of the algorithm at this setting of noise. The accuracy surfaces of SVM, LDM and F-LDM jittered significantly after adding noise. In contrast, the accuracy of FUPLDM keeps a smoother surface, whether increasing the number of single-type noise points or both types together.

B. Experiments on UCI Datasets

In this subsection, we conducted a series of experiments on the University of California Irvine UCI [42] repository of machine learning datasets, and selected datasets with a different number of sample features. In the "Monk1", "Monk2", "Monk3" and "Spect" data, UCI provides training and test sets. For the other parts, we randomly split the data into two parts, one for training (50% - 60%) and the other for testing. The concrete settings are tabulated in Table II. The experiment was repeated 10 times on each data set, and took the average value as the recorded result.

First, we compare the performance of FUPLDM with SVM, Pin-SVM, UPSVM, LDM, FLDM, FSVM, FTSVM and CDFTSVM in terms of classification accuracy. By using the quadratic programming function embedded in MATLAB, the optimization problem of FUPLDM and other algorithms can be effectively solved. The experimental results and the optimal value of hyperparameter C in the above datasets can be seen in Table III. As we can see, FUPLDM obtains the highest accuracy on most data sets. Specially, FSVM (79.81% accuracy) outperforms SVM (78.89% accuracy), and FLDM (84.82% accuracy) surpasses LDM (84.45% accuracy), highlighting the importance of fuzzy membership in describing potential sample contamination. Moreover, incorporating the pinball loss function leads to superior performance compared to not utilizing it. UPSVM achieves 84.93% accuracy while SVM achieves 78.89%. Similarly, FUPLDM achieves 87.31% accuracy while FLDM achieves 84.82%. This observation suggests that the pinball loss function exhibits better noise resistance than the hinge loss function.

Second, the proposed FUPLDM score ranks first and is much higher than the other algorithms, which demonstrates the high performance of FUPLDM (Refer to Supporting document). To fully illustrate this gap, the percentage of each algorithm's score to the total score is calculated. It can be found that SVM has the lowest percentage of 9.09%. Pin-SVM, UPSVM and LDM all have higher percentages, the highest of which is LDM with 21.62%, while the proposed FUPLDM has a percentage of 32.82%. It outperformed the second-ranked algorithm by over 50%, which is a great improvement compared to the other five algorithms. Therefore, FUPLDM achieves good comprehensive performance on the datasets.

It's worth noting that the average classification accuracy of all algorithms falls within the expected range. However, in the case of LDM, there is a noticeable difference in performance on the Patients and Wine datasets compared to other algorithms. To delve deeper into this phenomenon, we first calculated the imbalance levels between positive and negative classes in each dataset, followed by applying the t-SNE method to perform dimensionality reduction on these datasets. t-SNE effectively maps high-dimensional data to a lower-dimensional space while preserving the local similarity structure among data points [43].

Analysis of the results reveals the following insights. For the Patients dataset, with 255 positive class samples and 1376 negative class samples, data imbalance is the primary factor affecting LDM's classification performance. LDM, during training, emphasizes maximizing the marginal mean, which causes the classification boundary to shift closer to the minority class samples. In contrast, SVMs only consider support vector information and are less influenced by the distribution of minority or majority class samples. As for our proposed FUPLDM algorithm, we introduced a balancing weight factor into the objective function, effectively addressing the issue of data imbalance.

For the Wine dataset, although a minor data imbalance is present, the predominant factor influencing the classification performance of the algorithms is the inherent data distribution. Analysis of the results reveals that data points in the Wine dataset are distributed on or near the same plane, with positive and negative classes on the left and right sides of the plane, respectively. In such data scenarios, LDM, during training, focuses on minimizing marginal variance, resulting in a very small angle between the classification hyperplane and the plane where the data points are located. Conversely, SVMs primarily consider boundary information and are less influenced by the sample distribution. Our FUPLDM algorithm, however, adopts the UP loss function, which computes quantile distances instead of the shortest distances, while optimizing both global distance information and loss information. This not only leads to outstanding generalization performance but also effectively alleviates the issues arising from sample distribution.

In order to statistically illustrate the effectiveness of our algorithm, the following statistical analysis was done:

1) Statistical analysis based on Friedman test theory: The Friedman test measures the differential performance of an algorithm on multiple datasets [44], [45]. According to this method, each algorithm is assigned an initial ranking by its classification accuracy, with the best performing algorithm ranked 1 and the rest in increasing order. This indicates that the lower the rank of the classifier, the better is its performance.

Assuming that on the i^{th} dataset, r_i^j is used to denote the rank of the j^{th} classifier, the Friedman statistic is $\chi_F^2 = \frac{12N}{k(k+1)}\left[\sum_j \bar{R}_j^2 - \frac{k(k+1)^2}{4}\right]$, where N and k represent the number of datasets and the number of binary classifiers, respectively. When N and k are large enough, it follows the χ_F^2 distribution with $(k-1)$ degrees of freedom. Since the Friedman statistic is conservative, one can use $F_F = \frac{(N-1)\chi_F^2}{N(k-1)-\chi_F^2}$ which follows $F_F((k-1),(k-1)(N-1))$. By this method, the effectiveness of the binary classifier can be evaluated and the significant difference between our algorithm and other methods can be obtained. Based on the rankings calculated from Table III (see the supporting documentation for details), brought to $N = 20$ and $k = 9$, we can calculate $\chi_F^2 = 63.2003$ and $F_F = 12.4050$. The results of the calculations are shown in Table.V-B. When $\alpha = 0.05$, the critical value $F_{\alpha=0.05}(8,152) = 2.00$, which shows the effectiveness of our classifier and demonstrates the significant difference between our algorithm and others.

TABLE IV
RESULTS OF FRIEDMAN TEST.

Dataset	χ_F^2	F_F	$F_{\alpha=0.05}(8,152)$
20 UCI	63.2003	12.4050	2.00

C. Experiments on UCI Datasets with Noise

To further validate the effectiveness and noise resistance of FUPLDM, and to conduct a comparative performance analysis among SVM, Pin-SVM, UPSVM, LDM, F-LDM, and FUPLDM, Gaussian white noise is added to the UCI datasets such that each feature of the sample is corrupted by the same distribution of white noise. For simplicity, we set the mean value to 0 and the feature noise variance to r = 0.05, 0.1 and 0.5, respectively. In order to clearly demonstrate the advantages of the FUPLDM algorithm, we calculated the accuracy gain (AG) of FUPLDM over other algorithms as follows:

$$AG = \frac{\Delta Acc}{Acc}, \tag{43}$$

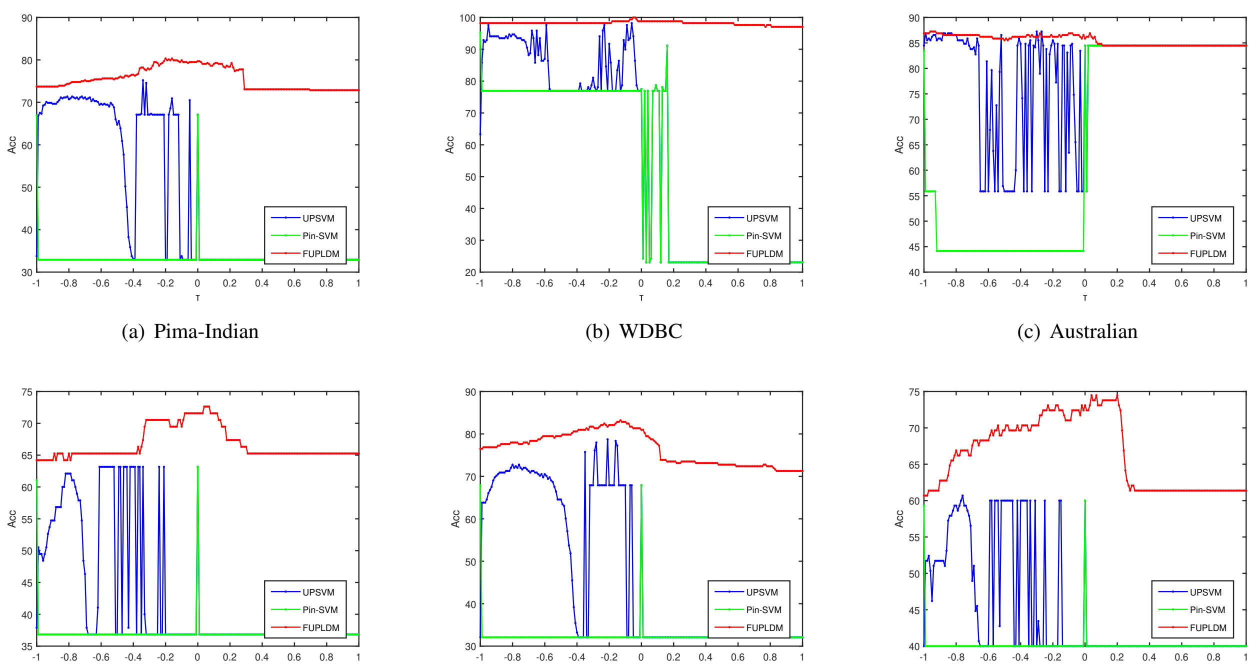
where ΔAcc represents the difference of FUPLDM with respect to the accuracy of the current algorithm and Acc represents the accuracy of the current algorithm. Specifically, we calculated the ratio of the accuracy difference between FUPLDM and the comparison algorithm to the accuracy of the comparison algorithm.

The experimental results of the UCI datasets with noise are shown in Tables V. To clearly reflect the advantages of the algorithm, we use bold numbers to indicate the highest classification accuracy, the lowest training time and the highest AG in each dataset. It can be seen that in most cases, FUPLDM has better classification accuracy when the datasets are corrupted by different degrees of noise. This is because FUPLDM takes into account the distribution of sample points for crossover datasets, which improves the generalization performance. As the proportion of noise increases, the accuracy of LDMs (LDM, F-LDM and FUPLDM) is higher than that of SVMs (UPSVM, Pin-SVM and SVM) in most cases. It indicates that the mechanism of LDM can effectively eliminate the interference of noisy samples and improve the classification accuracy. In addition, the classifier with pinball loss function (Pin-SVM, UPSVM, FUPLDM) has higher classification accuracy than hinge loss (SVM, LDM, F-LDM), which shows that the UP loss function can effectively improve the noise resistance of the model. In general, FUPLDM achieves better anti-noise performance than all other algorithms.

TABLE V
ACCURACY COMPARISON OF SVM, PIN-SVM, UPSVM, LDM, F-LDM AND FUPLDM ON THE NOISY UCI DATASET.

Dataset	SVM Accuracy/τ Time/C	AG	Pin-SVM Accuracy/τ Time/C	AG	UPSVM Accuracy/τ Time/C	AG	LDM Accuracy/τ Time/C	AG	F-LDM Accuracy/τ Time/C	AG	FUPLDM Accuracy/τ Time/C
Monk 1(r=0.05)	57.87/0 1.16/0.25	**10.80%**	62.04/0.07 1.31/0.06	3.35%	60.65/-0.75 0.15/0.25	5.72%	61.11/0 **0.04/4.00**	4.93%	63.43/0 **0.04/2.00**	1.09%	**64.12/-0.37** **0.04/4.00**
r=0.1	57.64/0 **0.87/0.25**	**12.04%**	59.95/-0.29 0.98/0.06	7.72%	**65.05/-0.34** 0.98/4.00	-0.72%	61.11/0 0.89/4.00	5.68%	63.89/0 0.98/8.00	1.08%	64.58/-0.29 **0.87/0.50**
r=0.5	57.64/0 0.99/0.06	**12.04%**	58.33/0.06 1.09/0.06	10.71%	**65.28/-0.33** 1.09/4.00	-1.07%	61.34/0 0.04/2.00	5.28%	64.35/0 **0.02/2.00**	0.36%	64.58/-0.24 0.03/4.00
Monk 2(r=0.05)	**67.13/0** 1.60/0.02	0.00%	**67.13/-0.99** 1.81/0.02	0.00%	**67.13/-0.71** 0.23/0.02	0.00%	**67.13/0** **0.05/0.02**	0.00%	**67.13/0** **0.05/0.02**	0.00%	**67.13/-1** **0.05/0.02**
r=0.1	**67.13/0** 1.27/0.03	0.00%	**67.13/-1** 1.45/0.03	0.00%	**67.13/-0.49** 1.40/0.03	0.00%	**67.13/0** 1.28/0.03	0.00%	**67.13/0** 1.28/0.03	0.00%	**67.13/-1** **1.26/0.03**
r=0.5	**67.13/0** 1.23/0.03	0.00%	**67.13/-0.99** 1.39/0.03	0.00%	**67.13/-0.49** 1.34/0.03	0.00%	**67.13/0** **0.04/0.03**	0.00%	**67.13/0** 0.05/0.03	0.00%	**67.13/-1** **0.04/0.03**
Pima-Indian(r=0.05)	67.09/0 3.13/0.02	**17.84%**	69.87/-1 3.97/0.13	13.15%	67.09/-0.37 0.58/0.02	**17.84%**	78.85/0 0.15/4.00	0.27%	78.85/0 **0.13/2.00**	0.27%	**79.06/-0.14** **0.13/1.00**
r=0.1	67.09/0 2.54/0.03	**17.84%**	67.09/-0.21 3.27/0.03	**17.84%**	76.71/-0.28 3.30/2.00	3.06%	78.85/0 **2.49/8.00**	0.27%	**79.06/0** 2.50/4.00	0.00%	**79.06/-0.21** 2.50/1.00
r=0.5	67.09/0 2.53/0.03	**18.15%**	69.87/-1 3.09/0.03	13.45%	74.79/-0.2 3.08/4.00	5.99%	79.06/0 0.08/4.00	0.27%	79.06/0 0.09/4.00	0.27%	**79.27/0.01** **0.07/4.00**
WDBC(r=0.05)	91.72/0 2.17/0.02	**5.80%**	91.72/-1 4.64/0.03	**5.80%**	95.86/-0.62 0.85/0.02	1.23%	96.45/0 **0.30/8.00**	0.61%	95.86/0 **0.30/0.02**	1.23%	**97.04/0.32** 0.33/8.00
r=0.1	90.53/0 **1.67/0.03**	**7.19%**	95.86/0.32 4.19/1.00	1.23%	96.45/-0.79 4.09/2.00	0.61%	96.45/0 3.69/8.00	0.61%	95.86/0 3.69/0.03	1.23%	**97.04/0.32** 3.69/8.00
r=0.5	89.94/0 1.55/0.03	**7.89%**	94.08/-1 3.55/0.13	3.15%	96.45/-0.78 3.59/4.00	0.61%	96.45/0 0.19/8.00	0.61%	95.86/0 **0.16/0.03**	1.23%	**97.04/0.39** 0.19/8.00
Echo(r=0.05)	76.47/0 0.12/0.02	**25.64%**	90.2/0.05 0.18/0.02	6.52%	90.2/-0.29 0.03/0.02	6.52%	92.16/0 **0.02/0.02**	4.25%	94.12/0 **0.02/0.50**	2.08%	**96.08/-0.11** **0.02/2.00**
r=0.1	70.59/0 **0.09/0.03**	**36.11%**	90.2/-0.11 0.15/2.00	6.52%	92.16/-0.32 0.14/0.50	4.25%	94.12/0 0.20/0.06	2.08%	94.12/0 0.20/0.13	2.08%	**96.08/-0.11** 0.20/4.00
r=0.5	70.59/0 0.09/0.03	**36.11%**	94.12/0.09 0.13/2.00	2.08%	94.12/0.09 0.13/2.00	2.08%	94.12/0 **0.01/0.06**	2.08%	94.12/0 **0.01/0.13**	2.08%	**96.08/-0.11** **0.01/4.00**
Average of AG		13.83%		6.10%		3.08%		1.80%		0.87%	-

* The traversal range of τ is [-1:0.01:1]. 'Time' denotes the CPU time consumption of the algorithm.(For more details, please refer to: https://github.com/DDDDDDDH/FUPLDM.git)



(a) Pima-Indian (b) WDBC (c) Australian

(d) Bupa-Liver (e) Daibetes (f) BUPA

Fig. 7. The influence of τ on the accuracy of Pin-SVM, UPSVM and FUPLDM algorithms respectively. (For more details, please refer to: https://github.com/DDDDDDDH/FUPLDM.git)

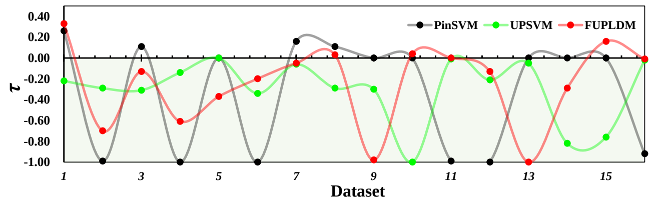


Fig. 8. The waveform of parameter τ. The dataset number is plotted on the horizontal axis, while the vertical coordinate represents the τ value at which the algorithm achieves the highest accuracy within a given dataset.

D. Parameter Sensitivity Analysis

To investigate the stability of the proposed FUPLDM, parameter sensitivity tests are performed. Although Pin-SVM, UPSVM and FUPLDM have many parameters, the core parameter is τ, which is investigated in this section. We compare the performance of the three algorithms when the value of τ ranges from 0 to 1 with a step size of 0.01.

Fig. 7 shows the variation of the accuracy as the value of τ changes ($r = 0$). To reduce randomness, we list the performance of these algorithms on 15 datasets. From Fig. 7, we can obtain the following observations.

1) Accuracy: The proposed FUPLDM obtains the highest accuracy in almost all the datasets. Specifically, FUPLDM achieves the best performance for most τ values.

2) Stability: As the value of τ changes, the proposed FUPLDM exhibits the best stability in all the datasets. In the process, the accuracy of Pin-SVM and UPSVM fluctuates greatly, while FUPLDM changes smoothly in most cases. UPSVM is unstable throughout the interval, while Pin-SVM varies dramatically when $\tau > 0$. Moreover, it can be found that the curves of Pin-SVM and UPSVM coincide when $\tau > 0$, which shows the correctness of the reformulated UP loss.

3) Application: The fluctuation curve of the parameter τ is plotted as shown in Fig. 8. As evident from the results, in a majority of cases, τ attains negative values at the point where the algorithm achieves optimal accuracy, which indicates that the gain for the correctly identified samples is effective. This discovery lays the foundation for further exploration and expansion of our research.

According to all the above experimental analysis, we can draw the following conclusions.

- Due to the improved loss function, UPSVM outperforms PinSVM and SVM, but FUPLDM outperforms all the algorithms. (See Section V-B for details)
- Integrating the 2-FM function proposed in this paper, FUPLDM demonstrates superior capability in identifying feature noise and reducing its impact on the classification hyperplane, consequently achieving better noise immunity performance.(See Section V-C for details)
- As FUPLDM applies a higher penalty factor to small class samples, it enhances the significance of these samples and partially addresses the issue of unbalanced classification tasks.
- FUPLDM has better stability and higher classification accuracy than other algorithms in terms of the key parameter τ, which reflects the robustness of our proposed model. (See Section V-D for details)
- FUPLDM provides some mitigation for unbalanced datasets, but further enhancements can still be made. (See Section Supporting Document for details)
- While FUPLDM does not introduce additional time complexity compared to LDM, it still exhibits a relatively high level, indicating the necessity for further improvement. (See Supporting Document for details)

VI. Conclusion

In this paper, a novel FUPLDM algorithm is constructed to improve the noise immunity and generalization performance. To the best of our knowledge, this is the first time that a combination of the UP loss and the fuzzy affiliation is used to improve the performance of LDM. The properties of ideal classifiers, including noise insensitivity, maximization of inter-class distance and minimization of intra-class scattering, have also been investigated theoretically. In addition, the validity and reliability of our proposed FUPLDM are also demonstrated by numerical experiments. The performance of the proposed FUPLDM is evaluated on both an artificial dataset and 20 benchmark UCI datasets. In order to establish a thorough benchmark, we compare FUPLDM against eight related algorithms, namely SVM, FSVM, FLDM, PinSVM, UPSVM, FTSVM, and CDFWVM. The results obtained reveal the exceptional performance of FUPLDM, surpassing the competing algorithms in terms of effectiveness, noise immunity and accuracy. FUPLDM also performs the most outstandingly on the noise-added UCI datasets. While FUPLDM exhibits notable advancements in binary classification scenarios, directly extending its application to multiclassification tasks does not yield substantial performance advantages.

Our future research endeavors will be directed towards exploring innovative approaches specifically tailored for multi-classification scenarios. It is also intriguing to explore the improvements in unbalanced categorization, such as constructing adaptive equilibrium functions, etc.

References

[1] J. Zhao, Y. T. Xu, and H. Fujita, "An improved non-parallel universum support vector machine and its safe sample screening rule," *Knowledge-Based Systems*, vol. 170, pp. 79–88, 2019.

[2] P. Saigal, S. Chandra, and R. Rastogi, "Multi-category ternion support vector machine," *Engineering Applications of Artificial Intelligence*, vol. 85, pp. 229–242, 2019.

[3] M. Avolio and A. Fuduli, "A semiproximal support vector machine approach for binary multiple instance learning," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 32, no. 8, pp. 3566–3577, 2020.

[4] L. Reyzin and R. E. Schapire, "How boosting the margin can also boost classifier complexity," in *Proceedings of the 23rd International Conference on Machine Learning*, 2006, pp. 753–760.

[5] T. Zhang and Z. H. Zhou, "Optimal margin distribution clustering," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 32, no. 1, 2018, pp. 4474–4481.

[6] L. Liu, M. X. Chu, R. F. Gong, and L. Zhang, "An improved nonparallel support vector machine," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 32, no. 11, pp. 5129–5143, 2020.

[7] W. Gao and Z. H. Zhou, "On the doubt about margin explanation of boosting," *Artificial Intelligence*, vol. 203, pp. 1–18, 2013.

[8] L. W. Wang, M. Sugiyama, Z. X. Jing, C. Yang, Z. H. Zhou, and J. F. Feng, "A refined margin analysis for boosting algorithms via equilibrium margin," *The Journal of Machine Learning Research*, vol. 12, pp. 1835–1863, 2011.

[9] T. Zhang and Z. H. Zhou, "Large margin distribution machine," in *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2014, pp. 313–322.

[10] V. Vapnik, *The nature of statistical learning theory*. Springer Science & Business Media, 1999.

[11] F. Cheng, J. Zhang, and C. H. Wen, "Cost-sensitive large margin distribution machine for classification of imbalanced data," *Pattern Recognition Letters*, vol. 80, pp. 107–112, 2016.

[12] S. Abe, "Unconstrained large margin distribution machines," *Pattern Recognition Letters*, vol. 98, pp. 96–102, 2017.

[13] R. Khemchandani, S. Chandra *et al.*, "Twin support vector machines for pattern classification," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 5, pp. 905–910, 2007.

[14] X. L. Huang, L. Shi, and J. A. Suykens, "Sequential minimal optimization for svm with pinball loss," *Neurocomputing*, vol. 149, pp. 1596–1603, 2015.

[15] Y. T. Xu, Z. J. Yang, and X. L. 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, 2016.

[16] X. L. Huang, L. Shi, and J. A. Suykens, "Support vector machine classifier with pinball loss," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 5, pp. 984–997, 2013.

[17] X. L. Huang, L. Shi, and J. A. Suykens, "Solution path for pin-svm classifiers with positive and negative τ values," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 28, no. 7, pp. 1584–1593, 2016.

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

[19] W. X. Zhu, Y. Y. Song, and Y. Y. Xiao, "Support vector machine classifier with huberized pinball loss," *Engineering Applications of Artificial Intelligence*, vol. 91, 2020, 103635.

[20] P. Anand, R. Rastogi, and S. Chandra, "Improvement over pinball loss support vector machine," *arXiv preprint arXiv:2106.01109*, 2021.

[21] X. Zhang, C. Mei, D. Chen, and J. Li, "Feature selection in mixed data: A method using a novel fuzzy rough set-based information entropy," *Pattern Recognition*, vol. 56, pp. 1–15, 2016.

[22] Q. Fan, Z. Wang, D. Li, D. Gao, and H. Zha, "Entropy-based fuzzy support vector machine for imbalanced datasets," *Knowledge-Based Systems*, vol. 115, pp. 87–99, 2017.

[23] S. Rezvani, X. Z. Wang, and F. Pourpanah, "Intuitionistic fuzzy twin support vector machines," *IEEE Transactions on Fuzzy Systems*, vol. 27, no. 11, pp. 2140–2151, 2019.

[24] M. N. Ahmed, S. M. Yamany, N. Mohamed, A. A. Farag, and T. Moriarty, "A modified fuzzy c-means algorithm for bias field estimation and segmentation of mri data," *IEEE Transactions on Medical Imaging*, vol. 21, no. 3, pp. 193–199, 2002.

[25] X. W. Yang, G. Q. Zhang, J. Lu, and J. Ma, "A kernel fuzzy c-means clustering-based fuzzy support vector machine algorithm for classification problems with outliers or noises," *IEEE Transactions on Fuzzy Systems*, vol. 19, no. 1, pp. 105–115, 2010.

[26] C. Zhang, H. X. Li, Y. Gao, and C. L. Chen, "Weakly-supervised enhanced semantic-aware hashing for cross-modal retrieval," *IEEE Transactions on Knowledge and Data Engineering*, 2022.

[27] D. Liu, "The effectiveness of three-way classification with interpretable perspective," *Information Sciences*, vol. 567, pp. 237–255, 2021.

[28] L. Zhang, Q. Jin, S. Fan, and D. Liu, "A novel dual-center based intuitionistic fuzzy twin bounded large margin distribution machines," *IEEE Transactions on Fuzzy Systems*, pp. 1–14, 2023.

[29] D. c. Liang, D. Liu, W. Pedrycz, and P. Hu, "Triangular fuzzy decision-theoretic rough sets," *International Journal of Approximate Reasoning*, vol. 54, no. 8, pp. 1087–1106, 2013.

[30] L. Zhang, Q. Jin, S. Fan, and D. Liu, "A novel dual-center based intuitionistic fuzzy twin bounded large margin distribution machines," *IEEE Transactions on Fuzzy Systems*, 2023.

[31] Z. N. Wu, H. G. Zhang, and J. H. Liu, "A fuzzy support vector machine algorithm for classification based on a novel pim fuzzy clustering method," *Neurocomputing*, vol. 125, pp. 119–124, 2014.

[32] L. A. Zadeh, "Fuzzy logic," *Computer*, vol. 21, no. 4, pp. 83–93, 1988.

[33] L. Sun, L. Wang, W. Ding, Y. Qian, and J. Xu, "Feature selection using fuzzy neighborhood entropy-based uncertainty measures for fuzzy neighborhood multigranulation rough sets," *IEEE Transactions on Fuzzy Systems*, vol. 29, no. 1, pp. 19–33, 2021.

[34] T. Wang, H. Li, Y. Qian, B. Huang, and X. Zhou, "A regret-based three-way decision model under interval type-2 fuzzy environment," *IEEE Transactions on Fuzzy Systems*, vol. 30, no. 1, pp. 175–189, 2022.

[35] J. C. Bezdek, R. Ehrlich, and W. Full, "FCM: The fuzzy c-means clustering algorithm," *Computers & Geosciences*, vol. 10, no. 2-3, pp. 191–203, 1984.

[36] J. X. Zhu, X. L. Ma, L. Martínez, and J. Zhan, "A probabilistic linguistic three-way decision method with regret theory via fuzzy C-means clustering algorithm," *IEEE Transactions on Fuzzy Systems*, 2023.

[37] W. Ding, S. Chakraborty, K. Mali, S. Chatterjee, J. Nayak, A. K. Das, and S. Banerjee, "An unsupervised fuzzy clustering approach for early screening of covid-19 from radiological images," *IEEE Transactions on Fuzzy Systems*, vol. 30, no. 8, pp. 2902–2914, 2022.

[38] L. Hu, X. Pan, Z. Tang, and X. Luo, "A fast fuzzy clustering algorithm for complex networks via a generalized momentum method," *IEEE Transactions on Fuzzy Systems*, vol. 30, no. 9, pp. 3473–3485, 2022.

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

[40] C. Memo, "T. poggio, s. mukherjee, r. rifkin, a. rakhlin and a. verri," 2001.

[41] D. Gupta, P. Borah, and M. Prasad, "A fuzzy based lagrangian twin parametric-margin support vector machine (fltpmsvm)," in *2017 IEEE symposium series on computational intelligence (SSCI)*. IEEE, 2017, pp. 1–7.

[42] A. Asuncion and D. Newman, "UCI Machine Learning Repository," 2007.

[43] L. Van der Maaten and G. Hinton, "Visualizing data using t-sne." *Journal of machine learning research*, vol. 9, no. 11, pp. 2579–2605, 2008.

[44] J. Demšar, "Statistical comparisons of classifiers over multiple data sets," *The Journal of Machine learning research*, vol. 7, pp. 1–30, 2006.

[45] M. Ganaie, M. Tanveer, and C.-T. Lin, "Large-scale fuzzy least squares twin svms for class imbalance learning," *IEEE Transactions on Fuzzy Systems*, vol. 30, no. 11, pp. 4815–4827, 2022.

Libo Zhang (Member, IEEE) received the B.S. degree in applied mathematics in Zhejiang University of Technology in 2013, and the Ph.D. degree in Management Science and Engineering from Nanjing University, Nanjing, China, in 2019.

From 2017-2018, he was a visiting Ph.D. student in the Australian National University. He has been with the College of Artificial Intelligence, Southwest University, Chongqing, since 2019. His research interests include fuzzy set theory, pattern recognition, deep learning, and three-way decision theory.

Denghao Dong (Student Member, IEEE) is currently pursuing the bachelor's degree with the College of Artificial Intelligence, Southwest University, Chongqing, China.

His research interests include fuzzy set theory, pattern recognition, deep learning, and three-way decision theory.

Lianyi Luo is currently pursuing the undergraduate degree with the College of Artificial Intelligence, Southwest University, Chongqing, China.

Her research interests include fuzzy set theory, pattern recognition, deep learning, and three-way decision theory.

Dun Liu (Member, IEEE) received the B.S. degree in mathematics and Ph.D. degree in management science and engineering from Southwest Jiaotong University, Chengdu, China, in 2005 and 2011, respectively.

From 2012 to 2014, he was a Postdoctoral Researcher with the School of Economics and Management, Tsinghua University, Beijing, China. In 2009, he was also a Visiting Scholar with the University of Regina, Regina, SK, Canada, and Carnegie Mellon University, Pittsburgh, PA, USA, in 2016. He is currently a Professor with the School of Economics and Management, Southwest Jiaotong University. Since 2005, he has been an Author of more than 140 research papers in refereed journals and conferences. His research interests include Uncertainty modeling and applications, three-way decisions, granular computing, information management, and decision support system.

Dr. Liu is a Member of ACM, and a Senior Member of IRSS and China Computer Federation.