Iranian Journal of Fuzzy Systems

Volume 18, Number 2, (2021), pp. 93-107

Original paper

A novel interval-valued fuzzy multiple twin support vector machine

H. Ju[1], W. Qiang[2] and L. Jing[3]

[1,3] *College of Science, China Agricultural University, Beijing, 100083, China*
[1] *School of Information, Beijing WUZI University, Beijing, 101149, China*
[2] *Institute of Software Chinese Academy of Sciences, Beijing, 100190, China*

juhongmei@bwu.edu.cn, a01114115@163.com, jingling@cau.edu.cn

Abstract

Multiple twin support vector machine (MTSVM) which evaluates all the training data into a "one-versus-rest" structure is a multi-class classification algorithm. It has extensive applications in the multi-class classification problems. Like twin support vector machine (TSVM), MTSVM treats all sample points equally because it lacks the ability to judge the importance of different sample points. In order to improve the classification performance of MTSVM, a new method of adding interval-valued fuzzy membership degree to sample points is proposed. In this way, a novel interval-valued fuzzy multiple twin support vector machine (IVF-MTSVM) is established in this paper. Previous methods of adding fuzzy membership degree to sample points are totally based on their importance to the class, while the method in this paper emphasizes the importance of sample points to the classification model, and takes into account the importance to the class to some extent. This is a new perspective to establish fuzzy membership degree to sample points in support vector machines since it is different from the previous methods in thinking. Then the solution to IVF-MTSVM is derived. Experiments on UCI datasets show that this new method has certain advantages over other multi-class twin support vector machine methods in "one-versus-rest" structure and other fuzzy multiple twin support vector machine established by some previous methods. Finally, Friedman test and Benferroni-Dunn test are used to verify the statistical significance of this new method.

Keywords: Multi-class classification problem, multiple twin support vector machine, "one-versus-rest" structure, interval-valued fuzzy membership degree.

1 Introduction

SVM is a computationally powerful tool for pattern classification and has drawn much attention in recent years [6, 9, 26, 31]. Because of its many advantages, it quickly becomes the research hotspot for researchers and has been successfully applied in many fields [3, 18, 23]. After SVM, other parallel support vector machines are introduced. For the least squares support vector machine (LSSVM) [28], least square function with equality constraints is used as a loss function instead of the complex QP problem of SVM, so the solving speed is relatively fast. For the -support vector machine (SVM) [27], a parameter ν lets one effectively control the number of support vectors. In recent years, nonparallel classifiers which are different from those searching for two parallel support hyperplanes have been proposed. The nonparallel hyperplanes for each class can describe the distribution of different classes, thus applicable to wider problems. For the generalized eigenvalue proximal support vector machine (GEPSVM) [22], data points of each class are proximal to one ofthe two nonparallel planes. The nonparallel planes are eigenvectors corresponding to the smallest eigenvalues of two related generalized eigenvalue problems. For the twin support vector machine (TWSVM) [14], it seeks two nonparallel proximal hyperplanes such that each hyperplane is closer to one of the two classes and is at least one distance from the other. The strategy results that TWSVM solves two smaller QPPs, whereas SVM solves one larger QPP, which increases the TWSVM training speed by approximately four-fold compared to that of SVM. As an

Corresponding Author: L. Jing
Received: October 2019; Revised: August 2020; Accepted: October 2020.

important extension of SVM, TWSVM is welcomed due to its superior performance and is widely used in many fields [4, 7, 16].

As is known to all, SVM and these extended classification models mentioned above treat all sample points equally. But in fact, different sample points have different effects on the classification due to their difference in distribution. If we want to give a quantitative analysis of these differences and attach different values to different sample points to achieve better classification results, we need to use the tool of fuzzy mathematics. The combination of fuzzy mathematics and various extended support vector machines has produced many new classification models, such as fuzzy support vector machine [19], fuzzy least squares support vector machine [30], fuzzy proximal support vector classification via generalized eigenvalues [13], fuzzy twin support vector machine [12], intuitionistic fuzzy twin support vector machines [25], etc. How to determine the membership degree of these samples is the key point. Due to the higher classification accuracy of these extended models, they have been used in many fields [5, 15, 20, 32, 38].

In the real world, many pattern recognition problems are multi-class classification problems. For example, webpage data classification [35], hyperspectral images classification [2, 24] and sentiment classification [21] are all very popular multi-class classification problems. In the TWSVM family framework for k-class classification problem, "one-versus-one" [1], "one- versus-rest" [10, 29, 33] and "one-versus-one-versus-rest" approaches [34] are usually used to resolve it. "One-versus-one" structure needs to construct $k(k-1)/2$ binary classifications where each classifier is involved with the training data of two classes. In this case, the information of the remaining samples is omitted in each binary classification so that unfavorable results may be received. "One-versus-one-versus-rest" structure constructs $k(k-1)/2$ classifiers with output $\{-1, 0, 1\}$ and all training data is utilized in constricting the decision function. In this case, its computational complexity is higher. Compared with the "one-versus-one" structure, all the sample points in "one-versus-rest" structure participate in each classification training, so the decision-making process is more reasonable. Compared with the "one-versus-one-versus-rest" structure, the training complexity of "one-versus-rest" structure is relatively low, especially when the number of classes is greater than three.

Based on the above background, this paper proposes a novel multi-class classification method, which is called interval-valued fuzzy multiple twin support vector machine (IVF-MTSVM). It's the combination of interval-valued fuzzy set theory (IVFS) and MTSVM. The method of adding interval-valued membership degree to sample points in MTSVM is different from previous methods in thinking. It emphasizes the importance of sample points to classification model, and also takes into account the importance of sample points to class. Then the solution to IVF-MTSVM is derived. Experiments on UCI datasets show that this new method has certain advantages over other multi-class twin support vector machine methods in "one-versus-rest" structure and other fuzzy multiple twin support vector machine established by some previous methods. Finally, Friedman test and Benferroni-Dunn test are used to verify the statistical significance of this new method.

The rest of this paper is organized as follows. Section 2 is the preliminaries. In section 3, interval-valued fuzzy multiple twin support vector machine is presented in detail. Experimental results and discussions are given in section 4. The last section is the conclusion.

2 Preliminaries

In this section, fuzzy related concepts and multiple twin support vector machine are introduced respectively.

2.1 Fuzzy related concepts

Definition 2.1. [36] *Let X be a non-empty set, $F = \{(x, \mu_F(x))|x \in X\}$ is called a fuzzy set (FS) in X. $\mu_F : X \to [0, 1]$ is called the membership degree function of F, $\mu_F(x)$ represents the membership degree of x belonging to F, and $0 \le \mu_F(x) \le 1$.*

For the classical set, the membership degree of x is either 0 or 1. For the fuzzy set, the membership degree of x is a number in $[0, 1]$. The classical set is a special case of fuzzy set, and fuzzy set is an extension of the classical set.

Definition 2.2. [37] *Let X be a non-empty set, $F = \{(x, [\mu_{F^-}(x), \mu_{F^+}(x)])|x \in X\}$ is called an interval-valued fuzzy set in X. $\mu_{F^-}, \mu_{F^+} : X \to [0, 1]$ are called the interval-valued membership degree functions of F, $[\mu_{F^-}(x), \mu_{F^+}(x)]$ represents the interval of membership degree of x belonging to F, and $0 \le \mu_{F^-}(x) \le \mu_{F^+}(x) \le 1$.*

The difference $\pi_F(x) = \mu_{F^+}(x) - \mu_{F^-}(x)$ is called the interval-valued fuzzy index. It shows the hesitant degree of x belonging to F. It is the most important and original idea distinguishing the interval-valued fuzzy set from the fuzzy set.

2.2 Multiple twin support vector machine

This paper considers the multi-class classification problem with the sample set $S = \{(x_1, y_1), (x_2, y_2), \ldots, (x_l, y_l)\}$, where $x_j \in R^n (j = 1, \ldots, l)$ are inputs, $y_j \in \{1, 2, \ldots, k\} (j = 1, \ldots, l)$ are the corresponding outputs and k is the number of classes. The following algorithm evaluates all training points into a "one-versus-rest" structure. Let S_1^i be the sample set with label "i" and l_1^i be the corresponding sample number. The points in it are denoted in rows and the features of each point are shown as n columns in matrices $A_i (\in R^{l_1^i \times n})$. Let S_2^i be the sample set except for label "i'' and l_2^i be the corresponding sample number. The points in it are denoted in rows and the features of each point are shown as n columns in matrices $\tilde{A}_i \in (R^{l_2^i \times n})$. Obviously, $S = S_1^i \cup S_2^i$ and $l_1^i + l_2^i = l$. Let $A = [A_i, \tilde{A}_i]$ be the matrix of all samples in S.

Multiple twin support vector machine (MTSVM) [30] turns a multi-class classification problem into k binary classification problems. As illustrated in Figure 1, it aims to find k hyperplanes:

$$K(A_i, A^T)w_i + b_i = 0, \quad i = 1, \ldots, k. \tag{1}$$

For the ith binary classification problem, MTSVM regards the sample points with label "i" as positive points, and the residual points belonging to the other $k-1$ classes as negative points. It requires that the positive hyperplane $K(A_i, A^T)w_i + b_i = 0$ should be as close as possible to all the positive points, while the negative points should be at a distance of at least 1 from the positive hyperplane.

The k hyperplanes can be solved by the following QPPs

$$\begin{aligned} & min_{w_i, b_i, \delta_i, \xi_i} \frac{1}{2} \delta_i^T \delta_i + c_i (e_i^-)^T \xi_i \\ \text{s.t.} \quad & K(A_i, A^T)w_i + e_i^+ b_i = \delta_i, \; i = 1, \ldots, k, \\ & K(\tilde{A}_i, A^T)w_i + e_i^- b_i \leq -e_i^- + \xi_i, \; i = 1, \ldots, k, \\ & \xi_i \geq 0, \; i = 1, \ldots, k, \end{aligned} \tag{2}$$

where $K(,)$ is a kernel function, $c_i (> 0, i = 1, \ldots, k)$ are penalty parameters, $\xi_i, \delta_i (i = 1, \ldots, k)$ are vectors of slack variables associated with samples, and e_i^+, e_i^- are vectors with all 1's of appropriate dimensions.

For a new test point x, MTSVM determines its class label by the following decision function

$$\text{Class } i = \arg \min_{i=1,\ldots,k} \frac{|K(x^T, A^T)w_i + b_i|}{\sqrt{w_i^T K(A, A^T) w_i}}. \tag{3}$$

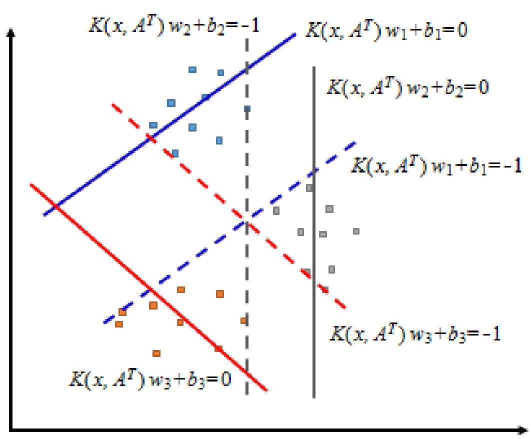

Figure 1: MTSVM

3 Interval-valued fuzzy multiple twin support vector machine

Like any other kind of multi-class support vector machine, MTSVM lacks the ability to determine the importance of different sample points. Prior knowledge plays an important role in improving the performance of multi-class classification because it can tell us the importance of different sample points. There are usually two ways to add membership degree to a sample point. One is based on the distance between a sample point and the center of the class. That is to say, if the sample point is close to the center of the class, it will be given a larger membership degree, while if the sample point is far away from the center of the class, it will be given a smaller membership degree. The other one is based on the relative density of a sample point. That is to say, if the sample point is relatively dense, it will be given a higher membership degree, while on the contrary, it will be given a lower membership degree. In this paper, a novel method of adding membership degree to sample points is introduced which is different from previous work. It emphasizes the importance of sample points to classification model and also considers the importance of sample points to class to some extent. The following is a detailed explanation on how to add such interval-valued fuzzy membership to MTSVM and establish the interval-valued fuzzy multiple twin support vector machine (IVF-MTSVM). The specific flow chart of IVF-MTSVM is shown in Figure 2.

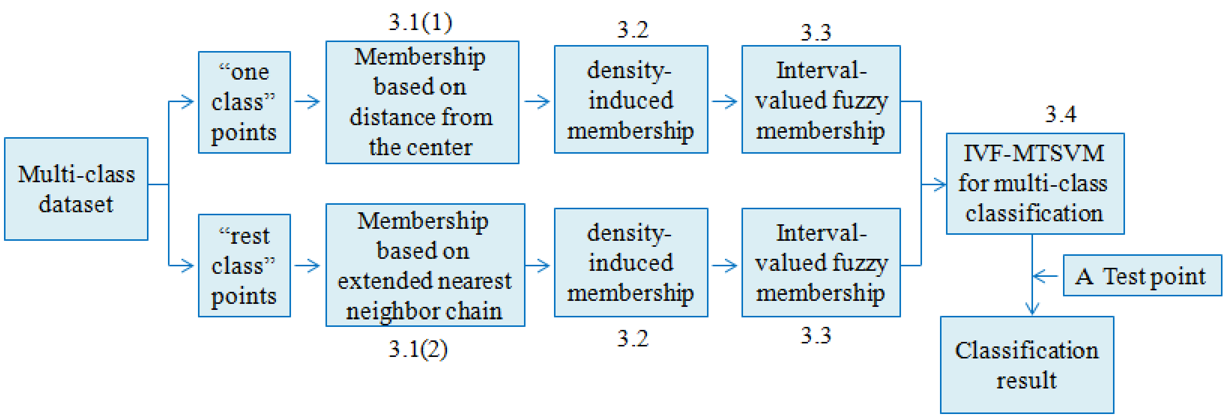

Figure 2: The flow chart of IVF-MTSVM

3.1 Establishing the initial membership degree for training points

MTSVM requires the decision hyperplane of the i-th class to pass through such points as evenly as possible. Therefore, this paper establishes the initial membership degree of the ith class points according to the distances between the sample points and the class center. At the same time, MTSVM requires the distance between the residual class points and the decision hyperplane of the i-th class be at least 1, so this paper establishes the initial membership degree of residual class points based on the extended nearest neighbor chain.

(1) The initial membership degree for $x(x \in S_1^i)$[19]

Determine the center of class "i"

$$C_1^i = \frac{\sum_{x \in S_1^i} x}{l_1^i}, \tag{4}$$

Then define the initial membership degree for training point $x(x \in S_1^i)$

$$\mu_1^i(x) = 1 - \frac{d(x, C_1^i)}{\max_k d(x_k, C_1^i)}, \quad x_k \in S_1^i. \tag{5}$$

Here $d(x, z) = ||x - z||_2$ is defined as the distance between x and z.

Remark 3.1. *In S_1^i, we consider the importance of sample points to the classification hyperplane to add the membership degree. Of course, for this class of points, the method of establishing membership degree is applicable whether we consider the importance of sample points to the classification model or to the class.*

(2) The initial membership degree for $x(x \in S_2^i)$[39]

MTSVM requires that training points other than class "i" to be at least 1 away from the hyperplane of class "i". The hyperplane $K(\tilde{A}_i, A^T)w_i + b_i = -1$ only depends on support vectors in S_2^i, whereas all other sample points are irrelevant to this decision function. This is the same thing as a binary support vector machine. But we don't know which sample points are support vectors until we classify them, so we find the points that lie nearest to the hyperplane first by the nearest neighbor chain method. Because they are more important for the determination of the hyperplane $K(\tilde{A}_i, A^T)w_i + b_i = -1$, they are assigned with higher membership degree.

The extended nearest neighbor chain of $x (x \in S_2^i)$ consists of a sequence nodes $x_1, x_2, \ldots, x_m$, where m is a user-specified integer. The corresponding labels are $y_1, y_2, \ldots, y_m$. The first node x_1 is x itself and its label is -1. The other node $x_j (j = 2, \ldots, m)$ is the nearest neighbor in $S - \{x_1, \ldots, x_{j-1}\}$ with label y_j. When j is even, $y_j = 1$. Otherwise, $y_j = -1$. That is to say, the second node x_2 is the nearest neighbor of x_1 in S_1^i, the third node x_3 is the nearest neighbor of x_2 in $S_2^i - \{x_1\}$, the fourth node x_4 is the nearest neighbor of x_3 in $S_1^i - \{x_2\}$, the fifth node x_5 is the nearest neighbor of x_4 in $S_2^i - \{x_1, x_3\}$, and so on.

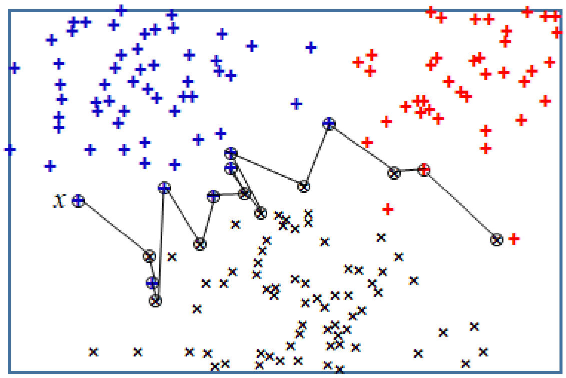

Figure 3: An illustration of the extended nearest neighbor chain in a triple-class classification problem

Figure 3 is an illustration of the extended nearest neighbor chain in a triple-class classification problem. The black x-marks represent class S_1^i, while blue pluses and red pluses represent class S_2^i. When we choose x in S_2^i and find the chain of it, the triple-class problem is converted into binary problems via "one-versus-rest" strategy.

In order to utilize the above information, we define a distance sequence $(d_{1,2}, d_{2,3}, \ldots, d_{m-1,m})$, $d_{j,j+1}$ is defined as the distance between the j-th node x_j and the $j+1$-th node x_{j+1},

$$d_{j,j+1} = ||x_j - x_{j+1}||_2, \quad j = 1, \ldots, m-1. \tag{6}$$

We use the longest edge of the chain to estimate how far the example x_i is away from the decision plane $K(\tilde{A}_i, A^T)w_i + b_i = -1$.

The weight assigned to x in S_2^i can be represented as

$$\mu_2^i(x) = \frac{E_{margin}}{\max_{j=1,\ldots,m-1} d_{j,j+1}}. \tag{7}$$

Here E_{margin} is the mean of the edges except the longest one and can be represented as

$$E_{margin} = \frac{\sum_{j=1}^{m-1} d_{j,j+1} - \max_{j=1,\ldots,m-1} d_{j,j+1}}{m-2}. \tag{8}$$

Since $E_{margin} \leq \max_{j=1,\ldots,m-1} d_{j,j+1}$, the weights satisfy $0 \leq \mu_2^i(x) \leq 1$. When x locates far away from the margin, $E_{margin} \ll \max_{j=1,\ldots,m-1} d_{j,j+1}$, $\mu_2^i(x)$ is close to 0; when x locates near the margin, $E_{margin} \approx \max_{j=1,\ldots,m-1} d_{j,j+1}$, the weight $\mu_2^i(x)$ is close to 1.

Remark 3.2. *In S_2^i, this method of establishing membership degree values the importance of the sample points to the classification model rather than the usual importance to the class.*

Now in S_1^i, the fuzzy membership degree defined by distance distinguishes the sample points near and far from the center of the class. In S_2^i, the sample points near and far from the classification edge are distinguished based on the fuzzy membership degree defined by the nearest neighbor chain. Next, we further excavate the density information of the sample points and use it to adjust the membership degree just established in S_1^i and S_2^i.

3.2 Establishing the density-induced membership degree for training points [17]

According to the density distribution of a given data set, a data point may locate in a dense region and has a higher density degree, or in a non-dense region and has a lower density degree. The initial membership degree for training points does in 3.1 does not take into account the density distribution, so it may not get the optimal results. In this paper, we use the KNN method to extract relative density information and assign them to the corresponding sample points.

Assume x is a sample point, x^K is the K-th nearest neighbor of x in the corresponding class. Given the value of K, $d(x, x^K)$ represents the distance between x and x^K, the relative density degree is defined by

$$\bar{\mu}(x) = \exp\{\omega \times \frac{MEAN^K}{d(x, x^K)}\}, \tag{9}$$

where $MEAN^K = \frac{1}{s}\sum d(x, x^K)$, s is the number of training points in the corresponding dataset, $0 \leq \omega \leq 1$ is a weighting factor. Note that under this definition, if the data points are in a dense area, a greater density will be achieved; if the data points are in a non-dense area, a lower density will be achieved.

To be specific, if x is a sample point with label "i",

$$\bar{\mu}_1^i(x) = \exp\{\omega \times \frac{MEAN^K}{d(x, x^K)}\}, \tag{10}$$

where $MEAN^K = \frac{1}{l_1^i}\sum_{x\in S_1^i} d(x, x^K)$.

If x is a sample point except for label "i",

$$\bar{\mu}_2^i(x) = \exp\{\omega \times \frac{MEAN^K}{d(x, x^K)}\}, \tag{11}$$

where $MEAN^K = \frac{1}{l_2^i}\sum_{x\in S_2^i} d(x, x^K)$.

3.3 Establishing the interval-valued fuzzy membership degree for training points

In order to take into account the distribution characteristics of sample points, the initial membership of all the sample points is modified according to the local density of sample points, so as to obtain the interval-valued fuzzy membership of all the sample points.

If x is a sample point with class label "i",

$$\text{let} \quad \tilde{\mu}_1^i(x) = \mu_1^i(x) + t\bar{\mu}_1^i(x), 0 \leq t \leq 1, \tag{12}$$

we get the revisionary interval-valued fuzzy membership degree $[\mu_1^i(x), \tilde{\mu}_1^i(x)]$.

If x is a sample point except for label "i",

$$\text{let} \quad \tilde{\mu}_2^i(x) = \mu_2^i(x) + t\bar{\mu}_2^i(x), 0 \leq t \leq 1, \tag{13}$$

we get the revisionary interval-valued fuzzy membership degree $[\mu_2^i(x), \tilde{\mu}_2^i(x)]$.

In the following, we add the revisionary interval-valued fuzzy membership degree into MTSVM to establish IVF-MTSVM.

3.4 Establishing and solving IVF- MTSVM model

Let

$$\mu_1^{(i)}(x) = \frac{\mu_1^i(x) + \tilde{\mu}_1^i(x)}{2}, \ x \in S_1^i \tag{14}$$

$$\mu_2^{(i)}(x) = \frac{\mu_2^i(x) + \tilde{\mu}_2^i(x)}{2}, \ x \in S_2^i \tag{15}$$

$$W^{(i)} = diag([\mu_1^{(i)}(x_1), \ldots, \mu_1^{(i)}(x_{l_1^i})]), \ x_1, \ldots, x_{l_1^i} \in S_1^i \tag{16}$$

$$\tilde{W}^{(i)} = diag([\mu_2^{(i)}(x_1), \ldots, \mu_2^{(i)}(x_{l_2^i})]), \ x_1, \ldots, x_{l_2^i} \in S_2^i \tag{17}$$

The interval-valued fuzzy membership of the training points in S_1^i and S_2^i is written in the form of weight matrix $W^{(i)}$ and $\tilde{W}^{(i)}$ respectively. The hyperplanes of linear IVF-MTSVM are

$$(w_i \cdot x) + b_i = 0, \ i = 1, \ldots, k. \tag{18}$$

They can be achieved by resolving the following QPPs:

$$\begin{aligned} & min_{w_i,b_i,\delta_i,\xi_i} \frac{1}{2}\delta_i^T\delta_i + c_i(e_i^-)^T\xi_i \\ \text{s.t.}\quad & W^{(i)}(A_iw_i + e_i^+b_i) = \delta_i,\ i = 1,\dots,k, \\ & \tilde{W}^{(i)}[-(\tilde{A}_iw_i + e_i^-b_i) - e_i^-] \geq -\xi_i,\ i = 1,\dots,k, \\ & \xi_i \geq 0,\ i = 1,\dots,k \end{aligned} \tag{19}$$

Introduce the Lagrange function

$$L(w_i,b_i,\delta_i,\alpha_i,\beta_i) = \frac{1}{2}||W^{(i)}(A_iw_i + e_i^+b_i)||^2 + c_i(e_i^-)^T\xi_i - \alpha_i^T\{\tilde{W}^{(i)}[-(\tilde{A}_iw_i + e_i^-b_i - e_i^-) + \xi_i]\} - \beta_i^T\xi_i \tag{20}$$

Here $\alpha_i = (\alpha_{i1},\alpha_{i2},\dots,\alpha_{il_1^i})^T$, $\beta_i = (\beta_{i1},\beta_{i2},\dots,\beta_{il_2^i})^T$ are Lagrange multipliers, according to KKT,

$$\begin{aligned} & W^{(i)}[A_i^T(W^{(i)}A_iw_i + W^{(i)}e_i^+b_i)] + \tilde{W}^{(i)}\tilde{A}_i^T\alpha_i = 0, \\ & W^{(i)}[(e_i^+)^T(W^{(i)}A_iw_i + W^{(i)}e_i^+b_i)] + \tilde{W}^{(i)}(e_i^-)^T\alpha_i = 0, \\ & c_ie_i^- - \alpha_i - \beta_i = 0, \\ & \tilde{W}^{(i)}[-(\tilde{A}^{(i)}w_i + e_i^-b_i) - e_i^-] + \xi_i \geq 0, \\ & \xi_i \geq 0, \\ & \alpha_i^T\{\tilde{W}^{(i)}[-(\tilde{A}^{(i)}w_i + e_i^-b_i) - e_i^-] + \xi_i\} = 0, \\ & \beta_i^T\xi_i = 0, \\ & \alpha_i \geq 0,\ \beta_i \geq 0, \\ & 0 \leq \alpha_i \leq c_i, \end{aligned} \tag{21}$$

$$[(W^{(i)}A_i)^T(W^{(i)}e_i^+)^T][(W^{(i)}A_i)(W^{(i)}e_i^+)][w_i \quad b_i]^T + [(\tilde{W}^{(i)}\tilde{A}_i)^T(\tilde{W}^{(i)}e_i^-)^T]\alpha_i = 0.$$

Let $E = [(W^{(i)}A_i)(W^{(i)}e_i^+)], F = [(\tilde{W}^{(i)}\tilde{A}_i)(\tilde{W}^{(i)}e_i^-)], u_i = [w_i \quad b_i]^T$, $E^TEu_i + F^T\alpha_i = 0$, then $u_i = -(E^TE)^{-1}F^T\alpha_i$,

$$u_i = -(E^TE + \varepsilon I)^{-1}F^T\alpha_i. \tag{22}$$

Using Wolfe theorem, this optimization problem can be transformed into its dual problem

$$\begin{aligned} & \max_{\alpha_i}(e_i^-)^T - \frac{1}{2}\alpha_i^TF(E^TE)^{-1}F^T\alpha_i \\ \text{s.t.}\quad & 0 \leq \alpha_i \leq c_i, \end{aligned} \tag{23}$$

Once this problem is solved, $u_i = [w_i \quad b_i]^T$ is derived and the decision function is established. For a new test point x, its class label can be determined by the following decision function:

$$\text{Class } i = \arg\min_{i=1,\dots,k} \frac{|(w_i \cdot x) + b_i|}{||w_i||}. \tag{24}$$

In the nonlinear case, $K(,)$ is the kernel function and the hyperplanes of IVF-MTSVM are

$$K(A_i, A^T)w_i + b_i = 0, \quad i = 1,\dots,k. \tag{25}$$

The model corresponding to (27) can be written as:

$$\begin{aligned} & min_{w_i,b_i,\delta_i,\xi_i} \frac{1}{2}\delta_i^T\delta_i + c_i(e_i^-)^T\xi_i \\ \text{s.t.}\quad & W^{(i)}(K(A_i,A^T)w_i + e_i^+b_i) = \delta_i,\ i = 1,\dots,k, \\ & \tilde{W}^{(i)}[-(K(\tilde{A}_i,A^T)w_i + e_i^-b_i) - e_i^-] \geq -\xi_i,\ i = 1,\dots,k, \\ & \xi_i \geq 0,\ i = 1,\dots,k. \end{aligned} \tag{26}$$

Similar to the linear case, solve the Lagrange dual problem

$$L(w_i, b_i, \xi_i, \alpha_i, \beta_i) = \frac{1}{2}||W^{(i)}(K(A_i, A^T)w_i + e_i^+ b_i)||^2 + c_i(e_i^-)^T\xi_i - \alpha_i^T\{\tilde{W}^{(i)}[-(K(\tilde{A}_i, A^T)w_i + e_i^- b_i) - e_i^-] + \xi_i\} - \beta_i^T\xi_i \tag{27}$$

Here $\alpha_i = (\alpha_{i1}, \alpha_{i2}, \dots, \alpha_{il_1^i})^T$, $\beta_i = (\beta_{i1}, \beta_{i2}, \dots, \beta_{il_2^i})^T$ are Lagrange multipliers, according to KKT,

$$u_i = -(E^TE + \varepsilon I)^{-1}F^T\alpha_i, \tag{28}$$

where $E = [(W^{(i)}K(A_i, A^T))(W^{(i)}e_i^+)], F = [(\tilde{W}^{(i)}K(\tilde{A}_i, A^T))(\tilde{W}^{(i)}e_i^-)], u_i = [w_i \ b_i]^T$. Using Wolfe theorem, this optimization problem can be transformed into its dual problem

$$\begin{aligned} &\max_{\alpha}(e_i^-)^T\alpha_i - \frac{1}{2}\alpha_i^TF(E^TE)^{-1}F^T\alpha_i \\ \text{s.t.} \quad &0 \le \alpha_i \le c_i, \end{aligned} \tag{29}$$

Once this problem is solved, $u_i = [w_i \ b_i]^T$ is derived and the decision function is established. For a new test point x, its class label can be determined by the following decision function:

$$\text{Class } i = \arg\min_{i=1,\dots,k} \frac{|K(x^T, A^T)w_i + b_i|}{\sqrt{w_i^TK(A, A^T)w_i}}. \tag{30}$$

The specific **IVF-MTSVM Algorithms** are shown in **Algorithm 1** and **Algorithm 2**.

Algorithm 1 Linear IVF-MTSVM Algorithm

Input: The training matrices $A_i(\in R^{l_1^i \times n})$, $\tilde{A}_i(\in R^{l_2^i \times n})$ and $A(\in R^{l \times n})$, the nearest neighbor chain parameter m, the density parameter ω, K, the interval-valued fuzzy parameter t and the penalty parameters $c_i > 0$.

Output: The decision function Class $i = \arg\min_{i=1,\dots,k} \frac{|(w_i \cdot x) + b_i|}{||w_i||}$.

For Training: For $i = 1$ to k, where k is number of classes in dataset.

1a. Establish IVF membership degree $[\mu_1^i(x), \tilde{\mu}_1^i(x)]$ by (5), (10), (12) and $[\mu_2^i(x), \tilde{\mu}_2^i(x)]$ by (7), (11), (13);

1b. Define weight matrices $W^{(i)}$ and $\tilde{W}^{(i)}$ as (16) and (17);

1c. Construct original problems of linear IVF-MTSVM (19);

1d. Select all optimal parameters on the basis of validation;

1e. Solve the dual problem (23) to obtain w_i and b_i;

1f. Generate hyperplanes using (18).

For Testing:
Training phase generates k hyperplanes one for each class. Then, a class is assigned to test point by using (24).

Algorithm 2 Nonlinear IVF-MTSVM Algorithm

Input: The training matrices $A_i(\in R^{l_1^i \times n})$, $\tilde{A}_i(\in R^{l_2^i \times n})$ and $A(\in R^{l \times n})$, the nearest neighbor chain parameter m, the density parameter ω, K, the interval-valued fuzzy parameter t and the penalty parameters $c_i > 0$.

Output: The decision function Class $i = \arg\min_{i=1,\dots,k} \frac{|K(x^T, A^T)w_i + b_i|}{\sqrt{w_i^TK(A, A^T)w_i}}$.

For Training: For $i = 1$ to k, where k is number of classes in dataset.

1a. Establish IVF membership degree $[\mu_1^i(x), \tilde{\mu}_1^i(x)]$ by (5), (10), (12) and $[\mu_2^i(x), \tilde{\mu}_2^i(x)]$ by (7), (11), (13);

1b. Define weight matrices $W^{(i)}$ and $\tilde{W}^{(i)}$ as (16) and (17);

1c. Select a kernel function K and construct original problems of nonlinear IVF-MTSVM (26);

1d. Select all optimal parameters on the basis of validation;

1e. Solve the dual problem (29) to obtain w_i and b_i;

1f. Generate hyperplanes using (25).

For Testing:
Training phase generates k hyperplanes one for each class. Then, a class is assigned to test point by using (30).

---

4 Numerical experiments

4.1 Experimental results on UCI datasets

In order to validate the performance of our newly-designed IVF-MTSVM for multi-class classification, we compare it with other multiple twin support vector machine methods in "one-versus-rest" structure and other fuzzy multiple twin support vector machine established by some previous methods. All the experiments are carried out on eight datasets which come from UCI repository. In order to prove the effectiveness of IVF-MTSVM in reducing the influence of outliers, we add a number of artificial error mark data points in the eight datasets respectively, which account for 5% of the sample points. Thus, we have a total of sixteen datasets, eight of which are UCI original datasets, and eight of which are new datasets with 5% noises. The experiments are implemented in MATLAB 2017 on a PC with Intel(R) Core(TM) i7-3520M CPU 2.90GHz processor and 8GB RAM.

Macro F1 and Micro F1 are adopted as the evaluation indices of classification in this paper. In a binary classification task, $Accuracy = \frac{TP+FN}{TP+FP+TN+FN}$, $Precision = \frac{TP}{TP+FP}$, $Recall = \frac{TP}{TP+FN}$ and $FI = \frac{2Precision \cdot Recall}{Precision+Recall}$ are commonly used evaluation indicators to measure the quality of the classifier, where TP, FP, TN and FN are the number of true positive, false positive, true negative and false negative on the testing samples, respectively. Accuracy cannot reasonably reflect the predictive ability of the model when sample imbalance occurs. Usually, there is an inverse relationship between Precision and Recall. F1 is a composite indicator that considers both Precision and Recall. In a multi-class classification problem, Macro F1 and Micro F1 are used to measure the average performance for the whole classes. Macro F1 is computed by averaging F1 of every class, while Micro F1 is calculated by averaging the Precision and Recall of every point. Since Macro F1 gives the weight to all classes equally, it will mainly be influenced by the performance of rare classes. In contrast, Micro F1 treats all points equally and it will be dominated by the performance of common classes.

To make the results more convincing, we utilize five-fold cross-validation [11] to deal with each experiment. The training set is randomly partitioned into five subsets which are roughly of equal size, and one of those subsets is reserved as the testing set whereas the remaining serve as the training set. This process is repeated five times until all of the five subsets have been set to be a testing one once. Moreover, all the datasets are standardized such that the features scale is in $[0, 1]$ before training. Here, we use RBF kernel to classify these datasets.

Table 1: Comparison of Macro F1 and Micro F1 of four classification methods on UCI datasets (RBF)

Datasets	Noise Ratio(%)	MLSTSVM		MWLTSVM		MTSVM		IVF-MTSVM	
		Macro F1	Micro F1	Macro F1	Micro F1	Macro F1	Micro F1	Macro F1	Micro F1
Wine	0	**99.48**	**99.44**	98.30	98.32	97.76	97.74	98.90	98.86
($178 \times 13 \times 3$)	5	93.31	93.25	**96.72**	**96.81**	90.04	89.37	94.18	94.07
Iris	0	**97.99**	**98.00**	96.70	96.67	96.67	96.67	97.31	97.33
($150 \times 4 \times 3$)	5	87.28	87.33	**93.98**	**94.00**	85.84	86.05	92.09	92.16
Glass	0	64.02	65.39	64.18	65.87	66.73	66.36	**68.09**	**68.00**
($214 \times 9 \times 6$)	5	56.63	57.44	58.63	59.29	54.09	53.73	**59.41**	**60.39**
E coli	0	82.28	82.91	**86.69**	**87.15**	82.54	83.19	84.92	85.26
($327 \times 7 \times 5$)	5	80.34	81.04	**82.94**	83.14	80.15	79.81	82.86	**83.72**
Seeds	0	93.70	93.71	93.01	92.86	94.42	94.29	**94.43**	**94.29**
($210 \times 7 \times 3$)	5	91.49	91.42	**92.89**	**92.85**	90.13	90.00	91.81	91.65
Balance	0	91.76	91.84	**97.00**	**97.12**	94.79	95.00	95.87	96.02
($625 \times 4 \times 3$)	5	90.41	90.56	91.08	91.14	89.63	89.76	**92.77**	**92.93**
Tae	0	54.31	58.78	54.61	59.53	59.59	63.98	**61.04**	**65.29**
($151 \times 5 \times 3$)	5	52.78	56.26	51.02	55.06	54.54	58.82	**55.27**	**59.68**
Hayesroth	0	75.85	77.37	75.97	77.35	73.89	75.53	**77.64**	**78.38**
($132 \times 5 \times 3$)	5	62.32	63.67	64.32	65.86	62.41	63.31	**66.71**	**66.52**
Average		79.62	80.53	81.13	82.06	79.58	80.23	**82.08**	**82.78**

Remark 4.1. *In table 1, MTSVM [33], multiple least square twin support vector machine (MLSTSVM) [29] and multiple weighted linear loss twin support vector machine (MWLTSVM) [10] are all twin-related multi-class classification models in "one-versus-rest" structure.*

Table 2: Comparison of training time of four classification methods on UCI datasets (RBF)

Datasets	Noise Ratio (%)	MLSTSVM	MWLTSVM	MTSVM	IVF-MTSVM
Wine	0	0.0410	**0.0216**	0.2703	0.2735
Iris	0	0.0313	**0.0123**	0.2270	0.2299
Glass	0	0.1272	**0.0411**	1.3223	1.3250
E coli	0	0.2300	**0.0751**	2.3137	2.3171
Seeds	0	0.0534	**0.0183**	0.3941	0.3978
Balance	0	0.7060	**0.2270**	7.0513	7.0565
Tae	0	0.0330	**0.0153**	0.2519	0.2553
Hayesroth	0	0.0250	**0.0126**	0.2078	0.2106

Table 3: Comparison of Macro F1 and Micro F1 of four classification methods on UCI datasets (RBF)

Datasets	Noise Ratio(%)	MTSVM		F1-MTSVM		F2-MTSVM		IVF-MTSVM	
		Macro F1	Micro F1	Macro F1	Micro F1	Macro F1	Micro F1	Macro F1	Micro F1
Wine	0	97.76	97.74	98.45	98.44	**99.40**	**99.37**	98.90	98.86
$(178 \times 13 \times 3)$	5	90.04	89.37	93.41	93.34	**96.08**	**95.89**	94.18	94.07
Iris	0	96.67	96.67	**97.33**	**97.39**	96.19	96.24	97.31	97.33
$(150 \times 4 \times 3)$	5	85.84	86.05	**94.35**	**94.40**	92.01	92.10	92.09	92.16
Glass	0	66.73	66.36	65.66	65.91	66.95	67.04	**68.09**	**68.00**
$(214 \times 9 \times 6)$	5	54.09	53.73	54.37	54.76	58.74	58.38	**59.41**	**60.39**
E coli	0	82.54	83.19	**85.54**	**85.62**	84.41	84.91	84.92	85.26
$(327 \times 7 \times 5)$	5	80.15	79.81	**83.06**	**84.10**	81.88	82.04	82.86	83.72
Seeds	0	94.42	94.29	94.28	94.25	93.79	93.71	**94.43**	**94.29**
$(210 \times 7 \times 3)$	5	90.13	90.00	90.75	90.73	90.68	90.65	**91.81**	**91.65**
Balance	0	94.79	95.00	95.76	95.87	95.79	95.91	**95.87**	**96.02**
$(625 \times 4 \times 3)$	5	89.63	89.76	91.37	91.48	91.11	91.24	**92.77**	**92.93**
Tae	0	59.59	63.98	60.99	64.11	59.83	64.05	**61.04**	**65.29**
$(151 \times 5 \times 3)$	5	54.54	58.82	55.71	59.45	**56.37**	**60.76**	55.27	59.68
Hayesroth	0	73.89	75.53	74.53	75.63	74.85	75.87	**77.64**	**78.38**
$(132 \times 5 \times 3)$	5	62.41	63.31	62.72	63.93	65.17	66.36	**66.71**	**66.52**
Average		79.58	80.23	81.14	81.84	81.45	82.16	**82.08**	**82.78**

Remark 4.2. *In table 2, F1-MTSVM is a distance-based fuzzy MTSVM by adding only the membership in formula (11) to all sample points in MTSVM. Similarly, F2-MTSVM is a density-induced fuzzy MTSVM by adding only the membership degree in formula (15) to all sample points in MTSVM.*

Table 4: Comparison of training time of four types of MTSVM methods on UCI datasets (RBF)

Datasets	Noise Ratio (%)	MTSVM	F1-MTSVM	F2-MTSVM	IVF-MTSVM
Wine	0	**0.2703**	0.2726	0.2727	0.2735
Iris	0	**0.2270**	0.2293	0.2292	0.2299
Glass	0	**1.3223**	1.3249	1.3249	1.3260
E coli	0	**2.3137**	2.3163	2.3163	2.3171
Seeds	0	**0.3941**	0.3964	0.3966	0.3978
Balance	0	**7.0513**	7.0548	7.0547	7.0565
Tae	0	**0.2519**	0.2542	0.2542	0.2563
Hayesroth	0	**0.2078**	0.2095	0.2095	0.2116

Table 5: Comparison of parameter adjusting time of membership of three fuzzy MTSVM methods on UCI datasets (RBF)

Datasets	Noise Ratio (%)	F1-MTSVM	F2-MTSVM	IVF-MTSVM
Wine	0	**0.0016**	0.0289	0.1107
Iris	0	**0.0015**	0.0213	0.0817
Glass	0	**0.0021**	0.0361	0.1619
E coli	0	**0.0030**	0.0692	0.3513
Seeds	0	**0.0024**	0.0345	0.1481
Balance	0	**0.0045**	0.2356	1.2719
Tae	0	**0.0016**	0.0190	0.0810
Hayesroth	0	**0.0017**	0.0172	0.0644

4.2 Comparison and analysis

This subsection explains in detail all the comparisons from Table 1 to Table 5. Table 1 shows the comparison of Macro F1 and Micro F1 of four classification methods in "one-versus-rest" structure. Table 2 shows the comparison of training time of four classification methods in Table 1. In order to further compare the method of IVF-MTSVM proposed in this paper with the previous fuzzy MTSVM methods, the comparison in Table 3 is also made. Table 4 shows the comparison of training time of four types of MTSVM methods in Table 3. Table 5 shows the comparison of adjusting parameter time of membership of three fuzzy types of MTSVM methods in Table 3. The characteristics of each dataset including the number of samples, features and classes are also shown in Table 1 and Table 3. The highest Macro F1 and Micro F1 as well as the shortest calculation time in the tables are shown in bold. The following analysis is mainly based on Micro F1, because the trend of Macro F1 and Micro F1 is basically the same in the experiment in this paper. Through these comparisons, we can draw the following conclusions:

(1) In Table 1, for the multi-class twin support vector machine in "one-versus-rest" structure, except the method IVF-MTSVM, the Micro F1 of MWLTSM ranks first, MLSTSVM ranks second and MTSVM ranks last. IVF-MTSVM is an improvement on MTSVM. It improves the Micro F1 of MTSVM by 2.55%, making it the first of the four methods. This indicates that the multi-class classification model MTSVM with interval-valued fuzzy information has better classification performance than the original model.

(2) Again in Table 1, compared with IVF-MTSVM, the Micro F1 of MTSVM is improved by 1.33% on the 8 original data sets. On the other 8 datasets with 5% noise added, the Micro F1 of MTSVM is improved by 3.78%. It can be seen that this new classification method is more effective for datasets containing noise points. When the training data set is contaminated with outliers, this method successfully reduces the impact of outliers and produces a higher classification rate compared to standard multi-classifiers.

(3) Table 2 lists the corresponding training time in Table 1 (Noise Ratio = 0%). Of the four methods, MWLTSM ranks first, MLSTSVM ranks second, MTSVM ranks third, and IVF- MTSVM ranks last. The objective functions of MWLTSVM and MLSTSVM can eventually be converted into solving linear equations, so they have relatively low complexity. Specifically, the dimension of coefficient matrix of the linear equations in (5) and (8) is $(l+1)\times(l+1)$, so the complexity of directly solving this linear equation is $O((l+1)^3)$. But by using the conjugate gradient algorithm, the complexity is at most $O(2(l+1)^2\cdot r)$, where r is the iterations. Although the training time of IVF-MTSVM ranked the last, it is only 0.0031 seconds longer than the average time of MTSVM, which indicates that it still has a certain comparability of time. Moreover, because its Micro F1 is higher than the other three methods, its advantage is still obvious.

(4) In Table 3, Micro F1 of IVF-MTSVM method is 0.94% higher than that of F1-MTSVM method, and 0.62% higher than that of F2-MTSVM method. This shows that the interval-valued fuzzy membership proposed in this paper has certain advantages. When establishing the initial membership degree of sample points, this method mainly considers the importance of sample points to the model. When establishing the final membership degree of sample points, the density of sample points is also considered, which has a positive impact on the performance of the final classification model.

(5) Table 4 lists the corresponding training time in Table 4 (noise ratio = 0%). The training time of IVF-MTSVM is about the same as that of the other two fuzzy MTSVM, and neither is higher than that of MTSVM. This shows that adding various fuzzy memberships to MTSVM improves Micro F1 to some extent, but the calculation complexity of

the model has not changed. Specifically, suppose that the size of each class is roughly $1/k$, the complexity of MTSVM is $O((k-1)^3 l^3/k^2)$[33]. The complexity of the other three fuzzy MTSVM is also the same.

(6) Although fuzzy membership added before MTSVM training has almost no effect on the training complexity, the improved Micro F1 of these models is at the expense of time for parameter adjustment. Therefore, the adjustment times of different membership degrees are calculated in this paper and the results are placed in Table 5. We can see that the F1-MTSVM model has the shortest time because its membership is only related to the distance from the sample point to the center of the class. The membership of F1-MTSVM based on the nearest neighbor chain takes a little more time. IVF-MTSVM takes the longest time because for different types of sample points, different types of membership are established.

4.3 Statistical test

In this part, Friedman test and Benferroni-Dunn test [8] are used to test if there is a significant difference between the four TWSVM family multi-class classification methods in "one-versus-rest" structure in Table 1.

(1) Friedman test

Friedman test is proved to be a simple non parametric statistic method. To compute the Friedman statistic, the average ranking based on Micro F1 in Table 1 is listed in Table 6. Under the null hypothesis that all the algorithms are equivalent, the Friedman statistic can be computed as follows:

$$\chi_F^2 = \frac{12N}{m(m+1)}\left[\sum_j R_j^2 - \frac{m(m+1)^2}{4}\right], \tag{31}$$

where $R_j = \frac{1}{N}\sum_i r_i^j$, and r_i^j denotes the j-th of m algorithms on the ith of N datasets. Here $m = 4$ and $N = 16$. Then a more desirable statistic is derived:

$$F_F = \frac{(N-1)\chi_F^2}{N(m-1)-\chi_F^2}, \tag{32}$$

which is distributed according to the F-distribution with $m-1$ and $(m-1)(N-1)$ degrees of freedom.

According to (31) and (32), for the nonlinear case, we can obtain $\chi_F^2 = 18.84$ and $F_F = 9.69$, according to F-distribution with $(3, 45)$ degrees of freedom. As can be seen from the table of critical values for F-distribution, the critical value of F $(3, 45)$ is about 2.21 at the significance level $\alpha = 0.1$. Since the real value of F_F is much larger than the critical values, the null hypothesis which is all algorithms are performing equivalently is clearly rejected.

Table 6: Average ranking based on Micro F1 in Table 1

Datasets	Noise Ratio(%)	MLSTSVM	MWLTSVM	MTSVM	IVF-MTSVM
Wine	0	1	3	4	2
$(178 \times 13 \times 3)$	5	3	1	4	2
Iris	0	1	3.5	3.5	2
$(150 \times 4 \times 3)$	5	3	1	4	2
Glass	0	4	3	2	1
$(214 \times 9 \times 6)$	5	3	2	4	1
E coli	0	4	1	3	2
$(327 \times 7 \times 5)$	5	3	2	4	1
Seeds	0	3	4	1.5	1.5
$(210 \times 7 \times 3)$	5	3	1	4	2
Balance	0	4	1	3	2
$(625 \times 4 \times 3)$	5	3	2	4	1
Tae	0	4	3	2	1
$(151 \times 5 \times 3)$	5	3	4	2	1
Hayesroth	0	2	3	4	1
$(132 \times 5 \times 3)$	5	3	2	4	1
Average		2.94	2.28	3.31	1.47

(2) Benferroni-Dunn Test

Now, Bonferroni-Dunn test can be used to further analyze the relative performance among the comparing algorithms. Here, the difference between the average ranks of IVF-MTSVM and one baseline is compared with the following critical difference (CD):

$$CD = q_\alpha \sqrt{\frac{m(m+1)}{6N}}. \tag{33}$$

We have $q_\alpha = 2.128$ at significance level $\alpha = 0.1$, and thus $CD = 0.97(m = 4, N = 16)$. To visually show the relative performance of IVF-MTSVM comparing with other algorithms, Figure 4 provides the CD diagram, where the average ranks of each comparing algorithm are plotted along the axis. The lowest (best) rank on the axis is to the right since we perceive the algorithms on the right side as better. And any comparing algorithm with the average rank within one CD is interconnected with IVF-MTSVM. Otherwise, any other algorithm whose average rank is one CD outside IVF-MTSVM is considered to have significantly different performance with IVF-MTSVM. As shown in Figure 4, the average ranks of MLSTSVM ($2.94 - 1.47 = 1.47 > 0.97$) and MTSVM ($3.31 - 1.47 = 1.84 > 0.97$) are both one CD outside IVF-MTSVM, while the average rank of MWLTSVM ($2.28 - 1.41 = 0.87 < 0.97$) is less than but close to one CD. The results show that IVF-MTSVM is statistically significantly better than MLSTSVM and MTSVM, but not significantly better than MWLTSVM. However, compared to the other three methods in Table 3, IVF-MTSVM can obtain better Micro F1 on most data sets. This shows that IVF-MTSVM has better generalization ability.

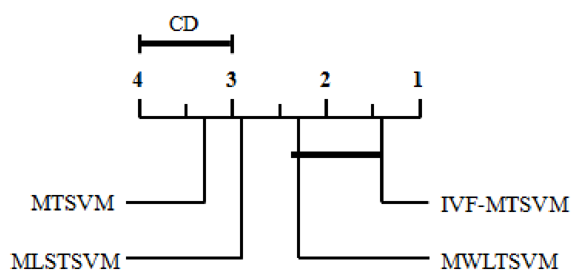

Figure 4: Comparison of the IVF-MTSVM against other three comparing algorithms with the Bonferroni-Dunn test

5 Conclusion

In order to improve the classification performance of MTSVM, IVF-MTSVM is proposed in this paper. A new method of adding fuzzy membership degree to sample points is introduced which is called the interval-valued fuzzy membership degree. Since the previous methods of adding fuzzy membership degree are based on the consideration of the importance of sample points to the class. The method in this paper emphasizes the importance of sample points to the classification model, and takes into account the importance of sample points to the class to some extent, so it is different from the previous methods in thinking. This is a new perspective to establish fuzzy membership degree for sample points in support vector machines. Because the influence of different sample points on the classification model is considered and the effect of outliers is also reduced, the Macro F1 and Micro F1 of MTSVM is improved.

In this paper, when adding interval-valued fuzzy membership to sampling points, the parameters in membership should be selected optimally. Although the grid search method can search for ideal parameters, this work will take some calculation time. In other words, the improvement of Macro F1 and Micro F1 is at the cost of parameter optimization time. Therefore, how to quickly and effectively select the best parameters can be further studied.

6 Acknowledgement

This work was supported by the National Natural Science Foundation of China (NO. 71771028), Beijing Intelligent Logistics System Collaborative Innovation Center, Famous Teacher Program of Beijing Higher Education Institutions and Major scientific research project of Beijing WUZI university (NO. 2019XJZD06).

References

[1] P. T. Berny, I. I. Mohammad, *Multiclass twin bounded support vector machine untuk pengenalan ucapan*, Prosiding Seminar Nasional Penelitian, Pendidikan dan Penerapan MIPA, **22** (2008), 106-111.

[2] C. J. Bo, H. C. Lu, D. Wang, *Hyperspectral image classification via JCR and SVM models with decision fusion*, IEEE Geoence and Remote Sensing Letters, **13**(2) (2016), 177-181.

[3] X. H. Cao, D. Wang, *Hyperspectral imagery classification with cascaded support vector machines and multi-scale superpixel segmentation*, International Journal of Remote Sensing, **41**(12) (2020), 4528-4548.

[4] M. X. Chu, R. F. Gong, S. Gao, *Steel surface defects recognition based on multi-type statistical features and enhanced twin support vector machine*, Chemometrics and Intelligent and Intelligent Laboratory Systems, **171** (2017), 140-150.

[5] C. C. Chuang, *Fuzzy weighted support vector regression with a fuzzy partition*, IEEE Transaction on Systems Man and Cybernetics Part B, **37**(3) (2007), 630-640.

[6] C. Cortes, V. N. Vapnik, *Support vector networks*, Machine Learning, **20**(3) (1995), 273-297.

[7] M. D. De Lima, J. Lima, et al, *Medical data set classification using a new feature selection algorithm*, Medical and Biological Engineering and Computing, **58**(3) (2020), 519-528.

[8] J. Demšar, *Statistical comparisons of classifiers over multiple data set*, Journal of Machine Learning Research, **7** (2006), 1-30.

[9] N. Y. Deng, Y. J. Tian, *Support vector machine-theory, algorithm and expansion*, Machine Learning, Beijing, Science press, 2009.

[10] S. F. Ding, X. K. Zhang, Y. X. An, *Weighted linear loss multiple birth support vector machine based on information granulation for multi-class classification*, Pattern Recognition, **67** (2017), 32-46.

[11] R. O. Duda, P. E. Hart, D. G. Stork, *Pattern classification*, 2ns Edition, John Wiley and Sons, 2001.

[12] B. B. Gao, J. J. Wang, et al, *Coordinate descent fuzzy twin support vector machine for classification*, IEEE International Conference on Machine Learning and Applications, (2015), 7-12.

[13] Jayadeva, R. Khemchandani, S. Chandra, *Fuzzy proximal support vector classification via generalized eigenvalues*, Pattern Recognition and Machine Intelligence, **3776** (2005), 360-363.

[14] Jayadeva, R. Khemchandani, S. Chandra, *Twin support vector machines for pattern classification*, IEEE Transactions on Pattern Analysis and Machine Intelligence, **29**(5) (2007), 905-910.

[15] B. Jiao, Q. Zhang, *A fast intuitionistic fuzzy support vector machine algorithm and its application in wind turbine gearboxes fault diagnosis*, Asia Simulation Conference/International Conference on System Simulation and Scientific Computing, **23** (2012), 297-305.

[16] M. P. Kumar, M. K. Rajagopal, *Detecting facial emotions using normalized minimal feature vectors and semi-supervised twin support vector machines classifier*, Applied Intelligence, **49**(12) (2019), 4150-4174.

[17] K. Y. Lee, D. W. Kim, et al, *Improving support vector data description using local density degree*, Pattern Recognition, **38**(10) (2005), 1768-1771.

[18] C. B. Li, S. S. Lin, et al, *Short-term wind power prediction based on data mining technology and improved support vector machine method: A case study in Northwest China*, Journal of Cleaner Production, **205** (2018), 909-922.

[19] C. F. Lin, S. D. Wang, *Fuzzy support vector machines*, IEEE Transaction on Neural Networks, **13** (2002), 464-471.

[20] Z. B. Liu, Y. Y. Gao, J. Z. Wang, *Automatic classification method of star spectra data based on manifold fuzzy twin support vector machine*, Spectroscopy and Spectral Analysis, **35**(1) (2015), 263-266.

[21] Y. Liu, J. Wu, *A method for multi-class sentiment classification based on an improved one-vs-one (OVO) strategy and the support vector machine algorithm*, Information Sciences, **394** (2017), 38-52.

[22] O. L. Mangasarian, E. W. Wild, *Multisurface proximal support vector machine classification via generalized eigenvalues*, IEEE Transactions on Pattern Analysis and Machine Intelligence, **28**(1) (2006), 69-74.

[23] C. D. Nascimento, S. D. Silva, *Breast tumor classification in ultrasound images using support vector machines and neural networks*, Research on Biomedical Engineering, **32**(3) (2016), 283-292.

[24] J. T. Peng, L. F. Zhang, *Regularized set-to-set distance metric learning for hyperspectral image classification*, Pattern Recognition Letters, **83** (2016), 143-151.

[25] S. Rezvani, X. Z. Wang, F. Pourpanah, *Intuitionistic fuzzy twin support vector machines*, IEEE Transaction on Fuzzy Systems, **27**(11) (2019), 2140-2151.

[26] B. Schölkopf, C. Burges, A. J. Smola, *Advances in kernel methods: Support vector learning*, Machine Learning, Cambridge, MIT Press, 1999.

[27] B. Schölkopf, A. J. Smola, et al, *New support vector algorithms*, Neural Computation, **12**(5) (2000), 1207-1245.

[28] J. A. K. Suykens, J. Vandewalle, *Least squares support vector machine classifiers*, Neural Process Lett, **9**(3) (1999), 293-300.

[29] D. Tomar, S. Agarwal, *An effective weighted multi-class least squares twin support vector machine for imbalanced data classification*, International Journal of Computational Intelligence Systems, **8**(4) (2015), 761-778.

[30] D. Tsujinishi, S. Abe, *Fuzzy least squares support vector machines*, Proceedings of the International Joint Conference on Neural Networks, (2003), 1599-1604.

[31] V. N. Vapnik, *Statistical learning theory*, New York, Wiley, 1998.

[32] Y. Q. Wang, S. Y. Wang, K. K. Lai, *A new fuzzy support vector machine to evaluate credit risk*, IEEE Transaction on Fuzzy Systems, **13**(6) (2005), 820-831.

[33] J. Y. Xie, J. Y. Hone, et al, *Extending twin support vector machine classifier for multi-category; classification problems*, Intelligent Data Analysis, **17**(4) (2013), 649-664.

[34] Y. Xu, R. Guo, L. Wang, *A twin multi-class classification support vector machine*, Cognitive Computation, **5**(4) (2013), 580-588.

[35] W. Ying, Z. O. Wang, J. L. An, *Study on multiclass text categorization method based on improved support vector machine*, Computer Engineering, **32**(16) (2006), 74-76.

[36] L. A. Zadeh, V. N. Vapnik, *Fuzzy sets*, Information and Control, **8** (1965), 338-353.

[37] L. A. Zadeh, V. N. Vapnik, *Outline of a new approach to the analysis of complex systems and decision processes, interval-valued fuzzy sets*, IEEE Transactions on Systems, Man, and Cybernetics, **3**(1) (1973), 28-44.

[38] S. L. Zhang, W. Lu, et al, *Using fuzzy least squares support vector machine with metric learning for object tracking*, Pattern Recognition, **84** (2018), 112-125.

[39] F. Zhu, J. Yang, et al, *Extended nearest neighbor chain induced instance-weights for SVMs*, Pattern Recognition, **60** (2016), 863-874.