JOURNAL OF INFORMATION SCIENCE AND ENGINEERING 35, 1365-1376 (2019)
DOI: 10.6688/JISE.201911_35(6).0012

Multiple Birth Support Vector Machine with Triplet Loss Function

SHI-FEI DING[1,2] AND YUE-XUAN AN[1,2]
[1]*School of Computer Science and Technology*
China University of Mining and Technology
Xuzhou, 221116 P.R. China
[2]*Mine Digitization Engineering Research Center*
Ministry of Education of the People's Republic of China
Xuzhou, 221116 P.R. China
E-mail: dingsf@cumt.edu.cn

TWSVM which can be regarded as an efficient binary classification algorithm has achieved much attention. However, when it comes to multi-class classification, scholars have to consider other algorithms. Therefore, many multi-class TWSVM has been proposed to deal with multi-class problems. In this paper, we use multiple birth support vector machines (MBSVM) which is an efficient algorithm for multi-class classification in which the decision criterion is the farthest distance of the test pattern to the hyper-planes, rather than the closest distance in multi-class TWSVM. The algorithm has much lower computational complexity and can be expected to be faster than the existing multi-class SVMs. However, when facing multi-class problem of imbalanced data, the MBSVM which adopts hinge loss is easily leads to instability for resampling. To enhance the performance of the MBSVM, we present a novel MBSVM with the triplet loss (tMBSVM) which deals with the imbalanced dataset problems and shows differences between positive data and negative data in one class. Numerical experiments on data sets demonstrate the feasibility and validity of our proposed method.

Keywords: SVM, TWSVM, MBSVM, multi-classification, machine learning

1. INTRODUCTION

Support vector machine (SVM) [1] based on statistical learning is proposed by Vapnik in 1995. Compared with other machine learning methods, it can use the principle of "margin maximization" by using the hinge loss function to reduce the structural risk and enhance its generalization capability. When facing non-linear problems, the kernel function can be applied to SVM to map the training data into higher-dimensional space and transform a nonlinear classification problem to into a linear classification problem in a high dimensional space. Due to its high performance, it has been widely applied and well-studied [2-5].

Twin support vector machine (TWSVM) is a variance of SVM [6], it can generate two nonparallel hyperplanes by solving a pair of smaller-sized quadratic programming (QP) problems rather than a single larger-sized QP problem. Therefore, TWSVM works 4 times faster than the standard SVM and can be applied to large-scale data sets. Many people have been devoted to the effective algorithm and proposed numerous improved TWSVMs [7-11]. However, compared with binary classification methods, the multi-

Received August 1, 2018; revised January 1, 2019; accepted March 25, 2019.
Communicated by Tzung-Pei Hong.

class classification methods can attain more extensive applications. Therefore, many scholars dedicate to extend TWSVM from binary classification to multi-class classification. Several methods have been proposed to extend TWSVM to multi-class classification field such as "one-versus-one" strategy, "one-versus-rest" strategy, "one-versus-one-versus-rest" strategy, "binary tree" based methods [12-14]. Furthermore, Yang [15] used the "all-versus-one" strategy and proposed a multiple birth support vector machine (MBSVM). The decision criterion in MBSVM is the farthest distance of the test pattern to the hyperplanes instead of the closest distance in TWSVM [16]. Compared with other multi-class TWSVMs, it has the characteristics of lower computational complexity and can be expected to be faster than the existing multi-class TWSVMs [17]. However, the MBSVM which adopts hinge loss is easily leads to instability for resampling when facing multi-class problem of imbalanced data. Furthermore, hyper-parameters in MBSVM which is always the results of artificial selection are hard to optimize and the chosen hyper-parameters are always not accurate.

Inspired by the studies of the MBSVM. In this paper, we introduce triplet loss function to MBSVM and proposed a novel triplet loss multiple birth support vector machine (tMBSVM), which can deal with the imbalanced dataset problems in MBSVM and show differences between positive data and negative data. Experimental results show that our tMBSVM achieves a significant performance.

This paper is organized as follows. Section 1 outlines the related works. The tMBSVM is proposed in Section 2. In section 3, the final algorithms proposed is verified through the experiments on the UCI benchmark datasets. The last section is the conclusion and the forecast part.

2. RELATED WORK

In this section, we introduce the mathematical model and geometric interpretation of twin support vector machines TWSVM, MBSVM. TWSVM is an improved algorithm of support vector machine proposed by Javadeva and Khemchandani in 2007. TWSVM needs to solve a hyperplane for each class separately. Therefore, when solving the two classification problems, TWSVM needs to solve two QP problems. However, the QP problems of TWSVM are small and its algorithm training speed is faster than that of SVM. The linear TWSVM and nonlinear TWSVM are described in detail below.

2.1 Twin Support Vector Machines

Given a training dataset $T = \{(x_1, y_1), (x_2, y_2), \ldots, ((x_N, y_N)\}$ in input space χ where $x_i \in \chi = \mathbf{R}^n$ and $y_i \in \{+1, -1\}$ is class label of x_i. Matrices A and B are comprised of the patterns of positive class and negative class in the training set respectively. For the linear separable binary classification problem, the goal of TWSVM is to find two non-parallel hyperplanes

$$\boldsymbol{x}^T\boldsymbol{w}_1 + b_1 = 0 \text{ and } \boldsymbol{x}^T\boldsymbol{w}_2 + b_2 = 0 \tag{1}$$

which make each hyperplane closer to the patterns in one of the two classes and as far as possible from the other. The hyperplanes are generally obtained by solving the following QP problems

$$\begin{aligned}&(\text{TWSVM1}) \quad \min \quad \frac{1}{2}(A\boldsymbol{w}_1+e_1b_1)^T(A\boldsymbol{w}_1+e_1b_1)+c_1e_2^T\xi \\ &\qquad\qquad \text{subject to} \quad -(B\boldsymbol{w}_1+e_2b_1)+\xi \geq e_2, \xi \geq 0\end{aligned} \tag{2}$$

$$\begin{aligned}&(\text{TWSVM2}) \quad \min \quad \frac{1}{2}(B\boldsymbol{w}_2+e_2b_2)^T(B\boldsymbol{w}_2+e_2b_2)+c_2e_1^T\eta \\ &\qquad\qquad \text{subject to} \quad (A\boldsymbol{w}_2+e_1b_2)+\eta \geq e_1, \eta \geq 0\end{aligned} \tag{3}$$

where c_1 and c_2 are penalty parameters, e_1 and e_2 are column vectors of ones, ξ and η are slack variables.

For non-linear problem, kernel function can map the data in the original space to a new space by a nonlinear transformation. Then TWSVM can learn the classification model from the training data in the new space with a linear classification method. $\phi(x)$ is defined as the mapping function from the input space χ to the feature space $\mathcal{H}$. $K(x, z)$ is defined as $K(x, z) = \phi(x)\phi(y)$. Generally, we use the Radial Basis Function (RBF) as the kernel function.

By introducing kernel function to TWSVM and constructing matric C, that is $C^T = [A\ B]^T$, the counterpart of the problem (2)-(3) should be

$$\begin{aligned}&(\text{TWSVM1}) \quad \min \frac{1}{2}\left\|K\left(A,C^T\right)\boldsymbol{u}_1+e_1b_1\right\|^2+c_1e_2^T\xi \\ &\quad \text{subject to} \quad -\left(K\left(B,C^T\right)\boldsymbol{u}_1+e_2b_1\right)+\xi \geq e_2, \xi \geq 0\end{aligned} \tag{4}$$

$$\begin{aligned}&(\text{TWSVM2}) \quad \min \frac{1}{2}\left\|K\left(B,C^T\right)\boldsymbol{u}_2+e_2b_2\right\|^2+c_2e_1^T\eta \\ &\quad \text{subject to} \quad \left(K\left(A,C^T\right)\boldsymbol{u}_2+e_1b_2\right)+\eta \geq e_1, \eta \geq 0\end{aligned} \tag{5}$$

where c_1, c_2 are penalty parameters, e_1, e_2 are column vectors of ones, ξ and η are slack variables.

The new sample which should be classified is assigned to one of the classes depending on which of the two planes it lies closest to. The decision function is given by

$$Label(x) = \arg\min_{k=1,2}\left(\left|K\left(x^T,C^T\right)\boldsymbol{u}_k+b_k\right| \Big/ \sqrt{\boldsymbol{u}_k^T K\left(C^T,C^T\right)\boldsymbol{u}_k}\right). \tag{6}$$

2.2 Multiple Birth Support Vector Machine

To extend TWSVM to multi-class classifications, scholars propose the theory of "one-versus-rest" (o-v-r) method, which constructs K binary classifiers for K-class classification problem. For one class, it takes the kth class as one class and considers the rest classes as the other class to construct a QP problem. MBSVM has the characteristics of lower computational complexity and can be expected to be faster than the existing multi-class TWSVMs for it extends TWSVM to multi-class classification by another theory "all-versus-one" which considers one of the classes as negative class and all the rest classes as positive class in turn to generate a serious of binary sub-classifiers to solve the multi-class classification problem [18, 19].

We take the K-classification problem is as an example, where A_k is comprised of the pattern of kth class, $B_k = \{A_1, A_2, \ldots, A_{k-1}, A_{k+1}, \ldots, A_{K-1}, A_K\}$ denotes all samples except the samples belonging to the pattern of kth class. Each class corresponds to one hyperplane and the hyperplane is assigned class label according to the pattern the hyperplane is farthest to. Therefore, MBSVM should deal with K QP problems and generate K hyperplanes. The QP problem of the kth class is given by

$$\begin{array}{ll} \min\limits_{w_k, b_k, \xi_k} & \frac{1}{2}\left\|B_k \boldsymbol{w}_k + e_{k1} b_k\right\|^2 + c_k e_{k2}^T \boldsymbol{\xi}_k \\ \text{subject to} & (A_k \boldsymbol{w}_k + e_{k2} b_k) + \boldsymbol{\xi}_k \geq e_{k2} \\ & \boldsymbol{\xi}_k \geq 0 \end{array} \tag{7}$$

where e_{k1} and e_{k2} are column vectors of ones, $\boldsymbol{w}_k$ and b_k are a normal vector and a bias term to the hyperplane respectively, and ξ_k is a slack variable. By introducing kernel function, the primal QP problem is turned to

$$\begin{array}{ll} \min\limits_{\boldsymbol{u}_k, b_k, \xi_k} & \frac{1}{2}\left\|K\left(B_k^T, C^T\right)\boldsymbol{u}_k + e_{k1} b_k\right\|^2 + c_k e_{k2}^T \boldsymbol{\xi}_k \\ \text{subject to} & (K\left(A_k^T, C^T\right)\boldsymbol{u}_k + e_{k2} b_k) + \boldsymbol{\xi}_k \geq e_{k2} \\ & \boldsymbol{\xi}_k \geq 0, \quad k = 1, 2, \ldots, K \end{array} \tag{8}$$

where u_k and b_k are variables corresponding to the hyperplane of the kth class.

When the samples are classified, the class label of one sample point is selected according to the class label of the hyperplane which the sample point is farthest to. The decision function is given by

$$Label(x) = \arg\min_{k=1,2,\ldots K}\left(\left|K\left(x^T, C^T\right)\boldsymbol{u}_k + b_k\right| \Big/ \sqrt{\boldsymbol{u}_k^T K\left(C^T, C^T\right)\boldsymbol{u}_k}\right). \tag{9}$$

The following figure is the illustration of MBSVM.

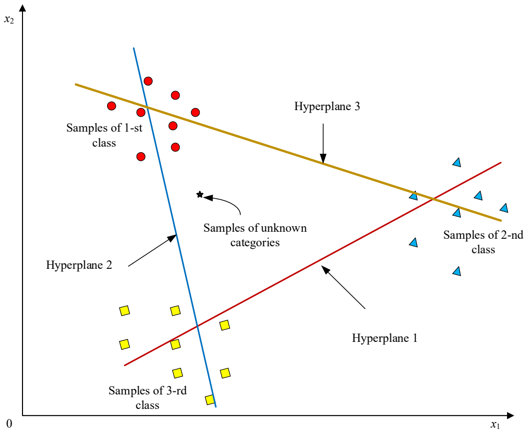

Fig. 1. Illustration diagram of MBSVM.

3. TRIPLET LOSS MULTIPLE BIRTH SUPPORT VECTOR MACHINE

MBSVM is an efficient algorithm which integrates the characteristics of TWSVM. However, the MBSVM is easily leads to instability for resampling when facing multi-class problem of imbalanced data, especially the number of classes is incredibly large. MBSVM could not well handle it and cannot achieve satisfactory results. In this section, we apply triplet function to MBSVM (tMBSVM) and introduce our improved MBSVM in detail.

3.1 Triplet Loss Function

Since the proposal of FaceNet [20], triplet loss function has attracted wide attention for it is able to adapt flexibly to various space shapes, achieve better feature and the threshold in triple loss can control the distance between positive samples and negative samples.

Triplet is made up of a three tuple: randomly select a sample from the training data set, which is called anchor data x^a, a positive data with the same label as anchor data and a negative data with different label. The two samples correspond to Positive (x^p) and Negative (recorded as x^n). The three tuple constitutes a (Anchor, Positive, Negative). The distances between the anchor data and positive data should be less than the distance between the anchor data and negative data. The purpose of triplet loss is to make the distance of feature expressions between x^a and x^p is as small as possible, the distance of feature expressions between x^a and x^n is as large as possible and there is a minimum margin between the distance between x^a and x^n and the distance between x^a and x^p. Assume we randomly select N anchor samples in the m dimension space. Adding a threshold α, all triplet should meet

$$\left\| f(x_i^a) - f(x_i^p) \right\|_2^2 + \alpha < \left\| f(x_i^a) - f(x_i^n) \right\|_2^2 \tag{10}$$

where ||*|| is Euclidean distance. This inequality essentially defines the distance relation between samples belonging to the same class and samples in different classes. The corresponding loss function is

$$\sum_{i}^{N} \left[\left\| f(x_i^a) - f(x_i^p) \right\|_2^2 - \left\| f(x_i^a) - f(x_i^n) \right\|_2^2 + \alpha \right]_+ \tag{11}$$

where $[*]_+ = \max(0, *)$. $\|f(x_i^a) - f(x_i^p)\|_2^2$ represents the Euclidean distance between the positive data selected and ith anchor data, $i = 1, 2, 3, \ldots, N$. $\|f(x_i^a) - f(x_i^n)\|_2^2$ represents the Euclidean distance between the negative data selected and ith anchor data, $i = 1, 2, 3, \ldots, N$.

3.2 The Model of tMBSVM

In Section 3.1, we discuss the triplet loss in detail. We find that triplet loss can make sure the distance of between samples in same class as small as possible and the distance of samples in different class is as large as possible and there is a minimum margin between the distance of data in same class and the distance of data in different class instead of the single constraint in MBSVM that the distance between data of one class and hyperplane of the other class is at least one. Therefore, the triplet loss function can take into accounts more information of data and control the differences between positive data and negative data. For we select the same number of positive and negative samples for calculation so it can deal with imbalanced dataset problems in MBSVM to some extent. Assume that there are K class (K hyperplanes). We take the kth class for example and add a triplet loss item into MBSVM. The tMBSVM constructs the kth hyperplane by solving the following problem.

$$\begin{aligned} \min_{w_k, b_k, \xi_k} \quad & \frac{1}{2}\|B_k \boldsymbol{w}_k + e_{k1} b_k\|^2 + c_k e_{k2}^T \boldsymbol{\xi}_k \\ \text{subject to} \quad & (A_k \boldsymbol{w}_k + e_{k2} b_k) + \boldsymbol{\xi}_k \geq e_{k2} \\ & \left\|(D_k \boldsymbol{w}_k + e_k b_k) - (A_k^D \boldsymbol{w}_k + e_k b_k)\right\|_2^2 + \alpha \\ & < \left\|(D_k \boldsymbol{w}_k + e_k b_k) - (B_k^D \boldsymbol{w}_k + e_k b_k)\right\|_2^2 \\ & \boldsymbol{\xi}_k \geq 0 \end{aligned} \tag{12}$$

where D_k is the anchor data randomly selected in kth class, A_k^D is also randomly selected in kth class (A_k) with the same data number of D_k. B_k^D is selected in B_k with the same data number of D_k. By constructing the Lagrange function, we can get the following functions derived from Eq. (12).

$$\begin{aligned} J_{t\ MBSVM}^{\ k} = & \frac{1}{2}\|B_k \boldsymbol{w}_k + e_{k1} b_k\|^2 + c_k e_{k2}^T \max(0, e_{k2} - (A_k \boldsymbol{w}_k + e_{k2} b_k)) \\ & + \lambda e_k^T \max(\left\|(C_k \boldsymbol{w}_k + e_k b_k) - (A_k^c \boldsymbol{w}_k + e_k b_k)\right\|_2^2 \\ & + \alpha - \left\|(C_k \boldsymbol{w}_k + e_k b_k) - (B_k^c \boldsymbol{w}_k + e_{k2} b_k)\right\|_2^2) \end{aligned} \tag{13}$$

where c_k and λ are penalty parameters, e_{k1}, e_{k2} and e_k are column vectors of ones.

Solving K times QP problem requires a lot of computation time. To solve this problem, we introduce the gradient-based optimization method to MBSVM, and use the gradient descent optimization method to optimize the parameters of MBSVM. Through this method, solving K times QP problem directly is avoided, and the efficiency of the algorithm is improved effectively.

The gradient descent optimization is used to update the parameters in Eq. (13). We can get the single step update of the parameter of the tMBSVM.

$$\boldsymbol{w}_k \leftarrow \boldsymbol{w}_k - \eta \frac{\partial J_{t-MBSVM}^k}{\partial \boldsymbol{w}_k} \tag{14}$$

$$b_k \leftarrow b_k - \eta \frac{\partial J_{t-MBSVM}^k}{\partial b_k} \tag{15}$$

where η is a learning rate.

By introducing kernel function, the primal QP problem is turned to

$$\begin{aligned}
&\min_{\boldsymbol{u}_k, b_k, \xi_k} && \frac{1}{2}\left\|K\left(B_k^T, C^T\right)\boldsymbol{u}_k + e_{k1}b_k\right\|^2 + c_k e_{k2}^T \boldsymbol{\xi}_k \\
&\text{subject to} && (K\left(A_k^T, C^T\right)\boldsymbol{u}_k + e_{k2}b_k) + \boldsymbol{\xi}_k \geq e_{k2} \\
& && \left\|(K\left(D_k^T, C^T\right)\boldsymbol{u}_k + e_k b_k) - (K\left(A_k^{DT}, C^T\right)\boldsymbol{u}_k + e_k b_k)\right\|_2^2 + \alpha \\
& && < \left\|(K\left(D_k^T, C^T\right)\boldsymbol{u}_k + e_k b_k) - K\left(B_k^{DT}, C^T\right)\boldsymbol{u}_k + e_k b_k)\right\|_2^2 \\
& && \boldsymbol{\xi}_k \geq 0,\ \ k = 1, 2, \ldots, K
\end{aligned} \tag{16}$$

where $\boldsymbol{u}_k$ and b_k are variables corresponding to the hyperplane of the kth class.

On substituting the constraints into the objective function, QP problem in Eq. (16) becomes

$$\begin{aligned}
J_{tMBSVMk} = \ & \frac{1}{2}\left\|K\left(B_k^T, C^T\right)\boldsymbol{u}_k + e_{k1}b_k\right\|^2 + c_k e_{k2}^T\left(e_{k2}\text{-}(K\left(A_k^T, C^T\right)\boldsymbol{u}_k + e_{k2}b_k)\right)\boldsymbol{\xi}_k \\
& +\lambda e_k(\left\|(K\left(D_k^T, C^T\right)\boldsymbol{u}_k + e_k b_k) - (K\left(A_k^{DT}, C^T\right)\boldsymbol{u}_k + e_k b_k)\right\|_2^2 \\
& + \alpha\text{-}\left\|(K\left(D_k^T, C^T\right)\boldsymbol{u}_k + e_k b_k) - K\left(B_k^{DT}, C^T\right)\boldsymbol{u}_k + e_k b_k)\right\|_2^2) \\
& k = 1, 2, \ldots, K.
\end{aligned} \tag{17}$$

The gradient descent optimization is used to update the parameters in Eq. (17).

$$\boldsymbol{u}_k \leftarrow \boldsymbol{u}_k - \eta \frac{\partial J_{t-MBSVM}^k}{\partial \boldsymbol{u}_k} \tag{18}$$

$$b_k \leftarrow b_k - \eta \frac{\partial J_{t-MBSVM}^k}{\partial b_k} \tag{19}$$

where η is a learning rate. In order to reduce the impact of the selection of anchor data on the optimization results, we set the algorithm to perform gradient descent optimization for many times to get the final result. So each sample has the same probability to be chosen. By choosing different data as anchor data, we can alleviate the impact over different sampled anchor data.

When the samples are classified, the class label of one sample point is selected according to the class label of the hyperplane which the sample point is farthest to. The decision function is given by

$$Label(x) = \arg\min_{k=1,2,\ldots K}\left(\left|K\left(x^T, C^T\right)\boldsymbol{u}_k + b_k\right| \Big/ \sqrt{\boldsymbol{u}_k^T K\left(C^T, C^T\right)\boldsymbol{u}_k} \right) \tag{20}$$

We can get the problem to be optimized as shown in Eq. (16) by introducing the kernel function. At this point, we can still optimize the parameters by gradient descent. Gradient descent is used to optimize the parameters.

The operation process of the MBSVM with triplet loss function can be described as follows. First, load dataset, and divide the dataset into two parts, training dataset and test dataset. Second, initialize the parameters of MBSVM, Input the training set into the tMBSVM. Then, train the model based on Eqs. (17)-(19). Finally, construct the decision Eq. (20).

The algorithm of tMBSVM is as follows:

```
Algorithm: MPSO-tMBSVM training algorithm
Given a training set $\mathcal{D} = \{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \ldots, (\mathbf{x}_m, y_m)\}$
        For $k \leftarrow 1$ to $K$:
do
    Load dataset, and divide the dataset into two parts, training dataset and test dataset.
    Initialize the parameters of tMBSVM.
    Input the training set into the tMBSVM
    train the model based on Eqs. (17)-(19).
    Construct the decision Eq. (20).
End for
```

When classifying new input samples, we first input samples into tMBSVM and finally get the classification according to Eq. (20).

4. EXPERIMENTS AND ANALYSIS

In this section, we present experimental results of our algorithms on several typical datasets in UCI Machine Learning Repository to verify the effectiveness of our algorithms. The UCI Machine Learning Repository is a collection of databases, domain theories, and data generators that are used by the machine learning community for the empirical analysis of machine learning algorithms. In order to ensure the reliability of experiments, ten-fold cross validation is adopted to this paper for experiments. We choose Radial Basis Function (RBF) as kernel function and the definition of kernel function is given by

$$K(x,z) = \exp\left(-\frac{\|x-z\|^2}{2\sigma^2}\right) \tag{21}$$

where σ is hyper parameters for controlling RBF kernel effect.

All detail characteristics of UCI datasets are shown in Table 1. It can be seen in Table 1 that balance, ecoli, customers, diabetes and glass which are imbalanced datasets are used to show that our algorithm can deal with the imbalanced problems. We also use the balanced dataset to illustrate our algorithm has robust in any environments.

All algorithms were implemented in Python 3.6.1 and Mxnet 1.0.0 on a PC with Intel i5-6300HQ quad core processor, 10 GB RAM and Microsoft Windows 10. Table 1 gives detailed information of the experimental datasets used in this paper.

Table 1. Detail characteristics of the UCI datasets.

Datasets	Number of sample	Dimension	Number of classes	The number of each class sample
balance	625	4	3	[49 288 288]
banknote	1372	4	2	[762 610]
cryotherapy	90	6	2	[42 48]
seeds	210	7	3	[70 70 70]
ecoli	336	7	8	[143 77 2 2 35 20 5 52]
customers	440	7	2	[298 142]
diabetes	768	8	2	[268 500]
glass	214	9	6	[70 76 17 13 9 29]
wine	178	13	3	[59 71 48]
australian	690	14	2	[307 383]

The experiments of the paper totally use 10 data sets. Before implementing experiments on the data sets, we need to implement a normalization processing and define the range of eigenvectors as [0,1]. The detailed characteristics of these problems are shown in Table 1. In the experiments, we select TWSVM, MBSVM, least square multiple birth vector machine (LSMBSVM) and weighted linear loss multiple birth vector machine (WLMBSVM) to be compared with our algorithm and compare the classification results of the proposed method with these algorithms. The experimental results are the average of the 10 experimental results in 10-fold cross validation. The hyper-parameters of the experiment are set as follows: learning rate $\eta = 0.01$, penalty parameter $C = 1.0$, the parameters of RBF kernel $\sigma^2 = 0.1$, and the maximum number of iterations is set to 50. The average accuracy of classification results are shown in Table 2.

Table 2. The accuracy of experimental results.

Datasets	TWSVM	MBSVM	LSMBSVM	WLMBSVM	tMBSVM
balance	86.37	87.52	87.71	89.92	90.22
banknote	80.76	79.30	53.52	48.55	82.05
cryotherapy	84.00	86.75	87.89	85.61	88.81
customers	75.94	76.36	55.27	57.26	77.96
ecoli	68.63	66.39	24.42	21.07	75.92
seeds	90.00	90.00	90.00	89.52	91.90
diabetes	67.07	67.19	47.36	52.50	70.05
glass	50.03	52.36	26.34	30.21	56.89
wine	97.77	97.71	97.22	97.74	97.77
australian	84.76	85.06	83.31	83.35	85.36

From the results shown in Table 2, we can find that the classification accuracies of tMBSVM are generally higher than the algorithms compared. tMBSVM always gets the highest accuracy. In addition, the performances of tMBSVM on balance, ecoli, customers, diabetes and glass which are representatives of imbalanced data sets approach the best performances. That means tMBSVM can show a much superiority on the problems which have imbalanced datasets. Meanwhile, the performances of tMBSVM in balanced datasets are not bad, which can be shown in Table 2. In general, the general performance of tMBSVM is better than the other algorithms compared. These results indicate that the proposed algorithm is efficient and robust. The reason is that tMBSVM carries on the good generalization ability of tMBSVM.

5. CONCLUSION AND DISCUSSION

For multiple birth support vector machine, it cannot take into accounts the relevant information between different samples of classes, which limits to further improve the classification accuracy of the problem. In this paper, we introduce triplet loss into MBSVM and propose a novel multiple birth support vector machine which is called tMBSVM. The tMBSVM can achieve better feature and the threshold in triple loss can control the distance between positive samples and negative samples. Meanwhile, our algorithm can achieve the effect of fast convergence. Therefore, the triplet loss multiple birth support vector machine (tMBSVM) can maximum the distance between data in one class and data in the rest class, reduce the influence of the imbalanced data to some extent. The experimental results show that our algorithm can efficiently enhance the classification accuracy and. In the future work, we will further improve the generality of the algorithm, the speed of parameter selection and the efficiency of the algorithm.

ACKNOWLEDGMENTS

This work is supported by the National Natural Science Foundation of China under Grant No. 61672522 and No. 61379101.

REFERENCES

1. V. N. Vapnik and C. Cortes, “Support vector networks,” *Machine Learning*, Vol. 20, 1995, pp. 273-297.
2. Y. X. An, S. F. Ding, and S. H. Shi, “Discrete space reinforcement learning algorithm based on support vector machine classification,” *Pattern Recognition Letters*, Vol. 11, 2018, pp. 30-35.
3. Y. Liu, K. W. Wen, Q. X. Gao, X. B. Gao, and F. P. Nie, “SVM based multi-label learning with missing labels for image annotation,” *Pattern Recognition*, Vol. 78, 2018, pp. 307-317.
4. J. X. Wu and H. Yang, “Linear regression-based efficient SVM learning for large-scale classification,” *IEEE Transactions on Neural Networks and Learning Systems*, Vol. 26, 2017, pp. 2357-2369.
5. B. Gu, Victor, and S. Sheng, “A robust regularization path algorithm for ν-support vector classification,” *IEEE Transactions on Neural Networks and Learning Systems*, Vol. 28, 2017, pp. 1241-1248.
6. D. Jayadeva, R. Khemchandani, and S. Chandra, “Twin support vector machines for pattern classification,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 29, 2007, pp. 905-910.

7. S. F. Ding, X. K. Zhang, and J. Z. Yu, "Twin support vector machines based on fruit fly optimization algorithm," *International Journal of Machine Learning and Cybernetics*, Vol. 7, 2016, pp. 193-203.
8. S. F. Ding, Y. X. An, X. Zhang, F. L. Wu, and Y. Xue, "Wavelet twin support vector machines based on glowworm swarm optimization," *Neurocomputing*, Vol. 225, 2016, pp. 157-163.
9. L. Cao and H. Shen, "Combining re-sampling with twin support vector machine for imbalanced data classification," in *Proceedings of International Conference on Parallel and Distributed Computing, Applications and Technologies*, 2016, pp. 325-329.
10. R. Rastogi and P. Saigal, "Tree-based localized fuzzy twin support vector clustering with square loss function," *Applied Intelligence*, Vol. 47, 2017, pp. 96-113.
11. R. Rastogi and S. Sharma, "Fast Laplacian twin support vector machine with active learning for pattern classification," *Applied Soft Computing*, Vol. 74, 2019, pp. 424-439.
12. S. F. Ding, X. Y. Zhao, J. Zhang, X. K. Zhang, and Y. Xue, "A review on multi-class TWSVM," *Artificial Intelligence Review*, 2017, DOI: 10.1007/s10462-017-9586-y.
13. C. N. Li, Y. F. Huang, and H. J. Wu, "Multiple recursive projection twin support vector machine for multi-class classification," *International Journal of Machine Learning & Cybernetics*, Vol. 7, 2016, pp. 729-740.
14. R. Khemchandani and A. Pal, "Tree based multi-category Laplacian TWSVM for content based image retrieval," *International Journal of Machine Learning and Cybernetic*s, Vol. 8, 2016, pp. 1-14.
15. Z. X. Yang, Y. H. Shao, and X. S. Zhang, "Multiple birth support vector machine for multi-class classification," *Neural Computing and Applications*, Vol. 22, 2013, pp. 153-161.
16. S. G. Chen and X. J. Wu, "Multiple birth least squares support vector machine for multi-class classification," *International Journal of Machine Learning and Cybernetics*, Vol. 8, 2017, pp. 1731-1742.
17. X. K. Zhang, S. F. Ding, and T. F. Sun, "Multi-class LSTMSVM based on optimal directed acyclic graph and shuffled frog leaping algorithm," *International Journal of Machine Learning and Cybernetics*, Vol. 7, 2016, pp. 241-251.
18. S. F. Ding, X. K. Zhang, Y. X. An, and Y. Xue, "Weighted linear loss multiple birth support vector machine based on information granulation for multi-class classification," *Pattern Recognition*, Vol. 67, 2017, pp. 32-46.
19. X. K. Zhang , S. F. Ding, and Y. Xue, "An improved multiple birth support vector machine for pattern classification," *Neurocomputing*, Vol. 225, 2017, pp. 119-128.
20. F. Schroff, D. Kalenichenko, and J. Philbin, "FaceNet: A unified embedding for face recognition and clustering," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 815-823.

Shi-Fei Ding (丁世飞) received his Ph.D. degree from Shandong University of Science and Technology in 2004. He received postdoctoral degree from Key Laboratory of Intelligent Information Processing, Institute of Computing Technology, and Chinese Academy of Sciences in 2006. He is a Professor and Ph.D. supervisor at China University of Mining and Technology. His research interests include intelligent information processing, pattern recognition.

Yue-Xuan An (安悦瑄) received her B.Sc. degree in Computer Science from Jiangsu Normal University in 2014. She is currently pursuing the Ph.D. degree in School of Computer Science and Technology, China University of Mining and Technology, and her supervisor is Prof. Shifei Ding. Her research interests include machine learning, and support vector machines *et al.*