IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received December 13, 2018, accepted January 3, 2019, date of publication February 8, 2019, date of current version March 1, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2897891

Twin Bounded Weighted Relaxed Support Vector Machines

FATEMEH ALAMDAR, FATEMEH SHEYKH MOHAMMADI, AND ALI AMIRI
Department of Computer Engineering, University of Zanjan, Zanjan 45371-38791, Iran

Corresponding author: Fatemeh Alamdar (fatemeh.alamdar@znu.ac.ir)

ABSTRACT Data distribution has an important role in classification. The problem of imbalanced data has occurred when the distribution of one class, which usually attends more interest, is negligible compared with other class. Furthermore, by the existence of outliers and noise, the classification of these data confronts more challenges. Despite these challenges, doing fast classification with good performance is desired. One of the successful classifier methods for dealing with imbalanced data and outliers is weighted relaxed support vector machines (WRSVMs). In this paper, the improved twin version of this classifier, which is called twin-bounded weighted relaxed support vector machines, is introduced to confront the mentioned challenges; besides, it performs in a significant fast manner and it is more accurate in most cases. This method benefits from the fast classification manner of twin-bounded support vector machines and outlier robustness capability of WRSVM in the imbalanced problems. The experimentally, the proposed method is compared with the WRSVM and other standard SVM-based methods on the public benchmark datasets. The results confirm the efficiency of the proposed method.

INDEX TERMS Twin support vector machines, weighted support vector machine, relaxed support vector machine, imbalanced data classification, fast classification, outliers.

I. INTRODUCTION

In machine learning, the purpose of classification is to learn a mathematical model by using a set of training instances that are capable of predicting the unknown class labels of testing instances with a good generalization ability.

Imbalanced classification occurs when the number of instances which represents one class is smaller than the other classes [1]. Moreover, the class with the lowest number of samples is usually the class of interest from the standpoint of the learning work [2]. The Classification of imbalanced data can also be known as skewed class distribution in the literature [3], [4]. Imbalanced data usually suffer from small sample size, class overlapping or small disjuncts, which is caused the classifier learning become arduous [5]–[7].

In the problem of binary classification, the number of instances of a class may extremely outnumber in comparison with the other class. The class with more instances is described as the majority class and the class with fewer instances is defined as a minority class. In the imbalanced problem, standard classifier learning algorithms have a bias towards the majority class. Therefore, the minority class instances are more often misclassified.

Imbalanced classification problem is of great interest since it found in many real-world classification problems such as fraud detection [8], remote sensing [9], pollution detection [10], risk management [11], face recognition [12], detection of oil spills [13], disease diagnosis [14]–[16], defect detection [17], email spam detection [18], customer segmentation and marketing [19], security surveillance [20], network intrusion detection [21], bioinformatics [22], manufacturing [23], power quality disturbance [24], and quality estimation [25].

The classification of imbalanced data in the presence of noise and outliers can be more arduous. Factors such as subjectivity, data-entry error or the lack of adequate information that used to label each sample caused the outlier in the data [26]. In different areas, the imbalanced classification problems in the attendance of noise or mislabeled data can be found. For example, in defect detection in manufacturing, data collected by sensors might be influenced by the environmental variables or provisional malfunctioning and this might affect the labeling of the data [27]. In medical

The associate editor coordinating the review of this manuscript and approving it for publication was Hengyong Yu.

diagnosis, wrong diagnosis or error in labeling patients often takes place and have a negative impact on medical decision-making [28].

Cost-sensitive learning is one of the standard approaches for imbalanced classification problems. This approach assigns various weights to each data sample based on its importance in the model and solves the weighted classification problem. The effect of outliers of the minority class is magnified under the cost-sensitive framework [29], [30].

Support vector machine (SVM) is one of the most popular supervised learning algorithms that first proposed by Vapnik [31] and is based on structural risk minimization. In this method, an optimal decision boundary is constructed which can separate two classes of samples with the maximal margin [32]. The SVM formulations are usually modeled and solved as a convex quadratic optimization problem for detecting an optimal hyperplane. SVM has advantages such as robustness, good generalization capability and unique global optimum solution in the case of the convex problem [33].

Noise and outliers can have a negative impact on the decision boundary. A number of studies in the field of SVM considered this issue and theoretically presented a robust formulation [34], [35]. Nevertheless, the computational challenges imposed by the resulting non-convex goals. Effective methods have been presented also for robust SVM with convex goals [36]–[40].

One of the proposed methods to reduce the effect of outliers on the separating hyperplane in SVM is RSVM [39]. The basic idea behind it is to provide a restricted amount of penalty-free slack for data samples that may prevent the classification performance. Free slack causes to support vectors partly be relaxed and push them towards their respective classes. Another method for this issue proposed in [40]. This method which is called WRSVM assigned different weights and free slack amount to the positive and negative class.

Twin support vector machine (TWSVM) [41] create two nonparallel hyperplanes such that each hyperplane is closer to one of two classes and is quite distant from the other class. It is implemented by solving two smaller quadratic programming problems (QPPs) instead of a single large QPP, which makes the training speed of TWSVM faster than the standard SVM. Furthermore, TWSVM has a good generalization capability.

Twin bounded support vector machine (TBSVM) [42], like TWSVM, constructs two nonparallel hyperplanes by solving two smaller QPPs. The considerable advantage of TBSVM over TWSVM is the structural risk minimization which is implemented by adding a regularization term with the purpose of maximizing the margin. This modification increases the performance of classification. TBSVM has better generalization ability than TWSVM.

There are many developed version of SVM which try to make it faster by constructing and solving two optimization problems instead of one [43]–[50], but TWSVM and TBSVM are the most outstanding well-known ones.

Being inspired by the success of TBSVM method, we present, in this study, a new Twin Bounded Weighted Relaxed Support Vector Machine (called TBWRSVM) which benefits from the advantages of TBSVM and WRSVM methods. Similar to WRSVM, TBWRSVM concurrently alleviates the impact of imbalance and outlier to enhance the classification performance. The proposed method like TBSVM uses two hyperplanes to determine the class of new data so that each hyperplane is closer to one of two classes at the same time and far away from the other class. This makes improvement in the algorithm's capability of generalization. The considerable advantages of our TBWRSVM are performing extremely faster and being more accurate mostly than WRSVM. Moreover, it reduces the effect of outliers on the decision boundary in training procedure.

We compare classification results from TBWRSVM with standard SVM, WRSVM, RSVM and some other robust classification algorithms on public benchmark datasets in KEEL and UCI machine learning repositories.

The remainder of this paper is organized as follows. In section 2, we briefly introduce the related works of imbalanced classification and classification in the presence of outliers. Section 3 reviews primary methods. The detail of TBWRSVM is described in section 4. The experiments and results are described in section 5. We conclude in section 6.

II. RELATED WORKS

Because of the importance of the imbalanced classification problem, many techniques have been developed to address this problem. These techniques are divided into four categories: 1) Algorithm level approaches, 2) Data level approaches, 3) Cost-sensitive learning, 4) Ensemble algorithms.

Algorithm level approaches attempt to adapt basic learning methods to bias the learning towards the minority class [51]–[53]. To understand why the classifier fails when the distribution of class is uneven in these methods, the specific knowledge of both the corresponding classifier and the application domain are required. For example in [54] a new decision tree is proposed which is robust and insensitive to the size of classes.

In data level approaches, the distribution of classes are balanced through resampling the data space [55], [56]. These techniques avoid the modification of the learning algorithm and try to reduce the result being affected by the imbalanced data through a preprocessing phase. Synthetic minority oversampling technique (SMOTE) [57], modified synthetic minority oversampling technique (MSMOTE) [58] and selective preprocessing of imbalanced data (SPIDER) [59] belongs to this category.

Cost-sensitive learning algorithms position between the data and the algorithm level. These algorithms allocate weights to data samples based on their importance. Many popular classification algorithms can be adapted under this framework. The SVM adaptation to the cost-sensitive learning framework is called weighted support vector machine (WSVM). In WSVM, different costs are associated with the minority and the majority class. Initial effort of applying weights to SVM is done in [60]. Fuzzy support vector machine is a type of support vector machine learning algorithm that was proposed in [61] under the cost-sensitive learning framework. The FSVM technique assigns various fuzzy membership to each data sample to consider their importance such that different data samples can make various portions to the learning of separating hyperplane. Another cost-sensitive method that is called z-svm proposed in [62] performing a weighted approach for the positive class. The purpose of this method is maximizing G-mean to regulate the hyperplane and diminish skew into the minority class. In [63] a cost-sensitive algorithm is extended that on the base of the data distribution adjusts the class boundary and the kernel matrix.

Ensemble learning is a method that incorporates the outputs of several base learners to obtain a new classifier. The main objective of ensemble learning is to improve the classification performance. Cost-sensitive Boosting such as AdaCost [64], RareBoost [65] and Boosting-based ensembles such as SMOTEBoost [66], RUSBoost [67] belongs to this category. Also, Bagging-based ensembles such as overBagging [68], underBagging [69] can be named.

The Classification of imbalanced data in the attendance of outliers is a very challenging work. In an imbalanced problem, the minority class might contain very few samples. Since under the cost-sensitive learning framework the minority class data points are associated with higher weights; therefore, the outliers receive high weights and contribute strongly to training. Consequently, the effect of outliers of the minority class will magnify [29], [30]. Hence, the use of these methods becomes problematic. This problem may also affect the performance of classifiers significantly [70]. Not only cost-sensitive methods but also preprocessing and ensemble learning methods suffer from sensitivity into noise or outliers [71].

Three popular approaches exist for dealing with outliers. The first approach contains extending algorithms that are less sensitive to noise and are robust such as Robust-C4.5 [72], ND-AdaBoost [73], RSVM [39], WRSVM [40], modified FSVM [74] and FSVM-CIL [30]. The pruning method is developed in [72] in order to completely eliminate the effect of outliers. In [73] a new version of boosting algorithm is extended that integrated a noise detection based loss function with AdaBoost to regulate the weight distribution at each iteration. In [39] a method is proposed in order to decrease the impact of outliers on the support vectors that is called relaxed support vector machine (RSVM). In this approach, for relaxing the support vectors, a limited amount of penalty-free slack variable is used and decision boundaries pressing into their class boundaries. Reference [40] proposed a cost-sensitive algorithm that improve RSVM with allocating various weights to the minority and majority classes which is called WRSVM. WRSVM incorporates the cost-sensitive approach of WSVM and the relaxation approach of RSVM. In [74] FSVM combined with kernel correction method which is done based on the Riemannian metric. Outliers have recognized in [30] by fuzzy membership values and their effects have reduced under the principle of cost-sensitive learning, then incorporated them in learning of imbalanced data with the use of a fuzzy support vector machine. In the second approach, the outliers are detected and removed by applying outlier detection and elimination techniques. Then the remaining data are used in the learning process. References [75]–[77] are some works in this approach. In the third approach, mislabeled data is correcting [78], [79].

III. PRIMARILIES

In this section, an outline of SVM, TWSVM, WSVM, RSVM and WRSVM is stated in summary.

Before these overviews, formally, let $\{x_k, y_k\}_{k=1}^{n}$ be the training data set which is a set of n column vectors of data points in the real d dimensional space $\Re^d$. k-th input pattern is x_k and k-th output pattern is y_k and $y_k \in \{-1, +1\}$ determines the label of k-th instance. $n = n_+ + n_-$ where n_+ and n_- are the size of positive and negative classes respectively. Suppose that the set of sample indices is denoted by I and I_+ and I_- indicate the set of sample indices for positive and negative classes, respectively.

A. SVM

Support Vector Machine (SVM) [31], [33] is a classification method that finds an optimized hyperplane to separate space of two classes which expressed by a normal vector $w \in \Re^d$ and a hyperplane offset $b \in \Re$. It is gained by maximizing the margin. SVM solution will be achieved by minimizing the following primal target function:

$$\min_{w,b,\xi} J(w, b, \xi) = \frac{1}{2}||w||^2 + c\sum_{i=1}^{n}\xi_i,$$
$$\text{with} \quad \forall i \begin{cases} y_i(w.\Phi(x_i) + b) \geq 1 - \xi_i, \\ \xi_i \geq 0, \end{cases}$$
$$i = 1, \ldots, n. \tag{1}$$

In this equation, $\Phi(.)$ is a function for mapping input space to feature space and $c > 0$ is a penalizing parameter. By slack variables ξ_i, misclassification permission is added to the formulation which measure the error of samples classification.

The Wolf dual of the primal problem can be written as follows:

$$\max_{\alpha} G(\alpha) \equiv \sum_{i=1}^{n}\alpha_i - \frac{1}{2}\sum_{i,j}\alpha_i\alpha_j y_i y_j K(x_i, x_j)$$
$$s.t. \ \sum_i \alpha_i y_i = 0,$$
$$0 \leq \alpha_i \leq c \quad i = 1, \ldots, n. \tag{2}$$

Optimization of the dual function leads to solution of the convex quadratic programming problem (QPP) of (1). In this equation, $K(x_i, x_j)$ is appropriate used kernel matrix and $\alpha \in \Re^n$ is a vector which signs Lagrangian coefficients.

Expressive parameters of discriminative hyperplane based on QPP solution can be computed as (3).

$$w = \sum_{i=1}^{n} \alpha_i^* y_i \Phi(x_i),$$
$$b = \frac{1}{N_{sv}} \sum_{j=1}^{N_{sv}} \left(y_j - \sum_{i=1}^{n} \alpha_i^* y_i K(x_i, x_j) \right) \quad (3)$$

Here α^* is the solution of dual problem and the number of support vectors (SVs) is indicated by N_{sv}. By finding separator hyperplane, class membership of a new untrained sample can be decided by the sign of the function $f(x) = \sum_{i=1}^{n} \alpha_i^* y_i K(x_i, x_j) + b$.

B. TWSVM AND TBSVM

The innovation of Twin Support Vector Machine (TWSVM) [41] is the generation of two nonparallel hyperplanes in a manner that each hyperplane is more nearby to one of the two classes and is as far as possible from the other. Instead of one (with all data in constraint), two smaller sized quadratic programming related SVM-type problems are solved by TWSVM, so as a result, TWSVM is faster.

Generally, definition of two separator hyperplanes is as follows:

$$K(x^T, C^T)u^{(1)} + b^{(1)} = 0, \text{ and}$$
$$K(x^T, C^T)u^{(2)} + b^{(2)} = 0, \quad (4)$$

where

$$C^T = [A\ B]^T. \quad (5)$$

where A and B are matrices containing d-dimensional data samples of classes 1 and -1, so the sizes of matrices A and B are $n_+ \times d$ and $n_- \times d$ respectively.

Equation 4 is generally expressed in non-linear mode. By considering $K(x^T, C^T) = x^T C, w^{(1)} = C^T u^{(1)}$ and $w^{(2)} = C^T u^{(2)}$, we have the linear form of it.

The first hyperplane is found by solving the following optimization problem:

$$\min_{u^{(1)}, b^{(1)}, \xi} \frac{1}{2} \left\| K(A, C^T)u^{(1)} + e_1 b^{(1)} \right\|^2 + c_1 e_2^T \xi$$
$$s.t. \ -(K(B, C^T)u^{(1)} + e_2 b^{(1)} + \xi \geq e_2, \quad \xi \geq 0. \quad (6)$$

Here $c_1 > 0$ is a constant between the maximum margin and the minimum classification error and e_1 and e_2 are vectors of ones of convenient dimensions.

Dual of the mentioned problem can be obtained by constructing the Lagrangian function and assuming KKT[1] conditions. Alike, supposing $c_2 > 0$ as a trade-off parameter, the second hyperplane can be stated as:

$$\min_{u^{(2)}, b^{(2)}, \xi} \frac{1}{2} \left\| K(B, C^T)u^{(2)} + e_2 b^{(2)} \right\|^2 + c_2 e_1^T \xi$$
$$s.t. \ (K(A, C^T)u^{(2)} + e_1 b^{(2)} + \xi \geq e_1, \quad \xi \geq 0. \quad (7)$$

[1] Karush–Kuhn–Tucker

Twin Bounded Support Vector Machine (TBSVM) [42] is a classifier extended based on TWSVM. In TBSVM, to maximize some margin and minimize the structural risk as a result, a regularization term is appended to the formulation. Generally, as TWSVM, definition of the optimization problems are as the following equations:

$$\min_{u^{(1)}, b^{(1)}, \xi} \frac{1}{2} c_1 (\left\| u^{(1)} \right\|^2 + (b^{(1)})^2)$$
$$+ \frac{1}{2} \left\| K(A, C^T)u^{(1)} + e_1 b^{(1)} \right\|^2 + c_2 e_2^T \xi$$
$$s.t. \ -(K(B, C^T)u^{(1)} + e_2 b^{(1)} + \xi \geq e_2, \quad \xi \geq 0, \quad (8)$$

and

$$\min_{u^{(2)}, b^{(2)}, \xi} \frac{1}{2} c_3 (\left\| u^{(2)} \right\|^2 + (b^{(2)})^2)$$
$$+ \frac{1}{2} \left\| K(B, C^T)u^{(2)} + e_2 b^{(2)} \right\|^2 + c_4 e_1^T \eta$$
$$s.t. \ (K(A, C^T)u^{(2)} + e_1 b^{(2)} + \eta \geq e_1, \ \eta \geq 0. \quad (9)$$

In this equations c_1, c_2, c_3, c_4 are trade-off constants and ξ, η are indicative of slack variables. Similar to TWSVM, linear case can be obtained, also, by finding the solution of the corresponding dual problems, the nonparallel proximal hyperplanes of (4) are attained.

C. WSVM

In standard SVM and many other versions of it, the importance of all data samples is equal in the training phase. When the data set is imbalanced, this importance equality might be distasteful. One way to address this problem is weighting of the samples. Applying the cost-sensitive learning framework to SVM leads to Weighted Support Vector Machine (WSVM) which was also found as Fuzzy SVM in some studies. Reference [61] is one of remarkable adapted algorithms to this approach. It is developed for achieving more flexible training process in which each sample is associated with a weight. Primal modified version of SVM was introduced by Veropoulos and colleagues [60] to classify imbalanced binary data. In this method, different costs are used corresponding to the positive (c^+) and negative (c^-) classes. By solving the following optimization problem, the optimal hyperplane was found in this case.

$$\min \frac{1}{2} ||w||^2 + c^+ \sum_{i \in I^+} \xi_i + c^- \sum_{i \in I^-}^{\xi_i},$$
$$s.t. \ y_i(w.x_i + b) \geq 1 - \xi_i \quad i = 1, \ldots, n$$
$$\xi_i \geq 0, \quad i = 1, \ldots, n \quad (10)$$

With the weighting parameters c^+ and c^-, dedication of different "importance" to the misclassification of classes is accomplished in this equation.

D. RSVM AND WRSVM

Performance of classification may be badly affected by attendance of some samples (misclassified samples or outliers).

Relaxed Support Vector Machine (RSVM) is presented by the purpose of influence reduction of these samples which is done by producing unpenalized (free) slack [39]. RSVM is given in (11).

$$\begin{aligned} \min_{w,b,\xi,\upsilon} \; & \frac{1}{2}||w||^2 + \frac{c}{2}\sum_{i\in I}^{\xi_i^2}, \\ s.t.\; & y_i(w.x_i + b) \geq 1 - \xi_i - \upsilon_i, \quad \forall i \in I \\ & \sum_{i\in I} \upsilon_i \leq n\Upsilon, \\ & \upsilon_i \geq 0 \quad \forall i \in I \end{aligned} \tag{11}$$

By $n\Upsilon$, forenamed restricted amount is given in optimization problem, where parameter Υ determines average slack per sample. υ_i are distributary variables of free slacks to samples.

Applying RSVM to imbalanced data may not entail desirable outcome, because positive and negative classes don't make difference for this method and the penalty term is identically assigned to all samples without considering relevant class sizes.

Due to the mentioned issues, weighted version of RSVM, WRSVM, which is tailored for imbalanced data, has been proposed [40], [80]. Combination of cost-sensitive attitude of WSVM and relaxation attitude of RSVM is occurred in WRSVM concurrently. Formulation of WRSVM is as follows:

$$\begin{aligned} \min_{w,b,\xi,\upsilon} \; & \frac{1}{2}\langle w.w\rangle + \frac{c}{2n^+}\sum_{i\in I^+}^{\xi_i^2} + \frac{c}{2n^-}\sum_{i\in I^-}^{\xi_i^2}, \\ s.t.\; & +(w.x_i + b) \geq 1 - \xi_i - \upsilon_i, \quad \forall i \in I^+ \\ & -(w.x_i + b) \geq 1 - \xi_i - \upsilon_i, \quad \forall i \in I^- \\ & \sum_{i\in I^+} \upsilon_i \leq n^+\Upsilon, \\ & \sum_{i\in I^-} \upsilon_i \leq n^-\Upsilon, \\ & \upsilon_i \geq 0, \quad \forall i \in I. \end{aligned} \tag{12}$$

As it is obvious, for compatibility to imbalance, the defined constraints in WRSVM attribute distinct amounts of total free slack to the positive and the negative classes.

IV. PROPOSED TWIN BOUNDED WEIGHTED RELAXED SUPPORT VECTOR MACHINE (TBWRSVM)

In this paper, we proposed a new improved version of WRSVM, which is called Twin Bounded Weighted Relaxed Support Vector Machine (TBWRSVM). TBWRSVM inherits specifications of cost-sensitive, relaxation and twining approach from methods of WSVM, RSVM and TBSVM, respectively. In this proposed method, to accelerate the training process and gain better performance, two smaller and simpler optimization problems are solved and two nonparallel hyperplanes are found. Also like TBSVM, the efforts, in TBWRSVM, have been made to minimize the structural risk by adding a regularization term to the formulation of optimization problems, which are given in succedent:

$$\begin{aligned} \min_{w_1,b_1,\xi,\upsilon} \; & \frac{1}{2}c_1(\langle w_1.w_1\rangle + b_1^2) \\ & + \frac{1}{2}(Aw_1 + e_1b_1)(Aw_1 + e_1b_1) + \frac{c}{2n^-}\xi^T\xi, \\ s.t.\; & -(Bw_1 + e_2b_1) \geq e_2 - \xi - \upsilon, \\ & e_2^T\upsilon \leq n^-\Upsilon, \\ & \upsilon \geq 0 \end{aligned} \tag{13}$$

and

$$\begin{aligned} \min_{w_2,b_1,\eta,\tau} \; & \frac{1}{2}c_2(\langle w_2.w_2\rangle + b_2^2) \\ & + \frac{1}{2}(Bw_2 + e_2b_2)(Bw_2 + e_2b_2) + \frac{c}{2n^+}\eta^T\eta, \\ s.t.\; & (Aw_2 + e_1b_2) \geq e_1 - \eta - \tau, \\ & e_1^T\tau \leq n^+\Upsilon, \\ & \tau \geq 0. \end{aligned} \tag{14}$$

Here, $w_1 \in \Re^{n^+}$ is the normal vector and $b_1 \in \Re$ is the offset of the hyperplane that is closer to the positive samples and is as far as possible from the negative one; similarly, $w_2 \in \Re^{n^-}$ is the normal vector and $b_2 \in \Re$ is the offset of the hyperplane pertaining to the negative samples. Other notations are as similar mentioned methods.

Dual of these problems can be obtained by use of Lagrangian function and KKT conditions. For the first problem, this computation is done as follows:

$$\begin{aligned} L = \; & \frac{1}{2}c_1(\langle w_1.w_1\rangle + b_1^2) + \frac{1}{2}(Aw_1 + e_1b_1)(Aw_1 + e_1b_1) \\ & + \frac{c}{2n^-}\xi^T\xi - \alpha^T(-(Bw_1 + e_2b_1) - e_2 + \xi + \upsilon) \\ & - \beta(n^-\Upsilon - e_2^T\upsilon) - \lambda^T\upsilon \end{aligned} \tag{15}$$

where α, β and λ are Lagrange multipliers vectors. The KKT necessary and sufficient optimality conditions [81] are stated as following:

$$\frac{\partial L}{\partial w_1} = c_1w_1 + A^T(Aw_1 + e_1b_1) + B^T\alpha = 0 \tag{16}$$

$$\frac{\partial L}{\partial b_1} = c_1b_1 + e_1^T(Aw_1 + e_1b_1) + e_2^T\alpha = 0 \tag{17}$$

$$\frac{\partial L}{\partial \xi} = \frac{c}{n^-}\xi - \alpha = 0 \tag{18}$$

$$\frac{\partial L}{\partial \upsilon} = -\alpha + \beta e_2 - \lambda = 0 \tag{19}$$

$$\alpha^T(-(Bw_1 + e_2b_1) - e_2 + \xi + \upsilon) = 0 \tag{20}$$

$$\beta(n^-\Upsilon - e_2^T\upsilon) = 0 \tag{21}$$

$$\lambda^T\upsilon = 0 \tag{22}$$

$$\alpha \geq 0, \quad \beta \geq 0, \quad \lambda \geq 0. \tag{23}$$

Due to $\lambda \geq 0$ and condition 19, we obtain

$$0 \leq \alpha \leq \beta. \tag{24}$$

Afterwards, (25) is resulted by combination of (16) and (17):

$$([A^T \; e_1^T][A \; e_1] + c_1 I)[w_1 \; b_1]^T + [B^T \; e_2^T]\alpha = 0. \quad (25)$$

Which can be rewritten as

$$(H^T H + c_1 I)u + G^T\alpha = 0 \Rightarrow u = -(H^T H + c_1 I)^{-1} G^T \alpha \quad (26)$$

by defining the follows

$$H = [A \; e_1], \; G = [B \; e_2], \; u = [w_1 \; b_1]^T. \quad (27)$$

The dual of the first problem is specified by use of these conditions and replacement in Lagrangian function:

$$\begin{aligned} \max_{\alpha,\beta} \; & \frac{-n^-}{2c}\alpha^T\alpha + e_2^T\alpha - \frac{1}{2}\alpha^T G(H^T H + c_1 I)^{-1} G^T \alpha \\ & - \beta n^- \Upsilon \\ s.t.\; & 0 \le \alpha \le \beta. \end{aligned} \quad (28)$$

For the second hyperplane, the dual problem and above equations can be reiterated similarly:

$$\begin{aligned} \max_{\gamma,\lambda} \; & \frac{-n^+}{2c}\gamma^T\gamma + e_1^T\gamma - \frac{1}{2}\gamma^T P(Q^T Q + c_2 I)^{-1} P^T \gamma \\ & - \rho n^+ \Upsilon \\ s.t.\; & 0 \le \gamma \le \rho, \end{aligned} \quad (29)$$

$$\begin{aligned} & v = [w_2 \; b_2]^T, \quad v = -(Q^T Q + c_2 I)^{-1} P^T \gamma, \\ & P = [A \; e_1], \quad Q = [B \; e_2]. \end{aligned} \quad (30)$$

Two computed problems are for linear case of TBWRSVM. In the following, nonlinear problems are described. Here, the kernel-generated surfaces (instead of hyperplanes) are determined as:

$$\begin{aligned} & K(x^T, C^T)u_1 + b_1 = 0, \text{ and} \\ & K(x^T, C^T)u_2 + b_2 = 0. \end{aligned} \quad (31)$$

In these surfaces, K is the predestinate kernel. The linear optimization problems are altered in this case, as follows:

$$\begin{aligned} \min_{u_1,b_1,\xi,\upsilon} \; & \frac{1}{2}c_1(\|u_1\|^2 + b_1^2) + \frac{1}{2}\left\|K(A, C^T)u_1 + e_1 b_1\right\|^2 \\ & + \frac{c}{2n^-}\xi^T\xi, \\ s.t.\; & -(K(B, C^T)u_1 + e_2 b_1) \ge e_2 - \xi - \upsilon, \\ & e_2^T\upsilon \le n^-\Upsilon, \\ & \upsilon \ge 0 \end{aligned} \quad (32)$$

and

$$\begin{aligned} \min_{u_2,b_1,\eta,\tau} \; & \frac{1}{2}c_2(\|u_2\|^2 + b_2^2) + \frac{1}{2}\left\|K(B, C^T)u_2 + e_2 b_2\right\|^2 \\ & + \frac{c}{2n^+}\eta^T\eta, \\ s.t.\; & (K(A, C^T)u_2 + e_1 b_2) \ge e_1 - \eta - \tau, \\ & e_1^T\tau \le n^+\Upsilon, \\ & \tau \ge 0. \end{aligned} \quad (33)$$

Lagrangian and KKT conditions can be determined as similar. Likewise, the dual of these problems are calculated by (34) to (37).

$$\begin{aligned} \max_{\alpha,\beta} \; & \frac{-n^-}{2c}\alpha^T\alpha + e_2^T\alpha \\ & - \frac{1}{2}\alpha^T R(S^T S + c_1 I)^{-1} R^T \alpha - \beta n^- \Upsilon \end{aligned} \quad (34)$$

$$\begin{aligned} s.t.\; & 0 \le \alpha \le \beta \\ & z_1 = [w_1 \; b_1]^T, \; z_1 = -(S^T S + c_1 I)^{-1} R^T \alpha, \\ & S = [K(A, C^T) \; e_1], \; R = [K(B, C^T) \; e_2] \end{aligned} \quad (35)$$

and

$$\begin{aligned} \max_{\gamma,\lambda} \; & \frac{-n^+}{2c}\gamma^T\gamma + e_1^T\gamma \\ & - \frac{1}{2}\gamma^T L(N^T N + c_2 I)^{-1} L^T \gamma - \rho n^+ \Upsilon \end{aligned} \quad (36)$$

$$\begin{aligned} s.t.\; & 0 \le \gamma \le \rho, \\ & z_2 = [w_2 \; b_2]^T, \; z_2 = -(N^T N + c_2 I)^{-1} L^T \gamma, \\ & L = [K(A, C^T) \; e_1], \quad N = [K(B, C^T) \; e_2]. \end{aligned} \quad (37)$$

V. EXPERIMENTS AND RESULTS

First of all, in this section, we introduce the used data sets and then represent the evaluation criteria for performance evaluation of algorithms. Finally, the proposed method is compared with its standard counterparts and the results are reported.

A. DATA SET

To validate the performance of TBWRSVM, we exert it on several benchmark data set for imbalanced classification which is commonly used in testing machine learning algorithms. The applied data set is given in Table 1. Intrinsic imbalance ratio (r_{int}) of each data set are imported in the last column of table. r_{int} is calculated as the number of majority class samples to number of all data. The two last datasets, Dermatology and Cleveland, are innately imbalanced data (without distribution altering). These data sets are available in UCI[2] and KEEL[3] data sets repositories.

B. EVALUATION CRITERIA

In this paper, In order to compare the methods, two common criteria were used: G-mean and CPU time. These criteria are defined as follows:

1) G-MEAN

Accuracy is one of the prevalent criterion for classifications performance measurement which is calculated by dividing the number of correctly classified samples to the total number of training samples [82]. Nevertheless, maybe it is not a suitable performance measurement for imbalanced classification problems since the majority class dominates the behavior of

[2]http://archive.ics.uci.edu/ml

[3]http://sci2s.ugr.es/keel/datasets.php

TABLE 1. Data sets statistics.

	Number of positive samples	Number of negative samples	Number of total samples	Feature	r_{int}
Diabetes	268	500	768	8	65.10
Pima	268	500	768	8	65.10
German	300	700	1000	20	70
Appendicitis	21	85	106	7	80.18
Flare solar	50	94	144	9	65.27
Heart	120	150	270	13	55.55
Thyroid	65	150	215	5	69.76
Breast cancer	77	186	263	9	70.72
Bupa	145	200	345	6	57.97
Haberman	81	225	306	3	73.52
Cleveland	13	160	173	13	92.48
Dermatology	20	338	358	34	94.41

this metric. For example, in an imbalanced problem where 95% of the data belong to the majority class and 5% to the minority, if all data assign to the majority class, classification accuracy will be 95%. Whereas, in imbalanced classification problems, the correct recognition of the minority class is more important. For this reason, some other proper criteria like G-mean [83], ROC Curve (AUC) [84] and F-measure [85] are being used in this field. In our evaluations, G-mean is the main criterion.

This criterion is based on two criteria namely specificity and sensitivity. These criteria are described by the TP, FP, FN, and TN instances, which stand for the numbers of true positive, false positive, false negative and true negative, respectively. The formulas for computing these measures are as follows:

$$Sensitivity = \frac{TP}{TP + FN} \tag{38}$$

$$Specificity = \frac{TN}{TN + FP} \tag{39}$$

G-mean is the geometric mean of specificity and sensitivity, which can be a reliable performance measure for imbalanced classification. Its formula is defined as follows:

$$\text{G-mean} = \sqrt{specificity \times sensitivity} \tag{40}$$

2) CPU TIME

This measure is the average time which is required to build hyperplanes.

C. MODEL SELECTION

The SVM and its diverse developments, such as WRSVM and TBWRSVM algorithms have some parameters that should be adjusted during the training phase; specifically, the regularization parameter and the Gaussian kernel width parameter for SVM and WSVM and also the total free slack parameter for WRSVM and TBWRSVM. For reducing the number of examinations in parameter combination, we used a nested uniform design model selection algorithm [86]. Nested uniform design methodology has been proposed for efficient, robust and automatic model selection for support vector machines.

The uniform design [87] is one type of space-filling designs that search its design points to be uniformly diffused in the experimental domain. This method searches the parameter space by probing the points that minimize a discrepancy function among their empirical distribution and the theoretical uniform distribution. In order to detect the close-to-optimal parameter set, this process can be applied iteratively in a nested manner. The optimal points are selected based on several classification evaluation measurements. What we use is L2-discrepancy measure.

Assume that there are N parameters over a domain C^n. Here the purpose is selecting a set of m points $P_m = \{x_1, \ldots, x_m\} \subset C^n$ so that these points are uniformly distributed on C^n. $F(x)$ is the cumulative uniform distribution function over C^n and $F_m(x)$ is the empirical cumulative distribution function of P_m. the L2-discrepency of P_m is defined as follows:

$$D_2\left(C^n, P_m\right) = \left[\int_{C^n} |F_m(x) - F(x)|^2 \, dx\right]^{1/2} \tag{41}$$

In order to determine the near-optimal parameter, this technique performs a parameter space search in a multiple-stage process. Because the pursued problem is imbalance, we select the optimal parameter set based on the highest G-mean value. The model selection process is as follows:

1) Select a parameter search domain and specify an appropriate number of levels (factors in design terminology) for each parameter.
2) Select an appropriate uniform design table to contain the number of parameters and levels (most UD tables

TABLE 2. The range of parameters for computational experiments.

Name	Symbol	Range
Penalization parameter	c	$[10^{-2}, 10^{+4}]$
Gaussian kernel width parameter	γ	$[-\frac{\log(0.999)}{k}, -\frac{\log(0.150)}{k}]$
Average free slack per sample	Υ	[0.05, 0.25]
Trade-off constant	c_1, c_2	$[10^{-3}, 10^{+4}]$
Note: k is the minimum distance from instance to average of instances.		

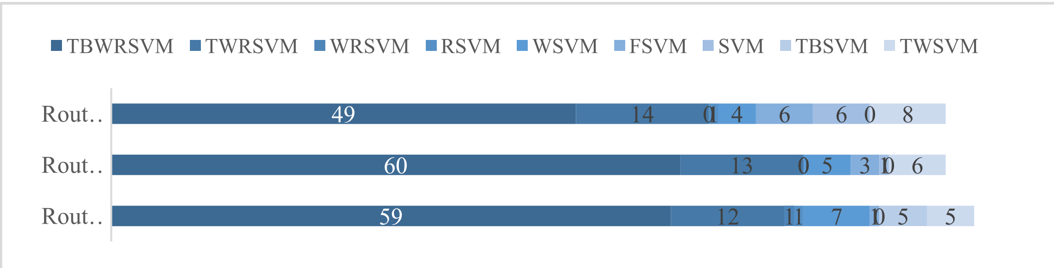

FIGURE 1. Summation of boldface results for each methods with R_{out} = 10, 15 and 20.

are available at UD-web. http://www.math.hkbu.edu.hk/UniformDesign)

3) From the uniform design table, determine the run order of experiments arbitrarily and then fill the uniform table using the performance evaluation of each parameter combination.
4) Amend the search around the point with the highest performance measure value by repeating step 1-3.

More details are available in [40] and [86].

D. EXPERIMENTAL RESULTS

In this section, we express the result of evaluation of TBWRSVM in comparison with TWRSVM, WRSVM, RSVM, TWSVM, TBSVM, FSVM, WSVM, and SVM models. TWRSVM (Twin Weighted Relaxed Support Vector Machine) is our other proposed method, which is an incorporation of WRSVM and TWSVM. All methods are implemented in MATLAB R2015a 64 bit on a 2.3 GHz quad-core CPU with 16 GB RAM and Windows 7. LIBSVM-3.12 is used to solve the SVM model and LIBSVM-weight-3.12 is used to solve the WSVM model. CPLEX 12.6 64 bit is used to solve the related optimization problems of TBWRSVM, TWRSVM, WRSVM, and RSVM. Evaluation criteria of each method are measured by the standard 10-fold cross-validation methodology. Selection of parameter in each iteration of the 10-fold cross validation is based on the nested uniform design [86] which is done on training data. For SVM, FSVM, WSVM, TWSVM, TBSVM, RSVM, WRSVM and TWRSVM, we use a 13-run UD sampling pattern at the first stage and 9-run UD sampling pattern at the second stage. We adopt a 5-run at the third stage. For TBWRSVM, we adopt a 17- and 13- run of the nested UD sampling pattern for the first and second stages. For the third stage of the nested UD for TBWRSVM, we use 9 points. The ranges of parameters in our computational experiments are summarized in Table 2. It should be noted that for defining the search range of γ (the width parameter of the Gaussian kernel), we used the proposed heuristic in [86].

Performance metrics are calculated using the test data. All data have zero mean and unitary standard deviation because of normalization which has been done just before their classification process. We use RBF kernel in our execution. Several imbalanced ratios (r_{imb}) of r_{int}, 80, 85, 90, 92, 95 and 97 are considered in our experimental implementations. As it was mentioned, r_{int} presents intrinsic imbalance ratio of dataset. Since we should consider different ratios, other ratios are computed by randomly omitting some minority samples. For adding artificial outliers to each dataset, class labels of some farthest samples in the majority class are varied to label of minority class. Outlier ratio (R_{out}) is determinant of this variation amount which is defined by a percentage of total samples. In the experiments, outlier ratio of 10%, 15% and 20% are utilized to investigate impact of different outlier levels ranging from low to high.

Tables 3, 4 and 5 respectively show the results of our proposed TBWRSVM in comparison with other methods, TWRSVM, WRSVM, RSVM, WSVM, FSVM, SVM, TWSVM and TBSVM in terms of G-mean criteria for R_{out} of 10, 15 and 20. The results are listed for all considered imbalanced ratio for each dataset. The highest G-mean values, the bold marked, are indicative of the best method for each dataset. By a Glance at the table, it is clear our proposed method, TBWRSVM, outperforms others. Totality of the best results for every method affirms it. This number for TBWRSVM is 59, 60 and 49 out of 75 in respect for

FIGURE 2. G-mean versus the outlier ratio for dataset of Heart, Flare solar, and Pima with imbalance ratios of 90 and 97 for left and right columns of plots, respectively.

outlier ratio of 10, 15 and 20. This amount is followed by TWRSVM with number of 12, 13 and 14 for outlier ratio of 10, 15 and 20 respectively. Fig. 1 shows these totality values for all methods as a bar diagram.

Outliers of the minority class impress on classification results. Amount of impression depends on two factors: the distribution and the geometry of classes [40]. Obviously, high outlier ratio has more impression. Fig. 2 shows this impression on G-mean for outlier ratios 0% to 50% for TBWRSVM in comparison with TWRSVM, WRSVM and TBSVM. The Figure illustrates this for three datasets Heart, Flare solar and Pima, which the left and right columns are plotted for imbalance ratios of 90% and 97%, respectively. By glancing at the plots, it is clear TBWRSVM is better than others in the most points. It is notable that G-mean value is not decreased so much by growth of outlier ratio and even

TABLE 3. G-mean for Rout = 10.

Dataset	r_{imb}	TBWRSVM	TWRSVM	WRSVM	RSVM	WSVM	FSVM	SVM	TBSVM	TWSVM
Diabetes	r_{int}	**75.94**	75.55	73.68	72.80	73.59	75.15	73.08	73.19	73.94
	80	**76.36**	75.65	74.28	67.70	74.98	75.42	73.35	74.04	73.60
	85	74.04	73.66	72.25	67.39	**74.88**	73.25	69.21	73.14	72.01
	90	**76.17**	75.86	71.60	63.69	75.38	74.82	73.00	73.63	72.40
	92	**79.29**	74.56	72.28	67.54	76.11	74.20	69.48	70.89	73.61
	95	**80.07**	**80.07**	79.59	67.28	78.92	74.79	68.33	74.22	79.43
	97	**79.18**	75.76	72.12	66.41	76.71	72.75	68.87	68.96	74.81
Pima	r_{int}	**75.60**	75.08	71.48	72.23	74.80	74.71	73.53	73.90	74.22
	80	77.86	77.29	75.16	71.48	**78.48**	78.00	76.94	75.59	78.40
	85	**76.75**	74.76	67.00	64.87	74.26	72.56	68.22	69.73	71.91
	90	**86.36**	83.79	78.52	77.20	83.65	81.85	81.71	83.42	80.70
	92	**85.87**	85.27	75.05	74.82	84.85	83.33	80.89	85.27	84.37
	95	**89.20**	89.13	74.75	75.84	87.80	85.77	80.28	87.35	85.33
	97	91.01	90.60	69.49	81.97	90.22	87.39	84.58	**91.59**	88.99
German	r_{int}	**74.14**	73.46	69.26	70.04	72.25	72.38	70.47	70.83	72.61
	80	**83.21**	**83.21**	78.93	78.20	81.68	81.12	80.82	80.70	81.79
	85	**83.12**	**83.12**	78.50	75.89	82.36	80.51	79.97	79.56	80.66
	90	**82.15**	81.20	73.66	75.15	79.73	79.23	77.26	77.26	80.25
	92	**86.93**	**86.93**	75.71	76.79	83.67	83.86	81.85	81.73	83.60
	95	**86.22**	84.49	75.44	76.16	83.49	81.13	80.10	85.10	82.44
	97	**90.45**	88.64	78.31	80.00	86.78	86.27	82.62	87.16	82.20
Appendicitis	r_{int}	80.77	81.60	78.61	73.15	82.67	81.31	76.05	76.43	**83.06**
	85	**93.58**	91.62	91.86	84.85	91.55	82.73	84.48	92.57	90.39
	90	**97.16**	86.42	87.07	86.42	95.68	86.42	86.42	95.68	91.21
	92	93.40	93.40	91.41	77.54	92.01	79.82	81.21	**94.29**	80.47
	95	**98.01**	88.70	87.07	76.42	95.87	85.90	76.42	95.22	85.77
	97	**98.66**	87.07	77.07	84.97	94.97	77.07	77.07	94.97	95.73
Flare solar	r_{int}	**68.69**	**68.69**	65.26	63.74	65.39	62.60	61.23	67.16	65.95
	80	**80.22**	78.54	77.86	53.12	72.82	73.91	67.72	74.79	76.29
	85	**84.84**	78.28	81.64	70.32	68.96	79.87	70.18	76.68	79.42
	90	**81.87**	75.43	70.10	58.30	80.10	76.41	57.10	78.42	68.33
	92	**78.57**	**78.57**	72.53	61.58	58.39	64.63	59.91	76.31	76.30
	95	76.03	72.92	72.54	26.42	64.05	71.66	44.14	**77.93**	71.40
	97	**77.85**	75.13	72.39	56.80	72.06	74.48	62.85	61.20	73.86
Heart	r_{int}	**84.97**	82.63	82.11	**84.97**	81.59	81.24	82.80	80.70	82.67
	80	**84.34**	79.70	77.36	80.51	77.41	79.22	74.80	76.38	81.34
	85	**86.04**	85.36	83.00	77.18	76.47	81.76	82.72	81.67	83.58
	90	**86.74**	80.57	80.42	66.09	80.25	79.15	79.15	77.61	77.97
	92	**77.86**	70.51	71.36	55.99	71.55	60.54	51.81	69.59	72.34
	95	**86.12**	80.55	78.18	54.22	68.58	77.97	64.97	79.41	79.89
	97	**79.01**	66.97	64.29	28.08	70.35	66.14	50.27	70.62	75.60
Thyroid	r_{int}	89.27	89.27	82.73	87.45	86.40	87.83	86.10	**89.51**	85.35
	80	**92.33**	**92.33**	81.39	80.01	88.21	85.20	84.54	86.12	79.76
	85	**90.26**	89.56	80.64	76.95	86.42	82.29	78.37	84.93	84.17
	90	**86.98**	**86.98**	78.91	76.50	85.24	84.94	74.68	65.62	78.56
	92	**93.84**	93.68	84.58	70.56	89.69	86.54	76.85	91.83	91.84
	95	91.95	91.95	86.10	63.70	90.00	87.29	76.99	33.70	**95.09**
	97	**91.99**	91.19	84.05	35.18	88.69	67.78	53.58	79.87	78.50
Breast cancer	r_{int}	**69.37**	68.87	63.14	64.08	65.12	66.82	66.78	66.55	67.21
	80	67.99	67.37	63.94	71.61	**72.21**	70.26	65.83	62.06	68.90
	85	**75.34**	71.84	67.88	60.69	73.05	69.11	74.93	67.96	72.93
	90	**74.03**	73.36	70.20	59.74	72.28	64.67	66.24	64.69	65.36
	92	**78.74**	71.85	59.91	57.07	69.29	62.28	59.81	71.48	63.54
	95	**82.24**	**82.24**	72.35	49.56	71.31	77.86	68.59	73.54	76.73
	97	79.17	76.44	79.87	46.17	76.06	73.13	61.04	72.96	**82.38**
Bupa	r_{int}	75.37	75.19	71.42	72.76	71.31	**75.63**	74.93	57.22	73.73
	80	72.02	69.58	69.65	68.99	**74.76**	69.66	70.12	62.61	70.58
	85	75.92	75.92	73.21	62.94	72.72	74.92	71.61	65.59	**77.55**
	90	**76.13**	73.58	68.74	57.46	75.89	72.46	72.09	70.06	73.24
	92	**82.83**	81.42	77.09	62.78	71.42	75.10	64.86	75.51	80.56
	95	78.10	77.32	75.43	58.82	**81.08**	76.67	77.12	75.28	78.55
	97	75.29	73.95	76.76	61.11	**80.09**	58.68	59.08	76.74	71.93
Haberman	r_{int}	**71.79**	70.58	64.83	63.95	69.75	67.47	62.06	58.78	68.19
	80	**76.61**	**76.61**	64.76	45.96	75.56	72.27	66.12	48.83	74.39
	85	**77.26**	76.12	56.26	58.82	71.43	73.90	62.62	59.83	70.00
	90	**71.30**	**71.30**	68.66	58.04	70.23	70.79	61.60	63.30	67.82
	92	80.44	80.44	55.26	48.41	**80.99**	75.82	52.50	44.84	67.21
	95	**79.32**	78.69	62.80	56.13	78.25	73.74	33.69	59.69	77.78
	97	**86.76**	85.86	80.55	53.88	85.36	79.07	52.05	48.48	85.46
Cleveland	r_{int}	75.73	72.89	68.02	52.80	72.10	69.89	58.50	74.46	**77.17**
	95	75.44	74.62	**77.67**	32.71	72.72	72.55	65.70	72.64	68.84
	97	**82.89**	80.3	76.79	30.04	73.42	72.75	52.81	75.88	77.68
Dermatology	r_{int}	**87.46**	86.97	85.98	84.35	87.25	84.56	73.25	84.85	80.94
	95	**91.24**	**91.24**	90.52	87.54	88.32	86.96	78.15	90.05	87.46
	97	88.47	87.71	87.67	84.15	85.83	85.95	75.23	**89.90**	83.01

TABLE 4. G-mean for $R_{out} = 15$.

Dataset	r_{imb}	TBWRSVM	TWRSVM	WRSVM	RSVM	WSVM	FSVM	SVM	TBSVM	TWSVM
Diabetes	r_{int}	**75.75**	75.27	72.15	73.65	72.68	74.98	74.10	71.75	72.84
	80	74.31	73.94	71.78	73.80	72.92	74.50	**74.72**	70.62	72.70
	85	**78.29**	77.44	74.75	71.51	77.09	77.24	74.69	74.55	77.02
	90	75.63	75.23	71.49	70.06	73.99	**76.43**	75.25	70.87	72.40
	92	**81.79**	80.23	78.29	71.13	77.66	80.67	78.32	79.36	79.72
	95	79.52	79.52	78.41	73.99	78.49	76.84	76.29	78.89	**79.78**
	97	83.03	81.03	81.63	73.09	82.31	77.99	77.25	79.21	**83.50**
Pima	r_{int}	**76.62**	74.15	72.85	74.30	73.66	75.09	74.28	70.59	74.70
	80	**76.76**	**76.76**	71.18	72.20	76.27	76.32	75.30	73.51	75.79
	85	**76.94**	75.92	69.88	70.92	75.84	75.28	73.64	71.48	75.25
	90	**82.30**	81.27	73.68	74.07	81.67	80.03	77.18	80.06	77.26
	92	**79.50**	**79.50**	68.41	69.74	77.09	76.69	74.42	76.02	72.21
	95	**81.98**	**81.98**	71.23	73.89	81.11	79.14	74.54	78.00	74.40
	97	**83.61**	82.86	68.16	71.05	82.29	79.59	75.47	83.04	75.50
German	r_{int}	**76.43**	76.07	69.88	68.58	74.67	75.39	74.63	70.04	75.83
	80	**79.59**	**79.59**	72.37	73.39	78.58	78.81	77.95	72.93	78.20
	85	**77.91**	**77.91**	67.13	73.38	76.96	77.36	75.93	72.86	77.81
	90	**83.84**	82.85	76.34	76.60	79.54	80.56	77.98	81.80	80.07
	92	**85.20**	**85.20**	72.40	77.22	80.68	79.36	77.45	82.22	79.70
	95	**88.66**	87.49	74.40	82.00	85.39	84.30	82.63	85.62	84.86
	97	**89.74**	88.77	84.27	83.94	88.76	87.88	86.73	87.57	87.47
Appendicitis	r_{int}	**84.72**	83.98	78.30	79.81	81.31	80.01	76.33	75.61	78.54
	85	**89.90**	82.96	82.38	70.99	83.58	79.75	74.77	80.20	81.69
	90	**91.68**	85.33	81.95	83.99	87.74	86.91	81.19	87.59	83.11
	92	**88.98**	86.09	79.17	69.90	84.57	84.93	72.11	84.28	75.94
	95	**89.89**	84.05	79.56	75.58	85.37	86.55	79.25	86.08	83.31
	97	**91.97**	90.59	82.86	86.76	91.18	89.08	87.02	85.96	87.33
Flare solar	r_{int}	**66.42**	64.94	63.67	61.96	57.65	63.04	63.86	59.63	64.27
	80	**80.41**	77.12	76.51	64.28	72.08	77.04	76.83	71.10	76.19
	85	**84.53**	82.50	81.09	67.48	75.25	79.19	79.64	73.16	79.29
	90	**79.28**	**79.28**	74.62	65.26	73.88	70.63	71.70	69.25	73.36
	92	**83.51**	82.16	80.73	62.20	74.91	79.98	76.21	77.94	77.52
	95	**88.69**	81.56	81.04	72.92	82.70	82.48	80.84	84.58	77.64
	97	**92.54**	78.15	76.72	70.97	81.73	76.48	75.20	76.22	89.71
Heart	r_{int}	81.03	79.67	80.57	80.17	**81.75**	78.55	79.97	78.50	81.14
	80	**79.51**	79.49	74.35	78.32	75.37	77.02	75.31	70.34	78.23
	85	**82.16**	81.26	74.44	72.76	72.96	76.09	78.02	73.35	80.23
	90	**81.95**	**81.95**	72.21	70.94	79.27	76.34	71.77	64.45	81.13
	92	**83.37**	78.34	73.13	71.40	68.37	71.87	69.88	72.84	77.94
	95	**80.27**	78.57	70.47	54.69	73.96	76.30	70.95	74.24	77.91
	97	**81.91**	80.39	74.87	59.67	73.67	72.38	69.80	64.94	77.64
Thyroid	r_{int}	87.28	87.28	81.53	80.96	**87.42**	85.12	85.47	82.65	78.28
	80	86.45	85.29	75.74	82.39	**88.81**	85.00	79.70	73.02	74.28
	85	86.74	83.08	80.01	75.90	**91.79**	85.66	86.20	85.54	84.17
	90	85.21	85.21	71.81	80.63	**88.89**	87.39	88.19	61.62	78.59
	92	**90.50**	88.76	80.13	83.11	89.05	88.78	89.28	85.17	85.78
	95	**93.65**	93.62	85.42	66.49	91.14	91.34	87.19	87.04	93.50
	97	91.35	90.92	86.88	84.25	92.15	88.53	90.83	86.63	**93.60**
Breast cancer	r_{int}	**69.85**	67.67	63.68	66.79	67.71	65.94	65.74	57.39	66.39
	80	**76.72**	76.09	71.36	73.01	75.73	74.19	73.05	66.77	74.11
	85	**75.84**	75.82	68.55	65.63	72.18	73.29	72.44	71.13	73.05
	90	**79.24**	**79.24**	71.87	76.44	74.80	72.86	71.03	72.44	74.96
	92	**79.06**	76.79	76.50	65.88	75.28	75.68	70.93	72.81	75.15
	95	76.27	74.91	72.80	75.62	74.77	76.30	71.15	71.72	**78.40**
	97	**81.08**	79.89	72.21	67.07	77.44	77.31	72.57	78.95	78.90
Bupa	r_{int}	**76.99**	**76.99**	70.12	73.66	74.81	75.08	75.59	58.71	74.00
	80	**77.95**	76.10	73.83	72.09	77.30	74.94	76.60	66.80	77.60
	85	**77.20**	75.01	70.57	72.62	75.51	75.59	76.48	68.22	76.11
	90	80.66	78.64	77.99	72.16	79.52	78.95	79.58	72.97	**82.67**
	92	83.29	80.14	75.79	64.81	83.24	**83.38**	80.46	77.60	81.90
	95	**83.89**	83.80	70.69	75.84	79.27	79.87	73.52	75.76	79.37
	97	**89.12**	88.36	84.25	77.33	86.37	85.13	84.61	83.96	85.99
Haberman	r_{int}	**75.20**	**75.20**	63.90	65.22	72.46	74.60	72.03	69.64	70.79
	80	**80.57**	79.10	74.21	72.90	78.08	79.88	79.54	74.69	79.48
	85	**81.72**	**81.72**	68.66	70.64	78.56	79.51	79.74	71.83	78.89
	90	74.44	74.34	66.70	74.33	73.64	**74.88**	72.24	67.97	71.43
	92	**83.71**	82.09	68.36	75.49	79.54	82.28	81.93	28.94	79.94
	95	**86.01**	85.57	76.84	79.58	83.08	81.66	74.62	76.83	83.06
	97	**88.56**	87.90	78.73	76.58	86.88	85.02	80.98	45.83	85.08
Cleveland	r_{int}	79.15	78.70	73.81	67.09	73.30	75.82	75.21	65.46	**80.93**
	95	**82.57**	**82.57**	75.91	43.67	73.35	74.03	68.94	66.52	75.10
	97	**79.90**	78.26	77.26	62.45	77.89	74.85	67.91	71.63	77.25
Dermatology	r_{int}	**88.77**	88.05	87.11	84.06	86.26	85.92	83.56	84.53	86.67
	95	**87.66**	86.89	85.38	82.99	86.25	86.93	83.98	83.01	86.37
	97	**89.45**	89.40	89.23	86.78	87.81	88.59	85.10	89.39	88.17

TABLE 5. G-mean for $R_{out} = 20$.

Data set	r_{imb}	TBWRSVM	TWRSVM	WRSVM	RSVM	WSVM	FSVM	SVM	TBSVM	TWSVM
Diabetes	r_{int}	75.51	74.39	71.41	74.64	75.25	**75.92**	75.24	64.14	74.57
	80	**75.52**	74.98	70.39	74.77	73.20	75.05	75.08	62.10	74.11
	85	**78.19**	76.85	77.34	76.27	77.06	75.69	73.56	76.59	75.69
	90	**76.34**	**76.34**	72.92	76.08	74.60	75.44	74.71	62.79	75.19
	92	**82.43**	82.14	79.61	78.87	80.61	81.77	80.82	77.35	81.48
	95	**84.35**	**84.35**	81.36	77.98	81.87	82.18	82.62	79.44	83.81
	97	**85.19**	83.16	83.56	78.55	82.00	82.18	82.08	81.04	83.64
Pima	r_{int}	**75.00**	74.95	70.72	73.66	73.33	74.89	74.56	65.20	74.33
	80	78.8	77.09	77.07	75.13	78.47	**80.98**	79.84	69.18	78.56
	85	**84.96**	84.83	78.71	79.03	84.13	83.12	82.48	82.96	81.81
	90	**80.76**	**80.76**	75.06	73.29	80.20	78.93	78.84	78.69	75.86
	92	**81.57**	80.95	71.62	76.90	81.02	80.68	79.25	79.89	78.92
	95	**81.26**	**81.26**	74.29	75.42	80.93	80.97	80.01	78.96	76.96
	97	**83.72**	**83.72**	68.63	72.59	81.98	80.84	80.17	80.07	76.95
German	r_{int}	76.75	77.07	72.20	71.06	76.33	**77.27**	76.64	70.20	76.25
	80	83.36	83.22	68.70	80.62	83.11	**83.52**	82.64	82.29	83.25
	85	**82.41**	**82.41**	68.06	78.58	81.45	82.18	81.55	80.47	81.96
	90	**84.83**	83.62	78.20	80.22	82.95	82.66	81.76	81.69	83.53
	92	**86.71**	**86.71**	80.59	84.10	86.56	85.70	84.93	84.10	85.65
	95	**90.28**	90.20	85.67	87.86	89.37	89.07	88.17	89.65	89.41
	97	**89.86**	89.83	84.67	84.28	87.38	87.32	86.76	88.80	88.60
Appendicitis	r_{int}	**84.63**	83.46	83.15	82.32	82.33	82.43	77.89	81.24	81.22
	85	87.34	87.03	79.97	81.41	87.45	**88.41**	83.67	80.55	82.34
	90	87.76	90.05	85.04	89.22	**91.17**	86.84	89.09	86.43	87.78
	92	**87.27**	**87.27**	86.50	83.17	82.93	83.31	85.74	80.04	80.56
	95	**92.47**	88.85	87.56	85.18	87.33	86.62	86.54	85.43	89.07
	97	**91.32**	89.01	78.41	90.82	90.04	88.74	88.40	87.31	90.04
Flare solar	r_{int}	**73.68**	69.13	72.27	64.79	68.86	69.91	67.66	63.12	72.31
	80	**83.48**	**83.48**	78.46	75.30	73.44	78.71	78.64	72.46	83.19
	85	84.68	84.79	80.95	77.82	78.81	81.40	80.14	82.47	**85.01**
	90	**82.59**	81.37	80.43	79.79	79.09	80.03	79.53	79.90	81.24
	92	**88.34**	82.19	82.19	82.66	81.59	83.14	82.59	86.37	84.65
	95	**87.05**	**87.05**	85.75	78.94	84.76	86.18	84.04	78.51	85.07
	97	**92.11**	91.26	90.37	86.77	88.66	86.97	86.62	84.80	90.97
Heart	r_{int}	**77.61**	76.71	76.13	74.45	75.60	76.11	74.93	71.51	76.50
	80	**80.17**	79.00	72.79	79.50	72.11	76.09	76.65	67.08	78.96
	85	78.01	77.37	73.73	73.77	77.47	77.06	74.59	63.50	**79.38**
	90	77.48	79.35	75.98	76.38	73.43	76.68	73.33	59.35	**79.60**
	92	**81.21**	80.30	76.31	76.89	73.34	79.55	80.06	65.32	79.22
	95	**79.75**	76.31	70.97	65.50	76.00	75.56	68.83	45.65	75.74
	97	**80.12**	78.34	75.34	66.79	74.67	78.14	75.51	62.25	78.15
Thyroid	r_{int}	85.69	85.57	81.47	85.29	86.51	**89.77**	89.22	79.71	75.12
	80	**89.95**	88.26	74.79	80.64	83.15	87.49	87.49	75.91	85.13
	85	86.98	86.93	82.92	80.92	91.31	91.28	**92.86**	82.33	87.65
	90	85.88	85.88	77.37	85.06	**90.19**	89.03	89.24	82.61	85.59
	92	**91.67**	90.19	83.14	84.91	89.67	88.61	89.22	80.05	88.44
	95	94.62	94.62	87.78	82.02	90.43	93.48	**96.20**	86.04	95.63
	97	**94.42**	93.47	92.13	91.23	90.79	92.08	87.64	88.97	92.13
Breast cancer	r_{int}	71.80	69.37	67.73	70.51	70.30	**72.09**	71.35	59.32	68.77
	80	**77.12**	75.90	75.58	77.01	73.86	75.36	75.21	67.72	75.86
	85	**78.27**	75.44	75.77	73.23	74.87	76.59	77.61	73.26	75.60
	90	78.53	78.32	76.62	**82.85**	78.33	78.59	76.63	72.13	76.39
	92	79.80	79.80	74.50	76.94	75.70	78.11	78.18	70.19	**80.70**
	95	**80.69**	79.10	75.30	74.69	78.30	79.23	76.81	72.50	79.06
	97	83.21	83.21	80.73	72.26	82.69	83.98	**85.16**	77.84	82.37
Bupa	r_{int}	**76.50**	75.16	70.94	74.54	75.60	76.42	74.53	59.26	74.56
	80	**77.69**	**77.69**	72.15	77.66	75.55	75.83	75.74	64.19	74.63
	85	79.55	79.55	75.42	77.24	80.34	79.76	**81.08**	69.51	79.67
	90	81.03	78.55	74.17	80.92	79.02	80.79	**81.60**	71.56	78.78
	92	**85.50**	85.39	80.43	80.57	82.68	85.22	84.27	75.70	83.59
	95	**84.57**	84.26	78.06	83.22	83.99	84.14	84.05	74.73	83.82
	97	**90.92**	89.78	88.67	84.56	83.51	89.11	85.92	87.81	89.72
Haberman	r_{int}	76.76	76.76	66.81	66.25	75.73	77.49	**78.25**	73.27	75.19
	80	**84.23**	84.10	75.44	83.28	82.54	82.88	83.49	79.99	83.16
	85	**73.83**	72.94	71.96	70.82	69.62	65.53	45.93	66.22	66.91
	90	**80.62**	**80.62**	69.85	80.13	78.27	78.31	80.14	74.53	76.45
	92	82.68	81.89	71.74	81.26	82.75	83.25	83.19	77.11	**84.19**
	95	87.39	86.25	80.01	85.06	**87.91**	87.22	87.52	81.26	87.04
	97	88.27	87.80	78.93	86.93	**90.68**	90.48	90.20	85.02	88.17
Cleveland	r_{int}	**80.15**	**80.15**	74.83	75.24	78.19	77.94	79.22	53.30	78.56
	95	**77.21**	76.58	75.21	72.68	71.14	77.05	73.41	68.62	76.01
	97	84.08	80.00	78.07	71.48	78.87	79.06	78.39	69.99	**84.51**
Dermatology	r_{int}	**87.25**	**87.25**	86.49	83.51	84.26	85.34	85.39	81.36	87.07
	95	87.48	87.12	87.41	87.02	86.17	86.74	86.79	85.76	**87.94**
	97	88.08	88.08	86.97	86.80	86.15	86.31	86.17	84.71	**88.15**

TABLE 6. CPU time for R_{out} 10, 15 and 20.

		$R_{out} = 10$			$R_{out} = 15$			$R_{out} = 20$		
Dataset	r_{imb}	TBWRSVM	TWRSVM	WRSVM	TBWRSVM	TWRSVM	WRSVM	TBWRSVM	TWRSVM	WRSVM
Diabetes	r_{int}	**1.2892**	1.3132	50.6399	**1.2251**	1.2400	51.1515	**1.2925**	1.3661	51.4858
	80	**1.2954**	1.2978	48.138	**1.1710**	1.1992	49.6759	**1.2869**	1.3076	47.7920
	85	**0.7981**	0.8911	28.1039	**1.4171**	1.5025	47.5611	**1.4171**	1.5025	47.5611
	90	**1.2155**	1.2836	47.1817	**1.2082**	1.2137	47.5827	**1.2766**	1.2892	47.8647
	92	**0.7913**	0.8638	23.3861	**0.6588**	0.6956	24.0945	**0.5401**	0.5613	23.4195
	95	**0.8072**	0.8864	22.8739	**0.6304**	0.6917	22.2685	**0.5555**	0.5809	22.3721
	97	**0.7936**	0.8620	20.4563	**0.6391**	0.6737	21.5024	**0.5197**	0.5316	21.8872
Pima	r_{int}	**1.2184**	1.2905	41.9774	1.2068	**1.2051**	40.5462	**1.2480**	1.3178	40.7942
	80	**0.8949**	0.9298	25.2388	**0.7622**	0.7962	24.8853	**0.7220**	0.7615	24.4526
	85	**0.8399**	0.8810	21.2651	**0.6977**	0.7663	20.5457	**0.7386**	0.7894	23.1764
	90	**0.8820**	0.8993	18.1791	**0.6751**	0.6844	19.6361	**0.5536**	0.5827	20.0321
	92	**0.8050**	0.8565	18.5994	**0.6315**	0.6504	18.1412	**0.5382**	0.5553	18.3455
	95	**0.8205**	0.8861	15.6033	**0.6256**	0.6771	16.2531	0.5843	**0.5603**	16.5883
	97	**0.8119**	0.9332	15.8620	**0.6445**	0.6861	16.8588	**0.5040**	0.5354	15.8804
German	r_{int}	**4.4225**	4.8037	81.1662	3.7771	**3.6272**	76.2071	**4.0270**	4.9670	69.9129
	80	**2.9822**	3.3227	50.3407	**2.4047**	2.4858	54.7613	**2.4973**	2.7267	50.4198
	85	3.1752	**3.0599**	44.0094	**2.2338**	2.2699	49.3827	**2.1861**	2.5209	47.3656
	90	**3.0540**	3.1040	36.7087	**2.3917**	2.5407	37.5701	**2.0463**	2.2751	37.8774
	92	**2.9806**	3.0740	41.0366	**2.4042**	2.4060	38.4786	**2.0215**	2.2456	36.7470
	95	**2.9175**	3.1967	35.5237	**2.3795**	2.4282	33.4704	**1.9786**	2.0627	32.4416
	97	**2.9689**	2.9716	35.6327	**2.4052**	2.4777	27.1541	**1.8996**	2.0836	32.6826
Appendicitis	r_{int}	**0.0356**	0.0463	0.7765	**0.0314**	0.0330	0.8745	**0.0309**	0.0365	0.7495
	85	**0.0309**	0.0327	0.6611	**0.0265**	0.0376	0.7542	**0.0277**	0.0290	0.6735
	90	**0.0343**	0.0398	0.4651	**0.0254**	0.0365	0.7013	**0.0250**	0.0291	0.5415
	92	**0.0281**	0.0377	0.6591	**0.0231**	0.0312	0.6875	**0.0232**	0.0281	0.6908
	95	**0.0257**	0.0464	0.4060	**0.0237**	0.0289	0.5514	**0.0237**	0.0330	0.5394
	97	**0.0276**	0.0420	0.4425	**0.0295**	0.0309	0.4745	**0.0290**	0.0299	0.4998
Flare solar	r_{int}	**0.0433**	0.0472	1.3160	**0.0432**	0.0447	1.2774	**0.0456**	0.0503	1.2724
	80	**0.0389**	0.0422	0.8871	**0.0351**	0.0363	1.0168	**0.0348**	0.0363	1.0456
	85	**0.0355**	0.0405	0.8485	**0.0332**	0.0359	0.9724	**0.0350**	0.0370	0.9958
	90	**0.0350**	0.0382	0.6837	**0.0302**	0.0336	0.8035	**0.0287**	0.0302	0.8237
	92	**0.0338**	0.0416	0.7904	**0.0315**	0.0341	0.7377	**0.0285**	0.0337	0.8177
	95	**0.0333**	0.0376	0.7262	**0.0299**	0.0321	0.8184	**0.0283**	0.0303	0.7756
	97	**0.0332**	0.0435	0.7762	**0.0283**	0.0315	0.7283	**0.0279**	0.0311	0.7575
Heart	r_{int}	**0.1561**	0.1864	5.0556	**0.1356**	0.1396	5.4779	**0.1271**	0.1435	5.3332
	80	**0.0919**	0.0997	2.2388	0.0840	**0.0813**	2.5491	**0.0690**	0.0725	2.6629
	85	**0.0835**	0.0924	2.0532	**0.0750**	0.0774	2.2039	**0.0623**	0.0654	2.0681
	90	**0.0786**	0.1030	1.9448	**0.0729**	0.0803	2.1869	**0.0590**	0.0647	1.9789
	92	**0.0842**	0.0947	1.7104	**0.0691**	0.0741	1.9326	0.0592	**0.0574**	1.8706
	95	**0.0780**	0.0975	1.7878	**0.0666**	0.0723	2.0548	**0.0566**	0.0579	1.8522
	97	**0.0729**	0.0976	1.5329	**0.0720**	0.0800	1.4690	**0.0516**	0.0584	1.6610
Thyroid	r_{int}	0.1486	**0.1357**	3.6285	**0.2273**	0.2356	3.7964	**0.1711**	0.1968	3.6646
	80	**0.1177**	0.1291	2.8427	0.1421	**0.1190**	2.7454	**0.1311**	0.1394	2.6772
	85	**0.1598**	0.1615	2.4362	**0.1223**	0.1611	2.4392	**0.1411**	0.1521	2.4939
	90	**0.1298**	0.1132	2.2900	**0.1077**	0.4016	2.2165	**0.0901**	0.1311	2.2113
	92	**0.1113**	0.1172	2.1292	**0.1074**	0.1542	2.1144	**0.1014**	0.1113	2.1095
	95	**0.1333**	0.1579	2.0387	0.1139	**0.1138**	2.0463	**0.1043**	0.1341	1.9668
	97	**0.1748**	0.1780	1.9067	**0.1219**	0.1746	1.8702	**0.0928**	0.1366	1.9814
Breast cancer	r_{int}	**0.1405**	0.1491	5.6638	**0.1442**	0.1690	5.5002	**0.1525**	0.1560	5.4152
	80	**0.1104**	0.1274	4.4403	0.1628	**0.1211**	4.3349	**0.1215**	0.1319	4.4773
	85	**0.0978**	0.1164	3.7365	**0.1160**	0.1250	3.9504	**0.1049**	0.1169	4.0914
	90	**0.1078**	0.1285	3.3579	**0.1190**	0.1429	3.6065	**0.0947**	0.0983	3.6569
	92	**0.0982**	0.1074	3.1497	**0.0930**	0.1039	3.5061	**0.0926**	0.0942	3.6377
	95	**0.0916**	0.1092	2.9027	**0.0901**	0.1075	2.6426	**0.0842**	0.0927	3.0880
	97	**0.0866**	0.0990	2.7294	0.1043	**0.0982**	3.0636	**0.0859**	0.1014	2.7114
Bupa	r_{int}	**0.1950**	0.2017	8.8727	**0.2027**	0.2154	8.5615	**0.2200**	0.2345	8.4377
	80	**0.1302**	0.1367	5.2346	**0.1454**	0.1910	4.8808	**0.1329**	0.1473	5.0003
	85	**0.1174**	0.1299	4.6933	**0.1135**	0.1193	4.3346	**0.1228**	0.1361	4.1923
	90	**0.1084**	0.1220	3.5585	**0.1077**	0.1146	4.1368	**0.1123**	0.1231	4.3244
	92	**0.1041**	0.1133	3.4082	0.1105	**0.1087**	3.5503	**0.1045**	0.1166	3.9197
	95	**0.1009**	0.1177	3.7716	**0.0956**	0.1008	3.5291	**0.1028**	0.1181	3.1500
	97	**0.1063**	0.1189	2.9035	**0.0936**	0.1013	3.1212	**0.1078**	0.1225	3.4407
Haberman	r_{int}	**0.1798**	0.1932	6.8998	**0.1714**	0.1870	7.7233	**0.1615**	0.1714	7.6418
	80	**0.1635**	0.1641	6.7197	0.1633	**0.1533**	6.4735	**0.1841**	0.2202	6.5038
	85	**0.1340**	0.1761	5.5929	**0.1468**	0.1627	5.9279	**0.1880**	0.2043	7.2879
	90	**0.1553**	0.1730	7.6466	**0.1711**	0.1777	7.7575	**0.1611**	0.1692	7.8358
	92	**0.1197**	0.1388	4.9554	**0.1140**	0.1264	5.0471	**0.1191**	0.1218	4.8552
	95	**0.1306**	0.1461	4.7779	**0.1180**	0.1298	4.6523	**0.1123**	0.1158	4.3352
	97	**0.1186**	0.1413	4.5439	**0.1101**	0.1205	4.4583	**0.1044**	0.1132	4.4942
Cleveland		**0.0919**	0.1075	1.9377	**0.0806**	0.0875	2.2146	**0.0679**	0.0783	2.0349
	r_{int} 97	**0.0960**	0.1082	1.9089	**0.0728**	0.0836	2.1773	**0.0634**	0.0665	1.6947
		0.0925	0.1125	1.8509	**0.0748**	0.0838	1.6906	**0.0622**	0.0678	2.0352
Dermatology	r_{int}	0.3709	**0.3624**	9.5568	**0.2835**	0.2854	10.7446	**0.0679**	0.0783	2.0349
	95	**0.3430**	0.3439	10.4293	**0.2684**	0.2933	10.5862	**0.0634**	0.0665	1.6947
	97	**0.3256**	0.3372	8.6680	**0.2740**	0.3031	9.4013	**0.0622**	0.0678	2.0352

TABLE 7. Mean rank of Friedman test for R_{out} 10, 15 and 20.

	$R_{out} = 10$	$R_{out} = 15$	$R_{out} = 20$
TBWRSVM	3.81	3.89	3.78
TWRSVM	2.97	3.04	3.01
WRSVM	1.93	1.59	1.53
RSVM	1.28	1.48	1.68

TABLE 8. Wilcoxon signed-rank test.

Ranks

		N	Mean Rank	Sum of Ranks
TWRSVM – TBWRSVM	Negative Ranks	2[a]	31.00	62.00
	Positive Ranks	73[b]	38.19	2788.00
	Ties	0[c]		
	Total	75		

a. TWRSVM < TBWRSVM
b. TWRSVM > TBWRSVM
c. TWRSVM = TBWRSVM

Test Statistics[a]

	TWRSVM – TBWRSVM
Z	-7.198[b]
Asymp. Sig. (2-tailed)	.000

a. Wilcoxon Signed Ranks Test
b. Based on negative ranks.

in some cases it is increased, which significantly affirms the suitability of our proposed method for imbalanced datasets.

TBWRSVM is expected to perform much faster in comparison with WRSVM. CPU time of TBWRSVM and WRSM and also TWRSVM are given in Table 6. It is obvious, TBWRSVM is faster in the most cases and it has significant results in comparison with WRSVM, something about more than 20 times faster on average.

Here, the Friedman test [88] was utilized to determine the superiority of TBWRVM in comparison with TWRSVM, WRSVM and RSVM in a more accurate manner. This test is a non-parametric statistical test and its basis is considered ranking. Its usage is for difference detecting in treatments across various test endeavors. The results of the Friedman test in terms of G-maen for outlier ratios of 10, 15 and 20 are listed in Table 7. Based on this test, our proposed TBWRSVM is superior. High amount of this value for TBWRSVM and its difference with others implies great improvement of it. It should be noted that RSVM has a low amount of the ranking value in this test which may be associated with its development's goal which make it to be outlier robustness but not to be proper for imbalanced data in particular.

Wilcoxon signed-rank test [89] is another non-parametric statistical test which is exploited for statistical affirmation of CPU time results. In this test, the null hypothesis is that the two methods are similar in terms of CPU time and 0.05 is the regulated significance level in this test.

According to Table 6, very high difference between TBWRSVM and WRSVM in CPU time is thoroughly evident. So we apply Wilcoxon signed-rank test for comparing TBWRSVM and TWRSVM which are close to each other. Results of this test for outlier ratio of 20 are given in Table 8, which null hypothesis is disapproved ($0.001 < 0.05$) based on it and the CPU time for TBWRSVM is less than TWRSVM. It's obvious, TBWRSVM is faster.

VI. CONCLUSION

In this paper, we proposed a new Twin Bounded Weighted Relaxed Support Vector Machine (called TBWRSVM), which can concurrently deal with the impact of imbalanced data and outlier noise, enhance classification performance and do it in significant fast manner.

In the presented method, instead of one optimization problem, two problems are utilized to achieve high-speed runtime. Moreover, to increase the generalization ability, a regularization term is added to the primal problem with the aim of maximizing margin. In addition, to deal with the imbalanced data and to reduce the effect of noise and outliers, the penalization cost and a limited value of free slack are used respectively.

Experiments have been made on several data sets, indicating that the proposed method is premier to the other methods such as TWRSVM, WRSVM, RSVM and other robust classification methods. This superior is demonstrated in both computation time and classification accuracy in most data sets for both low and high outlier ratio.

ACKNOWLEDGMENT

The authors would like to thank the High Performance Computing Research Center (HPCRC) of Amirkabir University of Technology for providing computer facilities.

REFERENCES

[1] N. Japkowicz, "The class imbalance problem: Significance and strategies," in *Proc. Int. Conf. Artif. Intell.*, 2000, pp. 111–117.
[2] N. V. Chawla, N. Japkowicz, and A. Kotcz, "Editorial: Special issue on learning from imbalanced data sets," *ACM SIGKDD Explor. Newslett.*, vol. 6, no. 1, pp. 1–6, 2004.
[3] Y. Sun, A. K. Wong, and M. S. Kamel, "Classification of imbalanced data: A review," *Int. J. Pattern Recognit. Artif. Intell.*, vol. 23, no. 4, pp. 687–719, 2009.
[4] G. M. Weiss, "Mining with rarity: A unifying framework," *ACM SIGKDD Explorations Newslett.*, vol. 6, no. 1, pp. 7–19, Jun. 2004.
[5] G. M. Weiss and F. Provost, "Learning when training data are costly: The effect of class distribution on tree induction," *J. Artif. Intell. Res.*, vol. 19, no. 1, pp. 315–354, Jul. 2003.
[6] N. Japkowicz and S. Stephen, "The class imbalance problem: A systematic study," *Intell. Data Anal.*, vol. 6, no. 5, pp. 429–449, Oct. 2002.
[7] D. A. Cieslak and N. V. Chawla, "Start globally, optimize locally, predict globally: Improving performance on imbalanced data," in *Proc. 8th IEEE Int. Conf. Data Mining*, Dec. 2009, pp. 143–152.
[8] W. Wei, J. Li, L. Cao, Y. Ou, and J. Chen, "Effective detection of sophisticated online banking fraud on extremely imbalanced data," *World Wide Web*, vol. 16, no. 4, pp. 449–475, 2012.
[9] D. P. Williams, V. Myers, and M. S. Silvious, "Mine classification with imbalanced data," *IEEE Geosci. Remote Sens. Lett.*, vol. 6, no. 3, pp. 528–532, Jul. 2009.
[10] W.-Z. Lu and D. Wang, "Ground-level ozone prediction by support vector machine approach with a cost-sensitive classification scheme," *Sci. Total Environ.*, vol. 395, nos. 2–3, pp. 109–116, 2008.
[11] Y.-M. Huang, C.-M. Hung, and H. C. Jiau, "Evaluation of neural networks and data mining methods on a credit assessment task for class imbalance problem," *Nonlinear Anal., Real World Appl.*, vol. 7, no. 4, pp. 720–747, 2006.

[12] Y.-H. Liu and Y.-T. Chen, "Total margin based adaptive fuzzy support vector machines for multiview face recognition," in *Proc. IEEE Int. Conf. Syst., Man Cybern.*, vol. 2, Oct. 2005, pp. 1704–1711.

[13] M. Kubat, R. C. Holte, and S. Matwin, "Machine learning for the detection of oil spills in satellite radar images," *Mach. Learn.*, vol. 30, nos. 2–3, pp. 195–215, 1998.

[14] M. A. Mazurowski, P. A. Habas, J. M. Zurada, J. Y. Lo, J. A. Baker, and G. D. Tourassi, "Training neural network classifiers for medical decision making: The effects of imbalanced datasets on classification performance," *Neural Netw.*, vol. 21, nos. 2–3, pp. 427–436, 2008.

[15] A. Freitas, A. Costa-Pereira, and P. Brazdil, "Cost-sensitive decision trees applied to medical data," in *Data Warehousing and Knowledge Discovery* (Lecture Notes in Computer Science), vol. 4654, I. Y. Song, J. Eder, and T. Nguyen, Eds. Berlin, Germany: Springer, 2007, pp. 303–312.

[16] M. You, R.-W. Zhao, G.-Z. Li, and X. Hu, "MAPLSC: A novel multi-class classifier for medical diagnosis," *Int. J. Data Mining Bioinf.*, vol. 5, no. 4, pp. 383–401, 2011.

[17] Y. Ma, G. Luo, J. Li, and A. Chen, "Combating class imbalance problem in semi-supervised defect detection," in *Proc. IEEE Int. Conf. Comput. Problem-Solving (ICCP)*, Oct. 2011, pp. 619–622.

[18] T. Fawcett, "'*In vivo*' spam filtering: A challenge problem for KDD," *ACM SIGKDD Explor. Newslett.*, vol. 5, no. 2, pp. 140–148, 2003.

[19] G. Kim, B. K. Chae, and D. L. Olson, "A support vector machine (SVM) approach to imbalanced datasets of customer responses: Comparison with other customer response models," *Service Bus.*, vol. 7, no. 1, pp. 167–182, 2013.

[20] G. Wu, Y. Wu, L. Jiao, Y.-F. Wang, and E. Y. Chang, "Multi-camera spatio-temporal fusion and biased sequence-data learning for security surveillance," in *Proc. 11th ACM Int. Conf. Multimedia*, 2003, pp. 528–538.

[21] D. A. Cieslak, N. V. Chawla, and A. Striegel, "Combating imbalance in network intrusion datasets," in *Proc. IEEE Int. Conf. Granular Comput.*, 2006, pp. 732–737.

[22] A. Al-Shahib, R. Breitling, and D. Gilbert, "Feature selection and the class imbalance problem in predicting protein function from sequence," *Appl. Bioinf.*, vol. 4, no. 3, pp. 195–203, 2005.

[23] A. Adam, L. C. Chew, M. I. Shapiai, L. W. Jau, Z. I. Shapiai, and M. Khalid, "A hybrid artificial neural network-Naive Bayes for solving imbalanced dataset problems in semiconductor manufacturing test process," in *Proc. 11th Int. Conf. Hybrid Intell. Syst. (HIS)*, 2011, pp. 133–138.

[24] G.-S. Hu, F.-F. Zhu, and Z. Ren, "Power quality disturbance identification using wavelet packet energy entropy and weighted support vector machines," *Expert Syst. Appl.*, vol. 35, nos. 1–2, pp. 143–149, 2008.

[25] S. Suresh, R. V. Babu, and H. J. Kim, "No-reference image quality assessment using modified extreme learning machine classifier," *Appl. Soft Comput.*, vol. 9, no. 2, pp. 541–552, 2009.

[26] C. E. Brodley and M. A. Friedl, "Identifying mislabeled training data," *J. Artif. Intell. Res.*, vol. 11, no. 1, pp. 131–167, 1999.

[27] S. Zhong, W. Tang, and T. M. Khoshgoftaar, "Boosted noise filters for identifying mislabeled data," Dept. Comput. Sci. Eng., Florida Atlantic Univ., Boca Raton, FL, USA, 2005.

[28] M. Pechenizkiy, A. Tsymbal, S. Puuronen, and O. Pechenizkiy, "Class noise and supervised learning in medical domains: The effect of feature extraction," in *Proc. 19th IEEE Int. Symp. Comput.-Based Med. Syst. (CBMS)*, Jun. 2006, pp. 708–713.

[29] N. V. Chawla, "Data mining for imbalanced datasets: An overview," in *Data Mining and Knowledge Discovery Handbook*. New York, NY, USA: Springer, 2009, pp. 875–886.

[30] R. Batuwita and V. Palade, "FSVM-CIL: Fuzzy support vector machines for class imbalance learning," *IEEE Trans. Fuzzy Syst.*, vol. 18, no. 3, pp. 558–571, Jun. 2010.

[31] V. Vapnik, *The Nature of Statistical Learning Theory*. New York, NY, USA: Springer-Verlag, 1995.

[32] N. Cristianini and J. Shawe-Taylor, "*An Introduction to Support Vector Machines and Other Kernel-Based Learning Methods*. Cambridge, U.K.: Cambridge Univ. Press, 2000.

[33] C. J. C. Burges, "A tutorial on support vector machines for pattern recognition," *Data Mining Knowl. Discovery*, vol. 2, no. 2, pp. 121–167, 1998.

[34] L. Mason, J. Baxter, P. L. Bartlett, and M. Frean, "Functional gradient techniques for combining hypotheses," in *Advances in Neural Information Processing Systems*. Cambridge, MA, USA: MIT Press, 2000, pp. 221–246.

[35] P. L. Bartlett and S. Mendelson, "Rademacher and Gaussian complexities: Risk bounds and structural results," *J. Mach. Learn. Res.*, vol. 3, pp. 463–482, Mar. 2003.

[36] Q. Song, W. Hu, and W. Xie, "Robust support vector machine with bullet hole image classification," *IEEE Trans. Syst., Man, Cybern. C, Appl. Rev.*, vol. 32, no. 4, pp. 440–448, Nov. 2002.

[37] J. Weston and R. Herbrich, "Adaptive margin support vector machines," in *Advances in Large-Margin Classifiers*, A. J. Smola, P. L. Bartlett, B. Schölkopf, and D. Schuurmans, Eds. Cambridge, MA, USA: MIT Press, 2000, pp. 281–295.

[38] L.-J. Chien, Y.-J. Lee, Z.-P. Kao, and C.-C. Chang, "Robust 1-norm soft margin smooth support vector machine," in *Proc. 11th Int. Conf. Intell. Data Eng. Automat. Learn.*. Berlin, Germany: Springer, 2010, pp. 145–152.

[39] O. Şeref, W. A. Chaovalitwongse, and J. P. Brooks, "Relaxing support vectors for classification," *Ann. Oper. Res.*, vol. 216, no. 1, pp. 229–255, 2014.

[40] O. Şeref, T. Razzaghi, and P. Xanthopoulos, "Weighted relaxed support vector machines," *Ann. Oper. Res.*, vol. 249, no. 1, pp. 235–271, 2017.

[41] R. Khemchandani, Jayadeva, and S. Chandra, "Twin support vector machines for pattern classification," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 5, pp. 905–910, May 2007.

[42] Y. H. Shao, C. H. Zhang, X. B. Wang, and N. Y. Deng, "Improvements on twin support vector machines," *IEEE Trans. Neural Netw.*, vol. 22, no. 6, pp. 962–968, May 2011.

[43] O. Mangasarian and E. Wild, "Multisurface proximal support vector machine classification via generalized eigenvalues," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 1, pp. 69–74, Jan. 2006.

[44] R. Khemchandani, S. Chandra, and Jayadeva, "Fuzzy twin support vector machines for pattern classification," in *Mathematical Programming and Game Theory for Decision Making*. Singapore: World Scientific, 2008, pp. 131–142.

[45] M. A. Kumar and M. Gopal, "Least squares twin support vector machines for pattern classification," *Expert Syst. Appl.*, vol. 36, no. 4, pp. 7535–7543, 2009.

[46] S. Ghorai, A. Mukherjee, and P. K. Dutta, "Nonparallel plane proximal classifier," *Signal Process.*, vol. 89, no. 4, pp. 510–522, 2009.

[47] X. Peng, "A ν-twin support vector machine (ν-TSVM) classifier and its geometric algorithms," *Inf. Sci.*, vol. 180, no. 20, pp. 3863–3875, 2010.

[48] S.-G. Chen and X.-J. Wu, "A new fuzzy twin support vector machine for pattern classification," *Int. J. Mach. Learn.*, vol. 9, no. 9, pp. 1553–1564 2018.

[49] Z. Qi, Y. Tian, and Y. Shi, "Structural twin support vector machine for classification," *Knowl.-Based Syst.*, vol. 43, no. 2, pp. 74–81, May 2013.

[50] X. Xie, "Regularized multi-view least squares twin support vector machines," *Appl. Intell.*, vol. 48, no. 9, pp. 1–8, 2018.

[51] B. Liu, Y. Ma, and C. K. Wong, "Improving an association rule based classifier," in *Proc. Eur. Conf. Princ. Data Mining Knowl. Discovery*, 2000, pp. 293–317.

[52] Y. Lin, Y. Lee, and G. Wahba, "Support vector machines for classification in nonstandard situations," *Mach. Learn.*, vol. 46, pp. 191–202, Jan. 2002.

[53] R. Barandela, J. S. Sánchez, V. García, and E. Rangel, "Strategies for learning in class imbalance problems," *Pattern Recognit.*, vol. 36, no. 3, pp. 849–851, Mar. 2003.

[54] W. Liu, S. Chawla, D. A. Cieslak, and N. V. Chawla, "A robust decision tree algorithm for imbalanced data sets," in *Proc. SIAM Int. Conf. Data Mining*. Philadelphia, PA, USA: SIAM, 2010, pp. 766–777.

[55] G. E. Batista, R. C. Prati, and M. Monard, "A study of the behavior of several methods for balancing machine learning training data," *ACM SIGKDD Explor. Newslett.*, vol. 6, no. 1, pp. 20–29, 2004.

[56] K. Napierała, J. Stefanowski, and S. Wilk, "Learning from imbalanced data in presence of noisy and borderline examples," in *Proc. Int. Conf. Rough Sets Curr. Trends Comput.*, 2010, pp. 158–167.

[57] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic minority over-sampling technique," *J. Artif. Intell. Res.*, vol. 16, no. 1, pp. 321–357, 2002.

[58] S. Hu, Y. Liang, L. Ma, and Y. He, "MSMOTE: Improving classification performance when training data is imbalanced," in *Proc. 2nd Int. Workshop Comput. Sci. Eng.*, vol. 2, 2009, pp. 13–17.

[59] J. Stefanowski and S. Wilk, "Selective pre-processing of imbalanced data for improving classification performance," in *Proc. Int. Conf. Data Warehousing Knowl. Discover*, 2008, pp. 283–292.

[60] K. Veropoulos, C. Campbell, and N. Cristianini, "Controlling the sensitivity of support vector machines," in *Proc. Int. Joint Conf. Artif. Intell.*, 1999, pp. 55–60.
[61] C.-F. Lin and S.-D. Wang, "Fuzzy support vector machines," *IEEE Trans. Neural Netw.*, vol. 13, no. 2, pp. 464–471, Mar. 2002.
[62] T. Imam, K. M. Ting, and J. Kamruzzaman, "Z-SVM: An SVM for improved classification of imbalanced data," in *Proc. Australas. Joint Conf. Artif. Intell.*, 2006, pp. 264–273.
[63] G. Wu and E. Y. Chang, "KBA: Kernel boundary alignment considering imbalanced data distribution," *IEEE Trans. Knowl. Data Eng.*, vol. 17, no. 6, pp. 786–795, Jun. 2005.
[64] K. M. Ting, "A comparative study of cost-sensitive boosting algorithms," in *Proc. 17th Int. Conf. Mach. Learn.*, Stanford, CA, USA, 2000, pp. 983–990.
[65] M. Joshi, V. Kumar, and R. C. Agarwal, "Evaluating boosting algorithms to classify rare classes: Comparison and improvements," in *Proc. IEEE Int. Conf. Data Mining*, Nov./Dec. 2001, pp. 257–264.
[66] N. V. Chawla, A. Lazarevic, L. O. Hall, and W. K. Bowyer, "SMOTEBoost: Improving prediction of the minority class in boosting," in *Proc. Knowl. Discov. Databases*, 2003, pp. 107–119.
[67] C. Seiffert, T. M. Khoshgoftaar, J. Van Hulse, and A. Napolitano, "RUSBoost: A hybrid approach to alleviating class imbalance," *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 40, no. 1, pp. 185–197, Jan. 2010.
[68] S. Wang and X. Yao, "Diversity analysis on imbalanced data sets by using ensemble models," in *Proc. IEEE Symp. Comput. Intell. Data Mining*, Mar./Apr. 2009, pp. 324–331.
[69] R. Barandela, R. M. Valdovinos, and J. S. Sanchez, "New applications of ensembles of classifiers," *Pattern Anal. Appl.*, vol. 6, no. 3, pp. 245–256, Dec. 2003.
[70] H. Li, Y. Zong, K. Wang, and B. Wu, "A novel classification algorithm to noise data," in *Proc. Int. Conf. Adv. Swarm Intell.* Berlin, Germany: Springer, 2012, pp. 192–199.
[71] S. Wang, Z. Li, W. Chao, and Q. Cao, "Applying adaptive over-sampling technique based on data density and cost-sensitive SVM to imbalanced learning," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, 2012, pp. 1–8.
[72] G. H. John, "Robust decision trees: Removing outliers from databases," in *Proc. 1st Int. Conf. Knowl. Discovery Data Mining*, 1995, pp. 174–179.
[73] J. Cao, S. Kwong, and R. Wang, "A noise-detection based AdaBoost algorithm for mislabeled data," *Pattern Recognit.*, vol. 45, no. 12, pp. 4451–4465, 2012.
[74] S.-J. Zhao, H.-P. Zhang, and L. Li, "A new algorithm for imbalanced datasets in presence of outliers and noise," in *Proc. 8th Int. Conf. Natural Comput. (ICNC)*, 2012, pp. 30–34.
[75] S. Verbaeten and A. Van Assche, "Ensemble methods for noise elimination in classification problems," in *Multiple Classifier Systems*. Berlin, Germany: Springer, 2003, pp. 317–325.
[76] D. Guan, W. Yuan, Y.-K. Lee, and S. Lee, "Identifying mislabeled training data with the aid of unlabeled data," *Appl. Intell.*, vol. 35, no. 3, pp. 345–358, 2011.
[77] B. Sluban, D. Gamberger, and N. Lavrač, "Ensemble-based noise detection: Noise ranking and visual performance evaluation," *Data Mining Knowl. Discovery*, vol. 28, no. 2, pp. 265–303, 2012.
[78] C.-M. Teng, "Correcting noisy data," in *Proc. 16th Int. Conf. Mach. Learn.*. San Mateo, CA, USA: Morgan Kaufmann, 1999, pp. 239–248.
[79] J.-W. Sun, F.-Y. Zhao, C.-J. Wang, and S.-F. Chen, "Identifying and eliminating mislabeled training instances," in *Proc. Future Gener. Commun. Netw. (FGCN)*, vol. 1, 2007, pp. 244–250.
[80] T. Razzaghi and P. Xanthopoulos, "Cost-sensitive learning-based methods for imbalanced classification problems with applications," Ph.D. dissertation, Dept. Ind. Eng. Manage. Syst., Univ. Central Florida, Orlando, FL, USA, 2014.
[81] O. L. Mangasarian, *Nonlinear Programming*. Philadelphia, PA, USA: SIAM, 1994.
[82] J. Han, J. Pei, and M. Kamber, *Data Mining: Concepts and Techniques*, 2nd ed. San Mateo, CA, USA: Morgan Kaufmann, 2006.
[83] M. Kubat, R. Holte, and S. Matwin, "Learning when negative examples abound," in *Proc. Eur. Conf. Mach. Learn.*, 1997, pp. 146–153.
[84] A. P. Bradley, "The use of the area under the ROC curve in the evaluation of machine learning algorithms," *Pattern Recognit.*, vol. 30, no. 7, pp. 1145–1159, 1997.
[85] A. Estabrooks and N. Japkowicz, "A mixture-of-experts framework for learning from imbalanced data sets," in *Proc. Intell. Data Anal. Conf.*, 2001, pp. 34–43.
[86] C.-M. Huang, Y.-J. Lee, D. K. J. Lin, and S.-Y. Huang, "Model selection for support vector machines via uniform design," *Comput. Statist. Data Anal.*, vol. 52, no. 1, pp. 335–346, 2007.
[87] K.-T. Fang, "Uniform design: Application of number-theoretic methods in experimental design," *Acta Math. Appl. Sinica*, vol. 3, pp. 363–372, 1980.
[88] M. Friedman, "The use of ranks to avoid the assumption of normality implicit in the analysis of variance," *J. Amer. Statist. Assoc.*, vol. 32, no. 200, pp. 675–701, Dec. 1937.
[89] F. Wilcoxon, "Individual comparisons by ranking methods," *Biometrics Bull.*, vol. 1, no. 6, pp. 80–83, 1945.

FATEMEH ALAMDAR received the B.Sc. degree in computer software engineering from Shahid Bahonar University, Kerman, Iran, and the master's degree in computer engineering and artificial intelligence from Alzahra University, Iran, in 2011. She is currently pursuing the Ph.D. degree with the University of Zanjan, Iran. Her research interests include image processing, machine learning, and classification and clustering techniques.

FATEMEH SHEYKH MOHAMMADI received the bachelor's degree in computer software engineering from Imam Khomeini International University, Qazvin, Iran, in 2013, and the M.Sc. degree in computer software engineering from the University of Zanjan, Iran, in 2017.

ALI AMIRI was born in Zanjan, Iran. He received the B.Sc. degree in computer software engineering from the Zanjan Branch, Islamic Azad University, Zanjan, and the M.Sc. degree and the Ph.D. degree in computer engineering, artificial intelligence and robotic from the Iran University of Science and Technology, Tehran, Iran, in 2006 and 2011, respectively. He is currently an Assistant Professor with the Computer Engineering Department, University of Zanjan, Iran.