



# A Data Representation Approach to Support Imbalanced Data Classification Based on TWSVM

C. Jimenez<sup>(✉)</sup>, A. M. Alvarez, and A. Orozco

Automatics Research Group, Universidad Tecnologica de Pereira, Pereira, Colombia  
craljimenez@utp.edu.co

**Abstract.** Imbalance classification requires to represent the input data adequately to avoid biased results towards the class with the greater samples number. Here, we introduce an enhanced version of the famous twin support vector machine (TWSVM) classifier by incorporating an extended dual formulation of its quadratic programming optimization. Besides a centered kernel alignment (CKA)-based representation is used to avoid data overlapping. In particular, our approach, termed enhanced TWSVM (ETWSVM), allows representing the input samples in a high dimensional space (possibly infinite) after reformulation of the TWSVM dual form. Obtained results for binary classification demonstrate that our ETWSVM can reveal relevant data structures diminishing overlapping and biased classification results under imbalance scenarios. Moreover, ETWSVM notably adopts the lowest computational cost for training in comparison to state-of-the-art methods.

**Keywords:** TWSVM · Imbalanced data · Kernel enhancement

## 1 Introduction

Imbalanced data refer to datasets in which one of the classes have a higher number of samples than the others. The class with the highest number of samples is called the majority class, while the class with the lowest number of samples is known as the minority class. Traditional classification models tend to deal with the minority class as noise, that is, the minority class inputs are deemed as rare patterns. Besides, in several applications, i.e., natural disaster prediction, cancer gene expression discrimination, fraudulent credit card transactions, the non-identification of minority class samples yields to a massive cost [1].

Some machine learning approaches have been developed in the past decade to deal with imbalanced data, most of which have been based on sampling techniques, ensemble methods, and cost-sensitive learning [2]. The sampling techniques are applied over data to balance the number of samples between classes; this is done by eliminating samples from the majority class (under-sampling) and/or creating synthetic samples of the minority one (over-sampling). However, these techniques can generate a loss of information by eliminating majority

class samples, or overfitting due to the redundant information through generating synthetic minority class inputs [3]. Secondly, ensemble methods split the set of the majority class in several subsets with a size equal to the minority set. Then, this train as many classifiers as the number of majority class subsets. These approaches usually archive competitive performance; but, the computational cost is enormous since the number of classifiers to train [4]. In turn, cost-sensitive approaches modify the learning algorithm or the cost function by penalizing the misclassification of the minority class samples. Nevertheless, these techniques lead to overfitting, since it tends to bias the minority class [5]. In recent years, a twin support vector machines (TWSVM)-based algorithms take advantage of their generalization capacity and their low computational cost by constructing two non-parallel hyperplanes instead of only one hyperplane as the traditional support vector machine (SVM) [6]. Some TWSVM's extensions include a structural data representation a favor the generalization capacity of the classifier [7]. Additionally, TWSVM-based approaches have been proposed to counter the imbalanced data effect combining resampling techniques, coupling a between-class discriminant algorithm, and weighting the primal problem to prevent the overtraining [8,9]. Nonetheless, most of the TWSVM-based approaches do not include a suitable reproducing kernel Hilbert space (RKHS)-based representation, being difficult to incorporate an incorporate kernel approach within the optimization. Thus, they neglect the intrinsic formulation and virtues of a dual problem regarding kernel methods.

In this work, we propose a TWSVM-based cost-sensitive method that make a kernel-based enhancement of the TWSVM for imbalance data classification (ETWSVM), which allows representing the input samples in a high dimensional space. Also, we use a centered kernel alignment (CKA) method [10] with the objective of learning a kernel mapping to counteract inherent imbalanced issues, reduce the computational time and enhance the data representation. As benchmarks, we test the standard SVM classifier [11], a support vector machine with slack variables regulated [5], a SMOTE with SVM [12], and a weighted Lagrangian twin support vector machine (WLTSM) [13] for binary classification. Results obtained from benchmarks databases of the state-of-the-art show that our approach outperforms the baseline methods concerning both the accuracy and the geometric mean-based classification assessment.

The remainder of this paper is organized as follows: Sect. 2 describes the material and methods of the approach proposed for classification. Sections 3 and 4 describe the experimental set-up and the results obtained, respectively. Finally, the concluding remarks are outlined in Sect. 5.

## 2 Enhanced TWSVM

*Nonlinear Extension of the TWSVM Classifier.* Let  $\{\mathbf{x}_n \in \mathbb{R}^P, y_n \in \{+1, -1\}\}_{n=1}^N$  be an input training set, where each sample  $\mathbf{x}_n$  can belong to the minority class matrix  $\mathbf{X}^+ \in \mathbb{R}^{P \times N^+}$  ( $y_n = +1$ ), or to the majority class matrix  $\mathbf{X}^- \in \mathbb{R}^{P \times N^-}$  ( $y_n = -1$ ), being  $P$  the number of features and  $N = N^+ + N^-$

the number of samples. The well-know extension of the SVM classifier, termed twin support vector machine (TWSVM), relaxes the SVM's quadratic programming optimization (QPP) by employing two non-parallel hyperplanes, as follows:  $f^\ell(\mathbf{x}) = \mathbf{x}^\top \mathbf{w}^\ell + b^\ell = 0$ , where  $\ell \in \{+, -\}$ ,  $\mathbf{w}^\ell \in \mathbb{R}^P$  is a normal vector concerning  $f^\ell(\mathbf{x})$ , and  $b^\ell \in \mathbb{R}$  is a bias term. Traditionally, the non-linear extension of the TWSVM includes a kernel within the discrimination function as  $f^\ell(\mathbf{x}) = \sum_{n=1}^N \kappa(\mathbf{x}, \mathbf{x}_n) \tilde{w}_n^\ell + b^\ell$ , where  $\tilde{w}_n^\ell \in \mathbb{R}$  is the  $n$ -th element of the vector  $\tilde{\mathbf{w}}^\ell \in \mathbb{R}^N$ , and  $\kappa: \mathbb{R}^P \times \mathbb{R}^P \rightarrow \mathbb{R}$  is a kernel function. Nonetheless, such a non-linear extension does not consider directly a reproductive kernel Hilbert space (RKHS) within the empirical risk minimization-based cost function of the TWSVM, that is to say, it does not consider a mapping to a high dimensional (possible infinite) feature space.

So, we introduce a non-linear extension of the TWSVM, termed NTWSVM, optimization functional towards the mapping  $\varphi: \mathbb{R}^P \rightarrow \mathbb{R}^D$ , where  $D \rightarrow \infty$ . Thereby, the hyperplanes can be rewritten as:  $f^\ell(\mathbf{x}) = \varphi(\mathbf{x})^\top \mathbf{w}^\ell + b^\ell = 0$ . In particular, each hyperplane plays the role of a one-class classifier, aiming to enclose its corresponding class. Next, the NTWSVM's primal form yields:

$$\begin{aligned} \hat{\mathbf{w}}^\ell, \hat{b}^\ell, \hat{\boldsymbol{\xi}}^{\ell'} = \arg \min_{\mathbf{w}, b, \boldsymbol{\xi}} \frac{c_1}{2} (\|\mathbf{w}\|_2^2 + b^2) + \frac{1}{2} \|\mathbf{w}^\top \boldsymbol{\Phi}^\ell + \mathbf{1}^\ell b\|_2^2 + c_2 \mathbf{1}^{\ell'} \boldsymbol{\xi} \\ \text{s.t. } \left( \mathbf{w}^\top \boldsymbol{\Phi}^{\ell'} + b \mathbf{1}^{\ell'} \right) \boldsymbol{\Delta}^{\ell'} + \boldsymbol{\xi}^\top \geq \mathbf{1}^{\ell'}, \boldsymbol{\xi} \geq \mathbf{0}; \quad \forall \ell, \ell' \in \{+, -\}, \end{aligned} \quad (1)$$

where  $\boldsymbol{\Phi}^\ell = [\varphi(\mathbf{x}_n)]_{n=1}^{N^\ell} \forall \{\mathbf{x}_n \mid y_n = \ell\}$ ,  $\hat{\boldsymbol{\xi}}^{\ell'} \in \mathbb{R}^{N^{\ell'}}$  is a slack variable vector for the  $l$ -th class;  $c_1, c_2 \in \mathbb{R}^+$  are regularization parameters, the first one regularizes the slack variables and the second one the model parameters  $\hat{\mathbf{w}}^\ell$  and  $\hat{b}^\ell$  (rules the margin maximization). Besides,  $\mathbf{1}^\ell \in \mathbb{R}^{N^\ell}$  is an all ones row vector,  $\boldsymbol{\Delta}^{\ell'} = \ell' \mathbf{I}_{N^{\ell'}}$  being  $\mathbf{I}_{N^{\ell'}} \in \{0, 1\}^{N^{\ell'} \times N^{\ell'}}$  an identity matrix,  $\ell' = -\ell$ , and  $\|\cdot\|_2$  stands for the 2-norm. Later, the Wolfe dual form of Eq. (1) can be written as follows:

$$\hat{\boldsymbol{\alpha}}^{\ell'} = \arg \max_{\boldsymbol{\alpha}} - \frac{1}{2} \boldsymbol{\alpha}^\top \mathbf{M} \mathbf{B} \boldsymbol{\alpha} + \mathbf{1}^{\ell'} \boldsymbol{\alpha} \quad \text{s.t.} \quad \mathbf{0}^\top \leq \boldsymbol{\alpha}^\top \leq c_2 \mathbf{1}^{\ell'}, \quad (2)$$

where  $\hat{\boldsymbol{\alpha}}^{\ell'} \in \mathbb{R}^{N^{\ell'}}$  is a Lagrangian multiplier,  $\mathbf{M} \in \mathbb{R}^{N^{\ell'} \times N^{\ell'}}$  holds elements  $\mathbf{M}_{nn'} = \kappa(\mathbf{s}_n, \mathbf{s}_{n'} | \sigma^2 \mathbf{I}_{P+1})$ ,  $\kappa(\cdot, \cdot)$  is a multivariate Gaussian kernel,  $\mathbf{s}_n = [\mathbf{x}_n; 1]$  is an extended column vector of the matrix  $\mathbf{S}^{\ell'} \in \mathbb{R}^{(P+1) \times N^{\ell'}}$ ,  $\mathbf{B} \in \mathbb{R}^{N^{\ell'} \times N^{\ell'}}$  holds elements  $\mathbf{B}_{nn'} = \kappa(\mathbf{s}_n, \mathbf{s}_{n'} | \boldsymbol{\Sigma})$ ,  $\boldsymbol{\Sigma} = \left( c_1 \mathbf{I}_{D^{\ell'}+1} + \mathbf{S}^{\ell'} (\mathbf{S}^{\ell'})^\top \mathbf{S}^{\ell'} (\mathbf{S}^{\ell'})^\top \right)$ , and  $\mathbf{I}_{P+1}, \mathbf{I}_{D^{\ell'}+1}$  are identity matrix of proper size. Hence, the NTWSVM hyperplanes yields:  $f^\ell(\mathbf{x}) = \mathbf{d}^\top \mathbf{B} \boldsymbol{\Delta}^{\ell'} \hat{\boldsymbol{\alpha}}^{\ell'}$ , where  $\mathbf{d} \in \mathbb{R}^{N^{\ell'}}$  holds elements  $d_n = \kappa(\mathbf{s}, \mathbf{s}_n | \sigma^2 \mathbf{I}_{P+1})$  and  $\mathbf{s}_n \in \mathbf{S}^{\ell'}$ . Finally, the label of the new input is computed as the distance to the closest hyperplane as  $y^* = \arg \min_{\ell \in \{+, -\}} |f^\ell(\mathbf{x})|$ , where  $|\cdot|$  stands

for the absolute value. It worth noting that due to the one-class foundation of the NTWSVM approach, the distance to the hyperplanes are proportional to the score magnitude.

*CKA-Based Enhancement of the NTWSVM.* To avoid instability issues concerning the inverse of the matrix  $\Sigma$  and to reveal relevant data structures for imbalance classification tasks, we use a centered kernel alignment (CKA)-based approach to infer  $\Sigma^{-1} = \mathbf{E}\mathbf{E}^\top$ , where  $\mathbf{E} \in \mathbb{R}^{(P+1) \times P'}$  ( $P' \leq P+1$ ). In fact, we learn  $\mathbf{E}$  as a linear projection to match the kernels  $\mathbf{K}^x \in \mathbb{R}^{N \times N}$  and  $\mathbf{K}^y \in \mathbb{R}^{N \times N}$ , holding elements  $\mathbf{K}_{nn'}^x = \kappa(\mathbf{x}_n, \mathbf{x}'_{n'} | (\mathbf{E}\mathbf{E}^\top)^{-1})$ ,  $\mathbf{K}_{ij}^y = \delta(y_i, y_j)$ , being  $\delta(\cdot)$  the delta function. Besides, the empirical estimate of the CKA alignment between  $\mathbf{K}^x$  and  $\mathbf{K}^y$  is defined as [10]:

$$\hat{\rho}(\tilde{\mathbf{K}}^x, \tilde{\mathbf{K}}^y) = \frac{\langle \tilde{\mathbf{K}}^x, \tilde{\mathbf{K}}^y \rangle_{\mathbb{F}}}{\sqrt{\langle \tilde{\mathbf{K}}^x, \tilde{\mathbf{K}}^x \rangle_{\mathbb{F}} \langle \tilde{\mathbf{K}}^y, \tilde{\mathbf{K}}^y \rangle_{\mathbb{F}}}}, \quad (3)$$

where  $\tilde{\mathbf{K}}$  stands for the centered kernel matrix calculated as  $\tilde{\mathbf{K}} = \tilde{\mathbf{I}}\mathbf{K}\tilde{\mathbf{I}}$ , where  $\tilde{\mathbf{I}} = \mathbf{I} - \mathbf{1}\mathbf{1}^\top/N$ ,  $\mathbf{I} \in \mathbb{R}^{N \times N}$  is an identity matrix, and  $\langle \cdot, \cdot \rangle_{\mathbb{F}}$  denotes the matrix-based Frobenius inner product. Later, to compute the projection matrix  $\mathbf{E}$ , the following learning algorithm is employed:

$$\hat{\mathbf{E}} = \arg \max_{\mathbf{E}} \log \left( \hat{\rho}(\tilde{\mathbf{K}}^x, \tilde{\mathbf{K}}^y; \mathbf{E}) \right). \quad (4)$$

Lastly, given  $\Sigma^{-1} = \hat{\mathbf{E}}\hat{\mathbf{E}}^\top$ , the NTSVM is trained as above explained. For the sake of simplicity, we called our proposal as ETWSVM. A MatLab code of ETWSVM is publicly available<sup>1</sup>

### 3 Experimental Set-Up

*Datasets and ETWSVM Training.* A toy dataset comprising two classes belonging to Gaussian distribution with a mean vector of  $\mathbf{0} \in \mathbb{R}^2$  and a variance matrix of  $0.3\mathbf{I}_2$ , the minority samples are the samples with a probability greater than 0.7 and rest as majority one; and twelve benchmark datasets originating from the well-known UCI machine learning repository are used to test the ETWSVM proposed<sup>2</sup>. Table 1 displays the main properties of the studied UCI repository datasets; the minority class is marked and rest as the majority one. As quality assessment, we employ the classification performance regarding the accuracy (ACC) and the geometric mean (GM) measures. The ACC is computed as  $ACC = (TP + TN)/(TP + TN + FP + FN)$ , where  $TP$  is the number of samples belonging to class +1 and classified as +1,  $FN$  codes the number of samples belonging to class +1 and classified as -1,  $TF$  is the number of samples belonging to class -1 and classified as -1, and  $FP$  is the number of samples belonging to class -1 and classified +1. Traditionally, The ACC measure is used for classification, but it can be biased for imbalanced datasets. Therefore, the GM is utilized to test imbalance classification as follows:  $GM = \sqrt{\nu \times \zeta}$ ,

<sup>1</sup> <https://github.com/cralji/ETWSVM.git>.

<sup>2</sup> <https://archive.ics.uci.edu/ml/index.php>.

where  $\nu = TP/(TP + FN)$  and  $\zeta = TF/(TN + FP)$ . Besides, a nested 10-fold cross-validation scheme is carried out for ETWSVM training. Regarding the free parameter tuning, the regularization value  $c_2$  is fixed as  $c_2 = N^-/N^+$ , and the bandwidth is searched from the set  $\sigma \in \{0.1\sigma_0, 0.2\sigma_0, \dots, \sigma_0\}$ , where  $\sigma_0$  is the median of the input data Euclidean distances. Note that our CKA-based enhancement avoids the estimation of the  $c_1$  value in Eq. (1).

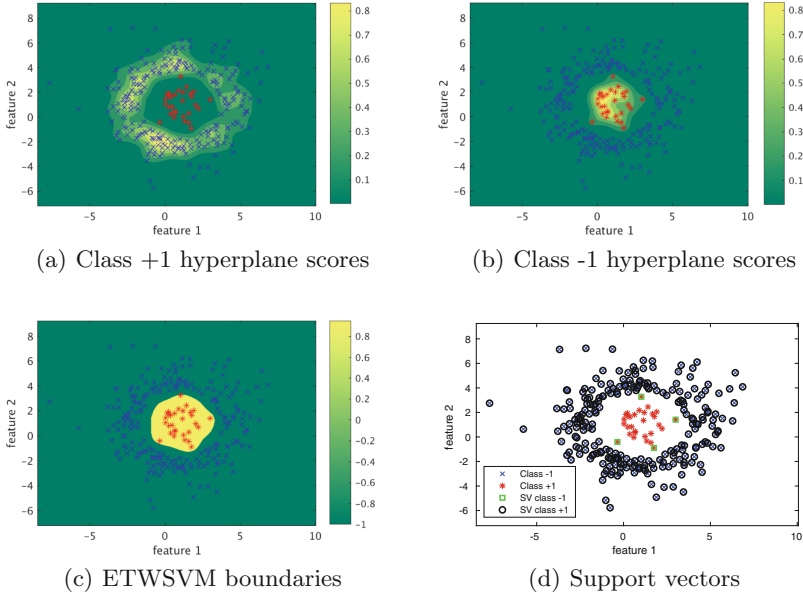
*Method Comparison.* As baseline, four classification approaches are tested: SVM [11], SVM with regularized slack variables (SVM reg-slack) [5], SVM with synthetic minority oversampling technique (SMOTE) [12], and a weighted Lagrangian twin support vector machine (WLTSVM) [13]. All provided methods include a Gaussian kernel. So, we infer the kernel bandwidth as in the ETWSVM training. Also, the regularization parameter value is computed from the set  $\{0.1, 1, 10, 100, 1000\}$  (for the WLTSVM technique we assume equal regularization parameter values). On the other hand, the  $k$  value for the SMOTE algorithm is fixed from the set  $\{3, 5, 9, 10, 15\}$  and the relationship between minority and majority samples is set to 1:1. All the classifiers are implemented in MATLAB R2016b environment on a PC with Intel i5 processor (2.7 GHz) with 6 GB RAM.

**Table 1.** Benchmark datasets from the UCI machine learning repository.

Datasets	Imbalance ratio	Features	Data size	Minority class
Haberman	0.3600	4	306	2
Housing	0.0743	13	506	1 (CHAS)
Vehicle	0.3076	18	846	Van
Transfusion	0.3123	4	748	1
Ionosphere	0.5600	34	351	b
Balance	0.0851	4	625	Balanced
Biodeg	0.5093	41	1055	RB
cmc	0.2921	9	1473	2:Long-term
Pima-Indians	0.5360	8	768	1
BankNote	0.8005	4	1372	1
Iris	0.5000	4	150	Iris Virginica
Wisconsin	0.5938	30	569	Malignant

## 4 Results and Discussion

Figure 1 presents the ETWSVM results on the toy dataset. As seen in Figs. 1(a) and (b), which show the generated score by the minority and majority hyperplanes, respectively; the majority hyperplane generates a higher score for the samples associated with the minority class. Such behavior occurs reciprocally between the minority hyperplane and the examples of the majority class. Indeed, each hyperplane function is fixed depending on the slack variables of its counterpart. Therefore, the minority class's support vectors are selected from the



**Fig. 1.** An illustrative example of the ETWSVM classifier. (a) and (b) show the scores hyperplane. (c) shows the decision boundary, where the yellow zone is classified as +1 and the blue as -1. (d) support vectors. (Color figure online)

majority hyperplane meanwhile the majority class’s support vectors from the minority one. Also, the Fig. 1(d) exposes the support vectors obtained, where the majority hyperplane only requires three support vectors, but the minority hyperplane chooses all the majority samples as support vectors to reveal the central data structure. As expected, our ETWSVM highlights the minority class data structure holding a higher number of support vectors than in the majority one. In turn, by combining the ETWSVM hyperplanes, Fig. 1(c) displays the classifier’s boundaries that encodes the class patterns correctly.

Next, the attained classification results on the real-world datasets are summarized in Table 2 concerning the ACC and the GM assessments. As seen, the average ACC performances for the SVM reg-slack, the SVM, the SMOTE-SVM, and our ETWSVM are closely similar. Nonetheless, the SMOTE-SVM and our proposal achieve more reliable results regarding the standard deviation values. Further, for the GM, which is preferred for evaluating imbalance data classification, demonstrates that the SMOTE-SVM and the introduced ETWSVM allow revealing discriminative patterns to avoid biased results. Again, the low standard deviation values of such approaches probe their dominance to cope with imbalance problems. Notably, the CKA-based kernel learning in Eq. (4),

**Table 2.** Classification results on the UCI repository datasets. ACC: accuracy, GM: geometric mean. Testing scheme: nested 10-fold cross-validation.

Datasets	SVM reg-slack [11]	SVM [4]	SMOTE-SVM [12]	WLTSVM [13]	ETWSVM
	ACC $\pm$ std (%)	ACC $\pm$ std (%)	ACC $\pm$ std (%)	ACC $\pm$ std (%)	ACC $\pm$ std (%)
	GM $\pm$ std (%)	GM $\pm$ std (%)	GM $\pm$ std (%)	GM $\pm$ std (%)	GM $\pm$ std (%)
Haberman	62.82 $\pm$ 08.52	71.53 $\pm$ 05.03	61.76 $\pm$ 05.20	59.49 $\pm$ 16.26	<b>71.90 <math>\pm</math> 08.88</b>
	53.42 $\pm$ 12.12	54.54 $\pm$ 09.53	56.24 $\pm$ 12.53	38.26 $\pm$ 22.86	<b>57.70 <math>\pm</math> 14.93</b>
Housing	<b>91.91 <math>\pm</math> 02.68</b>	88.55 $\pm$ 05.29	90.32 $\pm$ 03.65	44.86 $\pm$ 06.61	81.02 $\pm$ 07.61
	63.36 $\pm$ 16.72	67.73 $\pm$ 27.57	66.31 $\pm$ 25.13	46.18 $\pm$ 18.18	<b>72.11 <math>\pm</math> 15.95</b>
Vehicle	97.52 $\pm$ 01.42	<b>97.88 <math>\pm</math> 01.34</b>	97.63 $\pm$ 01.48	82.39 $\pm$ 04.62	92.19 $\pm$ 01.79
	97.66 $\pm$ 02.02	<b>97.71 <math>\pm</math> 01.79</b>	96.85 $\pm$ 02.39	82.20 $\pm$ 03.96	93.33 $\pm$ 02.34
Transfusion	<b>69.12 <math>\pm</math> 04.05</b>	68.04 $\pm$ 04.96	62.56 $\pm$ 10.27	35.95 $\pm$ 31.21	68.46 $\pm$ 06.65
	<b>64.78 <math>\pm</math> 05.91</b>	59.73 $\pm$ 22.00	51.91 $\pm$ 28.18	39.18 $\pm$ 33.89	63.92 $\pm$ 07.48
Ionosphere	<b>94.59 <math>\pm</math> 02.47</b>	94.01 $\pm$ 02.51	93.72 $\pm$ 04.96	82.40 $\pm$ 06.80	89.76 $\pm$ 06.29
	<b>93.56 <math>\pm</math> 03.11</b>	93.12 $\pm$ 03.29	92.23 $\pm$ 06.71	82.49 $\pm$ 07.61	85.64 $\pm$ 09.83
Balance	90.70 $\pm$ 03.34	<b>90.70 <math>\pm</math> 03.15</b>	83.83 $\pm$ 05.98	49.54 $\pm$ 20.70	59.71 $\pm$ 06.83
	<b>81.10 <math>\pm</math> 18.90</b>	57.32 $\pm$ 11.22	69.31 $\pm$ 13.40	52.49 $\pm$ 16.49	65.58 $\pm$ 12.86
Biodeg	87.02 $\pm$ 02.86	87.30 $\pm$ 02.98	<b>87.77 <math>\pm</math> 02.21</b>	77.82 $\pm$ 02.21	59.00 $\pm$ 40.78
	86.58 $\pm$ 02.78	86.49 $\pm$ 02.97	<b>86.84 <math>\pm</math> 02.71</b>	75.23 $\pm$ 02.23	59.83 $\pm$ 41.33
cmc	26.75 $\pm$ 04.74	33.20 $\pm$ 03.16	<b>67.82 <math>\pm</math> 06.23</b>	31.94 $\pm$ 22.44	61.71 $\pm$ 03.58
	19.88 $\pm$ 17.54	31.69 $\pm$ 04.05	58.76 $\pm$ 21.07	31.71 $\pm$ 22.56	<b>64.48 <math>\pm</math> 04.08</b>
Prima-Indians	25.26 $\pm$ 04.66	24.99 $\pm$ 04.30	72.78 $\pm$ 04.95	65.36 $\pm$ 06.95	<b>73.29 <math>\pm</math> 05.84</b>
	24.53 $\pm$ 05.56	24.47 $\pm$ 05.04	<b>72.71 <math>\pm</math> 04.60</b>	68.27 $\pm$ 07.09	71.87 $\pm$ 06.23
Banknote	88.26 $\pm$ 03.66	92.34 $\pm$ 12.99	<b>97.45 <math>\pm</math> 01.89</b>	91.84 $\pm$ 01.36	96.58 $\pm$ 01.65
	88.65 $\pm$ 03.62	86.64 $\pm$ 30.46	<b>97.42 <math>\pm</math> 02.11</b>	92.18 $\pm$ 01.28	96.72 $\pm$ 01.64
Iris	86.00 $\pm$ 09.14	99.33 $\pm$ 02.11	99.33 $\pm$ 02.11	99.33 $\pm$ 02.11	<b>100.00 <math>\pm</math> 00.00</b>
	73.92 $\pm$ 19.32	98.94 $\pm$ 03.34	98.94 $\pm$ 03.34	98.94 $\pm$ 03.34	<b>100.00 <math>\pm</math> 00.00</b>
Wisconsin	51.34 $\pm$ 05.77	<b>97.19 <math>\pm</math> 01.49</b>	97.18 $\pm$ 02.06	80.82 $\pm$ 05.55	97.01 $\pm$ 02.03
	46.55 $\pm$ 09.22	<b>96.96 <math>\pm</math> 01.72</b>	96.84 $\pm$ 02.30	81.54 $\pm$ 05.50	96.42 $\pm$ 02.31
Average	72.61 $\pm$ 25.83	78.76 $\pm$ 25.21	<b>84.35 <math>\pm</math> 14.35</b>	66.81 $\pm$ 22.34	79.22 $\pm$ 15.47
	66.17 $\pm$ 25.90	71.28 $\pm$ 26.02	<b>78.70 <math>\pm</math> 18.00</b>	65.72 $\pm$ 23.11	77.30 $\pm$ 15.99

favors the ETWSVM hyperplane representation through a non-linear matching between the input data dependencies and the output labels. Moreover, despite that SMOTE-SVM reports the best GM performance in average, it requires the higher computational time to train the classification system (see Table 3). On the other hand, the GM results for the ETWSVM are competitive but requires the lowest computational cost for training. The later can be explained because our ETWSVM optimization takes advantage of the one-class foundation of the straightforward TWSVM, while decreasing the class overlapping based on the CKA matching. Finally, it is worth noting to mention that the ETWSVM requires to fix few free parameters and solves two smaller sized QPP in comparison to the state-of-the-art algorithms.

**Table 3.** Classification results on the UCI repository datasets. Required time in seconds for one fold training.

Datasets	SVM reg-slack [11]	SVM [4]	SMOTE-SVM [12]	WLTSVM [13]	ETWSVM
HaberMan	06.50 ± 04.26	34.07 ± 32.22	114.73 ± 34.05	17.21 ± 00.42	<b>03.59 ± 00.76</b>
Balance	15.67 ± 02.91	<b>01.58 ± 00.06</b>	358.45 ± 190.86	42.53 ± 01.41	10.88 ± 01.62
Housing	111.11 ± 67.86	<b>03.17 ± 00.28</b>	1064.70 ± 126.78	21.78 ± 00.26	07.14 ± 01.44
Iris	00.29 ± 00.03	<b>00.22 ± 00.03</b>	01.36 ± 00.04	03.78 ± 00.07	01.72 ± 00.39
Vehicle	31.20 ± 16.03	<b>21.83 ± 09.59</b>	300.71 ± 49.30	75.96 ± 04.24	<b>22.45 ± 05.84</b>
Ionosphere	88.18 ± 78.22	102.79 ± 104.61	777.84 ± 161.20	12.60 ± 00.33	<b>04.19 ± 00.15</b>
Prima-Indians	162.53 ± 75.23	157.81 ± 50.10	1175.29 ± 141.30	60.93 ± 02.95	<b>13.97 ± 00.70</b>
Banknote	<b>01.18 ± 00.96</b>	01.96 ± 02.19	07.51 ± 01.98	278.07 ± 14.83	57.72 ± 09.77
Wisconsin	11.52 ± 07.18	12.16 ± 06.24	73.96 ± 27.75	31.97 ± 00.99	<b>08.60 ± 01.54</b>
biodeg	466.33 ± 126.19	581.15 ± 430.96	3688.18 ± 658.75	126.74 ± 09.69	<b>39.80 ± 09.08</b>
Transfusion	<b>00.67 ± 00.40</b>	03.95 ± 10.26	04.92 ± 01.11	50.68 ± 01.65	13.34 ± 03.63
cmc	02.34 ± 01.59	<b>01.60 ± 01.06</b>	15.20 ± 07.91	233.00 ± 09.44	67.65 ± 08.69
Average	74.79 ± 134.26	76.86 ± 166.28	631.90 ± 1051.79	79.60 ± 89.17	<b>20.92 ± 22.14</b>

## 5 Conclusions

In this study, we propose an enhanced version of the well-known twin support vector machine classifier. In fact, our approach, called ETWSVM, includes a direct non-linear mapping within the quadratic programming optimization of the standard TWSVM, which allows representing the input samples in a high dimensional space (possibly infinite) after reformulation of the TWSVM dual form. Besides, a centered kernel alignment-based approach is proposed to learn the ETWSVM kernel mapping to counteract inherent imbalanced issues. Attained results on both synthetic and real-world datasets probe the virtues of our approach regarding classification and computational cost assessments. In particular, the real-world results related to some UCI repository datasets, show that the ETWSVM-based discrimination outperforms most of the standard SVM-based methods for imbalance classification and obtains competitive results in comparison to the SMOTE-SVM. However, ETWSVM notably adopts the lowest computational cost for training. As future work, authors plan to couple the ETWSVM approach with resampling methods to improve the classification performance. Moreover, a scalable multi-class extension could be an exciting research line. Finally, a couple the ETWSVM approach with a stochastic gradient descent for large scale problems.

**Acknowledgments.** Under grants support by the project: “Desarrollo de un sistema de posorte clínico basado en el procesamiento estocástico para mejorar la resolución espacial de la resonancia magnética estructural y de difusión con aplicación al procedimiento de la ablación de tumores”, code: 111074455860, funded by COLCIENCIAS. Moreover, C. Jimenez is partially financed by the project E6-18-09: “Clasificador de máquinas de vectores de soporte para problemas desbalanceados con selección automática de parámetros”, funded by Vicerrectoria de Investigación, innovación y extension and by Maestría en Ingeniería Eléctrica, both from Universidad Tecnológica de Pereira.



## References

1. Haixiang, G., et al.: Learning from class-imbalanced data: review of methods and applications. *Expert Syst. Appl.* **73**, 220–239 (2017)
2. Loyola-González, O., et al.: Study of the impact of resampling methods for contrast pattern based classifiers in imbalanced databases. *Neurocomputing* **175**, 935–947 (2016)
3. Branco, P., Torgo, L., Ribeiro, R.P.: A survey of predictive modeling on imbalanced domains. *ACM Comput. Surv.* **49**(2), 31 (2016)
4. García, S., et al.: Dynamic ensemble selection for multi-class imbalanced datasets. *Inf. Sci.* **445**, 22–37 (2018)
5. Xanthopoulos, P., Razzaghi, T.: A weighted support vector machine method for control chart pattern recognition. *Comput. Ind. Eng.* **70**, 134–149 (2014)
6. Tang, L., Tian, Y., Yang, C.: Nonparallel support vector regression model and its SMO-type solver. *Neural Netw.* (2018)
7. Qi, Z., Tian, Y., Shi, Y.: Structural twin support vector machine for classification. *Knowl. Based Syst.* **43**, 74–81 (2013)
8. Liu, L., et al.: Between-class discriminant twin support vector machine for imbalanced data classification. In: *CAC 2017*, pp. 7117–7122. *IEEE* (2017)
9. Xu, Y.: Maximum margin of twin spheres support vector machine for imbalanced data classification. *IEEE Trans. Cybern.* **47**(6), 1540–1550 (2017)
10. Alvarez-Meza, A., Orozco-Gutierrez, A., Castellanos-Dominguez, G.: Kernel-based relevance analysis with enhanced interpretability for detection of brain activity patterns. *Front. Neurosci.* **11**, 550 (2017)
11. Anandarup, R., et al.: A study on combining dynamic selection and data preprocessing for imbalance learning. *Neurocomputing* **286**, 179–192 (2018)
12. Piri, S., Delen, D., Liu, T.: A synthetic informative minority over-sampling (SIMO) algorithm leveraging support vector machine to enhance learning from imbalanced datasets. *Decis. Support Syst.* **106**, 15–29 (2018)
13. Shao, Y.H., et al.: An efficient weighted lagrangian twin support vector machine for imbalanced data classification. *Pattern Recogn.* **47**(9), 3158–3167 (2014)