

# WIFROWAN: Wrapped Imbalanced Fuzzy-Rough Ordered Weighted Average Nearest Neighbor Classification

Mayte Guerra, Julio Madera

University of Camagüey,  
Department of Computer Science,  
Cuba

{mayte.guerra, julio.madera}@reduc.edu.cu

**Abstract.** In this paper we propose an ensemble method based on IFROWANN (Imbalanced Fuzzy-Rough Ordered Weighted Average Nearest Neighbor) algorithm to classify problems with imbalanced data. The ensemble generates many classifiers with different weight strategy and indiscernibility fuzzy relations. Classification is carried out selecting one of three strategies: I- to classify the new instance with the algorithm with best AUC in training. II- to average the memberships of the instance to the fuzzy-rough lower and upper approximation of each class given by the classifiers with best AUC. III- to average the memberships of the instance to the fuzzy-rough lower and upper approximation of each class of the all classifiers. Our method is validated by an extensive experimental study, showing statistically better results than 14 other state-of-the-art methods.

**Keywords.** Ensemble, imbalanced classification, fuzzy-rough sets.

## 1 Introduction

In recent years, class imbalance problems have emerged as one of the challenges in data mining community [28]. This kind of data appear in many real-world classification problems like fault diagnosis [29], anomaly detection [24], medical diagnosis [19], circuit breaker maintenance diagnosis [21], among others. In binary classification, this problem occurs when the number of instances of one class is much lower than the instances of the other class. The overrepresented class is called the majority or negative class, and the other class the minority or positive class. Traditional classifiers generally tend

to classify almost all instances as negative (i.e., the majority class) [22].

Many techniques for dealing with class imbalance have emerged: those that modify the data distribution by preprocessing techniques (data level solutions) [2, 6, 18, 23], those at the level of the learning algorithm which adapt a base classifier to deal with class imbalance (algorithm level solutions) [4, 16, 22], those that apply different costs to misclassification of positive and negative samples (cost-sensitive solutions) [8, 18, 25, 27], and ensemble based solutions that combine the previous solutions by means of an ensemble [10].

In this paper, we present an ensemble solution to classify imbalanced using the Imbalanced Fuzzy-Rough Ordered Weighted Average Nearest Neighbor (IFROWANN) algorithm [22]. This classifier combine fuzzy rough set theory and ordered weighted average aggregation to taking into account the imbalance of the classes. Each strategy of weight and fuzzy-rough indiscernibility relation produce different classification results. It is difficult to know the best configuration to obtain the best classification result in different datasets.

The ensemble generates the same classifier (IFROWANN), but with different weight strategy and different fuzzy relation. Final classification is given by one of these strategies. To evaluate the quality of our model, we have carried out an extensive experimental analysis on a collection of 66 imbalanced datasets with different imbalance ratios (IR) (between 1.82 and 129.44), originating from the SCI<sup>2</sup>S site ([sci2s.ugr.es](http://sci2s.ugr.es)). In the experiments, we have compared our algorithm with

the base IFROWANN proposal to show that it is similar to it without to test the weight and fuzzy relation strategies. Also, we have compared the ensemble with a set of 16 state-of-the-art methods designed for imbalanced classification, obtaining that it is better positioned than 14 and without significant differences with the rest. To assess the classification performance, we have used the Area Under the Curve (AUC) metric [22], and the significance of the results has been supported by the Friedman tests and post hoc Holm procedure.

The remainder of this paper is organized as follows. In Section 2, we introduce the imbalanced classification problem, including an overview of the state-of-the-art methods for solving it. In Section 3, we recall the standard IFROWANN algorithm. In Section 4, we introduce the WIFROWANN algorithm and present the proposed strategies to instance classification. In Section 5, we discuss the setup of the experimental study. In Section 6, we present and discuss the results. In Section 7, we draw some conclusions and future work about the study.

## 2 Classifications in Imbalanced Datasets

In this section, we first introduce the problem of imbalanced datasets in classification. Then, several techniques to address the class imbalance problem are presented. In binary classification, it is considered a set of data samples  $U$ , characterized by their values for the set  $A = \{a_1, \dots, a_m\}$  of attributes. Moreover,  $U = P \cup N$  where  $P$  represents the positive class, and  $N$  represents the negative class. We denote  $p = |P|$ ,  $n = |N|$ , and  $t = U \setminus p + n$ . The imbalance rate is then defined as  $n/p$ . The imbalanced classification problem can be tackled using four main types of solutions:

1) *Sampling (solutions at the data level)* [18]: This kind of solution consists of balancing the class distribution by means of a preprocessing strategy. Techniques at data level are undersampling, oversampling and hybrid methods. Some examples of this technique are *Synthetic Minority Oversampling Technique (SMOTE)* algorithm [6], *SMOTE-ENN* [2] and *SMOTE-RSB\** [23].

2) *Design of specific algorithms (solutions at the algorithmic level)* [4, 16]: Traditional classifier is adapted to deal directly with the imbalance between the classes. This is the case of *Imbalanced Fuzzy Rough Ordered Weighted Average Nearest Neighbor (IFROWANN)* [22].

3) *Cost-sensitive solutions* [8, 18]: These kind of methods incorporate solutions at data level, at algorithmic level, or at both levels together. They try to minimize higher cost errors where the cost of misclassifying a positive instance should be higher than the cost of misclassifying a negative one. The main examples are *Cost-sensitive C4.5 decision tree (CS-C4.5)* [25] and *Cost-sensitive support vector machine (CS-SVM)* [27].

4) *Ensemble solutions* [11]: Usually combine an ensemble learning algorithm and one of the techniques above, specifically, data level and cost-sensitive. For example the *EUSBOOST* algorithm [10], which uses evolutionary undersampling.

Next, we will discuss the evaluation of machine-learning algorithms in imbalanced domains.

## 3 Imbalanced Fuzzy-Rough Ordered Weighted Average Nearest Neighbor (IFROWANN)

In this section, we introduce the IFROWANN classification algorithm proposed in [22]. This algorithm is a variation of the Fuzzy-Rough Nearest Neighbor (FRNN) algorithm [17]. In order to predict the class of a new instance  $x$ , the IFROWANN algorithm calculate the memberships degrees of  $x$  to the fuzzy-rough lower and upper approximation of each class and assigns the instance to the class with higher degree.

More precisely, let  $U$  be the universe,  $I$  an impicator,  $T$  a t-norm defined by  $I(a, b) = \max(1 - a, b)$  and  $T(a, b) = \min(a, b)$ , for  $a, b$  in  $[0, 1]$ , and  $R$  a fuzzy relation that represents approximate indiscernibility between instances,  $W_P^l$  and  $W_N^l$  OWA weight vectors.

An impicator  $I$  is a  $[0, 1]^2 \rightarrow [0, 1]$  mapping that is decreasing in its first argument and increasing in its second argument, and that satisfies  $I(0, 0) = I(0, 1) = I(1, 1) = 1$ , and  $I(1, 0) = 0$ . The

membership degrees to the positive class  $\mu_P(x)$  and negative class  $\mu_N(x)$  are defined by:

$$\mu_P(x) = \frac{\underline{P}_{W_P^l}(x) + 1 - \underline{N}_{W_N^l}(x)}{2}, \quad (1)$$

$$\mu_N(x) = \frac{\underline{N}_{W_N^l}(x) + 1 - \underline{P}_{W_P^l}(x)}{2}, \quad (2)$$

where  $\underline{P}_{W_P^l}(x)$  is the lower approximation to the positive class  $P$  under the fuzzy relation  $R$  with the OWA weight vector  $W_P^l$ , and  $\underline{N}_{W_N^l}(x)$  is the lower approximation to the negative class  $N$  under  $R$  and the OWA weight vector  $W_N^l$  for negative class.  $x$  is classified to the positive class if  $\mu_P(x) \geq \mu_N(x)$ ; otherwise, it is classified to the negative class. The lower approximation for the negative and positive class are calculated by the equation 3 and 4:

$$\underline{P}_{W_P^l}(x) = \text{OWA}_{W_P^l}(\mathbf{I}(R(x, y), P(y))), \quad (3)$$

$$\underline{N}_{W_N^l}(x) = \text{OWA}_{W_N^l}(\mathbf{I}(R(x, y), N(y))), \quad (4)$$

where OWA is the operator to take into account the imbalance. Given a sequence  $A$  of  $t$  real values  $A = \langle a_1, \dots, a_t \rangle$ , and a weight vector  $W = \langle w_1, \dots, w_t \rangle$  such that  $w_i \in [0, 1]$  and  $\sum_{i=1}^t w_i = 1$  the OWA aggregation of  $A$  by  $W$  is given by:  $\text{OWA}_w(A) = \sum_{i=1}^t w_i b_i$  where  $b_i = a_j$  if  $a_j$  is the  $i$ th largest value in  $A$ .

IFROWANN algorithm has two fundamental factors: indiscernibility fuzzy function and weight vector. There is no weight and fuzzy relation strategies to obtain the best results for any IR. In order to obtain the best result of classification we must to try with several combinations of weight vectors and fuzzy relations. For this reason, this paper proposes an ensemble with the IFROWANN algorithm, which combine different weight strategies and fuzzy relations.

## 4 Extending the IFROWANN Algorithm

The Wrapper Imbalanced Fuzzy-Rough Ordered Weighted Average Nearest Neighbor (WIFROWANN) is an ensemble that uses the IFROWANN algorithm in base, with different configurations of OWA weight vector and fuzzy

relations. The description of the algorithm is divided in two main parts. First, we discuss the configurations to run the classifier, and second in Section 4.1, the different strategies to output the ensemble classification of the new instance.

To predict the class of a new instance  $x$ , the ensemble creates a set of classifiers  $C = \{C_1, \dots, C_L\}$ , each one with a weight strategy and an indiscernibility fuzzy relation strategy too. Given a vector with different weight strategies  $W = [W_1, \dots, W_n]$  and a vector with different configurations of indiscernibility fuzzy-rough relations  $R = [R_1, \dots, R_m]$ ; the ensemble build  $W \times R$  classifiers. Each classifier  $C_i$  computes the membership degree to the positive and negative class and the final result is given by the strategy of fusion or selection chose by us. Next section we explain these strategies. Weights strategy define which weight vector the classifier  $C_i$  will use to calculate the belong degree to the lower approximation to the positive and negative class in equations 3 and 4 respectively. The different weight vectors proposed in [22] are:

$$W_P^{l1} = \left\{ 0, \dots, 0, \frac{2}{n(n+1)}, \frac{4}{n(n+1)}, \dots, \frac{2(n-1)}{n(n+1)}, \frac{2}{n+1} \right\}, \quad (5)$$

$$W_P^{l2} = \left\{ 0, \dots, 0, \frac{1}{(2^n-1)}, \frac{2}{(2^n-1)}, \dots, \frac{2^{n-2}}{(2^n-1)}, \frac{2^{n-1}}{(2^n-1)} \right\}, \quad (6)$$

$$\begin{aligned} W_N^{l2} &= \left\{ 0, \dots, 0, \frac{1}{(2^p-1)}, \frac{2}{(2^p-1)}, \dots, \frac{2^{p-2}}{(2^p-1)}, \frac{2^{p-1}}{(2^p-1)} \right\} W_N^{l1} \\ &= \left\{ 0, \dots, 0, \frac{2}{p(p+1)}, \frac{4}{p(p+1)}, \dots, \frac{2(p-1)}{p(p+1)}, \frac{2}{p+1} \right\}, \end{aligned} \quad (7)$$

$$W_P^{l1,\gamma} = \left\{ 0, \dots, 0, \frac{2}{r(r+1)}, \frac{4}{r(r+1)}, \dots, \frac{2(r-1)}{r(r+1)}, \frac{2}{r+1} \right\}. \quad (8)$$

where  $p = |P|$  and  $n = |N|$ , are the number of instances of the Positive  $P$  and Negative  $N$  classes.

A variation of the  $W_P^{l1}$  vector is: Given  $0 \leq \gamma \leq 1$ :

**Table 1.** Description of the datasets used in the experimental evaluation

<b>Dataset</b>	<b>IR</b>	<b>Inst</b>	<b>Attr</b>	<b>Dataset</b>	<b>IR</b>	<b>Inst</b>	<b>Attr</b>
glass1	1,82	214	9	glass-0-4vs5	9,22	92	9
ecoli0vs1	1,86	220	9	ecoli-0-3-4-6vs5	9,25	205	7
wisconsin	1,86	683	7	ecoli-0-3-4-7vs5-6	9,28	257	7
pima	1,90	768	8	yeast-0-5-6-7-9vs4	9,35	528	8
iris0	2,00	150	4	ecoli-0-6-7vs5	10	220	6
glass0	2,06	214	9	vowel0	10,1	988	13
yeast1	2,46	1484	8	glass-0-1-6vs2	10,29	192	9
vehicle1	2,52	846	18	glass2	10,39	214	9
vehicle2	2,52	846	18	ecoli-0147vs2356	10,59	336	7
vehicle3	2,52	846	18	led7digit02456789vs1	10,97	443	7
haberman	2,68	306	3	glass-0-6vs5	11	108	9
glass-0123vs456	3,19	214	9	ecoli-0-1vs5	11	240	6
vehicle0	3,23	846	18	glass-0-1-4-6vs2	11,06	205	9
ecoli1	3,36	336	7	ecoli-0-1-4-7vs5-6	12,28	332	7
new-thyroid2	4,92	215	5	cleveland-0vs4	12,62	173	13
newthyroid1	5,14	215	5	ecoli-0-1-4-6vs5	13	280	6
ecoli2	5,46	336	7	ecoli4	13,84	336	7
segment0	6,01	2308	19	yeast-1vs7	13,87	459	7
glass6	6,38	214	9	shuttle-0-vs-4	13,87	1829	9
yeast3	8,11	1484	8	glass4	15,47	214	9
ecoli3	8,19	336	7	page-blocks-1-3vs4	15,85	472	10
page-blocks0	8,77	5472	10	abalone9-18	16,68	731	8
ecoli-0-3-4vs5	9	200	7	glass-0-1-6vs5	19,44	184	9
yeast-2vs4	9,08	515	7	shuttle-2-vs-4	20,5	129	9
ecoli-0-6-7vs3-5	9,09	222	7	yeast-1-4-5-8vs7	22,1	693	8
ecoli-0-2-3-4vs5	9,1	202	7	glass5	22,81	214	9
glass-0-1-5vs2	9,12	172	9	yeast-2vs8	23,1	482	8
yeast-0-3-5-9vs7-8	9,12	506	8	yeast4	28,41	1484	8
yeast-02579vs368	9,14	1004	8	yeast-1-2-8-9vs7	30,56	947	8
yeast-0256vs3789	9,14	1004	8	yeast5	32,78	1484	8
ecoli-0-4-6vs5	9,15	203	6	ecoli-0-1-3-7vs2-6	39,15	281	7
ecoli-0-1vs2-3-5	9,17	244	7	yeast6	39,15	1484	8
ecoli-0-2-6-7vs3-5	9,18	224	7	abalone19	129,44	4174	8

**Table 2.** Mean AUC for WIFROWANN vs best IFROWANN variants for different *IR* levels

Method	<i>IR</i> < 9	<i>IR</i> ≥ 9
W_All	0,9045	0,9038
W_Weights	0,9121	0,9115
W_W6W4W5	0,9202	0,9213
W_SF	0,9114	0,9126
W_F2	<b>0,9255</b>	0,9253
TL_W4	0,9141	0,9073
TL_W6	0,9086	0,9232
AV_W4	0,9204	0,9174
AV_W6	0,9110	<b>0,9343</b>
MIN_W4	0,9076	0,8833
MIN_W6	0,8902	0,8986
FRNN_MIN	0,8948	0,8716
FRNN_AV	0,9061	0,9089
FRNN_TL	0,9030	0,9020

**Table 3.** Average Friedman Ranking and adjusted p-values using Holm's procedure for *IR* < 9 datasets

Algorithm	Average Friedman Ranking	Adjusted p-value
W-All	9.2955	0.000001
MIN-W4	8.5455	0.000012
FRNN-AV	8.4318	0.000018
TL-W6	7.6136	0.000273
W-Weights	7.3409	0.000618
AV-W6	7	0.001614
W-SF	6.9773	0.001717
TL-W4	6.1364	0.013565
AV-W4	4.3864	0.279642
W-W6W4W5	4.25	0.330547
W-F2	3.0227	-

$$W_p^{l_{1,y}} = \left\langle 0, \dots, 0, \frac{2}{r(r+1)}, \frac{4}{r(r+1)}, \dots, \frac{2(r-1)}{r(r+1)}, \frac{2}{r+1} \right\rangle, \quad (9)$$

where  $r = [p + \gamma(n - p)]$ .

In particular, the first  $p$  positions in  $W_p^l$  can be put to 0, taking into account that this correspond to the highest values of  $I(R(x, y), P(y)) = \max(1 - R(x, y), 0) = 1 - R(x, y)$ , and thus to the  $p$  positive samples in the training data. The remaining  $n$  positions in the weight vector  $W_p^l$  correspond to the instances in  $N$ . In a completely analogous way, the weight vectors  $W_N^l$ , can we put 0 to the first  $n$  positions too.

We consider the following three alternatives for defining the fuzzy relation  $R$  Average t-norm, Łukasiewicz t-norm and Minimum t-norm  $R_{Av}, R_{TL}$  or  $R_{Min}$ :

$$R_{Min}(x, y) = \min(R_{a1}(x, y), \dots, R_{am}(x, y)), \quad (10)$$

$$\frac{R_{a1}(x, y), \dots, R_{am}(x, y)}{R_{TL}(x, y) = TL}, \quad (11)$$

$$R_{AV}(x, y) = \frac{R_{a1}(x, y) + \dots + R_{am}(x, y)}{m}, \quad (12)$$

where Łukasiewicz t-norm is defined by, for  $u_1, u_2, \dots, u_m$  in  $[0, 1]$ :

$$TL(u_1, u_2, \dots, u_m) = \max(u_1 + u_2 + \dots + u_m - m, 0), \quad (13)$$

and  $R_a$  is the similarity function between  $x$  and  $y$  instances with the attribute  $a$ .

For a quantitative attribute and nominal attribute, we use the equations respectively:

$$R_a(x, y) = 1 - \frac{|a(x) - a(y)|}{range(a)}, \quad (14)$$

$$R_a(a, y) = \begin{cases} 1 & \text{if } a(x) = a(y) \\ 0 & \text{en otro caso} \end{cases}, \quad (15)$$

We explain below in detail the strategies followed to build the ensemble.

#### 4.1 Strategies to Fusion Results

##### Strategy I: Select the Classifier with Best AUC in Training

This strategy selects the classifier with best AUC in training and the classification is made by the selected classifier. In other words, each classifier  $C_i$  classifies the training set  $U_{tr}$  and calculates the

**Table 4.** Average Friedman Ranking and adjusted p-values using Holm's procedure for  $IR \geq 9$  datasets

Algorithm	Average Friedman Ranking	Adjusted p-value
W-SF	7.2955	0.000039
W-Weights	7.2614	0.000046
TL-W4	6.7045	0.000555
AV-W4	6.375	0.002047
W-W6W4W5	5.4432	0.041491
TL-W6	4.8182	0.180954
W-F2	3.8864	0.769486
AV-W6	3.625	-

AUC for this training set. The ensemble selects the classifier  $D$  with the best AUC.

$$D = \arg\max_{C_k \in C} [AUC_{C_k}], \quad (16)$$

where  $W_i$  is the weight strategy followed by the classifier  $C_k$  and  $R_j$  is the fuzzy relation strategy take it into account by the classifier  $C_k$ . The new instance  $x$  takes the class given by the classifier  $D$ .

#### **Strategy IIa: Select the classifiers of maximum AUC and average the membership degrees to the positive and negative class**

In this strategy the ensemble selects a set of algorithms with maximum AUC value.

The selected set computes the membership degree to the positive and negative class of the instance  $x$ , then the calculated membership degrees are averaged. Finally  $x$  is classified as positive if the membership degree to the positive class is greater than the negative class  $\mu_P'(x) \geq \mu_N'(x)$ , and negative in the other case.

#### **Strategy IIb: Fusion the membership degrees to the positive and negative class**

First, each classifier  $C_i$  calculates the membership degree to the positive  $\mu_{P_i}(x)$  and negative  $\mu_{N_i}(x)$  class, and the outputs of the ensemble are the average degrees to the positive and negative class of all classifiers. Finally  $x$  is classified as positive if  $\mu_P'(x) \geq \mu_N'(x)$ , and it is classified as negative, otherwise.

## 5 Experimental Setups

In this section, we describe the experimental framework used to validate our proposal, including the benchmark datasets, the state-of-the-art methods, and the statistical tests used in order to carry out the performance comparison.

### 5.1 Datasets

We consider 66 datasets with different  $IR$  (between 1.82 and 129.44) to evaluate our proposal. The dataset was obtained from SCIS site. They are available online as part of the KEEL data set repository [1]. The characteristics of these datasets can be found in Table 1, showing the imbalanced rate  $IR$ , the number of instances (Inst), and the number of attributes (Attr) for each of them.

In our experimental study, we have also considered two subsets of the collection based on their  $IR$ :

1.  $IR < 9$  (low imbalance): This group contains 22 datasets, all with  $IR$  lower than 9.
2.  $IR \geq 9$  (high imbalance): This group contains 44 datasets, all with  $IR$  at least 9.

Furthermore, each dataset is partitioned in order to perform a fivefold cross validation.

### 5.2 Algorithms Analyzed in the Experimental Study

For the experimental study we consider the principal state-of-the-art methods: The IFROWANN algorithm with its competitive variants: AV-W6, AV-W4, TL-W6 and FRNN algorithm [22], preprocessing techniques, cost-sensitive and ensemble methods combined with a base classifier. We chose tree-based method C4.5 [20], support vector machines SVM [27], and lazy learner 1NN ( $k=1$ ) [5]. Preprocessing techniques are: SMOTE, SMOTE+RSB\*, SMOTE+ENN, Borderline-SMOTE, SafeLevel-SMOTE and DBSMOTE. Cost sensitive algorithms are: CS, MetaCost, CostSensitiveClassifier and CSWeighted. Ensemble methods are: AdaB-M1, AdaC2, RUSB, SBAG, Easy and EUSBOOT.

**Table 5.** Mean AUC for state-of-the-art methods

		<b>Methods</b>	<b>&lt;9</b>	<b>&gt;=9</b>
C4.5		SMOTE+ENN	0,8640	0,8164
		SMOTE+ RSB*	0,87	0,8232
		Borderline-SMOTE	0,8564	0,7998
		SafeLevel-SMOTE	0,8643	0,8106
		ADASYN	0,8604	0,8035
		SPIDER2	0,8499	0,7778
		DBSMOTE	0,8357	0,7638
Preprocessing techniques	SMO (SVM)	SMOTE	0,8574	0,8418
		SMOTE+ENN	0,8560	0,8412
		SMOTE+ RSB*	0,91	0,8815
		Borderline-SMOTE	0,8556	0,8317
		SafeLevel-SMOTE	0,8565	0,8358
		ADASYN	0,8545	0,8212
		SPIDER2	0,8269	0,6923
		DBSMOTE	0,8252	0,7192
		SMOTE	0,8478	0,8272
		SMOTE+ENN	0,8645	0,8342
1NN (KNN con k=1)		SMOTE+ RSB*	<b>0,92</b>	<b>0,9046</b>
		Borderline-SMOTE	0,8518	0,8007
		SafeLevel-SMOTE	0,8365	0,7861
		ADASYN	0,8526	0,8270
		CS	0,8578	0,8137
Cost sensitive algorithms	C4.5	MetaCost	<b>0,8617</b>	0,8246
		CostSensitiveClassifier	0,8487	0,7931
		CSWeighted	0,8597	0,8397
Cost sensitive algorithms	SMO	MetaCost	0,7289	0,6559
		CostSensitiveClassifier	0,8565	0,8304
		CSWeighted	0,8559	<b>0,8416</b>
Ensemble methods	1NN	MetaCost	0,8455	0,8147
		CostSensitiveClassifier	0,8367	0,7943
		AdaB-M1	0,8463	0,7877
		AdaC2	0,8649	0,7958
		RUSB	0,8747	0,8405
Ensemble methods	SMO	SBAG	0,8771	0,8431
		Easy	0,8711	0,8243
		AdaB-M1	0,8059	0,7392
		AdaC2	0,6487	0,6163
		RUSB	0,8270	0,7141
Ensemble methods	1NN	SBAG	0,8556	0,8406
		Easy	0,8501	0,8304
		AdaB-M1	0,8375	0,7948
		AdaC2	0,8370	0,7935
		RUSB	0,8562	0,8416
		SBAG	0,8599	0,8427
		Easy	0,8589	0,8365
		EUSBOOT	<b>0,93</b>	<b>0,9071</b>

**Table 6.** Preprocessing techniques vs WIFROWANN. Average Friedman Rankings and adjusted p-values using Holm's posthoc procedure for IR<9 datasets

Algorithm	Average Friedman Ranking	Adjusted p-value
RSB-C4.5	15.3864	0.000001
SMO(SMOTE)	15.1818	0.000001
W-All	7.1591	0.079631
RSB-kNN	6.5455	0.130075
RSB-SVM	6.2045	0.167274
W-Weights	5.5682	0.257154
W-SF	5.4091	0.284091
W-W6W4W5	3.7273	0.677354
W-F2	2.6591	-

**Table 7.** Preprocessing technique vs WIFROWANN. Average Friedman Rankings and adjusted p-values using Holm's posthoc procedure for IR≥9 datasets

Algorithm	Average Friedman Ranking	Adjusted p-value
RSB-SVM	9.5682	0.000121
RSB-kNN	6.3636	0.037686
W-All	5.875	0.070439
W-Weights	4.9659	0.190773
W-SF	4.4318	0.310542
W-W6W4W5	3.2727	0.707224
W-F2	2.5909	-

**Table 8.** Cost-sensitive algorithms vs WIFROWANN. Average Friedman Rankings and adjusted p-values using Holm's posthoc procedure for IR<9 datasets

Algorithm	Average Ranking	Friedman	Adjusted p-value
SMOCSWeighted	7.7045		0.000001
W-All	4.7273		0.016393
W-SF	3.8409		0.099929
W-Weights	3.7727		0.112483
W-W6W4W5	2.5682		0.574592
W-F2	1.9091		-

In order to compare the different algorithms appropriately, we will conduct a statistical analysis using nonparametric tests as suggested in [12, 14]. We first use Friedman's aligned-ranks test [9] and then Holm's post hoc test [15]. The post hoc procedure allows us to decide whether a hypothesis of comparison can be rejected at a specified level of significance  $\alpha$ . In this paper, we set  $\alpha = 0.05$ . The KEEL tool was used to perform the tests.

### 5.3 Parameters

The parameters for the ensemble and IFROWANN algorithm are the fuzzy relations Average t-norm, Łukasiewicz t-norm and Minimum t-norm (see equations 10, 11 and 12); similarity functions for quantitative and nominal attributes (see equations 14 and 15) and 6 variants of weight vectors combinations recommended in [22], which are described below:

$$1. W_1 = \langle W_P^{l_1}, W_N^{l_1} \rangle,$$

$$2. W_2 = \langle W_P^{l_1}, W_N^{l_2} \rangle,$$

$$3. W_3 = \langle W_P^{l_2}, W_N^{l_1} \rangle,$$

$$4. W_4 = \langle W_P^{l_2}, W_N^{l_2} \rangle,$$

$$5. W_5 = \langle W_P^{l_1, \gamma}, W_N^{l_1} \rangle \text{ with } \gamma = 0, 1,$$

$$6. W_6 = \langle W_P^{l_1, \gamma}, W_N^{l_2} \rangle \text{ with } \gamma = 0, 1.$$

In the Strategy I we prove three variants. (1) **W-All**: ensemble uses [*None*,  $W_1$ ,  $W_2$ ,  $W_3$ ,  $W_4$ ,  $W_5$ ,  $W_6$ ] weight strategy and [AV, TL, MIN] fuzzy relations; (2) **W-weights**: all classifiers have weight vectors, and (3) **W-W6W4W5**: builds classifiers with [ $W_6$ ,  $W_4$ ,  $W_5$ ] weight strategies and [AV, TL] fuzzy relations.

In the Strategy IIa we form one variant **W-SF**: ensemble uses [*None*,  $W_1$ ,  $W_2$ ,  $W_3$ ,  $W_4$ ,  $W_5$ ,  $W_6$ ] weights strategies and the three fuzzy relations and in the Strategy IIb the variant **W-F2**, ensemble uses [ $W_6$ ,  $W_4$ ,  $W_5$ ] weight strategies and [AV, TL] fuzzy relations.

**Table 9.** Cost-sensitive algorithms vs WIFROWANN. Average Friedman Rankings for  $IR \geq 9$  datasets

Algorithm	Average Friedman Ranking
SMO(MetaCost)	12.2273
1NN(CostSensitiveClassifier)	10.4545
1NN(MetaCost)	9.2841
C4.5CS	9.1477
SMO(CostSensitiveClassifier)	8.6364
C4.5(MetaCost)	8.4886
SMOCSWeighted	8.1591
1NNCSWeighted	7.9091
W-All	4.4545
W-Weights	3.8295
W-SF	3.6364
W-W6W4W5	2.6818
W-F2	2.0909

## 6 Experimental Results

In this section, we present the results of our experimental analysis.

In Section 6.1, we compare our proposal with IFROWANN baseline methods and its best configurations. Next, in Section 6.2, we compare the algorithms with the state-of-the-art methods for imbalanced classification.

### 6.1 Comparative Analysis of WIFROWANN with IFROWANN

Table 2 shows the mean AUC obtained for each variant of the ensemble and the best IFROWANN configurations with each block of datasets. It can be noticed that for the high imbalance datasets ( $IR \geq 9$ ), AV-W6 obtains the highest average AUC. However, for low imbalance datasets ( $IR < 9$ ), the ensemble with W-F2 reaches the highest value.

Next, we carry out a statistical analysis of our results for each block of datasets.

1. **Statistical Analysis for Low Imbalance Ratio Datasets:** Table 3 shows the Friedman test and Holm's procedure for  $IR < 9$  datasets.

For low imbalance the best ranking is obtained by W-F2 and Holm's procedure rejects those hypotheses that have  $p - value \leq 0.025$ . Holm posthoc test allows to conclude that W-F2 is no significantly better than AV-W4; but is significantly better than the rest of the variants of both algorithms. It remarkable to notice that W-F2 and W-W6W4W5 variants of the WIFROWANN are best positioned in the ranking than IFROWANN, and W-F2 has significant difference with AV-W6.

2. **Statistical Analysis for High Imbalance Ratio Datasets:** In this case, Holm's procedure rejects those hypotheses that have  $p - value \leq 0.016667$ . As we can observe (see Table 4), the best ranking is obtained by IFROWANN with AV-W6. The adjusted p-values indicates that the method AV-W6 significantly no outperforms the TL-W6 IFROWANN configuration and the variants of WIFROWANN methods W-F2 and W-W6W4W5.

### 6.2 Comparative Analysis with the State-of-the-art Methods

This section compares the ensemble variants with the state-of-the-art methods. The mean AUC results for the state-of-the-art methods are shown in Table 5. For every technique and every IR levels, the highest AUC value is marked in bold.

The mean AUC results for the ensemble variants are shown in Table 2 (the first five rows). From these results, we can observe that the best AUC values in all blocks are obtained by SMOTE + RSB\*, EUSBOOT and WIFROWANN with all its variants, and W-F2 obtains the highest AUC values, except for low  $IR$  datasets for which EUSBOOT gets the highest score. We carry out a statistical analysis of our results for each techniques and each block of datasets. In these cases, per block we show only the methods which obtain a  $p - value > 0$ . The rest of the state-of-the-art methods has a  $p - value = 0$ .

1. **Statistical analysis for preprocessing techniques:** Tables 6 and 7 show the Friedman test and Holm's procedure for  $IR < 9$  and  $IR \geq 9$ . In both cases, the best ranking is obtained by W-F2. For  $IR < 9$  Holm's

**Table 10.** Ensembles vs WIFROWANN. Average Friedman Rankings and adjusted p-values using Holm's posthoc procedure for IR<9 datasets

Algorithm	Average Friedman Ranking	Adjusted p-value
C4.5(RUSB)	11.5	0.000001
SMO(SBAG)	11.2955	0.000002
C4.5(SBAG)	10.5227	0.000012
W-All	5.8864	0.058076
W-Weights	4.9545	0.162399
W-SF	4.5455	0.238646
EUSBOOST	4.2727	0.301791
W-W6W4W5	3.0455	0.706474
W-F2	2.3409	-

**Table 11.** Average Friedman Rankings and adjusted p-values using Holm's posthoc procedure for IR≥9 datasets

Algorithm	Average Friedman Ranking	Adjusted p-value
W-All	5.4302	0.037028
W-Weights	4.6628	0.130545
EUSBOOST	4.5465	0.154139
W-SF	4.2326	0.23387
W-W6W4W5	3.0581	0.754418
W-F2	2.6395	-

procedure rejects those hypotheses that have  $p - value \leq 0.008333$ . W-F2 has significant difference with all preprocessing techniques, except for RSB-kNN and RSB-SVM. For  $IR \geq 9$  Holm's procedure rejects those hypotheses that have  $p - value \leq 0.01$ . W-F2 has significant difference with all preprocessing techniques too, except for RSB-kNN.

2. **Statistical analysis for Cost-Sensitive algorithms:** Tables 8 and 9 show the results of Friedman test and the Holm's procedure for  $IR < 9$ . For low imbalance, the associated p-value is 0.0125, which leads us to conclude that there are statistically significant differences among the cost-sensitive

compared methods compared with W-F2. For  $IR \geq 9$  Tables show only Friedman test because of the p-values of cost-sensitive algorithms was 0.

3. **Statistical analysis for ensemble techniques:** Table 10 and 11 show Friedman test and Holm's procedure for low and high imbalance datasets. The lowest Friedman rank is obtained by W-F2; however, there is no significant differences with EUSBOOST. W-F2 statistically outperforms the rest of the ensemble methods. We can conclude the same analysis for high imbalance.

## 6 Conclusions

In this paper, we have presented the WIFROWANN method, a new ensemble level solution for two-class imbalanced classification problems that is based on the IFROWANN algorithm. In particular, the W-All, W-Weights, W-SF, W-W6W4W5 and W-F2 variants of WIFROWANN method, considering six weighting strategies and no weighting strategy, combined with three different indiscernibility fuzzy relations. Our experimental results and statistical analysis have shown that:

W-F2 obtains better AUC mean respect to IFROWANN and the best position in the Friedman ranking for low imbalance datasets. Holm's procedure shows that this variant present significant difference with AV-W6 for low imbalance and no significant difference with the same IFROWANN configuration for high imbalance.

WIFROWANN outperforms 14 state-of-the-art representative algorithms that cover preprocessing level, cost-sensitive, and ensemble solutions specifically designed for imbalanced learning and similar behavior with IFROWANN, SMOTE+RSB and EUSBOOST methods.

For future work, we will consider extend WIFROWANN method for multiclass and multi-labels classification problems.

## Acknowledgements

The authors would like to thank to the Applied Informatics Mastery Program at the University of Camagüey.

## References

1. Alcalá-Fernández, J., Fernandez, A., Luengo, J., Derrac, J., García, S., Sánchez, L., & Herrera, F. (2010). KEEL datamining software tool: Dataset repository, integration of algorithms and experimental analysis framework. *Journal of Multiple-Valued Logic and Soft Computing*, Vol. 17, pp. 255–287.
2. Batista, G.E.A.P.A., Prati, R.C., & Monard, M.C. (2004). A study of the behaviour of several methods for balancing machine learning training data. *SIGKDD Explorations*, Vol. 6, No. 1, pp. 20–29. DOI:10.1145/1007730.1007735.
3. Bermejo, P., Gámez, J.A., Puerta, J.M. (2011). Improving the performance of naive bayes multinomial in e-mail foldering by introducing distribution-based balance of datasets. *Expert System with Applications*, Vol. 38, No. 3, pp. 2072–2080. DOI:10.1016/j.eswa.2010.07.146.
4. Cieslak, D.A., Hoens, T.R., Chawla, N.V., & Kegelmeyer, W.P. (2012). Hellinger distance decision trees are robust and skew-insensitive. *Data Mining Knowl. Discovery*, Vol. 24, pp. 136–158. DOI:10.1007/s10618-011-0222-1.
5. Cover, T. & Hart, P. (1967). Nearest neighbor pattern classification. *IEEE Transactions on Information Theory*, Vol. 13, No. 1, pp. 21–27. DOI:10.1109/TIT.1967.1053964.
6. Chawla, N.V., Bowyer, K.W., Hall, L.O., & Kegelmeyer, W.P. (2002). SMOTE: Synthetic minority over-sampling technique. *Journal of Artificial Intelligent Research*, Vol. 16, pp. 321–357. DOI:10.1613/jair.953.
7. Chawla, N.V., Japkowicz, N., & Kolcz, A. (2004). Special issue on learning from imbalanced data sets. Vol. 6, No. 1, pp. 1–6. DOI:10.1145/1007730.1007733.
8. Domingos, P. (1999). MetaCost: A general method for making classifiers cost sensitive. *Proceedings of Fifth International Conference on Knowledge Discovery and Data Mining (KDD99)*, pp. 155. DOI:10.1145/312129.312220.
9. Friedman, M. (1937). The use of ranks to avoid the assumption of normality implicit in the analysis of variance. *Journal of the American Statistical Association*, Vol. 32, No. 200, pp. 675–701.
10. Galar, M., Fernández, A., Barrenechea, E., & Herrera, F. (2013). EUSBoost: Enhancing ensembles for highly imbalanced data-sets by evolutionary undersampling. *Pattern Recognition*, Vol. 46, No. 12, pp. 3460–3471. DOI:10.1016/j.patcog.2013.05.006.
11. Galar, M., Fernández, A., Barrenechea, E., Bustince, H., & Herrera, F. (2012). A review on ensembles for the class imbalance problem: Bagging, boosting, and hybrid based approaches. *IEEE Transactions on Systems, Man, and Cybernetics Part C: Applications and Reviews*, Vol. 42, No. 4, pp. 463–484. DOI:10.1109/TSMCC.2011.2161285.
12. García, S., Fernández, A., Luengo, J., & Herrera, F. (2010). Advanced non-parametric tests for multiple comparisons in the design of experiments in computational intelligence and data mining: *Experimental Analysis of Power Information Sciences*, Vol. 180, No. 10, pp. 2044–206. DOI:10.1016/j.ins.2009.12.010.
13. García, S. & Herrera, F. (2009). Evolutionary undersampling for classification with imbalanced datasets: proposals and taxonomy. *Evolutionary Computation*, Vol. 17, No. 3, pp. 275–306. DOI:10.1162/evco.2009.17.3.275
14. García, S. & Herrera, F. (2008). An extension on statistical comparisons of classifiers over multiple datasets for all pairwise comparisons. *Journal of Machine Learning Research*, Vol. 9, pp. 2677–2694.
15. Holm, S. (1979). A simple sequentially rejective multiple test procedure. *Scandinavian Journal of Statistics*, Vol. 6, No. 2, pp. 65–70. DOI:10.2307/4615733.
16. Huang, Y.M., Hung, C.M., & Jiau, H.C. (2006). Evaluation of neural networks and data mining methods on a credit assessment task for class imbalance problem. *Real World Applications*, Vol. 7, No. 4, pp. 720–747. DOI:10.1016/j.nonrwa.2005.04.006.
17. Jensen, R. & Cornelis, C. (2011). Fuzzy rough nearest neighbour classification and prediction. *Theoretical Computer Science*, Vol. 412, No. 42, pp. 5871–5884. DOI:10.1016/j.tcs.2011.05.040.
18. López, V., Fernández, A., Moreno-Torres, J.G., & Herrera, F. (2012). Analysis of preprocessing vs. cost-sensitive learning for imbalanced classification open problems on intrinsic data characteristics. *Expert Systems with Applications*, Vol. 39, No. 7, pp. 6585–6608. DOI:10.1016/j.eswa.2011.12.043.

19. **Mazurowska, M.A., Habasa, P.A., Zuradaa, J.M., Lob, J.Y., Bakerb, J.A., & Tourassib, G.D. (2008).** Training neural network classifiers for medical decision making: The effects of imbalanced datasets on classification performance. *Neural Networks*, Vol. 21, No. 2-3, pp. 427–436. DOI: 10.1016/j.neunet.2007.12.031.
20. **Quinlan, J.R. (1983).** *C4.5 Programs for Machine Learning*. San Mateo, CA Morgan Kaufmann.
21. **Ramentol, E., Gondres, I., Lajes, S., Bello, R., Caballero, Y., Cornelis, C., & Herrera, F. (2016).** Fuzzy-rough imbalanced learning for the diagnosis of High Voltage Circuit Breaker maintenance: The SMOTE-FRST-2T algorithm. *Engineering Applications of Artificial Intelligence*, Vol. 48, pp. 134–139. DOI: 10.1016/j.engappai.2015.10.009.
22. **Ramentol, E., Vluymans, S., Verbiest, N., Caballero, Y., Bello, R., Cornelis, C., & Herrera, F. (2015).** IFROWANN: Imbalanced Fuzzy-Rough Ordered Weighted Average Nearest Neighbor Classification. *IEEE Transactions on Fuzzy Systems*, Vol. 23, No. 5, pp. 1622–1637. DOI: 10.1109/TFUZZ.2014.2371472.
23. **Ramentol, E., Caballero, Y., Bello, R., & Herrera, F. (2012).** SMOTE-RSB: a hybrid preprocessing approach based on oversampling and undersampling for high imbalanced data-sets using SMOTE and rough sets theory. *Knowledge and Information Systems an International Journal*, Vol. 33, pp. 245–265. DOI:10.1007/s10115-011-0465-6.
24. **Tavallaee, M., Stakhanova, N., & Ghorbani, A. (2010).** Toward credible evaluation of anomaly-based intrusion detection methods. *IEEE Trans. Systems, Man, and Cybernetics, Part C, (Applications and Reviews)*, Vol. 40, No. 5, pp. 516–524. DOI:10.1109/TSMCC.2010.2048428.
25. **Ting, K.M. (2002).** An instance-weighting method to induce cost-sensitive trees. *IEEE Transactions on Knowledge and Data Engineering*, Vol. 14, No. 3, pp. 659–665. DOI: 10.1109/TKDE.2002.1000348.
26. **Vapnik, V. (1995).** *The nature of statistical learning*. Springer.
27. **Veropoulos, K., Campbell, C., & Cristianini, N. (1999).** Controlling the sensitivity of support vector machines. *Proceedings of the International Joint Conference on AI*.
28. **Yang, Q. & Wu, X. (2006).** 10 challenging problems in data mining research. *International Journal of Information Technology and Decision Making*, Vol. 5, No. 4, pp. 597–604. DOI: 10.1142/S0219622006002258.
29. **Zhu, Z.B. & Song, Z.H. (2010).** Fault diagnosis based on imbalance modified kernel fisher discriminant analysis. *Chemical Engineering Research Design*, Vol. 88, No. 8, pp. 936–951. DOI: 10.1016/j.cherd.2010.01.005.

Article received on 04/11/2018; accepted on 19/11/2019.  
Corresponding author is Mayte Guerra.