Improving Pairwise Learning Classification in Fuzzy Rule Based Classification Systems Using Dynamic Classifier Selection

Alberto Fernández[1] **Mikel Galar**[2] **José Antonio Sanz**[2] **Humberto Bustince**[2] **Francisco Herrera**[3]

[1]Dept. of Computer Science, University of Jaén, Spain
[2]Dept. of Automática y Computación, Universidad Pública de Navarra, Spain
[3]Dept. of Computer Science and A.I., University of Granada, Spain

Abstract

Classification based on the One-vs-One decomposition strategy has shown a high quality for addressing those problems with multiple classes, even if the learning model enables the discrimination among several concepts. The main phase of the pairwise learning is the decision process, where the outputs of the binary classifiers are combined to give a single output. Recently, it has been shown that standard decision techniques do not take into account the influence of the non-competent classifiers, i.e. those that were not trained using the class of the query example, and this can deteriorate the performance of the model. In accordance with the former, a "Dynamic Classifier Selection" for the One-vs-One approach was proposed to alleviate this issue. It basically consists of finding those classifiers whose outputs are closest to the input example, and thus remove those ones which are not related with it. In this work, we want to analyse the goodness for the former approach using a fuzzy-type baseline classifier. Experimental results show that there is in fact a significant leap in the global performance when this model is applied, both versus the standard fuzzy rule based classification system, and the One-vs-One learning approach.

Keywords: Fuzzy Rule Based Classification Systems, Multi-classification, One-vs-One, Pairwise Learning, Dynamic Classifier Selection

1. Introduction

Classification is one of the most studied problems in machine learning and data mining [1]. It is a task that, from a supervised learning point of view, consists of inducing a mapping which allows to determine the class of a new pattern from a set of attributes. Most commonly used classifiers in Data Mining are intrinsically designed to deal with binary-class problems. However, multi-class problems are usually more difficult, since the complexity of finding the decision boundaries increases.

In this context, decomposition strategies [2] can be used to transform the original multi-class problem into binary subsets, which are easier to discriminate. The classifiers use to face the binary problems are referred to as base learners or base classifiers of the ensemble [3]. Even when these base classifiers are able to cope with multi-class problems, it has been shown that the use of binarization techniques allows the enhancement of the performance from the standard case [4, 5].

This contribution makes use of the extension of linguistic Fuzzy Rule Based Classification Systems (FRBCSs) [6] for a multi-classifier model by means of the One-vs-One (OVO) strategy [7]. This approach divides the original problem in as many pairs of classes as possible, ignoring the examples that do not belong to the related classes. Then, a single classifier is learnt for each binary-problem, and the outputs of these classifiers are finally combined in order to obtain the final class label for a given instance [4].

In order to aggregate the output for all binary classifiers, the simplest and most widely used method in pairwise learning is applying a "Weighted Voting" (WV) [8] so that the final class is assigned by taking the maximum vote among the summation of the scores for the binary classifiers associated to the same class.

The previous procedure has an inherent problem: all base classifiers will be fired for a given instance, even when they are not related with its output class. Therefore, these classifiers will submit an erroneous score that can be regarded as noise in the aggregation phase. This case is better known as the "non-competent classifiers problem" [8], which can mislead the correct labeling of the query example.

A simple yet effective way to overcome this problem, is to use a novel aggregation strategy based on Dynamic Classifier Selection (DCS) [9, 10], which could reduce the number of non-competent classifiers in the classification phase. This procedure analyzes the neighbourhood of the example prior to the decision step, and removes the output for those classifiers whose related class are "far enough" in the input space area. This new scheme is known as dynamic OVO [11], and it has shown to successfully improve the behaviour of the standard OVO approach for standard baseline classifiers.

The success of FRBCSs, among other Soft Computing techniques, is related to their smoothness when defining the borderline areas in complex problems [12], as well as their good interpretability due to the usage of linguistic variables, which are easier to understand for the experts or end-users [13]. In this contribution we aim at investigating whether the use of the dynamic OVO can enhance the performance with respect to the original FRBCS and the OVO approach, using the Chi et al.'s method [14] as baseline classifier. Specifically, we will show that this combination between both techniques may result on a more positive synergy, leading to a higher gap in the results when contrasted versus those shown by any other type of classifiers from the specialized literature (as shown in [11]).

The experiments carried out include a set of nineteen real-world problems from the KEEL data-set repository [15]. In addition to the usage of the standard accuracy rate to evaluate the performance of the classifiers, we include the kappa measure [16] accounting for the balance among the prediction of all classes. Finally, the comparisons among the results obtained are contrasted using the proper statistical tests [17, 18].

This contribution is arranged as follows. In Section 2, we provide a brief introduction to FRBCSs and the OVO learning scheme. Next, Section 3 describes the approach that overcomes the non-competent problem in an OVO scheme, i.e. the dynamic OVO. In Section 4, the experimental analysis is carried out. Finally, Section 5 concludes the paper.

2. Preliminaries: Fuzzy Rule Based Classification Systems and Pairwise Learning

This section introduces the main features of FRBCS (Subsection 2.1). Then, we recall the basis of OVO strategy and its simplest aggregation (Subsection 2.2), which will be later used to explain the DCS scheme.

2.1. A short overview on FRBCS

Any classification problem consists of m training patterns $x_p = (x_{p1}, \ldots, x_{pn}, C_p)$, $p = 1, 2, \ldots, m$ from M classes where x_{pi} is the ith attribute value ($i = 1, 2, \ldots, n$) of the p-th training pattern, and C_p the output class.

In this work we use fuzzy rules of the following form for our FRBCSs:

$$\begin{aligned}&\text{Rule } R_j : \text{ If } x_1 \text{ is } A_{j1} \text{ and } \ldots \text{ and } x_n \text{ is } A_{jn}\\&\text{then Class} = C_j \text{ with } RW_j\end{aligned} \quad (1)$$

where R_j is the label of the jth rule, $x = (x_1, \ldots, x_n)$ is an n-dimensional pattern vector, A_{ji} is an antecedent fuzzy set, C_j is a class label, and RW_j is the rule weight [19]. We use triangular membership functions as antecedent fuzzy sets.

When a new pattern x_p is selected for classification, then the steps of the fuzzy reasoning method are as follows:

1. **Matching degree**, that is, the strength of activation of the if-part for all rules in the Rule Base with the pattern x_p. In order to carry out this computation, a conjunction operator γ shall be applied. This operator is used to combine the membership degrees for every variable of the example, which were obtained by means of the μ function. Traditionally, a T-norm is selected for this purpose, although any aggregation operator can be employed [20]:

$$\begin{aligned}&\mu_{A_j}(x_p) = \gamma(\mu_{A_{j1}}(x_{p1}), \ldots, \mu_{A_{jn}}(x_{pn})),\\&\quad j = 1, \ldots, L\end{aligned} \quad (2)$$

2. **Association degree**. To compute the association degree of the pattern x_p with the M classes according to each rule in the Rule Base. To this end, a combination operator h is applied in order to combine the matching degree with the rule weight (RW). In our case, this association degree only refers to the consequent class of the rule (i.e. $k = Class(R_j)$).

$$\begin{aligned}&b_j^k = h(\mu_{A_j}(x_p), RW_j^k),\\&\quad k = 1, \ldots, M; \qquad j = 1, \ldots, L\end{aligned} \quad (3)$$

3. **Pattern classification soundness degree for all classes**. We use an aggregation function f, which combines the positive degrees of association calculated in the previous step.

$$\begin{aligned}&Y_k = f(b_j^k, j = 1, \ldots, L \text{ and } b_j^k > 0),\\&\quad k = 1, \ldots, M\end{aligned} \quad s \quad (4)$$

4. **Classification**. We apply a decision function F over the soundness degree of the system for the pattern classification for all classes. This function will determine the class label l corresponding to the maximum value.

$$F(Y_1, \ldots, Y_M) = \arg\max(Y_k), \quad [k = 1, \ldots, M] \quad (5)$$

Where L denotes the number of rules in the Rule Base and M the number of classes of the problem.

Fuzzy learning methods are the basis to build a FRBCS. The algorithm used in this work is the method proposed in [14], that we have called the Chi et al.'s rule generation.

To generate the fuzzy rule base this FRBCSs design method determines the relationship between the variables of the problem and establishes an association between the space of the features and the space of the classes by means of the following steps:

1. *Establishment of the linguistic partitions.* Once the domain of variation of each feature A_i is determined, the fuzzy partitions are computed.

This is carried out by means of triangular homogenous partitions within the range of the variable.

2. *Generation of a fuzzy rule for each example* $x_p = (x_{p1}, \ldots, x_{pn}, C_p)$. To do this is necessary:
   - 2.1 To compute the matching degree $\mu(x_p)$ of the example to the different fuzzy regions using a conjunction operator (usually modeled with a minimum or product T-norm).
   - 2.2 To assign the example x_p to the fuzzy region with the greatest membership degree.
   - 2.3 To generate a rule for the example, whose antecedent is determined by the selected fuzzy region and whose consequent is the label of class of the example.
   - 2.4 To compute the rule weight.

2.2. Decomposition strategies: One-vs-One

The use of decomposition strategies in multi-classification has shown to be of great interest in the research community [2, 4], including FRBCS [21, 22, 20]. The main idea for this learning scheme is to address a multiple classes problem by means of binary classifiers, following a divide and conquer paradigm. Finally, when a new example arrives the system, output is obtained by combining the confidence degrees of each single classifier. Therefore, the way the decision process is carried out has a strong influence in the classification performance [4].

OVO [7] and One-vs-All [23] decompositions are known to be the most common approaches. The former consists of learning a binary classifier to discern between each pair of classes, whereas the latter constructs a binary classifier to separate each single class from all other classes. Between both approaches, OVO is the most extended scheme, established by default in several widely used software tools [24, 25, 26].

OVO divides a m-class problem into $m(m-1)/2$ independent binary subproblems by contrasting all classes among them, each of which is learnt by a single classifier. In the classification stage, the input instance is presented to all classifiers, so that each one of them outputs a confidence degree r_{ij} and $r_{ji} \in [0, 1]$ in favor of their couple of classes C_i and C_j (usually $r_{ji} = 1 - r_{ij}$). Then, these confidence degrees are set within a score-matrix:

$$R = \begin{pmatrix} - & r_{12} & \cdots & r_{1m} \\ r_{21} & - & \cdots & r_{2m} \\ \vdots & & & \vdots \\ r_{m1} & r_{m2} & \cdots & - \end{pmatrix} \tag{6}$$

Different aggregations have been developed in order to compute the final class [4]. The simplest aggregation, yet powerful is the voting strategy, where each classifier votes for its predicted class, and the class obtaining the largest number of votes is predicted. However, in this work we aim to benefit from the features of fuzzy classifiers and to make use of the framework of fuzzy preference relations for classification [27]. In this scheme, the classification problem is translated into a decision making problem for determining the output among all predictions for the binary classifiers. Specifically, in this paper we consider the use of a maximal *Non-Dominance Criterion (ND)* [21] for the final decision process. This method predicts the class which is less dominated by all the remaining classes:

$$Class = \arg \max_{i=1,\ldots,m} \left\{ 1 - \sup_{j \in C} r'_{ji} \right\} \tag{7}$$

where r'_{ji} corresponds to the normalized and strict score-matrix.

3. Dynamic Classifier Selection for One-vs-One Strategy

The OVO approach has shown to be a very powerful strategy to improve the accuracy in multiple class problems, even when the baseline classifier can cope with them. However, since all binary classifiers always give an output score for a query instance, those values which are not directly related with the actual class of the instance may lead to noise in the decision process, and therefore to an erroneous classification. This is known as the "non-competence classifier problem" [8], and addressing this issue can lead to the enhancement of the final system.

The main hitch is that we cannot know a priori which are the competent classifiers for a given instance, but we can restrict the score matrix for a small subset of classes to whom membership is more probable. In the same way as in other DCS methods [28], we consider to use the neighbors of the instance to be classified in order to decide whether a classifier may be competent or not. The dynamic OVO procedure is composed of the steps shown in Algorithm 1.

In summary, this process carries out a pre-selection of the most-likely classes of the example among a large number of neighbors ($k = 3 \cdot m$), trying to provide a good trade-off between including enough "similar" classes and removing non-competent classifiers. Using the former value of k, it is quite improbable that the correct class is removed from the score-matrix and, even if that was the case, then we may be coping with an outlier or rare instance, so that the original OVO scheme would not predict it properly. The case of $6 \cdot m$ as a limit for the search procedure, which is hardly ever reached, is established aiming to not extend this search excessively in such rare cases.

Finally, any aggregation mechanism can be used to decide over the new post-processed score-matrix. Nevertheless, the original dynamic OVO procedure has shown a more robust behaviour when the *WV*

Algorithm 1 Dynamic Classifier Selection for OVO scheme

procedure DYNAMIC OVO(e, R)
  $k = 3 \cdot m$ ▷ m is the number of classes
  repeat
    $Neighbours \leftarrow kNN(e)$
    $C \leftarrow Classes(Neighbours)$ ▷ We select the class labels in the neighbourhood
    $k++$
  until $\#C > 1$ **or** $k == 6 \cdot m$
  if $C > 1$ **then**
    $R' \leftarrow [R - rows(i), cols(i)];\ i \notin C$
    return R' ▷ A subset of the score matrix
  else
    return R ▷ Standard OVO approach
  end if
end procedure

strategy is selected [11]. For this reason, we will include the combination of both WV and dynamic OVO mechanisms in our experimental study. We must recall that for the standard OVO approach, the ND criterion has been selected due to its robustness for FRBCSs.

The use of this approach has just a small computational load related to the use of the kNN classifier. Specifically, its complexity is of $O(n \cdot m)$, with m being the number of classes and n the number of examples in the training set.

In order to have a better understanding of the features of dynamic OVO, an illustrative example for the working procedure of this approach can be found at [11].

4. Experimental Study

In this section we will show the study of the performance for the Dynamic OVO approach in FRBCSs. The quality of this proposed methodology will be contrasted versus both the standard fuzzy learning algorithm and the OVO scheme.

In order to do so, we will first introduce the parameters used for the learning stage (Subsection 4.1). Then, we will describe the benchmark problems and the evaluation methodology (Subsection 4.2). Finally, we will present the experimental results and a brief discussion stressing the main findings achieved from this analysis (Subsection 4.3).

4.1. Base classifier and parameters

We consider the Chi et al.'s fuzzy rule learning [14] as base classifier to study the validity of the DCS methodology (Dynamic OVO aggregation). The confidences used in the score-matrices are obtained from the output values of the fuzzy inference in each class (step (4) of fuzzy reasoning method in Section 2.1). The configuration parameters considered are shown in Table 1, which are common for all problems, which is the default parameters' setting included in KEEL software [24] used to develop the experiments.

Table 1: Parameter specification for the Chi et al. base learner.

Parameter	*Value*
Number of Labels:	3 labels per variable
Conjunction operator:	Product T-norm
Rule Weight:	Penalized Certainty Factor [19]
Fuzzy Reasoning Method:	Winning Rule

As we have mentioned, in this contribution, we make use of the ND criterion as the most representative aggregation for FRBCS in OVO, as shown in [21]. Finally, with respect to the DCS, we use the Euclidean distance to find the neighbors of the instance (except when the data-set contains nominal values where we use the Heterogeneous Value Difference Metric, HVDM), and we will make use of the WV aggregation instead of ND, as it has shown a higher performance in a general framework [11].

4.2. Data-sets and classifiers' evaluation

We have used twenty data-sets from KEEL data-set repository [15], so that the same data partitions can used by other researchers. Additionally, instead of the commonly used cross-validation, and in order to correct the dataset shift [29, 30, 31], (training and test data do not follow the same distribution), we will use a recently published partitioning procedure called Distribution Optimally Balanced Cross Validation [32].

Table 2 summarizes the properties of these data-sets. They comprise a number of situations, from totally balanced data-sets to highly imbalanced ones, besides the different number of classes. Some of the largest data-sets (page-blocks, penbased, satimage, shuttle and thyroid) were stratified sampled at 10% in order to reduce the computational time required for training. In the case of missing values (autos and cleveland), we removed those instances from the data-set before doing the partitions. Finally, we will consider both the accuracy rate and the kappa metric to evaluate the performance of the classifiers.

In order to carry out the comparison of the classifiers appropriately, non-parametric tests should be considered, according to the recommendations for contrasting the results in Soft Computing approaches made in [17, 18]. In this contribution, we will consider the Friedman Aligned test for both computing the ranking of the algorithms according to its performance, and the p-value that determines significant differences among the results. Then, we will proceed with a Holm non-parametric statistical procedure for $1 \cdot n$ comparisons, obtaining the adjusted p-value (APV) associated with each comparison, which represents the lowest level of significance of a hypothesis that results in a rejection. Additionally, in order to perform comparisons between

Table 2: Summary description of data-sets.

Data-set	#Ex.	#Atts.	#Num.	#Nom.	#Cl.
Balance	625	4	4	0	3
Contraceptive	1473	9	9	0	3
Hayes-roth	132	4	4	0	3
Iris	150	4	4	0	3
NewThyroid	215	5	5	0	3
Tae	151	5	5	0	3
Thyroid	720	21	21	0	3
Wine	178	13	13	0	3
Vehicle	846	18	18	0	4
Cleveland	297	13	13	0	5
Page-blocks	548	10	10	0	5
Shuttle	2175	9	9	0	5
Autos	159	25	15	10	6
Glass	214	9	9	0	7
Satimage	643	36	36	0	7
Segment	2310	19	19	0	7
Ecoli	336	7	7	0	8
Penbased	1100	16	16	0	10
Yeast	1484	8	8	0	10
Vowel	990	13	13	0	11

two algorithms, we will use the Wilcoxon paired signed-rank test [33]. Any interested reader can find additional information on the thematic website http://sci2s.ugr.es/sicidm/, where software for the application of the statistical tests is provided.

4.3. Experimental results and analysis

The results from the experimentation are shown in Tables 3 and 4 for the accuracy and the kappa metrics respectively. From these tables of results, we observe that the pairwise learning approach improves the behaviour of the original fuzzy classifier. Additionally, we must stress the quality of the dynamic OVO scheme, as it excels over both the standard fuzzy algorithm and the OVO methodology according to accuracy (13 of 20 datasets) and kappa (14 of 20 datasets) metrics. When we contrast the individual results for each performance measure we observe that there is a clear correlation between both of them, thus providing a stronger support to the findings extracted. Finally, regarding the kappa metric we must point out that there are two specific problems in which we improve the results versus the standard FRBCS with respect to accuracy case, i.e. thyroid and page-blocks; this is due to the fact that dynamic OVO achieves a more balanced classification among all classes, instead of focusing on the majority class examples, as the former problems are inherently imbalanced.

In order to complement our experimental study, we proceed with a statistical analysis to stress the superiority of the dynamic OVO approach. With this aim, we compute the Friedman Aligned p-values in accuracy and kappa, which are near to zero ($6.3849E^{-4}$ and $6.6709E^{-4}$ respectively), which means that there are significant differences among the results.4 Therefore, we may run a Holm post-hoc test in order to detect which algorithms are outperformed by the control method, that is, the dynamic OVO approach (it achieves the highest rank in both cases). The results of the test are shown in Table 5. Additionally, in Table 6 we perform a Wilcoxon test to contrast the behaviour between OVO and dynamic OVO.

Table 3: Experimental results in training and test with the standard accuracy metric. From the leftmost to the rightmost column we show the results for the standard Chi et al.'s algorithm (Chi), the pairwise learning approach (OVO) and the dynamic OVO (dynOVO). The highest performance value per dataset is stressed in boldface.

Dataset	#Cl	Chi-Tr	Chi-Tst	OVO-Tr	OVO-Tst	dynOVO-Tr	dynOVO-Tst
balance	3	91.56	**90.24**	84.84	80.18	81.04	79.04
contraceptive	3	51.93	40.05	59.18	46.37	58.43	**46.84**
hayes	3	78.75	64.97	91.41	64.38	90.16	**65.67**
iris	3	93.67	93.33	96.33	**96.00**	96.33	**96.00**
newthyroid	3	85.93	84.65	95.35	93.02	96.16	**94.42**
tae	3	61.44	54.18	64.60	**57.12**	62.95	55.74
thyroid	3	92.97	**92.13**	53.07	52.55	72.32	72.81
wine	3	98.59	**92.15**	98.59	91.52	98.59	91.52
vehicle	4	66.11	61.36	73.23	62.43	74.38	**63.37**
cleveland	5	92.17	38.39	94.95	**53.88**	92.00	53.54
page-blocks	5	92.06	**91.98**	79.17	79.06	89.68	89.60
shuttle	5	80.17	80.16	83.53	83.47	99.43	**99.42**
autos	6	91.99	61.09	97.66	64.81	96.08	**65.34**
glass	7	66.24	59.02	73.38	59.86	74.08	**61.28**
satimage	7	48.32	48.28	74.41	71.98	80.37	**78.59**
segment	7	87.10	86.19	92.93	91.08	93.30	**91.99**
ecoli	8	75.83	72.39	84.00	**78.07**	82.59	77.74
penbased	10	98.24	97.85	98.50	98.05	98.50	**98.08**
yeast	10	29.68	28.98	57.26	55.21	58.44	**56.14**
vowel	11	55.73	53.23	92.70	89.49	93.54	**89.90**
Average	—	76.92	69.53	82.25	73.43	84.42	**76.35**

Table 4: Experimental results in training and test with the kappa metric. From the leftmost to the rightmost column we show the results for the standard Chi et al.'s algorithm (Chi), the pairwise learning approach (OVO) and the dynamic OVO (dynOVO). The highest performance value per dataset is stressed in boldface.

Dataset	#Cl	Chi-Tr	Chi-Tst	OVO-Tr	OVO-Tst	dynOVO-Tr	dynOVO-Tst
balance	3	.8466	**.8228**	.7517	.6721	.6901	.6545
contraceptive	3	.2948	.1422	3955	.2068	.3841	**.2135**
hayes	3	.6742	**.4835**	.8653	.4169	.8451	.4368
iris	3	.9050	.9000	.9450	**.9400**	.9450	**.9400**
newthyroid	3	.6399	.6005	.8987	.8481	.9154	**.8771**
tae	3	.4360	.3340	.4702	**.3583**	.4453	.3378
thyroid	3	.0950	.0602	.1137	.0968	.1951	**.1954**
wine	3	.9787	**.8825**	.9787	.8728	.9787	.8729
vehicle	4	.5494	.4868	.6437	.5001	.6590	**.5126**
cleveland	5	.8786	**.1872**	.9227	.1700	.8757	.1610
page-blocks	5	.4204	.4118	.3531	.3501	.5620	**.5608**
shuttle	5	.1207	.1205	.3855	.3838	.9837	**.9835**
autos	6	.8955	.5109	.9697	.5241	.9489	**.5307**
glass	7	.5205	.4253	.6479	.4624	.6525	**.4728**
satimage	7	.3584	.3580	.6930	.6642	.7630	**.7416**
segment	7	.8495	.8389	.9176	.8960	.9218	**.9066**
ecoli	8	.6456	.5933	.7809	**.6962**	.7598	.6921
penbased	10	.9804	.9761	.9833	.9784	.9834	**.9787**
yeast	10	.1442	.1348	.4560	.4282	.4697	**.4402**
vowel	11	.5136	.4860	.9197	.8844	.9289	**.8889**
Average	—	.5873	.4878	.7046	.5675	.7454	**.6199**

Table 5: Average results (test), Ranks (Friedman Aligned test) and APVs (Holm test). Control method is pointed out with asterisks.

Metric	Algorithm	Test Average	Ranking	APV
Accuracy	Chi	69.53 ± 2.194	41.500 (3)	.000411
	OVO	73.43 ± 3.085	29.000 (2)	.147457
	DynOVO	**76.35 ± 2.934**	21.000 (1)	******
Kappa	Chi	.4878 ± .0333	42.550 (3)	.000138
	OVO	.5115 ± .0428	28.375 (2)	.157845
	DynOVO	**.6199 ± .0445**	20.575 (1)	******

Observing the results of the tests, the superiority of dynamic OVO outstands. Whereas both accuracy and kappa are improved, rejecting the null hypotheses of equivalence with low p-values.

Table 6: Wilcoxon tests for the comparison between DynOVO and OVO approaches in accuracy and kappa metrics.

Comparison	Measure	R^+	R^-	p-value
DynOVO vs. OVO	Accuracy	173.5	36.5	0.009996
	Kappa	156.0	36.0	0.016647

R^+ are ranks in favor of DynOVO and R^- in favor of OVO-ND.

5. Concluding remarks

In this contribution, we have presented a study that show the success on the use of a DCS procedure to improve the behaviour of FRBCS in an OVO learning procedure. The idea behind this approach is to overcome the non-competent classifiers problem that is present in the decision process of the multi-classification. Specifically, this approach simplifies the final score-matrix by removing those classifiers whose learned classes are "far-away" from the query instance.

The good behaviour of this methodology, despite its simplicity, has been shown by means of a complete experimental study with a wide number of multi-class problems, and it has been contrasted with two different metric of performance, i.e. standard accuracy and kappa. Therefore, we must put emphasis in the positive synergy between both schemes, i.e. fuzzy learning and dynamic OVO, as it has enabled to improve the results in a high percentage, both regarding the standard FRBCS and the OVO methodology.

In the future, several works remain to be addressed. Among them, we must analyse the scalability of the dynamic OVO methodology must be studied. Clustering-based competence should also be studied in this framework, as well as different ways of DCS procedures.

Acknowledgment

This work was supported by the Spanish Ministry of Science and Technology under projects TIN-2011-28488, TIN-2012-33856, TIN2013-40765-P, and the Andalusian Research Plans P11-TIC-7765 and P10-TIC-6858.

References

[1] Jiawei Han, Micheline Kamber, and Jian Pei. *Data Mining: Concepts and Techniques*. Morgan Kaufmann, San Mateo, CA, USA, 3rd edition, 2011.

[2] A. C. Lorena, A. C. Carvalho, and J. M. Gama. A review on the combination of binary classifiers in multiclass problems. *Artificial Intelligence Review*, 30(1–4):19–37, 2008.

[3] J. Fürnkranz. Round robin classification. *Journal of Machine Learning Research*, 2:721–747, 2002.

[4] M. Galar, A. Fernández, E. Barrenechea, H. Bustince, and F. Herrera. An overview of ensemble methods for binary classifiers in multi-class problems: Experimental study on one-vs-one and one-vs-all schemes. *Pattern Recognition*, 44(8):1761–1776, 2011.

[5] Mikel Galar, Alberto Fernandez, Edurne Barrenechea, and Francisco Herrera. Empowering difficult classes with a similarity-based aggregation in multi-class classification problems. *Information Sciences*, 264:135–157, 2014.

[6] Hisao Ishibuchi, T. Nakashima, and M. Nii. *Classification and modeling with linguistic information granules: Advanced approaches to linguistic Data Mining*. Springer–Verlag, 2004.

[7] Trevor Hastie and Robert Tibshirani. Classification by pairwise coupling. *The Annals of Statistics*, 26(2):451–471, 1998.

[8] Eyke Hüllermeier and S. Vanderlooy. Combining predictions in pairwise classification: An optimal adaptive voting strategy and its relation to weighted voting. *Pattern Recognition*, 43(1):128–142, 2010.

[9] Veyis Gunes, Michel Menard, Pierre Loonis, and Simon Petit-Renaud. Combination, cooperation and selection of classifiers: A state of the art. *International Journal of Pattern Recognition and Artificial Intelligence*, 17(8):1303–1324, 2003.

[10] L. I. Kuncheva. *Combining Pattern Classifiers: Methods and Algorithms*. Wiley-Interscience, 1st edition, 2004.

[11] Mikel Galar, Alberto Fernandez, Edurne Barrenechea, Humberto Bustince, and Francisco Herrera. Dynamic classifier selection for one-vs-one strategy: Avoiding non-competent classifiers. *Pattern Recognition*, 46(12):3412–3424, 2013.

[12] Saleh Alshomrani, Abdullah Bawakid, Seong-O Shim, Alberto Fernandez, and Francisco Herrera. A proposal for evolutionary fuzzy systems using feature weighting: Dealing with overlapping in imbalanced datasets. *Knowledge-Based Systems*, 73:1–17, 2015.

[13] M.J. Gacto, R. Alcala, and F. Herrera. Interpretability of linguistic fuzzy rule-based systems: An overview of interpretability measures. *Information Sciences*, 181(20):4340–4360, 2011.

[14] Z. Chi, H. Yan, and T. Pham. *Fuzzy algorithms with applications to image processing and pattern recognition*. World Scientific, 1996.

[15] Jesús Alcalá-Fdez, Alberto Fernández, Julián Luengo, Joaquín Derrac, Salvador García, Luciano Sánchez, and Francisco Herrera. KEEL data–mining software tool: Data set repository, integration of algorithms and experimental analysis framework. *Journal of Multi–Valued Logic and Soft Computing*, 17(2-3):255–287, 2011.

[16] Arie Ben-David. Comparison of classification accuracy using cohen's weighted kappa. *Expert Systems with Applications*, 34(2):825–832, 2008.

[17] J. Demšar. Statistical comparisons of classifiers over multiple data sets. *Journal of Machine Learning Research*, 7:1–30, 2006.

[18] Salvador García and Francisco Herrera. An extension on "statistical comparisons of classifiers over multiple data sets" for all pairwise comparisons. *Journal of Machine Learning Research*, 9:2607–2624, 2008.

[19] Hisao Ishibuchi and T. Yamamoto. Rule weight specification in fuzzy rule–based classification systems. *IEEE Transactions on Fuzzy Systems*, 13:428–435, 2005.

[20] M. Elkano, M. Galar, J. Sanz, A. Fernandez, E. Barrenechea, F. Herrera, and H. Bustince Sola. Enhancing multi-class classification in farc-hd fuzzy classifier: On the synergy between n-dimensional overlap functions and decomposition strategies. *IEEE Transactions on Fuzzy Systems*, in press(99):1–1, 2015.

[21] Alberto Fernandez, Maria Calderon, Edurne Barrenechea, Humberto Bustince, and Francisco Herrera. Solving multi-class problems with linguistic fuzzy rule based classification systems based on pairwise learning and preference relations. *Fuzzy Sets and Systems*, 161(23):3064–3080, 2010.

[22] Salma Elhag, Alberto Fernandez, Abdullah Bawakid, Saleh Alshomrani, and Francisco Herrera. On the combination of genetic fuzzy systems and pairwise learning for improving detection rates on intrusion detection systems. *Expert Systems with Applications*, 42(1):193–202, 2015.

[23] Peter Clark and Robin Boswell. Rule induction with cn2: Some recent improvements. In Yves Kodratoff, editor, *EWSL*, volume 482 of *Lecture Notes in Computer Science*, pages 151–163. Springer, 1991.

[24] J. Alcalá;-Fdez, L. Sánchez, S. García, M. J. del Jesus, S. Ventura, J. M. Garrell, J. Otero, C. Romero, J. Bacardit, V. M. Rivas, J. C. Fernández, and F. Herrera. KEEL: a software tool to assess evolutionary algorithms for data mining problems. *Soft Computing*, 13:307–318, 2009.

[25] C.-C. Chang and C.-J. Lin. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2:27:1–27:27, 2011.

[26] Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, and Ian H. Witten. The WEKA data mining software: An update. *SIGKDD Explorations*, 11(1):10–18, 2009.

[27] Eyke Hüllermeier and Klaus Brinker. Learning valued preference structures for solving classification problems. *Fuzzy Sets and Systems*, 159(18):2337–2352, 2008.

[28] K. Woods, W. Philip Kegelmeyer, and K. Bowyer. Combination of multiple classifiers using local accuracy estimates. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(4):405–410, 1997.

[29] J. G. Moreno-Torres, T. Raeder, R. Aláiz-Rodríguez, N. V. Chawla, and F. Herrera. A unifying view on dataset shift in classification. *Pattern Recognition*, 45(1):521–530, 2012.

[30] J. Quiñonero Candela, M. Sugiyama, A. Schwaighofer, and N. D. Lawrence. *Dataset Shift in Machine Learning*. The MIT Press, 2009.

[31] Victoria Lopez, Alberto Fernandez, and Francisco Herrera. On the importance of the validation technique for classification with imbalanced datasets: Addressing covariate shift when data is skewed. *Information Sciences*, 257:1–13, 2014.

[32] Jose García Moreno-Torres, José A. Sáez, and Francisco Herrera. Study on the impact of partition-induced dataset shift on k-fold cross-validation. *IEEE Transactions On Neural Networks And Learning Systems*, 23(8):1304–1313, 2012.

[33] D. Sheskin. *Handbook of parametric and nonparametric statistical procedures*. Chapman & Hall/CRC, 2006.