On the influence of interval normalization in IVOVO fuzzy multi-class classifier

Mikel Uriz, Daniel Paternain, Humberto Bustince, Mikel Galar

Abstract IVOVO stands for Inverval-Valued One-Vs-One and is the combination of IVTURS fuzzy classifier and the One-Vs-One strategy. This method is designed to improve the performance of IVTURS in multi-class problems, by dividing the original problem into simpler binary ones. The key issue with IVTURS is that interval-valued confidence degrees for each class are returned and, consequently, they have to be normalized for applying a One-Vs-One strategy. However, there is no consensus on which normalization method should be used with intervals. In IVOVO, the normalization method based on the upper bounds was considered as it maintains the admissible order between intervals and also the proportion of ignorance, but no further study was developed. In this work, we aim to extend this analysis considering several normalizations in the literature. We will study both their main theoretical properties and empirical performance in the final results of IVOVO.

1 Introduction

In Machine Learning, classification problems consists in learning a model from a set of labelled training examples capable of predicting the class of new, previously unseen, examples. Classification problems are divided into two major groups depending on the number of classes that the learning algorithm should deal with: two-class (binary) and multi-class problems. The latter are usually considered to be more difficult due to the greater overlapping between decision boundaries. For-

Mikel Uriz, Daniel Paternain, Humberto Bustince, Mikel Galar
Department of Statistics, Computer Science and Mathematics
Public University of Navarre, Campus Arrosadia s/n, 31006 Pamplona, Spain
Institute of Smart Cities
Public University of Navarre, Campus Arrosadia s/n, 31006 Pamplona, Spain
e-mail: {mikelxabier.uriz, daniel.paternain, bustince, mikel.galar}@unavarra.es

tunately, multi-class problems can be reduced to binary ones using decomposition strategies [1]. Among them, the One-Vs-One (OVO) [2] strategy is widely used for this purpose. In OVO, the original problem is divided into as many binary problems as pairs of classes, which are solved by independent *base classifiers*. Consequently, new examples are classified by querying all base classifiers and aggregating their outputs. These kinds of strategies are not only useful for classifiers without inherent multi-class support, but also for those capable of managing multiple classes [2].

Fuzzy Rule-Based Classification Systems (FRBCSs) are state-of-the-art classifiers. Their main characteristic is that the model obtained is expressed by a number of rules using human-readable linguistic labels [3]. The OVO strategy has also shown to improve the accuracy of these models when addressing multi-class problems [4, 5, 6]. In [4] and [5], FARC-HD [7] was extended using OVO and proposing the usage of overlap functions to improve the final performance of the model. Afterwards, in [6], OVO was combined with IVTURS [8], handling interval-valued confidence outputs in the OVO aggregation phase for the first time. The new classifier was named as IVOVO and is center of attention in this work.

In [6], we addressed the two main obstacles when designing a OVO system dealing with interval-valued outputs: the usage of a normalization strategy for intervals and the adaptation of OVO aggregations for intervals. The interval normalization consisted in making the upper bounds sum to one, which preserves the order and ignorance. However, in the literature there is still no consensus on how intervals should be normalized and hence, other methods could have been considered. In this work, we aim to complement our previous work studying the influence of different interval normalization methods in the performance of IVOVO. We want to study whether this could be a key issue to achieve the best performance or if any of the studied alternatives is valid so as to achieve a competitive performance.

To do so, we will consider different ways for normalizing intervals [9] and study their effects in the resulting intervals both from the theoretical and applied point of view. That is, we will not only evaluate their performance in IVOVO, but we will also study whether the different normalizations are able to maintain the order established between intervals as well as other properties that may be expected after normalization.

The experimental study with IVOVO will consider twenty-two numerical datasets from the KEEL dataset repository [10]. The analysis will be supported by the usage of non-parametric statistical tests, as suggested in the specialized literature [11]. As aggregations in OVO, the voting [12] and Win Weighted Voting (WinWV) [4] will be considered.

The structure of this work is as follows. Section 2 describes preliminary concepts of FRBCSs, OVO and IVOVO required to understand the rest of the work. Section 3 details the different interval normalizations analyzed and studies their main properties. In Section 4 both the experimental framework and the corresponding experimental results are presented. Finally, in Section 5 we draw the conclusions.

2 IVOVO: Interval-Valued One-Vs-One

IVOVO stands for Interval-Valued One-Vs-One, and is based on the application of the OVO strategy to IVTURS fuzzy classifier, which outputs interval-valued confidence degrees instead of real-valued ones. For this reason, in this section we recall IVOVO and its main components: IVTURS and OVO.

2.1 Fuzzy Rule-Based Classification Systems: IVTURS

FRBCSs create models consisting of human-readable rules based on the usage of linguistic labels [3]. To generate a rule base, a learning algorithm is applied to a training set $\mathscr{D}_T$ having P labeled examples $x_p = (x_{p1}, \ldots, x_{pn}), p = \{1, \ldots, P\}$, where x_{pi} is the value of the i-th attribute $(i = \{1, 2, \ldots, n\})$ of the p-th training example. Each example belongs to a class $y_p \in \mathbb{C} = \{C_1, C_2, ..., C_m\}$, where m is the number of classes of the problem.

IVTURS algorithm [8] is based on FARC-HD (Fuzzy Association Rule-based Classification model for High-Dimensional problems) [7]. Both use rules with the following structure:

$$\text{Rule } R_j: \text{ If } x_1 \text{ is } A_{j1} \text{ and } \ldots \text{ and } x_{n_j} \text{ is } A_{jn_j} \text{ then Class} = C_j \text{ with } RW_j \qquad (1)$$

where R_j is the label of the j-th rule, $x = (x_1, \ldots, x_n)$ is a vector representing the example, $A_{ji} \in \mathbb{X}_i$ is a linguistic label modeled by a triangular membership function (where $\mathbb{X}_i = \{X_{i1}, \ldots, X_{il}\}$ is the set of linguistic labels for the i-th antecedent, being l the number of linguistic labels in this set), C_j is the class label and RW_j is the rule weight computed using the certainty factor defined in [13].

The main difference in the rule representation between FARC-HD and IVTURS is that the latter take advantage of Interval-Valued Fuzzy Sets (IVFSs) to model the uncertainty under the definition of the linguistic labels, and hence its membership functions are defined by IVFSs instead of FSs. Accordingly, the whole Fuzzy Reasoning Method (FRM) needs to be adapted to work with interval along all its steps. As a consequence, the confidence (association) degree for each class obtained in the final step is also an interval. Therefore, the final class is taken as the one with the largest confidence degree (according to an admissible order, see Section 2.2).

With respect to the rule learning algorithm, FARC-HD was composed of three steps (see [7] for more details): a fuzzy association rule extraction, a candidate rule pre-screening, and a genetic rule selection and lateral tuning. IVTURS makes use of FARC-HD for carrying out the rule extraction, but without performing the last step. Then, it introduces IVFSs and finally uses a genetic algorithm to tune the interval FRM and carry out a rule selection.

2.2 Admissible orders between intervals

The problem when dealing with intervals is the *a priori* non existence of a total order. Then, we are not always able to compare any pair of intervals, and therefore we cannot establish which is the greatest (or lowest) from a set of intervals. We recall that in IVTURS we must select the largest interval-valued confidence degree.

To solve this problem, we base ourselves on the concept of admissible orders, i.e. linear (total) orders with certain properties that can be defined in the interval setting (see, for example [14, 15]). Specifically, in this work we focus on the admissible order defined by Xu and Yager in [16], which is given as follows. Let $\mathbb{L}$ the set of all positive closed subintervals, i.e.

$$\mathbb{L} = \{x = [\underline{x}, \overline{x}] | \underline{x}, \overline{x} \in \mathbb{R} \text{ with } 0 \leq \underline{x} \leq \overline{x}\}.$$

Then, for each $x, y \in \mathbb{L}$, we have that $x \leq_{XY} y$ if and only if $\frac{\underline{x}+\overline{x}}{2} < \frac{\underline{y}+\overline{y}}{2}$ or ($\frac{\underline{x}+\overline{x}}{2} = \frac{\underline{y}+\overline{y}}{2}$ and $\overline{x} - \underline{x} \geq \overline{y} - \underline{y}$). For the sake of completeness, will also denote by $L([0,1])$ the set of all closed subintervals of the unit interval. Clearly, $L([0,1]) \subset \mathbb{L}$.

2.3 One-Versus-One (OVO)

In OVO the original m class problem is transformed into a $m(m-1)/2$ sub-problems (all possible pair of classes). Therefore, each base classifier will learn to distinguish a pair of classes $\{C_i, C_j\}$. To predict the class of a new examples, each classifier is expected to provide a pair confidence degrees $r_{ij}, r_{ji} \in [0,1]$ in favor of classes C_i and C_j, respectively. For simplicity, these outputs are stored in a *score-matrix* R. In the case of fuzzy classifiers, these pairs are rarely normalized [4, 5]. This fact requires a normalization step so that the outputs of all the base classifiers are in the same scale. Normalization with real-valued confidence degrees is direct, but it is not so straightforward with intervals.

2.4 IVOVO: Interval-Valued One-Vs-One

IVOVO [6] refers to the combination of IVTURS and OVO to enhance the performance of the former in multi-class problems. Nevertheless, there are two main issues when using OVO with IVTURS because the score-matrix is filled by interval confidence scores: 1) there is no consensus on which normalization strategy should be applied; 2) the aggregations needs to be adapted to work with intervals.

Hereafter we recall how these issues were addressed in [6]. Recall that the score-matrix is formed of intervals (R):

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

$r_{ij}, r_{ji} \in \mathbb{L}$ corresponding to the confidence degrees for classes C_i, C_j, respectively.

In IVOVO, the score-matrix R was normalized to a new score-matrix R^u in such a way that all the elements are closed sub-intervals in $[0,1]$, that is, $r^u_{ij} \in L([0,1])$ for every i, j, $i \neq j$ (according to the theory described in [8]). This was done by normalizing them according to the upper bounds:

$$r^u_{ij} = \begin{cases} \left[\dfrac{\underline{r}_{ij}}{\overline{r}_{ij}+\overline{r}_{ji}}, \dfrac{\overline{r}_{ij}}{\overline{r}_{ij}+\overline{r}_{ji}}\right] & \text{if} \quad \overline{r}_{ij} \neq 0 \quad \text{or} \quad \overline{r}_{ji} \neq 0 \\ [0.5,0.5] & \text{otherwise} \end{cases} \tag{3}$$

Interestingly, this normalization allows one to maintain the proportion of ignorance and the order between intervals. After normalizing, $\overline{r}^u_{ij} + \overline{r}^u_{ji} = 1$ holds. However, although this normalization presented good experimental results, no further analysis was developed on its suitability and influence with respect to other normalization strategies. This is why we will elaborate on this aspect.

Regarding the adaptation of the aggregations methods for OVO, they mainly consisted in using the interval arithmetic. We recall the voting strategy and the WinWV strategy as they will be the ones considered in the experimental study (notice that WV was shown to perform worse than WinWV when considering fuzzy classifiers and IVOVO).

- *Voting strategy (Vote)*: $Class = arg \max_{i=1,\ldots,m} \sum_{1 \leq j \neq i \leq m} s_{ij}$, where s_{ij} is 1 if $r^u_{ij} > r^u_{ji}$ and 0 otherwise.
- *WinWV*: $Class = arg \max_{i=1,\ldots,m} \sum_{1 \leq j \neq i \leq m} s_{ij}$, where s_{ij} is r^u_{ij} if $r^u_{ij} > r^u_{ji}$ and 0 otherwise.

3 Different approaches for the normalization of intervals

As we have stated in the introduction, one of the main problems when facing an interval-valued OVO decomposition strategy is how to perform a normalization of the interval-valued confidences. On the one hand, it is interesting to know how each normalization "transforms" the original intervals. For example, we want to know whether the resulting interval belongs to $L([0,1])$ (is bounded). On the other hand, and more interesting, we want to know if the normalization is able to keep the original ordinal structure of the interval confidences. That is, if $r_{ij} \leq_{XY} r_{ji}$, we wonder if this order is kept by the new normalized intervals.

As we have recalled in Section 2.4, the normalization adopted in the original IVOVO algorithm was done according to the upper bonds of the original intervals, i.e., making $\overline{r}^u_{ij} + \overline{r}^u_{ji} = 1$. One of the most interesting property of this method is the fact that the normalized intervals are always elements of $L([0,1])$. Moreover, it was also mentioned that the admissible order $\leq_{XY}$ between the original intervals is kept in the normalized ones, i.e., if $x \leq_{XY} y$, then $x^u \leq_{XY} y^u$.

3.1 Normalization by the lower bound and the middle point

While the original IVOVO normalization considered the upper bounds, in this subsection we analyze the usage of other points within the interval to perform the normalization. Specifically, we explore the normalization according to the middle points and to the lower bounds. Formally, if $x, y \in \mathbb{L}$, the normalization based on the middle points and the lower bounds are given, respectively, by

$$x^m = \begin{cases} \left[\dfrac{\underline{x}}{(\underline{x}+\overline{x}+\underline{y}+\overline{y})/2}, \dfrac{\overline{x}}{(\underline{x}+\overline{x}+\underline{y}+\overline{y})/2} \right] & \text{if } \underline{x}+\underline{y}+\overline{x}+\overline{y} \neq 0 \\ [0.5, 0.5] & \text{otherwise} \end{cases} \tag{4}$$

$$x^l = \begin{cases} \left[\dfrac{\underline{x}}{\underline{x}+\underline{y}}, \dfrac{\overline{x}}{\underline{x}+\underline{y}} \right] & \text{if } \underline{x}+\underline{y} \neq 0 \\ [0.5, 0.5] & \text{otherwise} \end{cases} \tag{5}$$

It is worth noting that if in the normalization by the upper bound we had that $\overline{x}^u + \overline{y}^u = 1$, now we have that $\frac{\underline{x}^m+\overline{x}^m}{2} + \frac{\underline{y}^m+\overline{y}^m}{2} = 1$ and that $\underline{x}^l + \underline{y}^l = 1$. Another important differences between x^m, x^l and x^u is the fact that x^m, x^l need not belong to $L([0,1])$, even if x, y do.

Proposition 1. *Let $x, y \in \mathbb{L}$. The following items hold:*

1. $x^m(y^m), x^l(y^l) \in \mathbb{L}$;
2. $\underline{x}^m(\underline{y}^m), \underline{x}^l(\underline{y}^l) \leq 1$ *for every* $x, y \in \mathbb{L}$;
3. $\overline{x}^m(\overline{y}^m) > 1$ *whenever* $\overline{x}(\overline{y}) > \underline{x}+\underline{y}+\overline{y}(\underline{y}+\underline{x}+\underline{x})$;
4. $\overline{x}^l(\overline{y}^l) > 1$ *whenever* $\overline{x}(\overline{y}) > \underline{x}+\underline{y}$.

However, even if the normalized intervals exceeds $L([0,1])$, the order relation between x and y is kept under these transformations.

Proposition 2. *Let $x, y \in \mathbb{L}$. If $x \leq_{XY} y$, then $x^m \leq_{XY} y^m$ and $x^l \leq_{XY} y_l$, with x^m, x^l being the normalized intervals given by Equations 4 and 5, respectively.*

Finally, we must notice that the normalization according to the lower bounds present an undesirable behavior when $\underline{x}+\underline{y} = 0$, since we will always have $x^l = y^l = [0.5, 0.5]$, discarding the information provided by the upper bounds of x and y.

Example 1. Let $x = [0, 0.6]$, $y = [0, 0.9]$. The normalized intervals according to the three normalization based on the upper bound, middle point and lower bound are given, respectively, by $x^u = [0.0, 0.4]$, $y^u = [0.0, 0.6]$, $x^m = [0.0, 0.8]$, $y^m = [0.0, 1.2]$, $x^l = [0.5, 0.5]$ and $y^l = [0.5, 0.5]$. Observe that, up to some extent, both x^u, y^u and x^m, y^m keep original information of x and y. This loss of information in x^l may be problematic in certain applications, specially if the property $\underline{x}+\underline{y} = 0$ frequently appears.

3.2 Other normalization methods

Apart from the normalizations based on values within the interval, such as the lower, middle and upper bound, in the literature one can find other approaches which were originally given for normalizing interval weighting vectors (see, for example [9]).

The first method (Other1) we recall here is the one based on interval arithmetic [17], which is the natural extension of the normalization of numbers. However, it is known to produce too wide intervals. The normalization is as follows: given $x, y \in \mathbb{L}$,

$$x^{o1} = \begin{cases} \left[\dfrac{\underline{x}}{\overline{x}+\overline{y}}, \dfrac{\overline{x}}{\underline{x}+\underline{y}}\right] & \text{if } \underline{x}+\underline{y} \neq 0 \text{ and } \overline{x}+\overline{y} \neq 0 \\ [0.5, 0.5] & \text{otherwise} \end{cases} \tag{6}$$

The normalized intervals following this methodology need not belong to $L([0,1])$, since $\overline{x}^{o1} > 1$ whenever $\overline{x} > \underline{x}+\underline{y}$. Moreover, it does not keep the ordinal structure of x and y under $\leq_{XY}$ and it shares the undesirable loss of information when $\underline{x}+\underline{y} = 0$.

Example 2. Let $x = [0.0, 0.3]$ and $y = [0.2, 0.2]$, where clearly $x <_{XY} y$. After applying the normalization, we have that $x^{o1} = [0.0, 1.5]$, $y^{o1} = [0.4, 1.0]$ and $y^{o1} <_{XY} x^{o1}$, so the order relation has been inverted.

The last two methods (Other2 and Other3) we analyze in this section are based on the "interval extended zero" method proposed in [18]. The normalization is performed by multiplying each original interval by an interval "weight", that is given by the following two formulae:

$$w^{o2} = \left[\frac{1}{\overline{x}+\overline{y}}, \frac{2}{\overline{x}+\overline{y}} - \frac{\underline{x}+\underline{y}}{(\overline{x}+\overline{y})^2}\right], \tag{7}$$

$$w^{03} = \left[\frac{1}{\underline{x}+\underline{y}} - \frac{y_{max}+y_{min}}{2(\underline{x}+\underline{y})}, \frac{1}{\overline{x}+\overline{y}} + \frac{y_{max}+y_{min}}{2(\overline{x}+\overline{y})}\right] \tag{8}$$

where

$$y_{max} = 1 - \frac{\underline{x}+\underline{y}}{\overline{x}+\overline{y}}, \quad y_{min} = \frac{\overline{x}+\overline{y}-\underline{x}-\underline{y}}{\underline{x}+\underline{y}+\overline{x}+\overline{y}}.$$

Finally, the normalized intervals are given by

$$x^{o2} = \begin{cases} w^{o2}x = [\underline{w}^{o2}\underline{x}, \overline{w}^{o2}\overline{x}], & \text{if } \overline{x}+\overline{y} \neq 0 \\ [0.5, 0.5] & \text{otherwise} \end{cases} \tag{9}$$

$$x^{o3} = \begin{cases} w^{o3}x = [\underline{w}^{o3}\underline{x}, \overline{w}^{o3}\overline{x}] & \text{if } \underline{x}+\underline{y} \neq 0 \text{ and } \overline{x}+\overline{y} \neq 0 \\ [0.5, 0.5] & \text{otherwise} \end{cases} \tag{10}$$

Here again, we cannot assure the belonging of x^{o2}, x^{o3} to $L([0,1])$, especially if x or y are very small intervals. However, they differ on how the special case

$x^{o2(o3)} = [0.5, 0.5]$ is obtained. Observe that in Equation 9, we must assure $\overline{x}+\overline{y} \neq 0$, which means $x, y \neq [0,0]$, while in Equation 10 we can obtain $[0.5, 0.5]$ as long as $\underline{x}+\underline{y} = 0$, with the consequent loss of information.

If we analyze the order according to $\leq_{XY}$, we have that its maintenance is violated by both approaches under certain conditions. Although in this work we have not fully analyze the conditions under which the order is kept, we have an interesting partial result that make us glimpse that it is mostly respected.

Proposition 3. *Let $x, y \in \mathbb{L}$ such that $x <_{XY} y$. If $\underline{x} > \underline{y}$, then $x^{o2(o3)} <_{XY} y^{o2(o3)}$.*

Example 3. Let $x = [0.1, 0.7]$ and $y = [0.4, 0.5]$, having $x <_{XY} y$. Applying equation 9 we have that $w^{o2} = [0.83, 1.32]$ and $x^{o2} = [0.083, 0.924]$, $y^{o2} = [0.332, 0.66]$ with the relation $y^{o2} <_{XY} x^{o2}$.

Now, let $x = [0.03, 0.44]$, $y = [0.14, 0.35]$. Applying equation 10 we have that $w^{o3} = [1, 6747, 3.3696]$ and $x^{o3} = [0.05, 1.483]$, $y^{o3} = [0.23, 1.18]$ and the relation $y^{o3} <_{XY} x^{o3}$.

4 Experimental study

The main goal of this experimental study is to empirically analyze the influence of the normalization in IVOVO. We want to assess the performance of the different normalizations using the same experimental framework as the one in [6]. To do so, as explained earlier, we will study the results using two OVO aggregations, Vote and WinWV. This is an interesting issue because the results in Vote will serve as a measure of how many times the order relation between intervals has been broken and how much this affects the results. Otherwise, WinWV will allow us to measure the quality of the final normalized intervals as they are directly used in the aggregation. Moreover, notice that in previous experiments [4, 5, 6], Vote always performed better than WinWV when dealing with fuzzy classifiers. One could expect that a better normalization could lead to better performance in WinWV, closing the gap between both aggregations.

4.1 Experimental framework

To carry out the experimental study we have considered twenty-two datasets from the KEEL dataset repository [10]. These are the same datasets as those considered in previous works [4, 5, 6]. In Table 1, we present a summary of all the datasets, indicating the number of examples (#Ex.), the number of attributes (#Atts.), the number of numerical (#Num.) and nominal (#Nom.) attributes, and the number of classes (#Class.).

We have used a *5-fold stratified cross-validation model* following the *Distribution Optimally Balanced Cross Validation* procedure [19]. Non-parametric statistical tests are used to support our conclusions as suggested in the specialized literature

Table 1 Summary description of the datasets.

Id.	Dataset	#Ex.	#Atts.	#Num.	#Nom.	#Class.	Id.	Dataset	#Ex.	#Atts.	#Num.	#Nom.	#Class.
aut	autos	159	25	15	10	6	bal	balance	625	4	4	0	3
cle	cleveland	297	13	13	0	5	con	contraceptive	1473	9	6	3	3
der	dermatology	358	34	1	33	6	eco	ecoli	336	7	7	0	8
gla	glass	214	9	9	0	7	hay	hayes-roth	132	4	4	0	3
iri	iris	150	4	4	0	3	lym	lymphography	148	18	3	15	4
new	newthyroid	215	5	5	0	3	pag	pageblocks	548	10	10	0	5
pen	penbased	1100	16	16	0	10	sat	satimage	643	36	36	0	7
seg	segment	2310	19	19	0	7	shu	shuttle	2175	9	9	0	7
tae	tae	151	5	3	2	3	thy	thyroid	720	21	21	0	3
veh	vehicle	846	18	18	0	4	vow	vowel	990	13	13	0	11
win	wine	178	13	13	0	3	yea	yeast	1484	8	8	0	10

[11]. More specifically, we use the Wilcoxon rank test to carry out pairwise comparisons and Aligned Friedman test to carry out multiple method comparison.

For IVTURS we used the configuration recommended by the authors: 5 fuzzy labels for each variable, 3 as maximum depth of the tree, a minimum support of 0.05, a minimum confidence of 0.8, 50 individuals as population size, 30 bits per gene for the Gray codification and a maximum of 20000 evaluations.

4.2 Influence of normalization strategies in IVOVO

Tables 2 and 3 show the classification accuracy obtained by each normalization method using both Vote and WinWV aggregations methods, respectively.

Attending at these results, there are several points to be highlighted:

- In Vote, NoNorm, Upper and Middle perform exactly the same. This was expected as Upper and Middle do not alter the order relation between intervals. Although the same could be expected by Lower, it needs to go through the else part (see Equation 5) in many more cases, making a lot of intervals to become $[0.5, 0.5]$, causing a decrease in accuracy.
- Also in Vote, the rest of the normalizations provides different results for different datasets, but looking at the overall performance Other2 and Other3 seems to behave better than Other1. There are datasets such as satimage, shuttle or vehicle, where differences are clear. Notice the three of them break the order relation in some cases. We should point out that, in general, the greater the number of times the order relation is broken, the greater the loss of performance is. Other1 and Other3 also suffer the same problem as Lower with the else part.
- With respect to WinWV, Middle is the best performer. This result may suggest that Upper (used in IVOVO) is not the most adequate for this purpose. However, we cannot make such a claim without carrying out the proper statistical analysis.
- Anyway, WinWV shows the importance of a good normalization. Lower and Other1 achieve the worst results, with performances far from the rest. Other2 and Other3 are able to overcome Upper in terms of overall accuracy and looking at NoNorm the need for normalization can be observed.

These statements needs to be validated performing the proper statistical analysis. First, Table 4 shows the results of the Aligned Friedman Ranks tests, one for each OVO aggregation to focus on the differences among normalizations.

Table 2 Classification accuracy obtained by IVOVO in testing (Vote).

Dat	NoNorm	Upper	Lower	Middle	Other1	Other2	Other3
aut	0.7713	0.7713	0.7652	0.7713	0.7592	0.7652	0.7652
bal	0.8512	0.8512	0.8030	0.8512	0.8045	0.8608	0.8094
cle	0.5457	0.5457	0.5457	0.5457	**0.5524**	0.5492	0.5457
con	0.5364	0.5364	0.5364	0.5364	0.5323	0.5330	0.5337
der	0.9529	0.9529	0.9529	0.9529	0.9529	0.9529	0.9529
eco	0.8167	0.8167	0.8137	0.8167	0.8286	0.8286	0.8195
gla	0.7098	0.7098	0.7148	0.7098	0.6911	**0.7202**	0.7188
hay	0.7445	0.7445	0.7445	0.7445	0.7445	0.7445	0.7445
iri	**0.9533**	**0.9533**	**0.9533**	**0.9533**	**0.9533**	**0.9533**	**0.9533**
lym	0.8052	0.8052	0.7983	0.8052	0.7983	0.8052	0.7983
new	**0.9488**	**0.9488**	**0.9488**	**0.9488**	0.9116	**0.9488**	0.9442
pag	0.9435	0.9435	0.9435	0.9435	0.9435	0.9435	0.9435
pen	0.9519	0.9519	0.9410	0.9519	0.9282	0.9473	0.9391
sat	0.8198	0.8198	0.8198	0.8198	0.7169	0.7481	0.8028
seg	**0.9216**	**0.9216**	0.9178	**0.9216**	0.9056	0.9173	0.9156
shu	**0.9435**	**0.9435**	**0.9435**	**0.9435**	0.8634	0.8721	0.9055
tae	0.5677	0.5677	0.5677	0.5677	0.5613	0.5742	0.5742
thy	**0.9417**	**0.9417**	**0.9417**	**0.9417**	0.9403	0.9403	0.9403
veh	0.7091	0.7091	0.7091	0.7091	0.6558	0.6913	0.7032
vow	**0.8990**	**0.8990**	0.8949	**0.8990**	0.8768	0.8980	0.8919
win	0.9663	0.9663	0.9663	0.9663	0.9609	0.9609	0.9663
yea	0.5957	0.5957	0.5944	0.5957	0.5829	0.5951	**0.5977**
AVG	**0.8134**	**0.8134**	0.8098	**0.8134**	0.7938	0.8068	0.8075

According to the tests, NoNorm, Middle and Upper are equally effective with Vote. Other2, Lower and Other3 get lower ranks, although only statistical differences are found with Other1. In WinWV, Middle is the best performer in terms of ranks, and in this case significant differences are found against NoNorm, Lower and Other1. The others give a p-value of 1.0 due to the much greater differences with the rest in the comparison. For this reason, we carry out pairwise Wilcoxon tests to compare the best four alternatives in this case (Table 5).

From these tests we can conclude that Middle outperforms all the other contenders. Therefore, with WinWV using the Middle point as normalization factor seems to be beneficial.

Our last comparison will compare the results between Vote and WinWV considering both the normalization used in the original IVOVO [6] (Upper) and the best performer in this work (Middle). We will compare the best WinWV alternative versus the Vote with NoNorm (which is the same as any normalization not altering the order relation). The results of the comparison are presented in Table 6.

The outputs of the Wilcoxon tests allows us to conclude that normalization is crucial to achieve the best performance. In our previous work [6], although WinWV allowed us to increase the performance of WV it did not allowed to overcome simple Vote (first test). However, with a better normalization (in this case using Middle), statistical differences in favour of Vote are transformed into a comparison won by WinWV (although without statistical differences).

Table 3 Classification accuracy obtained by IVOVO in testing (WinWV).

	NoNorm	**Upper**	**Lower**	**Middle**	**Other1**	**Other2**	**Other3**
aut	0.7195	0.7585	0.6400	0.7652	0.6402	0.7592	**0.7719**
bal	**0.8625**	0.8254	0.8033	0.8447	0.8064	0.8543	0.8157
cle	0.3974	0.5458	0.4785	0.5389	0.4885	0.5457	0.5422
con 0.5161	**0.5371**	0.5350	0.5364	0.5303	0.5344	0.5343	
der	0.9274	**0.9669**	**0.9669**	**0.9669**	**0.9669**	**0.9669**	**0.9669**
eco	0.7839	0.8137	0.4296	0.8316	0.4443	**0.8373**	0.8283
gla	0.5456	0.6453	0.5261	0.6966	0.5257	0.7064	0.7055
hay	0.7440	**0.7522**	**0.7522**	**0.7522**	**0.7522**	**0.7522**	**0.7522**
iri	**0.9533**	**0.9533**	0.8067	**0.9533**	0.8067	**0.9533**	**0.9533**
lym	0.7838	**0.8123**	0.8054	**0.8123**	0.8054	**0.8123**	0.8054
new	0.9163	0.9395	0.9395	**0.9488**	0.9023	**0.9488**	0.9442
pag	0.5444	0.8801	0.8404	**0.9471**	0.8387	0.9453	0.9453
pen	0.8375	0.9393	0.4514	**0.9538**	0.4551	0.9474	0.9529
sat	0.6717	0.8183	0.5465	**0.8245**	0.4810	0.7762	0.8106
seg	0.6723	0.8931	0.5979	0.9164	0.5931	0.9117	0.9130
shu	0.7872	0.8946	0.3941	0.9426	0.3771	0.8717	0.9050
tae	0.5279	0.5679	0.5364	**0.5744**	0.5297	0.5742	**0.5744**
thy	0.9251	0.9348	0.9389	**0.9417**	0.9250	0.9403	0.9403
veh	0.6250	0.7103	0.6618	**0.7161**	0.6203	0.6972	0.7079
vow	0.7202	0.8424	0.3545	0.8707	0.3354	0.8667	0.8687
win	0.9609	**0.9717**	0.8942	0.9663	0.8996	0.9609	0.9663
yea	0.4509	0.5803	0.1986	0.5937	0.1986	0.5910	0.5971
AVG	0.7215	0.7992	0.6408	**0.8134**	0.6328	0.8070	0.8092

Table 4 Aligned Friedman test

Method	Vote **Rank (p-value)**	WinWV **Rank (p-value)**
NoNorm	59.00 (-)	98.16 (0.0003+)
Middle	59.00 (1.0)	45.25 (-)
Upper	59.00 (1.0)	53.70 (1.0)
Other2	77.18 (0.5604)	48.00 (1.0)
Lower	78.84 (0.5604)	121.61 (0.0000+)
Other3	89.55 (0.1156)	49.91 (1.0)
Other1	119.93 (0.0000+)	125.86 (0.0000+)

+ near the p-value means that statastical differences are found at 95% confidence.

Table 5 Wilcoxon test for WINWV aggregation method

		Middle	Other2	Other3	Upper
Middle	(W/T/L)	-	13/5/4	13/5/4	15/4/3
	p-value	-	0.0973*	0.0531*	0.0012+
Other2	(W/T/L)		-	10/5/7	12/4/6
	p-value		-	0.3801	0.1309
Other3	(W/T/L)			-	12/3/7
	p-value			-	0.0001+

* and + near the p-value mean that statistical differences are found at 90% and 95% confidence, respectively.
(W/T/L) stands for (Wins/Ties/Losses)

Table 6 Wilcoxon test for best VOTE method and best WINWV method

Comparison	R+	R-	Hypothesis	p-value
NoNorm_Vote (IVOVO) vs Upper_WinWV	194.5	58.5	Rejected for NoNorm_Vote	0.0273
NoNorm_Vote vs Middle_WinWV	113.5	139.5	Not rejected	0.6726

5 Conclusions

In this paper we have focused on analyzing the influence of different normalization methods for intervals in IVOVO. To do so, we have considered five ways of normalizing interval and we have analyzed some of their main properties, such as whether they maintain the order relation between intervals (considering Xu and Yager's admissible order). Then, we have carried out an experimental study were the high influence of normalization has been shown. Overall, the normalization based on the middle point has shown to perform well with both Vote and WinWV aggregations. More interestingly, the usage of this normalization has allowed us for the first time to improve the performance of Vote strategy using the confidences given by a fuzzy classifier.

For future work we aim to carry out a deeper study including more normalization methods. From a theoretical point of view, we are interested in analyzing whether all the normalization methods based on internal points (lower, middle and upper) satisfy the usual properties demanded to normalized interval-valued vector. Moreover, we want to extend the maintenance of not only the Xu and Yager's order, but many other admissible orders. From an applied point of view, we will check if new methods allow us to outperform the results presented in this work, specially when considering WinWV.

Acknowledgment

This work has been partially supported by the Spanish Ministry of Science and Technology under the project TIN2016-77356-P and the Public University of Navarre under the project PJUPNA13.

References

1. A. Lorena, A. Carvalho, and J. Gama, "A review on the combination of binary classifiers in multiclass problems," *Artif. Intell. Rev.*, vol. 30, no. 1-4, pp. 19–37, 2008.
2. 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 Recogn.*, vol. 44, no. 8, pp. 1761 – 1776, 2011.
3. H. Ishibuchi, T. Nakashima, and M. Nii, *Classification and modeling with linguistic information granules: Advanced approaches to linguistic Data Mining*. Springer-Verlag, 2004.
4. M. Elkano, M. Galar, J. A. Sanz, A. Fernández, E. Barrenechea, F. Herrera, and H. Bustince, "Enhancing multiclass classification in FARC-HD fuzzy classifier: On the synergy between *n*-dimensional overlap functions and decomposition strategies," *IEEE Trans. Fuzzy Syst.*, vol. 23, no. 5, pp. 1562–1580, 2015.

5. M. Elkano, M. Galar, J. Sanz, and H. Bustince, "Fuzzy rule-based classification systems for multi-class problems using binary decomposition strategies: On the influence of n-dimensional overlap functions in the fuzzy reasoning method," *Information Sciences*, vol. 332, pp. 94–114, 2016.
6. M. Elkano, M. Galar, J. Sanz, G. Lucca, and H. Bustince, "IVOVO: A new interval-valued one-vs-one approach for multi-class classification problems," in *17th Int. Fuzzy Sys. Assoc. (IFSA)*, 2017, pp. 1–6.
7. J. Alcalá-Fdez, R. Alcalá, and F. Herrera, "A fuzzy association rule-based classification model for high-dimensional problems with genetic rule selection and lateral tuning," *IEEE Trans. Fuzzy Syst.*, vol. 19, no. 5, pp. 857–872, 2011.
8. J. Sanz, A. Fernández, H. Bustince, and F. Herrera, "IVTURS: A linguistic fuzzy rule-based classification system based on a new interval-valued fuzzy reasoning method with tuning and rule selection," *IEEE Trans. Fuzzy Syst.*, vol. 21, no. 3, pp. 399–411, 2013.
9. O. Pavlačka, "On various approaches to normalization of interval and fuzzy weights," *Fuzzy Sets and Syst.*, vol. 243, pp. 110 – 130, 2014.
10. J. Alcalá-Fdez, A. Fernandez, J. Luengo, J. Derrac, S. García, L. Sánchez, and F. Herrera, "KEEL data-mining software tool: Data set repository, integration of algorithms and experimental analysis framework," *Journal of Multiple-Valued Logic and Soft Computing*, vol. 17:2-3, pp. 255–287, 2011.
11. S. García, A. Fernández, J. Luengo, and F. Herrera, "A study of statistical techniques and performance measures for genetics-based machine learning: Accuracy and interpretability," *Soft Comput.*, vol. 13, no. 10, pp. 959–977, 2009.
12. J. Friedman, "Another approach to polychotomous classification," Department of Statistics, Stanford University, Tech. Rep., 1996.
13. H. Ishibuchi, T. Yamamoto, and T. Nakashima, "Hybridization of fuzzy GBML approaches for pattern classification problems," *IEEE Trans. System, Man and Cybernetics B*, vol. 35, no. 2, pp. 359–365, 2005.
14. H. Bustince, J. Fernandez, A. Kolesárová, and R. Mesiar, "Generation of linear orders for intervals by means of aggregation functions," *Fuzzy Sets and Syst.*, vol. 220, pp. 69–77, 2013.
15. D. Paternain, L. D. Miguel, G. Ochoa, I. Lizasoain, R. Mesiar, and H. Bustince, "The interval-valued choquet integral based on admissible permutations," *IEEE Trans. Fuzzy Syst.*, In Press.
16. Z. S. Xu and R. R. Yager, "Some geometric aggregation operators based on intuitionistic fuzzy sets," *Int. J. General Syst.*, vol. 35, no. 4, pp. 417–433, 2006.
17. R. Xu, "Fuzzy least-squares priority method in the analytic hierarchy process," *Fuzzy Sets and Syst.*, vol. 112, pp. 395 – 404, 2000.
18. P. Sevastjanov, L. Dymova, and P. Bartosiewicz, "A new approach to normalization of interval and fuzzy weights," *Fuzzy Sets and Syst.*, vol. 198, pp. 34 – 45, 2012.
19. J. Moreno-Torres, J. Saez, and F. Herrera, "Study on the impact of partition-induced dataset shift on k-fold cross-validation," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 23, no. 8, pp. 1304–1312, 2012.