A Fuzzy Ensemble of Features Selectors through SMART-or Aggregation and Yager fuzzy Ordering*

Marco Baioletti[1 [0000−0001−5630−7173]], Andrea Capotorti[1,2 [0000−0002−1337−8315]], and Alessio Troiani[1 [0000−0002−1192−7309]]

[1] Dipartimento di Matematica e Informatica, University of Perugia, Perugia, Italy
[2] member of GNAMPA-INdAM group

Abstract. We propose a novel feature selection (FS) method based on peculiar fuzzy set generation, aggregation, and ordering. In particular, here we propose to elicit the fuzzy membership by a proper probability-possibility transformation of frequencies stemming from the bootstrap application of different filter FS methods. At the same time, we aggregate such vague scores of each feature via the recently introduced SMART-or fuzzy aggregation operator. Finally, to rank the features for the selection proposal we adopt Yager's ordering. Empirical results on benchmark databases show an overperformance of our approach with respect to different generation techniques, or aggregation functions and orderings.

Keywords: fuzzy Ensemble Feature Selection, SMART-or aggregation, Yager's ordering

1 Motivations, state of the art and comparison with other approaches

The need to extract useful information from an extremely large dataset where thousands of features describe each data point is, nowadays, ubiquitous. Examples span a large array of fields from healthcare to environmental monitoring to marketing to include a few examples.

* The work of MB and AC has been partially supported by PRIN 2022 project "Models for dynamic reasoning under partial knowledge to make interpretable decisions" (grant number 2022AP3B3B) funded by the European Union – Next Generation EU. The work of AT has been partially supported by PRIN 2022 PNRR: "RETINA: REmote sensing daTa INversion with multivariate functional modeling for essential climAte variables characterization" (Project Number: P20229SH29, CUP: J53D23015950001) funded by the European Union under the Italian National Recovery and Resilience Plan (NRRP) of NextGenerationEU, under the Italian Ministry of University and Research (MUR). The work of all authors has been partially supported by their Department grant "Ricerca di Base".

Though in principle datasets with a vast number of features could lead to more accurate predictions, the abundance of figures comes together with higher computational cost and complexity to process the data.

In addition, it is common that the accuracy of predictions is mostly driven by a small, but unfortunately unknown, set of features.

In this framework, determining the most relevant features to obtain accurate predictions would be beneficial in two respects: on one hand, it would reduce the complexity of the problem; on the other hand, it would help to improve the explainability of the Artificial Intelligence model being used (see, e.g., [10]).

A Feature Selection (FS) method assigns a score to each feature producing an importance rank that allows discarding the least relevant ones. However, though several approaches for feature selection have been developed, there is no unique feature selection method suitable for all possible applications. Moreover, these methods are very sensitive with respect to the input data. Consequently, even in the context of a specific application, the results produced by each selection method, considered individually, prove to be highly unstable.

To improve the stability of the feature selection procedure, ensembles of feature selection methods have been recently proposed and are gaining wide popularity (see, e.g., [3,13]). An ensemble method consists of a suitable combination of the results of several *pure* feature selection methods. To produce this combination, two different techniques are common: one consists in merging the different feature rankings stemming from the pure methods; the other aggregates the specific scores of each feature assigned by each pure method. In line with [20,21] we choose to follow the latter approach, obtaining a final feature rank by aggregating the scores obtained by four different pure feature selection methods.

Since one of the main goals of the ensemble technique is to have a robust method, in line again with [20,21], we choose the four different feature selection methods among the so-called *filter* ones. We made such a choice because, differently from the other common types of selectors, namely the *wrapped* and the *embedded* ones (see [15]), filter methods are *model agnostic*, i.e. they assign the scores to each feature independently of any subsequent learning algorithm used for classification or clustering. This renders the proposal generally applicable. To compare our method with those presented in [20,21], we will choose one different filter-based traditional feature selection algorithm in each of the four main groups (for more details refer again to [15]):

- Correlation-based Feature Selection (CFS) [11] among the statistical-based methods;
- ReliefF Feature Selection (ReliefF) [16] among the similarity-based methods;
- Mutual Information based Feature Selection (MIFS) [1] among the information-theoretical-based methods;
- Supervised Infinite Feature Selection (Inf-FS$_S$) [17] among the graph-based methods.

Note that, among the graph-based methods, we choose the Inf-FS$_S$ instead of the similar IFS method adopted in [20] or the ILFS method used in [21] because of its superior performance as shown in [17].

While aggregation attenuates the different behaviors of the chosen method, the instability mentioned above of every single method can be attenuated by iterating computations on different subsets of the original datasets and considering the variability of the results. This latter variability is usually obtained through bootstrapping techniques and by expressing the vagueness of the results through fuzzy sets, as done again in [20,21].

In their latter contribution, Shen et al. conclude with the need to explore fuzzy set generation and aggregation methods different from those they used: in [20] fuzzy memberships coincide with relative frequencies of empirical scores, whereas in [21] they are estimated through Gaussian shapes based on scores' mean and variance.

Here, in order to determine fuzzy membership, we will follow the principle of maximum specificity through a probability-possibility transformation as proposed in [8] (for more details about its maximum specificity refer to [9]).

We propose an alternative method, w.r.t. the aforementioned authors also for the aggregation and the defuzzification/feature-ranking steps. Indeed, instead of the weighted fuzzy combination and center of average defuzzifier used in [20], or the drastic sum aggregation and centroid defuzzification used in [21], here we will apply the recent SMART-or [2] as aggregation operator and the Yager's ordering [22] to obtain the feature-ranking.

This is because on one side - differently from the weighted combination or the drastic sum - the SMART-or operator does not need any weight choice, and hence no *training*, and takes into the right consideration of the different degrees of agreement/disagreements among the different fuzzy sets; on the other side - differently from the center of average or the centroid defuzzifiers - Yager's ordering is more sensitive to the specific shapes of the membership functions we obtained. Moreover, the use of such ordering has produced more accurate and stable classification results with respect to other proposals, like those using ideal benchmarks (see e.g. [5,6]).

For the sake of comparing our method with the one proposed by Shen et al. in [21], we will adopt the same bootstrapping scheme, the same four FS methods, with the slight aforementioned difference for the graph-based one, and the same data frames taken from the UCI machine learning repository [7] for the empirical result. Anyhow, we remark that our method can be applied to an arbitrary ensemble method based on arbitrary filter-based FS algorithms.

The rest of the paper is organized as follows: in Section 2 we formalize the notation and the main concepts adopted in the rest of the paper; in Section 3 we will detail the elicitation procedure for the membership functions, the recently introduced SMART-or operator to aggregate different fuzzy numbers and its generalization used here for fuzzy quantities, together with the different orderings that can be used to arrive at a final ranking among the features; finally in Section 4 we will report the performances in terms of accuracy, measured by the area under the Receiver Operating Curve (AUC) [18], and stability, measured through the specific consistency index introduced in [14].

2 Notation and main steps

Let us introduce the quantities involved in our method. Let D be a given dataset containing S data samples (instances), each expressing N values of different features $X_1, \ldots, X_N$ plus one class label Y. The goal is to build an order (rank) of the N features from the least to the most relevant to predict (classify) the class value. Such ranking will be obtained by combining (aggregate) the outputs of M feature selection methods FS_j, $j = 1, \ldots, M$.

The procedure follows the same three main steps introduced in [21]:

1. a bootstrap phase, where scores $S^i_{j,l} \in [0, 1]$ are generated in each l-th run of the bootstrap sampling, $l = 1, \ldots, L$, for each i-th feature, $i = 1, \ldots, N$, by each method FS_j, $j = 1, \ldots, M$, and transformed into fuzzy membership functions μ^i_j of the unit interval $[0, 1]$;
2. an aggregation phase using a fuzzy operator obtaining a single membership $\overline{\mu}^i$ for each feature, $i = 1, \ldots, N$;
3. a final ranking $r_1, \ldots, r_N$ obtained through some defuzzification method and expressing the relevance, from the least to the most important, of the features.

As already mentioned and as we will detail in the next two sections, for the first step we will use the same variability generation of [21], i.e. the scores are obtained by the application of the $M = 4$ filter feature selectors CFS, ReliefF, MIFS, Inf-FS$_S$ - with the slight difference in Inf-FS$_S$ already described in the previous section - in each of the $L = 100$ bootstrap samplings. On the contrary, our approach is completely different from the aforementioned proposal about the fuzzy membership transformation. Indeed, while in [21] relative frequencies of the scores are transformed into Gaussian-shaped memberships based on scores' mean and variance, here we will transform them through a so-called probability/possibility transformation (see details in the next section).

In the other two procedural steps, we further differentiate from [21].

In particular, to aggregate the four memberships μ^i_j, $j = 1, \ldots, M = 4$, of each i-th feature we used the recently introduced SMART-or operator $\veebar$ [2] because more expressive of the agreements/disagreements among the different inputs with respect to the drastic sum adopted in [21].

Moreover, for the last ordering step, we have taken into account several orders among fuzzy sets to see which one produces the best performance:

- the centroid-based ordering already used in [21]
- Chen's ordering based on comparison with maximizing set and minimizing set [4]
- an ad-hoc ordering implemented by us based on similarity with respect to the MIN operator among fuzzy numbers [12]
- Yager's ordering among fuzzy quantities on the unit interval [22]

As we will see in empirical result Section 4, Yager's ordering has resulted in higher effectiveness in terms of accuracy and stability results.

3 Fuzzy sets generation, aggregation and ranking

The bootstrap sampling consists of $L = 100$ generation of random samples with replacement of the same size S of the original dataset D. Hence, in each run, some instances could be present several times, while others could not be present.

By applying one specific feature selection method FS_j to a specific sample we obtain 100 scores $S^i_{j,l}$ for each feature. These scores are normalized into the $[0, 1]$ interval with the usual transformation

$$\hat{S}^i_{j,l} = \frac{S^i_{j,l} - \min \mathbf{S_j}}{\max \mathbf{S_j} - \min \mathbf{S_j}} \tag{1}$$

where $\min \mathbf{S_j}$ and $\max \mathbf{S_j}$ are the minimum and the maximum of all the scores obtained with FS_j for all the N features.

These S normalized scores are then discretized into a 100 equally spaced class distribution with intervals $i_k = [(k-1)/100, k/100[$ for $k = 1, \ldots, 99$, and $i_{100} = [0.99, 1]$, obtaining a vector $\mathbf{f}^i_j$ of 100 relative frequencies $f^i_{1,j}, \ldots, f^i_{100,j}$. These relative frequencies can be thought of as probability mass function values and consequently transformed into a fuzzy membership function μ^i_j through a probability-possibility transformation among those proposed in [8]. In particular, we use the probability-possibility transformation proposed in [8, Section 2.3] for the finite case. This is because of the discretization we performed through the previous distribution in classes.

In fact, we can define the membership functions, thought of as possibility distributions, by the transformation

$$\mu^i_j(x) = \sum_{l \in L_k} f^i_{l,j} \quad \forall x \in i_k \text{ with } L_k = \{l \in 1, \ldots, 100 \,|\, f^i_{l,j} \leq f^i_{k,j}\}. \tag{2}$$

Such transformations dominate $\mathbf{f}^i_j$, i.e. $\mu^i_j(x) \geq f^i_{k,j}$ for any $x \in i_k$; they are order-equivalent to $\mathbf{f}^i_j$, i.e. $\mu^i_j(x_1) \leq \mu^i_j(x_2)$ iff $x_1 \in i_{k_1}$ and $x_2 \in i_{k_2}$ such that $f^i_{k_1,j} \leq f^i_{k_2,j}$; and they are maximally specific, i.e. any other possibility distribution π^i_j dominating $\mathbf{f}^i_j$ and ordering equivalent is such that $\pi^i_j(x) \geq \mu^i_j(x)$.

These M fuzzy member functions μ^i_j, $j = 1, \ldots, M = 4$, can be aggregated together through the SMART-or $\veebar$ operator, obtaining a single fuzzy membership function

$$\overline{\mu}^i = \mu^i_1 \veebar \ldots \veebar \mu^i_M \tag{3}$$

for each feature, $i = 1, \ldots, N$.

The SMART-or $\veebar$ operates as a weighted average of the extrema of the different alpha-cuts, with weights tuned to obtain a specifically aimed behavior of the merging, i.e, towards the more external values (in line with the canonical max t-conorm if applied "vertically"). This behavior emphasizes the disagreement, in terms of weak overlapping, of the different alpha-cuts. To obtain this, fixing an alpha-cut, the weights of the $M - 1$ outer extrema, with indexes in O_l for the

left extrema and in O_r for the right ones, are $\frac{1}{n}(1+\epsilon_j)$, with

$$\epsilon_j = \begin{cases} \frac{\sum_{f=1}^{M} \frac{1}{f} \pi_f^j}{\Delta} & \text{if } \Delta \neq 0 \\ 0 & \text{otherwise} \end{cases} \tag{4}$$

with π_f^j the length of the various overlapping and Δ the range of the alpha-cuts, as depicted in Fig.1.

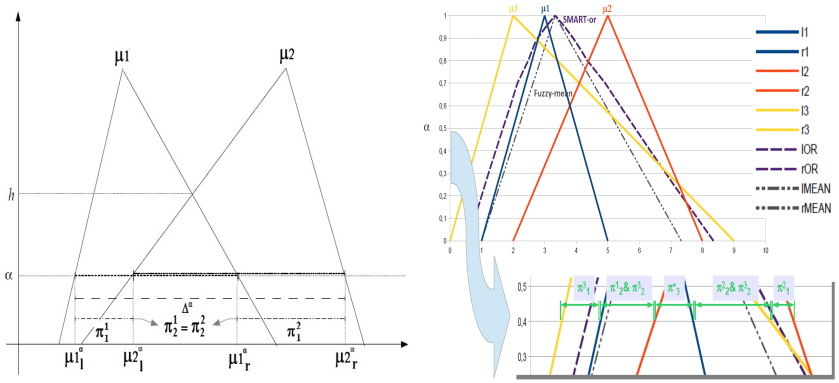


Fig. 1: Overlapping lengths π_f^j and range Δ of alpha-cuts involved in the weights ϵ_j for the SMART-or $\veebar$ aggregation: on the left between two memberships; on the right among three.

For the two inner extrema, i.e. the largest left extremum and the lowest right extremum, the weights are simply $\frac{1}{n}(1-\sum_{j\in O_*}\epsilon_j)$ (for a full description of this operator $\veebar$ refer to [2]).

Note that, since the previously described discretization of the normalized scores into the class distribution, the membership functions μ_j^i are step functions, hence with a finite number of distinct alpha-cuts. This renders the SMART-or operator $\veebar$ easily applicable and effectively implementable for our purposes, even if the original formulation in [2] was only for fuzzy numbers, while here we deal with more general fuzzy quantities.

In Fig.2 it is possible to appreciate the difference between the SMART-or aggregation and the drastic-sum conorm

$$\text{Drastic Sum}(\mu_A(x), \mu_B(x)) = \begin{cases} \mu_A(x) \text{ if } \mu_B(x) = 0 \\ \mu_B(x) \text{ if } \mu_A(x) = 0 \\ 1 \qquad \text{otherwise} \end{cases} \tag{5}$$

adopted in [21] for a feature in one of the datasets used in the empirical results (see Tab.1 in the next Sec.4).

Once we obtain the aggregated fuzzy membership functions (3) for all the N features, as shown e.g. in Fig.3, we can rank them in some ascending order.

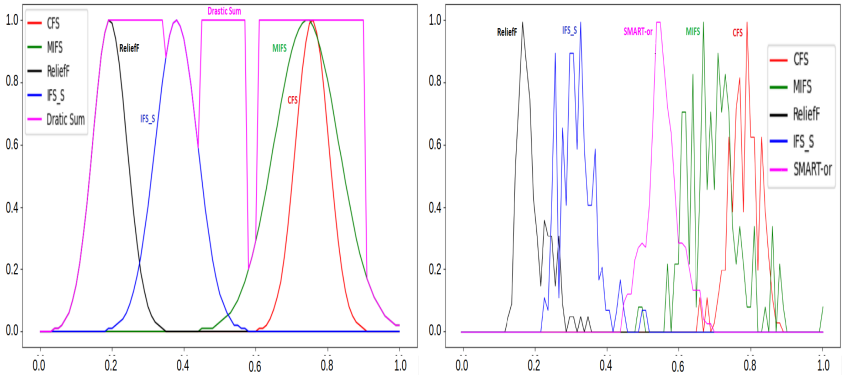


Fig. 2: fuzzy memberships and their aggregation for feature X_4 in the Glass datasets: Gaussian-shaped memberships aggregated with drastic sum (left); probability/possibility transformed membership aggregated through the SMART-or $\veebar$ (right).

In literature, there are several reasonable possible orderings of fuzzy sets in the unit interval, as ours are. The most popular, and for this reason used in [21], is the ordering based on the defuzzification through the centroids (centers of gravity). Although being very practical and easily computable, this ordering is very insensitive with respect to the shapes of the input memberships. Hence, to compare the performances of different orderings, we also compute different indexes.

More sensible orderings are those in line with [4,5,6] that compute an index measuring a distance with two benchmark minimizing and maximizing fuzzy sets (as those shown in Fig.3 with dashed/dotted lines). We tried an ordering based on the performance index proposed in [6]:

$$P_i = \frac{d_i^-}{d_i^- + d_i^+} \tag{6}$$

with

$$d_i^- = \int_{\text{supp}\overline{\mu}^i \bigcup \text{supp}\mu_{\min}} |\overline{\mu}^i(x) - \mu_{\min}(x)|dx \quad d_i = \int_{\text{supp}\overline{\mu}^i \bigcup \text{supp}\mu_{\max}} |\overline{\mu}^i(x) - \mu_{\max}(x)|dx \tag{7}$$

but with the two benchmark sets defined as in [4]:

$$\mu_{\min}(x) = \left(\frac{x_{\max} - x}{x_{\max} - x_{\min}}\right)^k \quad \mu_{\max}(x) = \left(\frac{x - x_{\min}}{x_{\max} - x_{\min}}\right)^k \tag{8}$$

where x_{min} and $x_{\max}$ are the minimum and maximum, respectively, of the union of the supports of all the memberships $\overline{\mu}^i$, $i = 1, \ldots, N$, while we toned the parameter k to 2 for better distinguish mostly overlapping memberships close to the benchmarks.

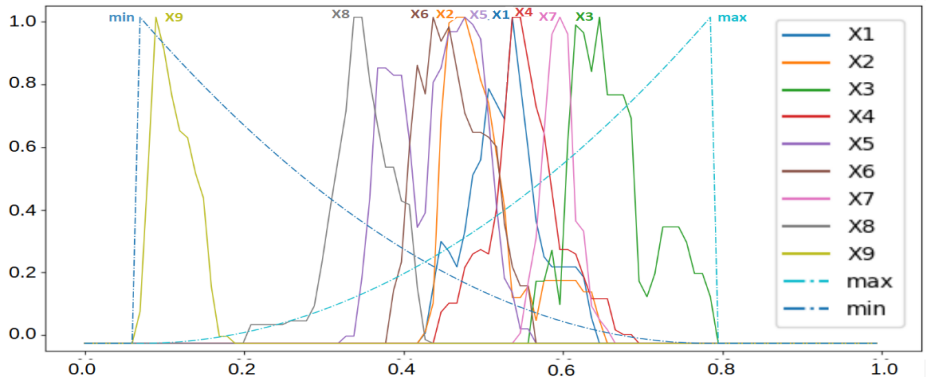


Fig. 3: Aggregated fuzzy memberships for features $X_1, \ldots, X_9$ in Glass dataset and the two min and max benchmarks.

Since the benchmark sets (8) depend only on the supports of the input memberships and their shapes, we decide to evaluate also a different reference set, specifically that obtained by applying the min operators among a set of fuzzy numbers (for more details see e.g. [12]). So we recursively compute such minimum among a subset with indexes $I \subseteq N$

$$\mathrm{MIN}_{\mu^I}(x) = \sup_{x=\min\{x_i | i \in I\}} \min\{\overline{\mu}^i(x_i) | i \in I\}. \tag{9}$$

We then rank at the lower available position the feature X_i with membership $\overline{\mu}^i$ more similar to MIN_{μ^I}, with similarity between fuzzy sets measured in the canonical way

$$\mathrm{SIM}(\overline{\mu}^i, \mathrm{MIN}_{\mu^I}) = \frac{\int_0^1 \left(\overline{\mu}^i(x) \wedge \mathrm{MIN}_{\mu^I}(x)\right) dx}{\int_0^1 \left(\overline{\mu}^i(x) \vee \mathrm{MIN}_{\mu^I}(x)\right) dx}. \tag{10}$$

The index i of such feature is removed from the index set I and the procedure is iterated.

We finally also implemented the ordering among fuzzy quantities in the unit interval - as our $\overline{\mu}^i$ are - proposed by Yager in [22] that is based on the following function

$$F(\overline{\mu}^i) = \int_0^1 M(C^i_\alpha) d\alpha \quad i = 1, \ldots, N \tag{11}$$

where $M(C^i_\alpha)$ are the mean values of the elements in the alpha-cuts C^i_α of the memberships $\overline{\mu}^i$.

4 Empirical results

For comparison reasons, we implemented four different ensembles of feature selection: one that mimics exactly the framework used in [21], named "Drastic sum/centroid", and the other three implementing our framework described

in the previous section, differentiated by the final ordering method and hence named "SmartOR/minmax", "SmartOR/SIMmin", and "SmartOr/Yager", respectively.

The comparisons have been conducted by measuring the performances in terms of classification accuracy and stability of the four different ensembles applied to the same eight datasets used in [21] as testing data (since our framework is parameter-free we don't need any training data repository). Such datasets are taken from the UCI machine learning repository [7] and are usually adopted as benchmarks. Their description is reported in Tab.1.

Table 1: General description of testing data repository

Dataset	n. class labels (C)	n. features (N)	n. instances (S)	$\frac{S}{CN}$
Appendicitis	2	7	106	7.57
BCC	2	9	116	6.44
Breast Tissue	6	9	106	2.0
CMSC	2	18	540	15.0
Glass	6	9	214	4.0
Musk	2	166	476	1.43
WDBC	2	30	569	9.48
Yeast	10	8	1484	18.55

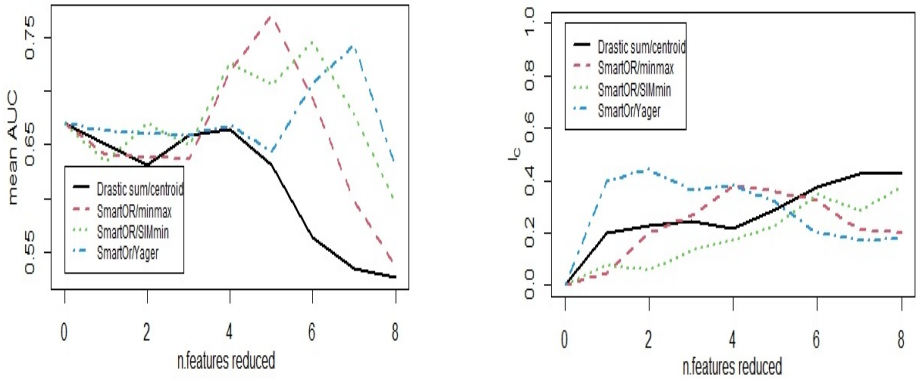
We have designed a 10-fold cross-validation framework with backward feature elimination. This means that each dataset is randomly split into 10 subsets and, cyclically, one of them is used to perform classification predictions by removing one feature per time on the basis of the rankings estimated on the rest of the dataset through the four alternative ensembles.

As a classification algorithm, we preferred the Naive Bayes (NB) with respect to other possibilities since its good performances are widely recognized, but especially because it is based on the assumption of stochastic independence among the features and our filter features selectors score the feature individually, hence independently from each other.

As far as the performance metrics are concerned, we chose to measure the accuracy of the predictions through the widely adopted area under the Receiver Operating Curve (AUC) [18]. Since the 10-fold cross-validation schema, we obtain one value of such AUC in any one of the 10 runs. Hence, for each dataset and each ensemble, we ended up with a mean AUC for each removed feature, as illustrated, e.g., in Fig.4a for the BCC dataset.

About stability, we adopted the consistency index specifically introduced in [14] for the feature selection

$$I_C(A,B) = \frac{r - \frac{k^2}{N}}{k - \frac{k^2}{N}} = \frac{rN - k^2}{k(N-k)} \tag{12}$$



(a) mean AUC (b) I_C

Fig. 4: mean AUC and I_C paths for the BCC dataset

where A and B are proper subset of the features $X_1, \ldots, X_N$ with same cardinality k and cardinality of common elements (intersection) r. In our case, the feature subsets are composed of the best k, $k = N-1, \ldots, 1$, features stemming in each learning phase of the cross-validation. Hence, the index I_C is computed pairwise among them and for each method and each dataset we have a I_C value for each removed feature, as illustrated, e.g., in Fig.4b for the BCC dataset.

Since the main goal of any feature selector is to decrease the space complexity, for each dataset in Tab.1, we ranked the four different ensembles on the base of either AUC and I_C index globally, as overall weighted mean indexes, and on three peculiar cut points: at only 2, at $\log_2(N)$, and at $\sqrt{N}$ features remaining, respectively. In the overall weighted mean indexes, each value of the mean AUC and I_C is multiplied by the percentage of the number of features reduced in each iteration of the backward elimination.

Since the performances of the ensembles change in each dataset, we synthesized them with mean ranks. Note that these ranks go from the best to the worst, hence best-performance ensembles have lower ranks.

Having two decision criteria, mean AUC and I_C based ranks, we firstly select the best ensembles based on the classification accuracy (mean AUC based ranks) and later we decide the best ensemble on the basis of the stability (I_C based mean ranks).

Results for the first criterion are reported in Tab.2, where it results that our ensembles with orders based on min and max benchmarks (8) and on Yager's index (11) outperform the other two.

Since there is no dominance between the last two ensembles, we passed to consider for them the I_C based mean ranks at the same points, obtaining the results reported in Tab.3.

From these results, we can assert that our framework based on the SMART-or aggregation operator and Yager's ordering outperforms the others.

Table 2: Global and at cut points AUC based mean ranks for the four ensembles with aggregation operator and ordering reported in the first column

ensemble	AUC based mean ranks			
	global	2 features	$\log_2(N)$ features	$\sqrt{N}$ features
Drastic sum/centroid	2.625	2.625	2.75	2.375
SmartOR/SIMmin	3.375	3.25	3.375	3.625
SmartOR/minmax	2.25	2.0	1.9375	2.0
SmartOr/Yager	1.75	2.125	1.9375	2.0

Table 3: Global and at cut points I_C based mean ranks for the ensembles with SMART-or aggregation operator in both, and with ordering based on min and max benchmarks (8) or on Yager's index (11), respectively.

average/ordering	I_C based mean ranks			
	global	2 features	$\log_2(N)$ features	$\sqrt{N}$ features
SmartOR/minmax	1.75	1.8125	1.75	1.75
SmartOr/Yager	1.25	1.1875	1.25	1.25

5 Conclusion and future developments

We propose a fuzzy ensemble of filter feature selections based on the recently introduced SMART-or operator for the aggregation step and Yager's ordering of fuzzy sets in the unit interval for the ranking.

Comparison with other similar approaches [20,21] and other orderings applied to benchmark datasets has shown the overperformance of the proposal.

Our proposal can be extended to other ambits, e.g. to mathematical physics to select main contributions to lattice-gas energy fluctuations ([19]), or to decision theory whenever there is a poll of N experts, or sources of information, that express fuzzy grades on M attributes. For each attribute, the N fuzzy grades are aggregated through the SMART-or operator, obtaining a list of M final grades so that a ranking among attributes can be performed.

In the future, it could be helpful to explore other fuzzy orderings that could produce even more stable results. Moreover, due to the variability in performance indexes, a different fully fuzzy analysis of the feature selection results could be performed.

References

1. Brown, G., Pocock, A., Zhao, M.J., Luján, M.: Conditional likelihood maximisation: A unifying framework for information theoretic feature selection. Journal of Machine Learning Research **13**(2), 27–66 (2012)

2. Capotorti, A., Figà-Talamanca, G.: Smart-or and smart-and fuzzy average operators: A generalized proposal. Fuzzy Sets and Systems **395**, 1–20 (2020)
3. Chen, C.W., Tsai, Y.H., Chang, F.R., Lin, W.C.: Ensemble feature selection in medical datasets: Combining filter, wrapper, and embedded feature selection results. Expert Systems **37**(5) (2020)
4. Chen, S.H.: Ranking fuzzy numbers with maximizing set and minimizing set. Fuzzy Sets and Systems **17**(2), 113–129 (1985)
5. Chou, S.Y., Dat, L.Q., Yu, V.F.: A revised method for ranking fuzzy numbers using maximizing set and minimizing set. Computers & Industrial Engineering **61**(4), 1342–1348 (2011)
6. Deng, H.: Comparing and ranking fuzzy numbers using ideal solutions. Applied Mathematical Modelling **38**(5), 1638–1646 (2014)
7. Dua, D., Graff, C.: Uci machine learning repository (2017)
8. Dubois, D., Foulloy, L., Mauris, G., Prade, H.: Probability-possibility transformations, triangular fuzzy sets, and probabilistic inequalities. Reliable Computing **10**(4), 273–297 (2004)
9. Dubois, D., Prade, H., Sandri, S.: On Possibility/Probability Transformations, pp. 103–112. Springer Netherlands, Dordrecht (1993)
10. Dunn, J., Mingardi, L., Zhuo, Y.D.: Comparing interpretability and explainability for feature selection. CoRR **abs/2105.05328** (2021)
11. Hall, M.: Correlation-based feature selection for machine learning. Department of Computer Science **19** (06 2000)
12. Hong, D.H., Kim, K.T.: An easy computation of min and max operations for fuzzy numbers. Journal of Applied Mathematics and Computing **21**(1/2), 555–561 (2006)
13. Kim, J., Kang, J., Sohn, M.: Ensemble learning-based filter-centric hybrid feature selection framework for high-dimensional imbalanced data. Knowledge-Based Systems **220**, 106901 (2021)
14. Kuncheva, L.I.: A stability index for feature selection. In: Proceedings of the 25th Conference on Proceedings of the 25th IASTED International Multi-Conference: Artificial Intelligence and Applications. p. 390–395. AIAP'07, ACTA Press, USA (2007)
15. Li, J., Cheng, K., Wang, S., Morstatter, F., Trevino, R.P., Tang, J., Liu, H.: Feature selection: A data perspective. ACM Comput. Surv. **50**(6) (2017)
16. Robnik-Sikonja, M., Kononenko, I.: Theoretical and empirical analysis of relieff and rrelieff. Machine Learning **53**, 23–69 (2003)
17. Roffo, G., Melzi, S., Castellani, U., Vinciarelli, A., Cristani, M.: Infinite feature selection: A graph-based feature filtering approach. IEEE Transactions on Pattern Analysis and Machine Intelligence **0**, 1–1 (07 2020)
18. Sammut, C., Webb, G.I. (eds.): Encyclopedia of Machine Learning and Data Mining. Springer (2017)
19. Scoppola, B., Troiani, A.: Gaussian mean field lattice gas. Journal of Statistical Physics **170**(6), 1161–1176 (2018)
20. Shen, Z., Chen, X., Garibaldi, J.M.: A novel weighted combination method for feature selection using fuzzy sets. In: 2019 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE). pp. 1–6. IEEE (2019)
21. Shen, Z., Chen, X., Garibaldi, J.M.: A fuzzy aggregation based ensemble framework for accurate and stable feature selection. In: 2021 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE). pp. 1–7 (2021)
22. Yager, R.R.: A procedure for ordering fuzzy subsets on the unit interval. Information Sciences **24**, 143–161 (1981)