Applied Soft Computing 111 (2021) 107689

Contents lists available at ScienceDirect

Applied Soft Computing

journal homepage: www.elsevier.com/locate/asoc

Review article

# Impact of the learners diversity and combination method on the generation of heterogeneous classifier ensembles

M. Paz Sesmero, José Antonio Iglesias [*], Elena Magán, Agapito Ledezma, Araceli Sanchis

*Computer Science Department, Universidad Carlos III de Madrid, Spain*

ARTICLE INFO

*Article history:*
Received 5 October 2020
Received in revised form 23 April 2021
Accepted 2 July 2021
Available online 15 July 2021

*Keywords:*
Ensemble of classifiers
Multiclass classification task
Labelling noise
Ensemble diversity

ABSTRACT

Ensembles of classifiers is a proven approach in machine learning with a wide variety of research works. The main issue in ensembles of classifiers is not only the selection of the base classifiers, but also the combination of their outputs. According to the literature, it has been established that much is to be gained from combining classifiers if those classifiers are accurate and diverse. However, it is still an open issue how to define the relation between accuracy and diversity in order to define the best possible ensemble of classifiers. In this paper, we propose a novel approach to evaluate the impact of the diversity of the learners on the generation of heterogeneous ensembles. We present an exhaustive study of this approach using 27 different multiclass datasets and analysing their results in detail. In addition, to determine the performance of the different results, the presence of labelling noise is also considered.

**Contents**

[*] Corresponding author.
*E-mail addresses:* msesmero@inf.uc3m.es (M.P. Sesmero), jiglesia@inf.uc3m.es (J.A. Iglesias), emagan@inf.uc3m.es (E. Magán), ledezma@inf.uc3m.es (A. Ledezma), masm@inf.uc3m.es (A. Sanchis).

https://doi.org/10.1016/j.asoc.2021.107689

## 1. Introduction

Machine learning uses the theory of statistics in building mathematical models, because the core task is making inferences from a sample [1]. Three major branches of machine learning are supervised, unsupervised and reinforcement learning. The goal of supervised learning is to create a model of the distribution of class labels in terms of features. This model is called classifier and it can be used to assign a class label to those instances in which this value is unknown [2]. Despite the variety and number of models that have been proposed, including artificial neural networks [3], decision trees [4], inductive logic programming [5], and Bayesian learning algorithms [6], the construction of a perfect classifier for any given task remains unobtainable [7].

The strategy of combining different classification models has attracted the interest of the Machine Learning Community. This strategy is known as mixture of experts, ensemble methods or ensemble of classifiers [8]. An ensemble of classifiers consists of a set of classifiers, named base learners, whose individual decisions are combined in some way to classify new examples [9]. According to Sagi [10], the reasons why ensemble methods often improve predictive performance are: avoid the overfitting, decrease the risk of obtaining a local minimum, and extend the hypothesis search space.

Similar to what happens when a human being has to make an important decision, ensembles of classifiers are based on the idea that decisions made collectively are more reliable than those based on a single opinion. In this sense, an ensemble of classifiers combines the results of a set of individual classifiers to obtain a more reliable result [9].

Several theoretical studies have demonstrated that the success of any ensemble of classifiers is related to the accuracy and diversity of the members of the ensemble [11]. Thus, an ensemble of classifiers could improve the accuracy of any of its individual members if they have a low error rate (are accurate) and their errors are not coincident (are diverse). However, obtaining base learners which satisfy both requirements simultaneously is not an easy task because the lower the number of errors, the higher its correlation is [12,13].

Generating an ensemble involves:

- the selection of a methodology for training and selecting the members of the ensemble (base learners),
- the choice of a method for combining their outputs.

The techniques used to generate a pool of base learners that are both accurate and diverse are based on the idea that the hypothesis of a classifier depends on both the training data and the learning algorithm that are used to generate each classifier. When base learners are obtained using different learning algorithms, ensembles are called heterogeneous. On the other hand, when the base learners are generated using a single learning algorithm and therefore the main sources of diversity are the modification of the training set and/or the use of different versions of the same learning algorithm, ensembles are called homogeneous.

The homogeneous ensembles include systems such as *Bagging* [14] or *Boosting* [15] where the base classifier are generated using different training datasets. Other systems categorized as homogeneous ensembles are the Random Subspace Selection [16] where each base classifier is created by using different feature sub-spaces, or systems as the proposed in [17] and [18], where each member of the pool is generated using different variants of the same learning algorithm.

On the other hand, the heterogeneous ensembles include systems as *Stacking* [19] and most of its variants [20] where the members of the pool are generated from different inducers, such as artificial neural networks, decision trees, support vector machines, Bayesian models, and so on. In this paper, it is studied the impact of the diversity among the learners in heterogeneous ensembles.

As was noted before, the second key aspect in the design of ensembles of classifiers is to define the strategy for combining the outputs of the base learners into a single output. In this sense, depending on the policy used to combine these outputs, the combination strategies proposed in the literature can be grouped into two techniques: fusion and selection.

- In classifier fusion, the decisions of all base classifiers are involved in the final ensemble decision. It is therefore a cooperative and competitive combination strategy, since all the decisions are involved and some decisions prevail over others. The decisions from all members can be combined using simple mathematical functions, such as the average, majority vote or weighted majority vote, or more complex techniques, such as meta-classifiers. A meta-classifier is a classifier trained to combine the outputs of the different base classifiers. Stacking [21] is perhaps the best known method that introduces the concept of a meta-classifier.
- In classifier selection, it is assumed that each classifier is an expert in some local region of the space [22]. Therefore, a new instance is classified by the decision of a single classifier. Depending on whether the region of competence of a classifier is defined during the training phase or during the qualification phase, selection techniques are divided into static or dynamic [23].

In light of this background, this paper is focused on the study of the diversity as method to select base learners of an heterogeneous ensemble, and the influence of the combination method on the accuracy of the ensemble.

## 2. Related works

Ensembles of classifiers is a proven approach in machine learning with a wide variety of research works. In this section, some related research works are described. In addition, since the combination of the different classifiers is essential in these methods, several research works about this issue are presented.

### 2.1. Ensembles of classifiers — applications

The first research works related to ensembles of classifiers date from several decades ago. In the 1990s, important ensemble methods such as *Bagging* [14], *Boosting* [15,24,25] or *Stacking* [19] were proposed. Those research works opened the door to a very promising approach in machine learning: ensembles of classifiers. In [26] a complete review of *Bagging* and *Boosting* methods and a large empirical study comparing several variants are presented. In 1997, Dietterich detailed in [27] that machine-learning research was making great progress in four different directions: (1) the improvement of classification accuracy by learning ensembles of classifiers, (2) methods for scaling up supervised learning algorithms, (3) reinforcement learning, and (4) the learning of complex stochastic models. It is remarkable that the first of those directions is related to the ensemble of classifiers. From those first research works, the interest in ensembles of classifiers has increased and nowadays, ensembles of classifiers are used in a wide range of environments and applications, as is detailed below.

Imbalance classification is a challenging research task in machine learning. Imbalanced datasets may negatively impact the predictive performance of most classical classification algorithms. In order to tackle this problem, many different methods based on ensembles of classifiers have been proposed [28]. In this sense, a study proposed in [29] conducts a *Bagging* based ensemble method to overcome the problem of class imbalance. The purpose of that research is to see the ability of some *Bagging* based ensemble methods on overcoming the class imbalance problem. In [30] is proposed a method for classifying five groups of imbalanced heartbeats. In that work, re-sampling techniques and *AdaBoost* ensemble classifier are used. In [31] is presented a method for the automatic classification of electrocardiograms (ECG) based on the combination of multiple Support Vector Machines and using a database highly imbalanced. Recently, it is proposed a new ensemble-based method, named Ensemble of Classifiers based on Multiobjective Genetic Sampling for Imbalanced Classification (E-MOSAIC), to deal with imbalanced multiclass classification tasks [32].

However, although the application of ensembles can deal with imbalanced classification problems, there are many other fields in which ensembles methods have been applied with good results.

In the intrusion detection field, there are several research works which tackle this problem by using different ensemble of classifiers and hybrid techniques [33]. In [34] is proposed a novel ensemble construction method that uses particle swarm optimization generated weights to create an ensemble of classifiers for intrusion detection. In [35] a novel framework to effectively and efficiently detect malicious apps and categorize benign apps is proposed. This framework is based on ensemble of classifiers and the experimental results show that it is more robust than the five base classifiers in the detection and categorization. The aim of [36] is to identify the critical features required in the construction of an intrusion detection model. The proposed model utilizes an approach based on ensembles of classifiers with minimum complexity to overcome the issues in the existing ensemble-based intrusion detection model.

In relation with the computerized detection of Alzheimer's disease, an approach [37] identifies persons with Alzheimer's disease using an ensemble of classifiers with Multi-Layer Perceptron (MLP), Support Vector Machine (SVM) and J48. In this field, in [38] is detailed the ability of an ensemble of machine learning models to implement classification strategies to discriminate among mild cognitive impairment, Alzheimer's disease and Cognitive Unimpaired. In [39] is proposed an ensemble framework to diagnose diabetes mellitus by optimally employing multiple classifiers based on *Bagging* and random subspace techniques. New advanced methods of image description and an ensemble of classifiers for recognition of mammograms in breast cancer are presented in [40].

In addition, ensembles of classifiers have been applied in other very different fields. In [41] is presented a sentiment analysis system for automatic recognition of emotions in text, using an ensemble of classifiers. An *Stacking* ensemble method is proposed in [42] for the detection of fake news. In [43] is proposed an activity recognition model which aim to detect the activities by employing ensemble of classifiers techniques using the Wireless Sensor Data Mining (WISDM).

Finally, in [44] visual analytic tools are presented to support a specialist user in interpreting the behaviour of an ensemble of classifiers and its underlying models.

### *2.2. Selecting the members of the ensemble*

A key point in the design of an ensemble of classifiers is the generation of the pool of classifiers that make up the ensemble. As was noted before, a necessary condition to obtain an accurate ensemble is that the base learners are both accurate and diverse.

Although most of the proposed models ensure diversity among base classifiers in an implicit way [45], there are some research works where diversity plays a key role in the design of the ensemble.

In [46] a method for generating ensembles of classifiers that emphasizes diversity among the ensemble members is proposed. In that research, the experimental results prove that accuracy of those ensembles based on diversity is higher than those ensembles based on the error rate. However, unlike our approach, the diversity in that study derives from using different feature subsets.

In [47], it is described an approach in which the members of the ensemble are selected based not only on the accuracy but also on diversity. For that purpose, the authors use a Multi Objective Evolutionary Algorithm in which the base learners are generated using *Bagging* and manipulating the input features. The results show that the generated multiple classifier ensembles outperform single classifiers. In that approach, the diversity is measured using Coincident Failure Diversity, Disagreement and Hamming Distance. In the approach proposed in this paper, we are using six different diversity measures (Section 3.2.1): *Q statistic* ($Q$)*, Correlation Coefficient* ($\rho$)*, Double fault measure (DF), Plain Disagreement measure (dis), Kappa-degree-of-agreement statistic* ($\kappa$)*, Ambiguity (amb).*

A method based on genetic algorithms is proposed in [48] for searching base learners that optimize not only the accuracy but also the diversity. As previous work, it is concluded that combinations of measures often resulting in better performance than a single measure. However, the goal of our proposal is the use of exhaustive search.

In [49] is analysed the efficiency of five diversity measures (plain disagreement, fail/non fail disagreement, Q statistic, correlation coefficient and kappa statistic) applied in the selection of feature subsets that promote the greatest disagreement among the base classifiers. In that research, it is quantified the correlation between each diversity measure and both the ensemble accuracy and the average accuracy of the base classifiers. The best correlations were shown by using the plain disagreement measure and the fail/non-fail disagreement measure. In our work, the diversity measures are used to select the subset of classifiers with the highest diversity value.

Finally, in [50] is studied in detail the relationships between different classifiers combination methods and several diversity measures. That proposal concludes the use of diversity measures in the design of ensemble classifiers is an open question. Our work is focused on that question.

## 3. Our approach: Base settings

As it was noted in Section 1, the main aim of this paper is to analyse the influence of different diversity measures and different integration methods on the design of heterogeneous ensembles of classifiers. This is not an easy task since many issues have to be taken into account. This section details the choices made in relation to the types of classifiers, the diversity measures and the integration methods used in the study.

### 3.1. Base learners

In order to obtain a pool of complementary base learners, five different types of classifiers have been selected. This selection has been done taking into account that the philosophy and operation of the classifiers should be as different as possible. The different types of classifiers are listed as follows:

1. *Linear Support Vector Machine* (LSVM): This is a binary classifier which finds a hyper-plane such that the margins between the two classes are maximized. The multiclass support is handled according to a one-vs-the-rest scheme [51].
2. *Decision Tree*: This classifier is a tree in which internal nodes are labelled by features.
3. *Neural Networks (Multi-layer Perceptron classifier)*: This model optimizes the log-loss function using stochastic gradient descent.
4. *K-Neighbours Classifier*: This classifier is an instance-based classifier.
5. *Gaussian Naïve Bayes*: This classifier uses a probabilistic approach.

### 3.2. Diversity measures

Diversity among the members of a set of classifiers is a key issue in classifier combination. However, measuring diversity is not straightforward because there is no generally accepted formal definition, and the diversity of an ensemble of classifier can be calculated in many ways [49,52]

In this approach, six different diversity measures have been selected. As in the selection of learners, this collection has been chosen in order to get a wide variety of measures that capture different properties. The selected measures can be divided in pairwise and non-pairwise measures, and they are listed below:

#### 3.2.1. Pairwise measures

These measures attempt to establish the diversity that exists between the predictions of two classifiers. Hence, when the set is made up of three or more base classifiers the total diversity is given by the average of the measures on all classifier pairs.

Since two classifiers are considered diverse when wrong decisions are made on different examples, it seems clear that the degree of diversity between two classifiers, $C_i$ and $C_j$, must be a function of:

- $N$: Number of examples.
- $N^{ab}$: Number of examples correctly classified (a = 1) or erroneously classified (a = 0) by the $C_i$ classifier, and correctly classified (b = 1) or erroneously classified (b = 0) by the $C_j$ classifier.

Based on this nomenclature, the pairwise measures, that are used in our approach, are listed and mathematically defined below:

- *Q statistic (Q)*: This measure quantifies the diversity between two classifiers analysing if both classifiers tend to correctly classify the same examples, or to commit errors on different patterns. This statistic is defined in Eq. (1).

$$Q_{ik} = \frac{N^{11}N^{00} - N^{01}N^{10}}{N^{11}N^{00} + N^{01}N^{10}} \tag{1}$$

- *Correlation Coefficient (ρ)*: This measure estimates quantitatively the relationship between the successes and errors made by two classifiers. Mathematically, this relationship is computed in Eq. (2).

$$\rho_{ij} = \frac{N^{11}N^{00} - N^{01}N^{10}}{\sqrt{(N^{11}+N^{10})(N^{01}+N^{00})(N^{11}+N^{01})(N^{10}+N^{00})}} \tag{2}$$

- *Double fault measure (DF)*: This measure, defined in Eq. (3), quantifies the relation between the wrongly classified examples by both classifiers and the total number of training examples.

$$DF_{ij} = \frac{N^{00}}{N^{11}+N^{00}+N^{01}+N^{10}} \tag{3}$$

In addition to this three measures which are based on the number of hits and errors for each pair of classifiers, two additional measures have been selected. These two measures are described below:

- *Plain Disagreement measure (dis)*: This measure quantifies the relation between the number of times the base classifiers assign the same class, and the total number of examples. It is defined in Eq. (4).

$$dis = \frac{1}{N}\sum_{k=1}^{N} Is(C_i(x_k) \neq C_j(x_k)) \tag{4}$$

where: $C_i(x_k)$ is the class assigned by classifier $i$ to the instance $k$. $Is()$ is a truth predicate.
- *Kappa-degree-of-agreement statistic,(κ)*: If $N_{ij}$ is the number of examples to which the first classifier assigns class $i$ and the second classifier assigns class $j$ and $N$ indicates the total of examples, then this measure is defined as in Eq. (5).

$$\kappa = \frac{\Theta_1 - \Theta_2}{1 - \Theta_2} \tag{5}$$

where: $\Theta_1 = \frac{\sum_{i=1}^{l} N_{ii}}{N}$ is the probability that both classifiers agree on their decisions, and $\Theta_2 = \sum_{i=1}^{l}(\frac{N_{i*}}{N}\frac{N_{*i}}{N})$ is a correction factor that estimates the probability of both classifiers matching in their decisions by chance.

It is worth mentioning that when the ensemble is made up of three or more base classifiers and diversity quantified using some pairwise measure, the total ensemble diversity will be calculated by averaging the dual values, as given in Eq. (6).

$$Diversity_{Ensemble} = \frac{2}{l(l-1)}\sum_{i=1}^{l-1}\sum_{j=i+1}^{l} Diversity_{i,j} \tag{6}$$

where: $l$ is the number of base learners in the corresponding ensemble.

#### 3.2.2. Non-pairwise measures

The aim of these measures is to estimate the diversity of the set of classifiers by considering it as a whole. This category includes, among others, the following measure:

- *Ambiguity (amb)*: The idea behind this measure is that a classification problem in which the examples belong to $K$ classes can be interpreted as $K$ regression problems. Therefore, the diversity of a set composed of $L$ base classifiers can be calculated by averaging the ambiguity of each example over the different regression problems. This measure is defined in Eq. (7).

$$amb = \frac{1}{LNK}\sum_{l=1}^{L}\sum_{n=1}^{N}\sum_{k=1}^{K}(Is(C_l(x_n)) = k) - \left(\frac{N_k^n}{L}\right)^2 \tag{7}$$

where: $N_k^n$ is the number of base learners that assign class $k$ to the example $x_n$, $C_l(x_n)$ is the class assigned by the classifier $l$ to the example $x_n$, and $L, N$ and $K$ are the total amount of base classifiers, examples and classes, respectively.

### *3.3. Combination methods*

Once the base classifiers that are comprising the ensemble have been fixed, the next step is to establish a procedure through which the individual decisions are combined to obtain a final hypothesis. There are several ways of combining the output of the base classifiers, from simple methods, such as majority voting, to more complex methods, such as meta-classifiers [19]. Since the combination method can affect the performance of the ensemble, in this research we analyse the impact of six different combination methods on the ensemble performance.

The selected combination methods can be divided in three categories according to the main algorithm used: simple majority voting, weighted majority voting, or use of a meta-classifier. The six combination methods are explained below:

#### *3.3.1. Methods based on simple majority voting*

Methods based on simple majority voting combine the outputs of the base classifiers in the simplest possible way, by performing a voting among the base classifiers without using any external elements or information. From this category, the following combination method is used:

- *Majority (unweighted) voting (MVOT)*: The ensemble takes a decision by counting how many base classifiers vote each class and selecting the most voted class. In the event of a tied vote, the ensemble will select randomly one of the most voted classes. Although this is a very simple method, according to the literature, unweighted vote is robust [53].

#### *3.3.2. Methods based on weighted majority voting*

In this case, the process is similar to the unweighted voting, since the ensemble adds the votes of the base classifiers and chooses the most voted class. However, instead of all votes having equal value, in a weighted majority voting the votes of each base classifier can have different values according to their reliability. In this work, the weights that modify the value of the votes are based on the evaluation measures of accuracy and precision that are described below:

- Global accuracy ($accuracy_C$): Samples correctly classified by the classifier $C$ divided by the total number of samples.
- Accuracy per class ($accuracy_{Ci}$): Samples of the class $i$ correctly classified by the classifier $C$ divided by the number of samples of the class $i$.
- Precision ($precision_{Ci}$): Samples of the class $i$ correctly classified by the classifier $C$ divided by the number of samples that the classifier has classified as class $i$.

According to this definition, the combination methods based on weighted majority voting that are used in our approach, are listed below:

- *Weighted voting using global accuracy (WVGA)*: The value of the weighted vote of a base classifier $C$ is $accuracy_C$.
- *Weighted voting using class accuracy (WVCA)*: The value of the weighted vote of a base classifier $C$ is $accuracy_{Ci}$, where $i$ is the class predicted by $C$.
- *Weighted voting using class precision (WVCP)*: The value of the weighted vote of a base classifier $C$ is $precision_{Ci}$, where $i$ is the class predicted by $C$.

#### *3.3.3. Methods based on meta-classifiers*

In this case, the predictions of the base classifiers are given to a meta-classifier which predicts the final output of the ensemble. This meta-classifier is trained on the training phase, so that it has the real class of the samples and it can learn accordingly. There are two methods based on meta-classifiers that are used in our approach, and both of them use a Gaussian Naïve Bayes classifier. The two meta-classifiers differ in the input that is provided to them, as explained below:

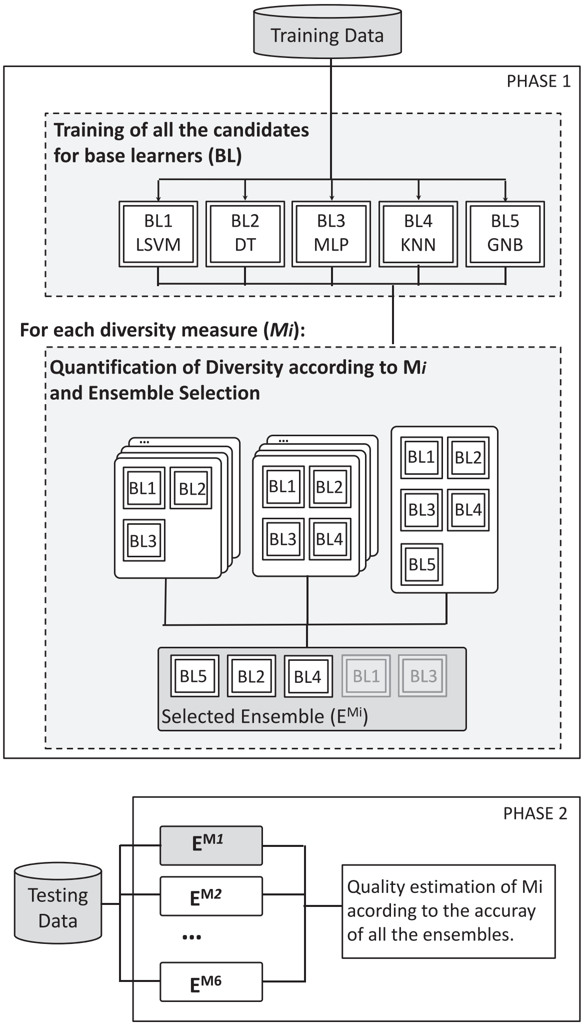
Training Data
PHASE 1
Training of all the candidates for base learners (BL)
BL1 LSVM
BL2 DT
BL3 MLP
BL4 KNN
BL5 GNB
For each diversity measure (Mi):
Quantification of Diversity according to Mi and Ensemble Selection
...
BL1
BL2
BL3
BL4
BL5
Selected Ensemble (E^Mi)
PHASE 2
Testing Data
E^M1
E^M2
...
E^M6
Quality estimation of Mi acording to the accuray of all the ensembles.

**Fig. 1.** Architecture of our approach.

- *Bayesian meta-classifier using predictions (PBAY)*: The meta-classifier receives samples formed only by the predictions of the classifiers.
- *Bayesian meta-classifier using original data and predictions (A+PBAY)*: The meta-classifier receives samples formed by a combination of the real sample, with its original attributes, and the predictions of the classifiers, that are introduced as new attributes.

## 4. Our approach: Designing the comparison architecture

In this section, it is explained in detail the proposed architecture to perform the comparative study. Fig. 1 shows the two phases in which this approach has been divided:

In the first phase, the selection of the base learners than make up the ensemble is performed. As it was detailed in the previous sections, this selection is done according to the diversity among the implemented classifiers (learners). Given that diversity among learners is quantified using six different diversity measures, the result of this phase is six different ensembles (one per measure).

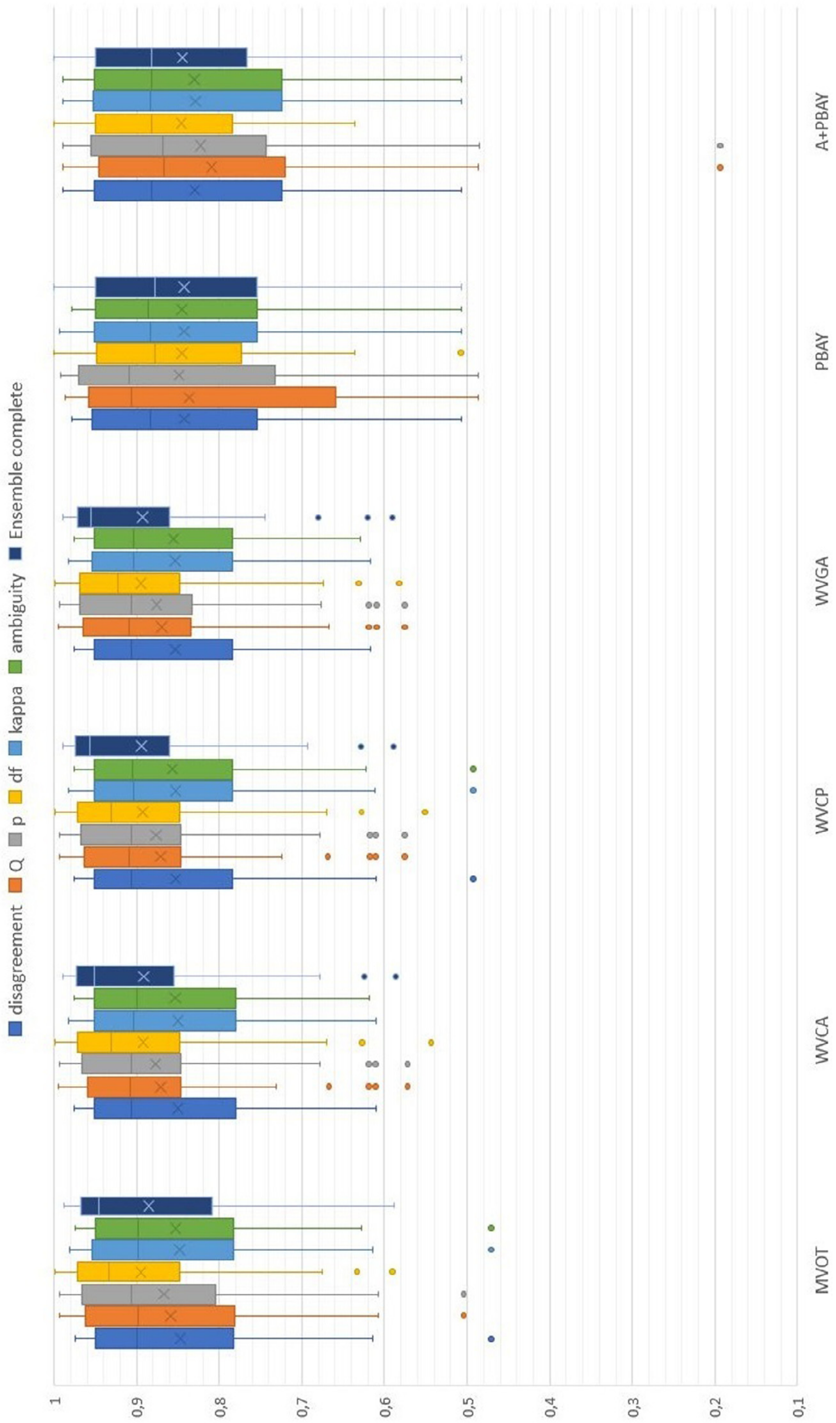

**Fig. 2.** Experimental Results. Accuracy of the different ensembles using different diversity measures and combination methods.

Then, in the second phase, it is estimated the quality of each measure according to the accuracy of all the ensembles. These phases are detailed in the following subsections.

### *4.1. Phase 1 — learners selection according to different diversity measures*

Phase 1 consists on both the generation of base learners, and the selection of the best pool of base learners according to different diversity measures. The process that will be followed to achieve these goals is summarized in Algorithm 1, and will be further explained in this section. To obtain all the different combinations, we use a procedure called *GenerateCombinations* (line 11, Algorithm 1), which is explained in Algorithm 2.

As shown in Fig. 1, the input of this phase is the training data, and it is divided in two different steps that are described as follows:

#### *4.1.1. Training of all the candidates for base learners (l):*

In this step, the five different classifiers that have been defined in Section 3.1 are trained. As it has been previously explained, all the classifiers are trained using the same dataset. Thus, the diversity among learners is obtained by using learners that have been implemented by applying different learning algorithms.

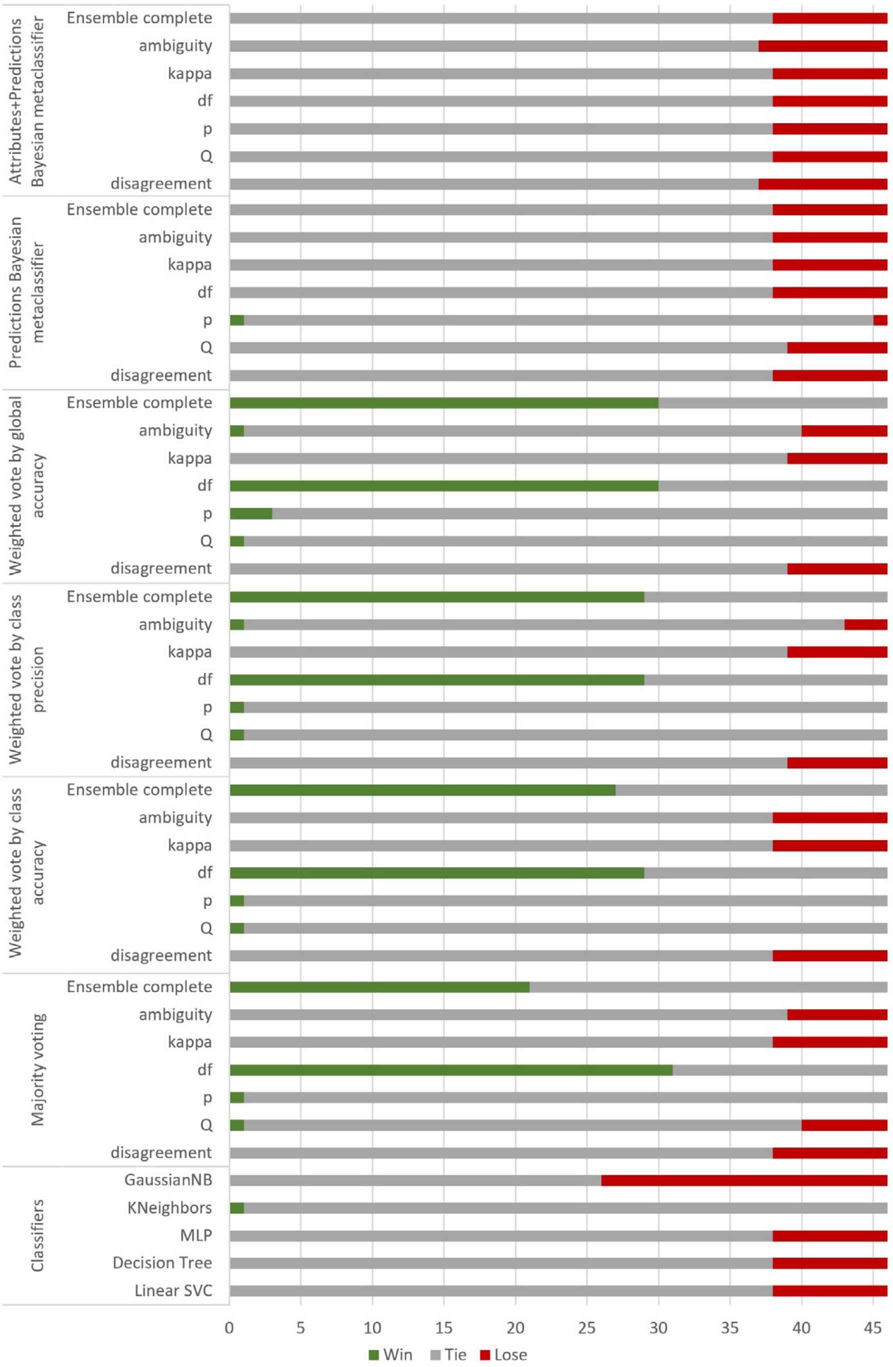

**Fig. 3.** Experimental Results. Friedman and Nemenyi Test — One vs all.

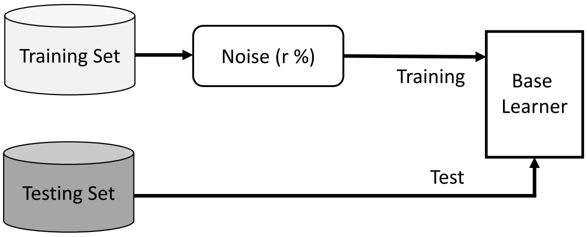

**Fig. 4.** Noise Injection Process.

Once the classifiers have been trained, the diversity among them will be quantified using the six diversity measures considered in this research.

#### 4.1.2. Diversity analysis and ensemble generation:

This step is repeated for each of the six diversity measures used in this proposal and detailed in Section 3.2. The inputs of this step are the learners trained in the previous step. From these pool of learners, several ensembles are generated. In this proposal, the number of possible ensembles to be evaluated depends on to the number of learners ($L$). In this sense, the number of created ensembles will be the combinations without repetition of $L$ elements taken $l$ by $l$.

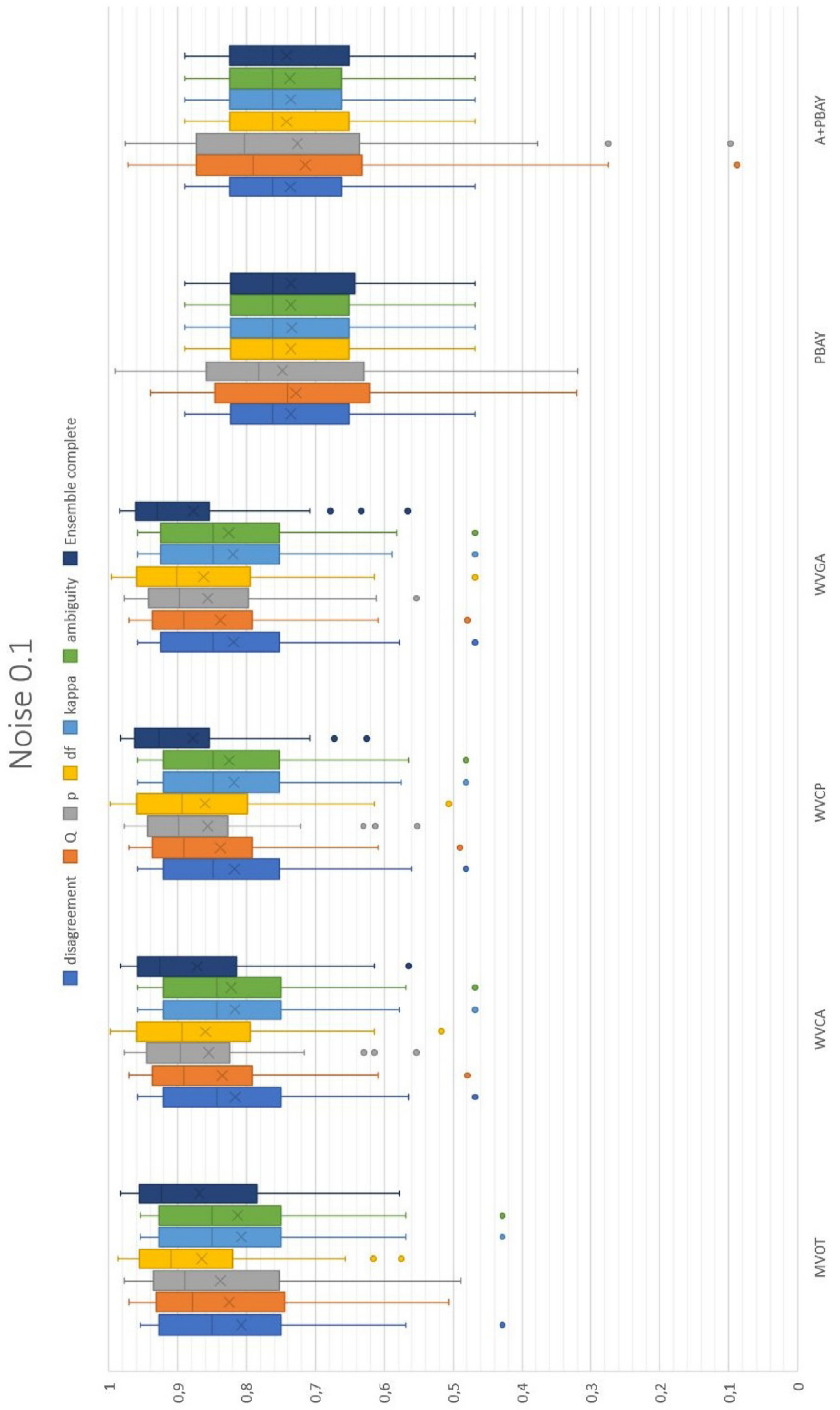

**Fig. 5.** Experimental Results. Accuracy of the different ensembles using different diversity measures and combination methods — Noise 10%.

Since the minimum number of base learners in an ensemble is three and the maximum is $L$, the value of $l$ will be all the values from three to $L$.

For mathematical reasons, ensembles of two classifiers are not considered since, in most of the cases, the pairwise ensembles would be chosen. The equation to define how many combinations without repletion of $L$ elements taken $l$ by $l$ is given in Eq. (8).

$$\sum_{l=3}^{L}\binom{L}{l}=\sum_{l=3}^{L}\frac{L!}{L!(l-L)!} \tag{8}$$

Fig. 1 helps to understand the idea behind the creation of the different ensembles.

Note that in Fig. 1, the number of learners is five ($L$=5) so the number of different ensembles ($N\_Ens_L$) can be calculated as follows:

$$N\_Ens_5=\binom{5}{3}+\binom{5}{4}+\binom{5}{5}=10+5+1=16 \tag{9}$$

Once all the possible ensembles have been designed, it is needed to quantify their diversity. As it was explained in Section 3.2, in this study diversity is quantified using six different

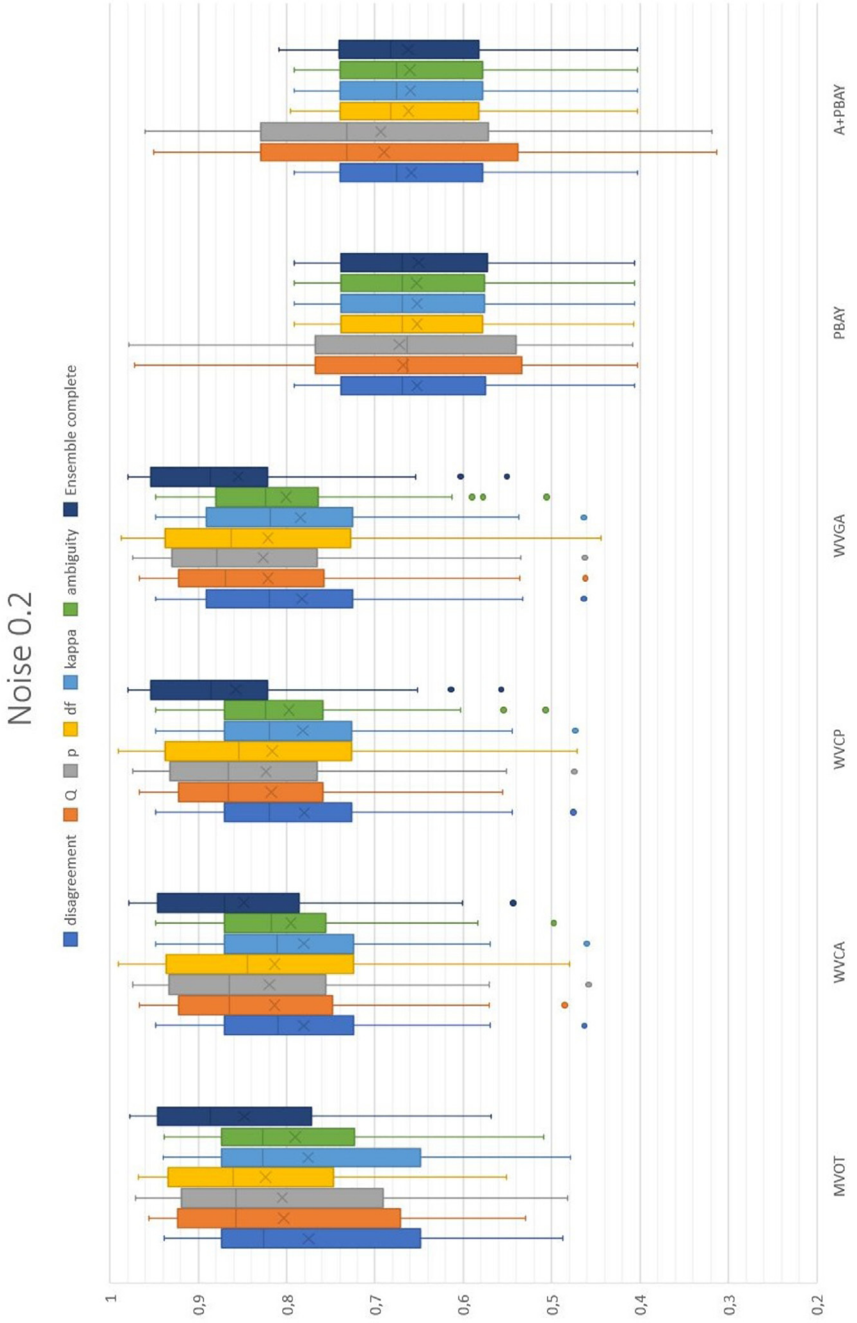

**Fig. 6.** Experimental Results. Accuracy of the different ensembles using different diversity measures and combination methods — Noise 20%.

measures: Q statistic (Q), Correlation Coefficient($\rho$), Double Fault measure (DF), Plain Disagreement measure (dis), Kappa-degree-of-agreement statistic($\kappa$) and Ambiguity (amb).

After quantifying the diversity of all the possible ensembles using a specific measure, the most diverse ensemble per measure is selected. Given that we are using six measures, at the end of this phase six different ensembles will be chosen. As it is observed in the proposed configuration (Fig. 1), the selected ensemble could be formed of three, four, or five base learners.

### 4.2. Phase 2 — quality estimation process

Once the members of the six different ensembles have been selected, they will be used to classify the new samples (testing data). After this, we will evaluate their accuracy to select the best ensemble. To do this, we will follow Algorithm 3.

In order to get the final classification, a combination method needs to be defined. As it was mentioned Section 3.3, in this approach, six different combination methods are applied and evaluated independently. However, the combination method could be changed without changing the idea behind this architecture.

The final goal of this phase (and mainly of this research) is to analyse the relation among the diversity measures and the combination methods on the accuracy of the ensemble. Thus, it is needed to determine if any of the diversity measures or any of the combination methods, are directly related with the generation of the most accurate ensemble.

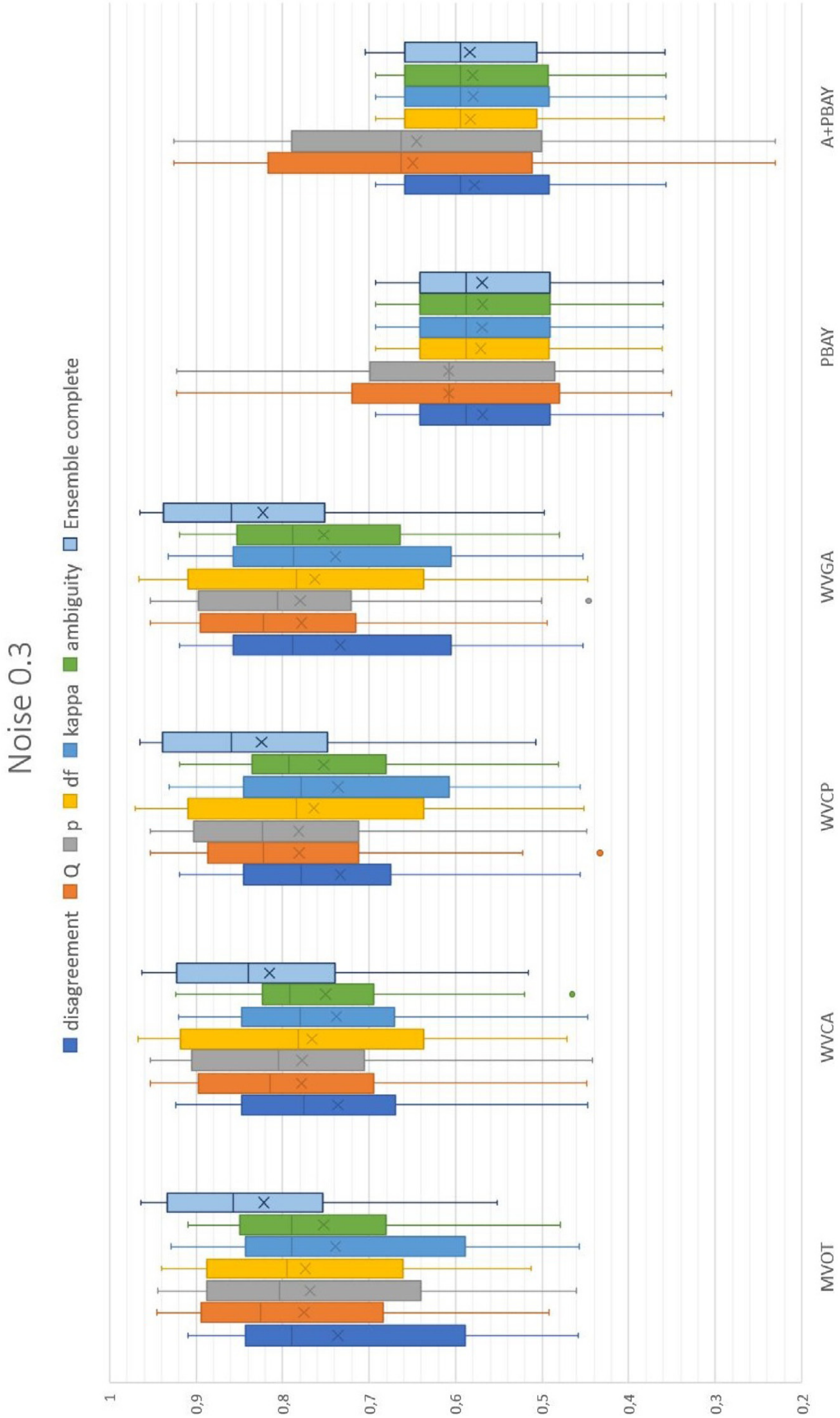

**Fig. 7.** Experimental Results. Accuracy of the different ensembles using different diversity measures and combination methods — Noise 30%.

With this purpose, the Friedman test will be applied. This non-parametric test will be used to analyse if the differences among the different ensembles generated (by applying different diversity measures and different combination methods) are statistically significant.

In case of statistically significant differences, a post hoc test well be applied to identify the ensembles that actually differ. For this purpose, the Nemenyi test will be applied, which will allow us to discover if the differences obtained as a result of the Friedman test are, indeed, significant. The formulation of these tests is detailed in Section 5.2.

## 5. Experimental setup and results

In this section, a detailed empirical evaluation of the proposed method is presented. This evaluation is done by using 27 benchmark datasets. In addition, an analysis in which training datasets are affected by labelling noise is presented.

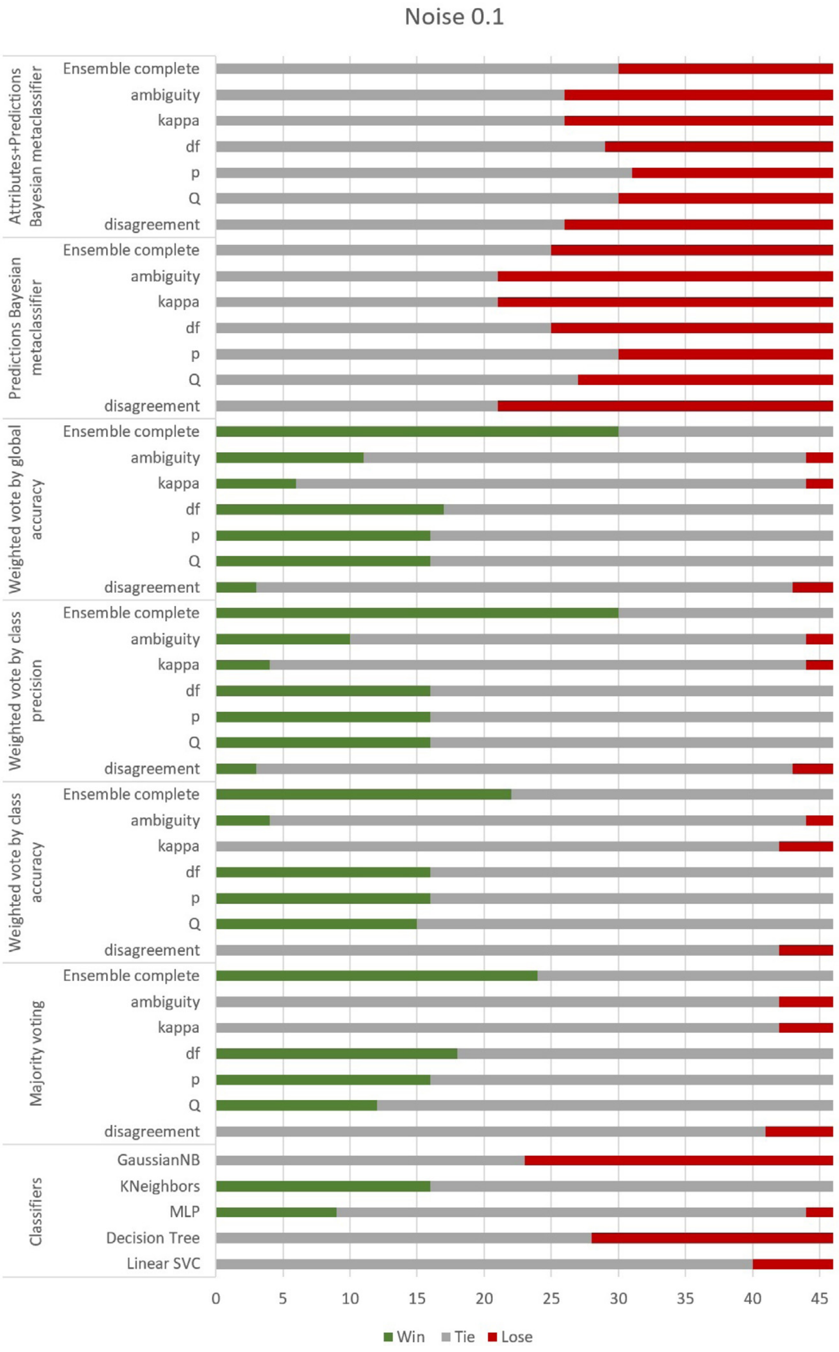

**Fig. 8.** Experimental Results. Friedman and Nemenyi Test — One vs all. Noise 10%.

### 5.1. Datasets

For testing the viability of the proposed method, 27 benchmark datasets from different repositories have been selected. Table 1 compiles the main characteristics of these datasets.

### 5.2. Experimental setup

To assess the predictive performance of the ensembles, the evaluation of this proposal has been carried out following the well-known stratified 10-fold cross validation. The data are randomly shuffled before the cross validation starts and, to prevent biased results, the whole process is performed 10 times. It is noted that all the models are created by using the same folds.

To compare all the ensembles obtained by applying not only the six different diversity measure, but also the six combination methods, a Friedman test [21] is applied. This test is an extension of the binomial sign test for two dependent samples to a design involving more than two dependent samples. The goal is to evaluate if in a set of $k$ dependent samples, there is at least two samples which represent different populations. In our experimental study, a sample is composed by the accuracies of the ensembles obtained by using a specific diversity measure and a specific combination method.

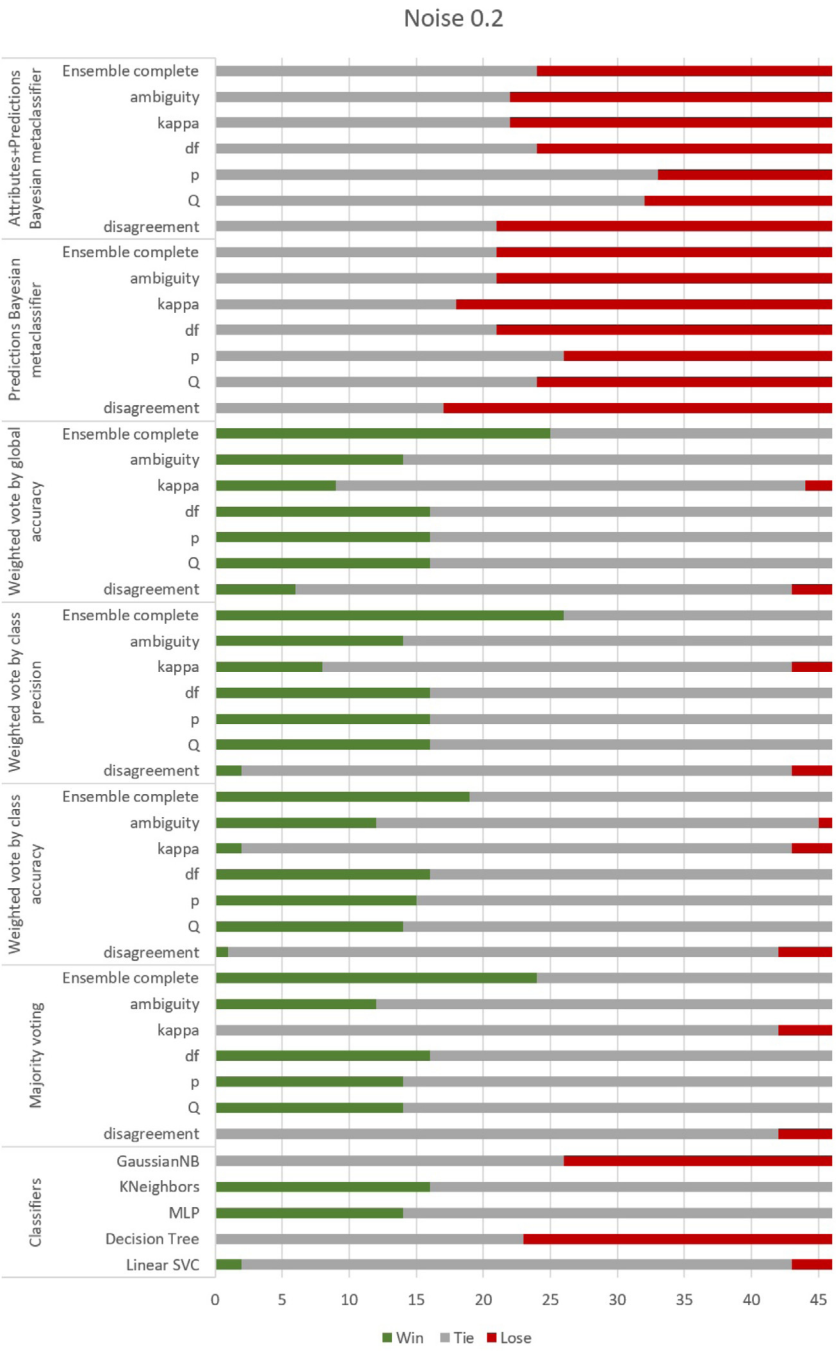

**Fig. 9.** Experimental Results. Friedman and Nemenyi Test — One vs all. Noise 20%.

In addition, we will analyse the results obtained (i) using the complete pool of learners combined with the different six combination methods and (ii) using the individual learners ("ensemble" composed by a single classifier). Thus, the total number of models that are compared in this research is: ((6+1)*6)+5 = 47.

In case of significant differences, a Nemenyi test is applied to compare the accuracy of the ensembles pairwise.

According to Friedman test, classification models (ensembles and simple classifiers) obtained applying $k$ (((6+1)*6)+5) different heuristics on $n$ (27) different datasets are statistically equivalent if the value obtained applying Eq. (10), is less than the tabled critical chi-square value at the pre-specified level of significance with k-1 degree of freedom.

$$\chi_F^2 = \frac{12N}{k(k+1)}\left[\sum_j R_j^2 - \frac{k(k+1)^2}{4}\right] \tag{10}$$

where: $R_j = \frac{1}{N}\sum_i r_i^j\, r_i^j$ is the rank of the $j$th classification model on the $i$th dataset. A rank of 1 is assigned to the measure of diversity which generates the highest accurate ensemble. In the case of tied scores, the average of the ranks is calculated.

For $k = 47$ and $\alpha = 0.05$ the tabled critical chi-square value is 60.83.

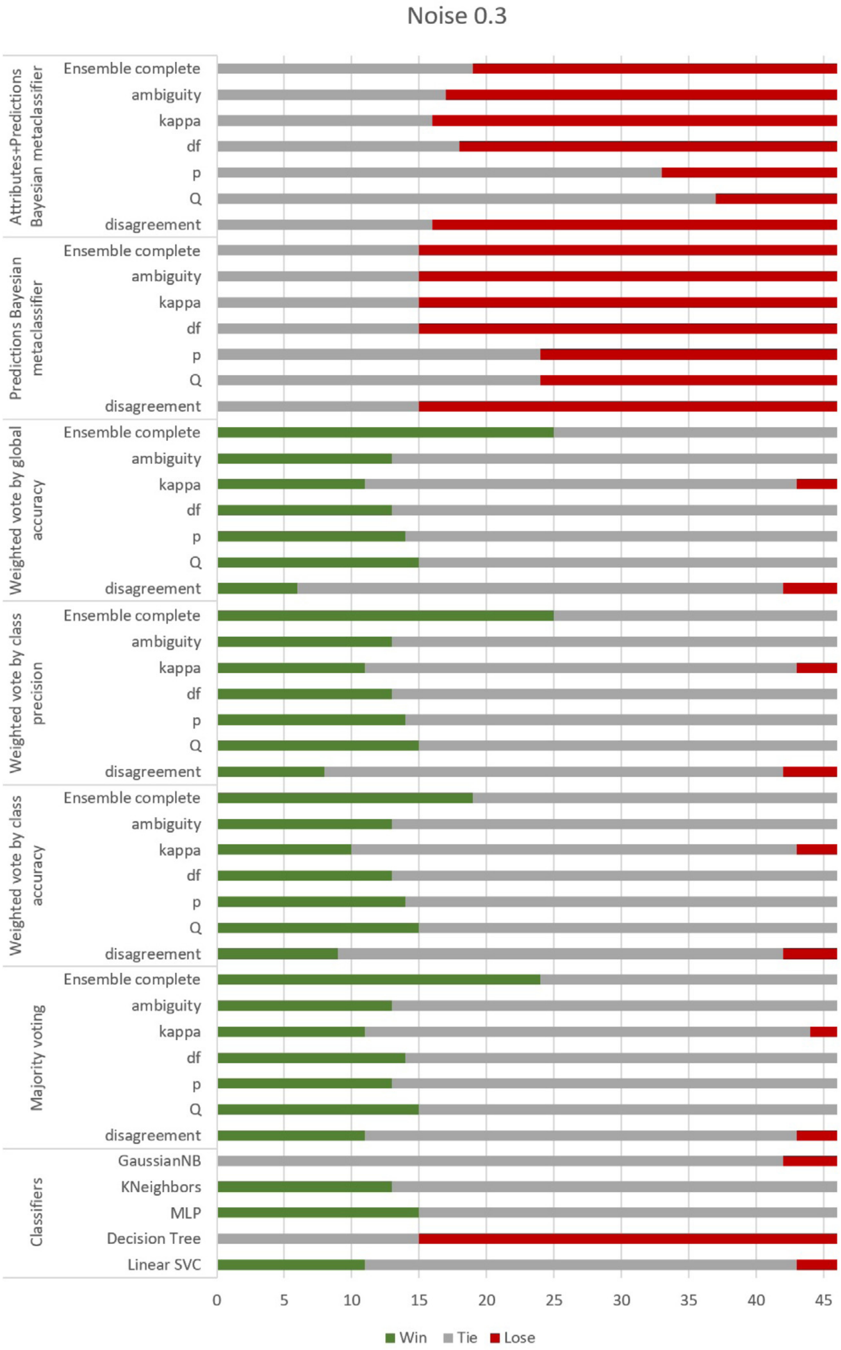

**Fig. 10.** Experimental Results. Friedman and Nemenyi Test — One vs all. Noise 30%.

When the null hypothesis (all evaluated models are equivalent) is rejected, the Nemenyi test will be applied. This post-hoc test will be used to determine if the accuracy differences between the evaluated classification models are statistically significant. To perform this statistical analysis, all the classifiers are compared to each other.

Finally, it is remarkable that all the experimental analyses were implemented using Scikit-learn [59], which is a machine learning library of Python.

### 5.3. Experimental results and discussion

Fig. 2 shows the accuracy rate of the different ensembles generated using a particular diversity measure and a specific combination method. The accuracy rate of the ensembles formed by the five base learners (*ensemble complete*) is also shown.

In addition, all the values of the accuracy rate of the different ensembles are available in http://www.caos.inf.uc3m.es/complementary-material/.

As we can observe in Fig. 2, some of the ensembles stand out thanks to their exceptionally good results. Specifically, ensembles built by using diversity measures $Q$, $\rho$ and $DF$ reach a high accuracy, matching or even improving the accuracy obtained by the combination of all of the individual classifiers, that is by the complete ensemble.

However, to assert if any of these combinations is actually better than the rest of them, we need to apply some statistical tests.

**Algorithm 1** Select diverse ensembles

```
$S$ is the training data
$BL$ is the set of candidates to base learners
$L$ is an integer that specifies the number of base learners ▷ (L=5)
$M$ is the set of measures that will quantify the diversity
$D$ is an integer that specifies the number of diversity measures ▷ (D=6)
procedure GET_ENSEMBLES($S$, $BL_i$, $M_i$)
    for $BL_i$ in $BL$ do                      ▷ (Each type of base learner selected)
        Train($BL_i$, $S$)
    end for
    for l=3,4,...,L do                         ▷ (Gets all combinations of at least 3 elements)
        $C^l$=GenerateCombinations(l, L)
    end for
    for $M_i$ in $M$ do                        ▷ (M={Q, $\rho$, DF, dis, $\kappa$, amb})
        for l=3,4,...L do                      ▷ (Gets ensembles from combinations of BL)
            for k=1,2,...,$\binom{L}{l}$ do
                $Pool^l_k$=$\phi$
                for j=1,2,...,l do
                    $Pool^l_k$=$Pool^l_k$ $\cup$ $BL_{C^l_k(j)}$ where $C^l_k(j)$ is the $j$-th element of $C^l_k$
                end for
                $diversity^{M_i}_{Pool^l_k}$=MeasureDiversity($Pool^l_k$, $M_i$)
            end for
        end for
        $E^{M_i}$=$max_{diversity}(Pool^l_k)$   ▷ (More diverse ensemble according to $M_i$)
    end for
    return SCE={$E^{M_1}$,$E^{M_2}$,...,$E^{M_D}$}
end procedure
```

**Algorithm 2** Generate combinations

```
$L$ is an integer that specifies the number of base learners ▷ (L=5)
$l$ is an integer that specifies the number of ensemble members
procedure GEN_COMBINATIONS(l, L)
    $C^l_1$ = $\phi$
    for i=1,2,...,l do
        $s_i = i$
    end for
    $C^l_1$ = {$s_1$,$s_2$,...,$s_i$}
    for i=2,3,...,$\binom{L}{l}$ do
        m=l
        max_val=L
        while $s_m$=max_val do
            m=m-1
            max_val=max_val-1
        end while
        $s_m$=$s_m$+1
        for j=m+1, m+2,...,l do
            $s_j$ = $s_{j-1}$+1
        end for$C^l_1$={$s_1$,$s_2$,...,$s_i$}
    end for
    return C                                   ▷ (All l-combinations for {1,2,...L})
end procedure
```

**Algorithm 3** Evaluate ensembles

```
$TD$ is the testing data
$C$ is the set of combination methods used to generate the final ensemble decision ▷ (C={MVOT, WVGA, WVCA, WVCP, PBAY, A+PBAY})
$K$ is an integer that specifies the number of combination methods ▷ (K=6)
procedure EVAL_ENSEMBLES($TD$, $SCE$, $M$, $C$)
    for t=1,2,...,T do
        Select $TD_t$ as testing data
        for i=1,2,...,D do
            for $C_k$ in C do
                $E^{M_i}_{C_k}$ = ApplyCombinationStrategy($C_k$,$E^{M_i}$)
                $acc_{ikt}$ = GetAccuracy($E^{M_i}_{C_k}$,$TD_t$)
            end for
        end for
    end for
    if FriedmanTest(acc) is not “All models are equivalent” then
        for each pair of $E^{M_i}_{C_k}$ $\rightarrow$ ($E_a$, $E_b$) do
            comparison = NemenyiTest($E_a$, $E_b$)
            if comparison is “Tied” then
                $result^{tie}_a$ = $result^{tie}_a$ + 1; $result^{tie}_b$ = $result^{tie}_b$ + 1;
            else
                if comparison is “A is significantly better” then
                    $result^{win}_a$ = $result^{win}_a$ + 1; $result^{lose}_b$ = $result^{lose}_b$ + 1;
                else
                    $result^{lose}_a$ = $result^{lose}_a$ + 1; $result^{win}_b$ = $result^{win}_b$ + 1;
                end if
            end if
        end for
    end if
    return result
end procedure
```

So, as it was noted in the previous section, to determine whether there are differences among the analysed models, the Friedman test will be applied. If the null hypothesis of this test (i.e. there is no difference in the performance of the classifiers) is rejected, the Nemenyi test will be applied to every pair of classification models. Applying this post hoc test, it will be possible to known whether the two compared models are equivalent or whether one of them outperforms the competition. If the models are equivalent, a tie will be considered. Otherwise, the outperforming model wins and the other model loses.

This way, all individual classifiers and all ensembles will be compared with the other 46 classification models, and the number of wins, losses and ties will be counted to reveal the best learner. These results are shown in Fig. 3.

According to these statistical values (Fig. 3), we can conclude that the highest accuracy values are obtained when the ensemble is composed of all the base learners, and when the base learners are selected according to the Double fault measure ($DF$). In addition, experimental results show that the accuracy decreases when the outputs of the base learners are combined using a meta-classifier.

In conclusion, it is important to remark that both diversity measures (used to select base learners) and combination methods (used to obtain the final decision of the ensemble) are related to the accuracy of the resulting ensemble. However, the experimental results show that the selection of base classifiers according to Double Fault measure and their combination using any voting method generates the most accurate ensembles.

**Table 1**
Description of the 27 datasets.

| Dataset | N. of instances | N. of attrib. | N. of classes | N. classes: max/min | Imbalance ratio | Source |
|---|---|---|---|---|---|---|
| Iris | 150 | 4 | 3 | 50/50 | 1.0 | [54] |
| Wine | 178 | 13 | 3 | 59/48 | 1.23 | [54] |
| Ionosphere | 351 | 34 | 2 | 225/126 | 1.78 | [54] |
| Letters | 20000 | 16 | 26 | 813/734 | 1.10 | [54] |
| Page Blocks | 5473 | 10 | 6 | 4913/28 | 175.46 | [54] |
| Pen Digits | 10992 | 16 | 10 | 1144/1055 | 1.08 | [54,55] |
| Landsat | 6435 | 36 | 6 | 1533/625 | 2.45 | [54] |
| Segmentation | 2310 | 18 | 7 | 330/330 | 1.0 | [54,56] |
| Shuttle | 58000 | 9 | 7 | 45586/10 | 4558.60 | [54,55] |
| Waveform | 5000 | 21 | 3 | 1667/1666 | 1.00 | [54] |
| Yeast | 1472 | 8 | 10 | 462/5 | 94.20 | [54] |
| Glass | 211 | 9 | 6 | 76/9 | 8.44 | [54,55] |
| Winered | 1599 | 11 | 6 | 681/10 | 68.10 | [54] |
| Vowel | 990 | 12 | 11 | 90/90 | 1.00 | [54,56] |
| Satimage | 6435 | 36 | 6 | 1533/626 | 2.45 | [54,56] |
| Texture | 5500 | 40 | 11 | 500/500 | 1.00 | [54,56] |
| Sensorless | 58483 | 48 | 11 | 5319/5314 | 1.00 | [54,55] |
| Synthetic | 600 | 60 | 6 | 100/100 | 1.00 | [54] |
| OptDigits | 5620 | 64 | 10 | 572/554 | 1.03 | [54,56] |
| Automobile | 159 | 75 | 6 | 48/3 | 3.05 | [54,56] |
| Libras | 360 | 90 | 15 | 24/24 | 1.00 | [54,56] |
| Mfeat-Fac | 2000 | 216 | 10 | 200/200 | 1.00 | [54] |
| Semeion | 1592 | 256 | 10 | 162/155 | 1.04 | [54] |
| Imb. Semeion | 1236 | 256 | 10 | 162/40 | 4.05 | [54,57] |
| Usps | 7291 | 256 | 10 | 1194/542 | 2.20 | [54,55] |
| Mnist | 60000 | 784 | 10 | 6742/5421 | 1.24 | [58] |
| Asistentur | 1006 | 1024 | 9 | 478/22 | 21.73 | [54,57] |

### *5.4. Tolerance to noise*

One of the most important requirements in any classification system is its tolerance to the noise [60]. For this reason, we present a comparison of the labelling noise effect in the ensemble generation using the proposed architecture.

Unlike other proposals [17,61], in this research work, the labelling errors are exclusively induced on the training instances [62]. To clarify this idea, Fig. 4 shows the noise injection process used in this experimentation.

In this experimental phase, we use the previously mentioned 27 benchmark datasets and three rates of noise: $r = 10\%$, $r = 20\%$ and $r = 30\%$.

### *5.5. Results and discussion (tolerance to noise)*

Figs. 5–7 show the accuracy rate of the generated ensembles using the six different diversity measures and the six different combination methods when the training set is corrupted by three different degrees of labelling noise.[1]

In addition, as in the previous section, we apply the Friedman and Nemenyi test in order to statistically analyse all the different ensembles in comparison to each other. The results are shown in Figs. 8–10.

According to these experimental results, the most accurate models are the ensembles obtained using the five base learners and whose final decision is obtained by using weighted vote by class precision (WVCP). It is worth mentioned that the ensembles based on learner selection that offer the highest accuracy values are those which are generated quantifying diversity according to Double Fault Measure and using Majority voting as combination method. However, for a noise level of 30% (Fig. 7), the choice of base classifiers based on $Q$ and $\rho$ values leads to ensembles with slightly higher accuracy than those obtained using DF.

## 6. Conclusion and future works

In this paper, we present an architecture for evaluating the impact of the diversity of the learners and the combination methods on the generation of heterogeneous classifier ensembles. This evaluation is done with and without presence of labelling noise. By exhaustively evaluating this architecture on different datasets, we apply two well-known statistical methods. According to these results, when datasets are free of noise, we can conclude that the best performance is achieved when the base learners are chosen according to the DF measure and combined using a vote mechanism. However, the experimental results obtained when learners are selected according to DF value are similar to those obtained using the pool of all the learners.

In addition, the ensembles with a worse performance are those in which the outputs of base learners are combined using a Bayesian meta-classifier. On the other hand, when the datasets are affected by labelling noise, the most accurate models are those ensembles obtained using the five base learners and combining their outputs using weighted vote by global accuracy (WVGA) and by class precision (WVCP).

In the light of these experimental results and when the number of base learners is relatively low, it seems difficult to conclude that “*many could be better than all*”.

As future work, we will increase the number of learners, and we will extend the study to homogeneous ensembles where diversity is achieved by varying the training datasets. With a different goal, we will focus on the use of criteria that combines not only the diversity but the accurate of the learners. Finally, it would be interesting to evaluate the results of our proposal by taking into consideration not only the impact of the labelling noise but also the size of the datasets.

## Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

## Acknowledgements

This work has been supported under projects PEAVAUTO-CM-UC3M–2020/00036/001, PID2019-104793RB-C31, and RTI2018-096036-B-C22, and by the Region of Madrid’s Excellence Program, Spain (EPUC3M17).

## References

[1] E. Alpaydin, Introduction To Machine Learning, MIT press, 2020.
[2] S.B. Kotsiantis, I.D. Zaharakis, P.E. Pintelas, Machine learning: a review of classification and combining techniques, Artif. Intell. Rev. 26 (3) (2006) 159–190, http://dx.doi.org/10.1007/s10462-007-9052-3.
[3] D.E. Rumelhart, J.L. McClelland, Parallel Distributed Processing: Explorations in the Microstructure of Cognition, Vol. 1: Foundations, MIT Press, 1986.
[4] J.R. Quinlan, Induction of decision trees, Mach. Learn. 1 (1) (1986) 81–106, http://dx.doi.org/10.1023/A:1022643204877.
[5] R.S. Michalski, A theory and methodology of inductive learning, in: R.S. Michalski, J.G. Carbonell, T.M. Mitchell (Eds.), Machine Learning: An Artificial Intelligence Approach, Springer Berlin Heidelberg, Berlin, Heidelberg, 1983, pp. 83–134, http://dx.doi.org/10.1007/978-3-662-12405-5_4.
[6] T.M. Mitchell, Machine Learning, McGraw-Hill, New York, 1997.

[1] All the values of the accuracy rate of the different ensembles are available in http://www.caos.inf.uc3m.es/impact-of-the-learners-diversity-on-the-generation-of-heterogeneous-ensembles/.

[7] R. Ranawana, V. Palade, Multi-classifier systems: Review and a roadmap for developers, Int. Journal of Hybrid Intell. Syst. 3 (2) (2006) 35–61.

[8] T.K. Ho, Multiple classifier combination: lessons and next steps, in: Hybrid Methods in Pattern Recognition, 2002, pp. 171–198, http://dx.doi.org/10.1142/9789812778147_0007, arXiv:https://www.worldscientific.com/doi/pdf/10.1142/9789812778147_0007, URL https://www.worldscientific.com/doi/abs/10.1142/9789812778147_0007.

[9] R. Polikar, Ensemble based systems in decision making, IEEE Circuits Syst. Mag. 6 (2006) 21–45, http://dx.doi.org/10.1109/MCAS.2006.1688199.

[10] O. Sagi, L. Rokach, Ensemble learning: A survey, Wiley Interdiscip. Rev.: Data Min. Knowl. Discov. 8 (4) (2018) 1–18, http://dx.doi.org/10.1002/widm.1249.

[11] T.G. Dietterich, An experimental comparison of three methods for constructing ensembles of decision trees: Bagging, boosting, and randomization, Mach. Learn. 40 (2) (2000) 139–157, http://dx.doi.org/10.1023/A:1007607513941.

[12] G. Valentini, F. Masulli, Ensembles of Learning Machines, in: Neural Nets WIRN Vietri-2002, Series Lecture Notes in Computer Sciences, vol. 2486, 2002, pp. 3–22, http://dx.doi.org/10.1007/3-540-45808-5_1.

[13] A. Chandra, H. Chen, X. Yao, Trade-off between diversity and accuracy in ensemble generation, in: Multi-Objective Machine Learning. Studies in Computational Intelligence, Vol. 16, (2006) 2006, pp. 429–464, URL http://link.springer.com/chapter/10.1007/3-540-33019-4{_}19.

[14] L. Breiman, Bagging predictors, Mach. Learn. 24 (2) (1996) 123–140, http://dx.doi.org/10.1023/A:1018054314350.

[15] R.E. Schapire, The strength of weak learnability, Mach. Learn. 5 (2) (1990) 197–227, http://dx.doi.org/10.1007/BF00116037.

[16] T.K. Ho, The random subspace method for constructing decision forests, IEEE Trans. Pattern Anal. Mach. Intell. 20 (1998) 832–844.

[17] T.G. Dietterich, Ensemble methods in machine learning, in: Proceedings of the First International Workshop on Multiple Classifier Systems, MCS '00, Springer-Verlag, Berlin, Heidelberg, 2000, pp. 1–15.

[18] J.F. Kolen, J.B. Pollack, Backpropagation is sensitive to initial conditions, in: Complex Systems, 1990.

[19] D.H. Wolpert, Stacked generalization, Neural Netw. 5 (2) (1992) 241–259, http://dx.doi.org/10.1016/S0893-6080(05)80023-1, URL http://www.sciencedirect.com/science/article/pii/S0893608005800231.

[20] M.P. Sesmero Lorente, A. Ledezma Espino, A. de Miguel, Generating ensembles of heterogeneous classifiers using stacked generalization, Wiley Interdiscip. Rev.: Data Min. Knowl. Discov. 5 (2015) http://dx.doi.org/10.1002/widm.1143.

[21] D.H. Wolpert, Stacked generalization, Neural Netw. 5 (2) (1992) 241–259, http://dx.doi.org/10.1016/S0893-6080(05)80023-1, URL http://www.sciencedirect.com/science/article/pii/S0893608005800231.

[22] X. Zhu, X. Wu, Y. Yang, Dynamic classifier selection for effective mining from noisy data streams, 2004, pp. 305–312, http://dx.doi.org/10.1109/ICDM.2004.10091.

[23] L. Kuncheva, Switching between selection and fusion in combining classifiers: An experiment, IEEE Trans. Syst. Man Cybern. B 32 (2002) 146–156, http://dx.doi.org/10.1109/3477.990871.

[24] H. Drucker, C. Cortes, L.D. Jackel, Y. LeCun, V. Vapnik, Boosting and other ensemble methods, Neural Comput. 6 (6) (1994) 1289–1301.

[25] H. Drucker, R. Schapire, P. Simard, Improving performance in neural networks using a boosting algorithm, in: Advances in Neural Information Processing Systems, 1993, pp. 42–49.

[26] E. Bauer, R. Kohavi, An empirical comparison of voting classification algorithms : Bagging, boosting, and variants, Mach. Learn. 36 (1996) 1–38.

[27] T.G. Dietterich, Machine-learning research, AI Mag. 18 (4) (1997) 97, http://dx.doi.org/10.1609/aimag.v18i4.1324, URL https://www.aaai.org/ojs/index.php/aimagazine/article/view/1324.

[28] M. Galar, A. Fernandez, E. Barrenechea, H. Bustince, F. Herrera, A review on ensembles for the class imbalance problem: bagging-, boosting-, and hybrid-based approaches, IEEE Trans. Syst. Man Cybern. C 42 (4) (2011) 463–484.

[29] L. Hakim, B. Sartono, A. Saefuddin, Bagging based ensemble classification method on imbalance datasets, Int. J. Comput. Sci. Netw. 6 (2017) 670–676.

[30] K.N. Rajesh, R. Dhuli, Classification of imbalanced ECG beats using re-sampling techniques and AdaBoost ensemble classifier, Biomed. Signal Process. Control 41 (2018) 242–254, http://dx.doi.org/10.1016/j.bspc.2017.12.004, URL http://www.sciencedirect.com/science/article/pii/S1746809417302872.

[31] V. Mondéjar-Guerra, J. Novo, J. Rouco, M. Penedo, M. Ortega, Heartbeat classification fusing temporal and morphological information of ECGs via ensemble of classifiers, Biomed. Signal Process. Control 47 (2019) 41–48, http://dx.doi.org/10.1016/j.bspc.2018.08.007, URL http://www.sciencedirect.com/science/article/pii/S1746809418301976.

[32] E.R.Q. Fernandes, A.C.P.L.F. de Carvalho, X. Yao, Ensemble of classifiers based on multiobjective genetic sampling for imbalanced data, IEEE Trans. Knowl. Data Eng. 32 (6) (2020) 1104–1115.

[33] A.A. Aburomman, M.B.I. Reaz, A survey of intrusion detection systems based on ensemble and hybrid classifiers, Comput. Secur. 65 (2017) 135–152.

[34] A.A. Aburomman, M.B.I. Reaz, A novel SVM-kNN-PSO ensemble method for intrusion detection system, Appl. Soft Comput. 38 (2016) 360–372.

[35] W. Wang, Y. Li, X. Wang, J. Liu, X. Zhang, Detecting android malicious apps and categorizing benign apps with ensemble of classifiers, Future Gener. Comput. Syst. 78 (2018) 987–994, http://dx.doi.org/10.1016/j.future.2017.01.019, URL http://www.sciencedirect.com/science/article/pii/S0167739X17300742.

[36] I.S. Thaseen, C.A. Kumar, A. Ahmad, Integrated intrusion detection model using chi-square feature selection and ensemble of classifiers, Arab. J. Sci. Eng. 44 (4) (2019) 3357–3368.

[37] E. Alickovic, A. Subasi, A.D.N. Initiative, et al., Automatic detection of alzheimer disease based on histogram and random forest, in: International Conference on Medical and Biological Engineering, Springer, 2019, pp. 91–96.

[38] S. Valladares-Rodríguez, L. Anido-Rifón, M.J. Fernández-Iglesias, D. Facal-Mayo, A machine learning approach to the early diagnosis of alzheimer's disease based on an ensemble of classifiers, in: International Conference on Computational Science and Its Applications, Springer, 2019, pp. 383–396.

[39] S. El-Sappagh, M. Elmogy, F. Ali, T. Abuhmed, S. Islam, K.-S. Kwak, A comprehensive medical decision–support framework based on a heterogeneous ensemble classifier for diabetes prediction, Electronics 8 (6) (2019) 635.

[40] B. Swiderski, S. Osowski, J. Kurek, M. Kruk, I. Lugowska, P. Rutkowski, W. Barhoumi, Novel methods of image description and ensemble of classifiers in application to mammogram analysis, Expert Syst. Appl. 81 (2017) 67–78, http://dx.doi.org/10.1016/j.eswa.2017.03.031, URL http://www.sciencedirect.com/science/article/pii/S0957417417301860.

[41] I. Perikos, I. Hatzilygeroudis, Recognizing emotions in text using ensemble of classifiers, Eng. Appl. Artif. Intell. 51 (2016) 191–201, http://dx.doi.org/10.1016/j.engappai.2016.01.012, URL http://www.sciencedirect.com/science/article/pii/S0952197616000166, Mining the Humanities: Technologies and Applications.

[42] J. Thorne, M. Chen, G. Myrianthous, J. Pu, X. Wang, A. Vlachos, Fake news stance detection using stacked ensemble of classifiers, in: Proceedings of the 2017 EMNLP Workshop: Natural Language Processing Meets Journalism, 2017, pp. 80–83.

[43] T. Daghistani, R. Alshammari, Improving accelerometer-based activity recognition by using ensemble of classifiers, Int. J. Adv. Comput. Sci. Appl. 7 (5) (2016) 128–133.

[44] P.C. Ribeiro, G.G. Schardong, S.D. Barbosa, C.S. de Souza, H. Lopes, Visual exploration of an ensemble of classifiers, Comput. Graph. 85 (2019) 23–41.

[45] E. Tang, P. Suganthan, X. Yao, An analysis of diversity measures, Mach. Learn. 65 (2006) 247–271, http://dx.doi.org/10.1007/s10994-006-9449-2.

[46] G. Zenobi, P. Cunningham, Using diversity in preparing ensembles of classifiers based on different feature subsets to minimize generalization error, in: L. De Raedt, P. Flach (Eds.), Machine Learning: ECML 2001, Springer Berlin Heidelberg, Berlin, Heidelberg, 2001, pp. 576–587.

[47] S. Gu, Y. Jin, Generating diverse and accurate classifier ensembles using multi-objective optimization, in: IEEE Symposium on Computational Intelligence in Multi-Criteria Decision-Making, MCDM, 2004, pp. 9–15.

[48] T. Löfström, U. Johansson, H. Boström, On the use of accuracy and diversity measures for evaluating and selecting ensembles of classifiers, in: Proceedings - 7th International Conference on Machine Learning and Applications, ICMLA 2008, 2008, pp. 127–132, http://dx.doi.org/10.1109/ICMLA.2008.102.

[49] A. Tsymbal, M. Pechenizkiy, P. Cunningham, Diversity in search strategies for ensemble feature selection, Inf. Fusion 6 (1) (2005) 83–98, http://dx.doi.org/10.1016/j.inffus.2004.04.003.

[50] C.A. Shipp, L.I. Kuncheva, Relationships between combination methods and measures of diversity in combining classifiers, Inf. Fusion 3 (2) (2002) 135–148, http://dx.doi.org/10.1016/S1566-2535(02)00051-9, URL http://www.sciencedirect.com/science/article/pii/S1566253502000519.

[51] M. Sesmero, J. Alonso-Weber, G. Gutierrez, A. Ledezma, A. Sanchis, A new artificial neural network ensemble based on feature selection and class recoding, Neural Comput. Appl. 21 (4) (2010) 771–783.

[52] L.I. Kuncheva, C.J. Whitaker, Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy, Mach. Learn. 51 (2) (2003) 181–207, http://dx.doi.org/10.1023/A:1022859003006.

[53] R.T. Clemen, Combining forecasts: A review and annotated bibliography, Int. J. Forecast. 5 (4) (1989) 559–583, http://dx.doi.org/10.1016/0169-2070(89)90012-5, URL http://www.sciencedirect.com/science/article/pii/0169207089900125.

[54] K. Bache, M. Lichman, UCI Machine Learning Repository, University of California, Irvine, School of Information and Computer Sciences, 2013, URL http://archive.ics.uci.edu/ml.

[55] C.-C. Chang, C.-J. Lin, LIBSVM: A library for support vector machines, ACM Trans. Intell. Syst. Technol. (TIST) 2 (2011) 27:1–27:27, Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm.

[56] 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, F. Herrera, KEEL: A software tool to assess evolutionary algorithms for data mining problems, Soft Comput. 13 (3) (2008) 307–318.

[57] M.P. Sesmero, A. Ledezma, J.M. Alonso-Weber, G. Gutierrez, A. Sanchis, Control Learning and Systems Optimization Group CAOS - Repository, Carlos III University of Madrid Spain, 2020, URL http://www.caos.inf.uc3m.es/datasets/.

[58] Y. LeCun, The MNIST Database of handwritten digits, URL http://yann.lecun.com/exdb/mnist.

[59] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, et al., Scikit-learn: Machine learning in Python, J. Mach. Learn. Res. 12 (Oct) (2011) 2825–2830.

[60] A study of the effect of different types of noise on the precision of supervised learning techniques, Artif. Intell. Rev. 33 (2010) 275–306, http://dx.doi.org/10.1007/s10462-010-9156-z.

[61] N. García-Pedrajas, C. García-Osorio, C. Fyfe, Nonlinear boosting projections for ensemble construction, J. Mach. Learn. Res. 8 (2007) 1–33.

[62] M.P. Sesmero, J.M. Alonso-Weber, A. Sanchis, CCE: An ensemble architecture based on coupled ANN for solving multiclass problems, Inf. Fusion 58 (2020) 132–152, http://dx.doi.org/10.1016/j.inffus.2019.12.015, URL http://www.sciencedirect.com/science/article/pii/S1566253519305469.