Hindawi
Complexity
Volume 2017, Article ID 9023970, 17 pages
https://doi.org/10.1155/2017/9023970

WILEY | Hindawi

Research Article

A New Robust Classifier on Noise Domains: Bagging of Credal C4.5 Trees

Joaquín Abellán, Javier G. Castellano, and Carlos J. Mantas

Department of Computer Science and Artificial Intelligence, University of Granada, Granada, Spain

Correspondence should be addressed to Joaquín Abellán; jabellan@decsai.ugr.es

Received 9 June 2017; Revised 10 October 2017; Accepted 2 November 2017; Published 3 December 2017

Academic Editor: Roberto Natella

The knowledge extraction from data with noise or outliers is a complex problem in the data mining area. Normally, it is not easy to eliminate those problematic instances. To obtain information from this type of data, robust classifiers are the best option to use. One of them is the application of bagging scheme on weak single classifiers. The Credal C4.5 (CC4.5) model is a new classification tree procedure based on the classical C4.5 algorithm and imprecise probabilities. It represents a type of the so-called *credal trees*. It has been proven that CC4.5 is more robust to noise than C4.5 method and even than other previous credal tree models. In this paper, the performance of the CC4.5 model in bagging schemes on noisy domains is shown. An experimental study on data sets with added noise is carried out in order to compare results where bagging schemes are applied on credal trees and C4.5 procedure. As a benchmark point, the known Random Forest (RF) classification method is also used. It will be shown that the bagging ensemble using pruned credal trees outperforms the successful bagging C4.5 and RF when data sets with medium-to-high noise level are classified.

1. Introduction

Supervised classification [1] is an important task in data mining, where a set of observations or cases, described by a set of *attributes* (also called *features* or *predictive variables*), have assigned a value or label of the variable to be classified, also called *class variable*. This variable must be discrete; in other cases, the learning process is called regression task. A classifier can be considered as a learning method from data to obtain a set of laws to predict the class variable value for each new observation. In order to build a classifier from data, different approaches can be used, such as classical statistical methods [2], decision trees [3], and artificial neural networks or Bayesian networks [4].

Decision trees (DTs), also known as classification trees or hierarchical classifiers, are a type of classifiers with a simple structure where the knowledge representation is relatively simple to interpret. The decision tree can be seen as a set of compact rules in a tree format, where, in each node, an attribute variable is introduced; and in the leaves (or end nodes) we have a label of the class variable or a set of probabilities for each class label. Hunt et al.'s work [5] was the origin of decision trees, although they began to gain importance with the publication of the ID3 algorithm proposed by Quinlan [6]. Afterwards, Quinlan proposed the C4.5 [3] algorithm, which is an improvement of the previous ID3 one and obtains better results. This classifier has the characteristic of the *instability*, that is, that few variations of the data can produce important differences on the model.

The fusion of information obtained via ensembles or combination of several classifiers can improve the final process of a classification task; this can be represented via an improvement in terms of accuracy and robustness. Some of the more popular schemes are bagging [7], boosting [8], or Random Forest [9]. The inherent instability of decision trees [7] makes these classifiers very suitable to be employed in ensembles.

Class noise, also known as *label noise* or *classification noise*, is named to those situations which appear when data sets have incorrect class labels. This situation is principally motivated by deficiencies in the data learning and/or test capture process, such as wrong disease diagnosis method and human errors in the class label assignation (see [10–12]). One of the most important procedures to have success

in a classification task in situations of noisy domains is the use or application of ensembles of classifiers. In the literature about classification on noisy domains, bagging scheme stands out as the most successful scheme. This ensemble scheme has characteristics that it reduces the variance and avoids overfitting. A complete and recent revision of machine learning methods to manipulate label noise can be found in [13].

On the other hand, until a few years ago, the classical theory of probability (PT) has been the fundamental tool to construct a method of classification. Many theories to represent the information have arisen as a generalization of the PT, such as theory of evidence, measures of possibility, intervals of probability, and capacities of order-2. Each one represents a model of imprecise probabilities (see Walley [14]).

The Credal Decision Tree (CDT) model of Abellán and Moral [15] uses imprecise probabilities and general uncertainty measures (see Klir [16]) to build a decision tree. The CDT model represents an extension of the classical ID3 model of Quinlan [6], replacing precise probabilities and entropy with imprecise probabilities and maximum of entropy. This last measure is a well-accepted measure of total uncertainty for some special type of imprecise probabilities (Abellán et al. [17]). In the last years, it has been checked that the CDT model presents good experimental results in standard classification tasks (see Abellán and Moral [18] and Abellán and Masegosa [19]). The bagging scheme, using CDT as base classifier, has been used for the particular task of classifying data sets about credit scoring (see Abellán and Castellano [20]). A bagging scheme that uses a type of credal tree different from the CDT presented in [15] will be described in this work. This new model achieves better results than the bagging of CDT shown in [20] when data sets with added noise are classified.

In Mantas and Abellán [21], the classical method of C4.5 of Quinlan [3] has been modified using similar tools to the ones used for the CDT method. The new algorithm is called Credal C4.5 algorithm (CC4.5). It is shown that the use of imprecise probabilities has some practical advantages in data mining: the manipulation of the total ignorance is coherently solved and the indeterminacy or inconsistency is adequately represented. Hence, on noisy domains, these classifiers have an excellent performance. This assertion can be checked in Mantas and Abellán [21] and Mantas et al. [22]. In [21], the new CC4.5 presents better results than the classic C4.5 when they are applied on a large number of data sets with different levels of class noise. In [22], the performance of CC4.5 with different values for its parameter s is analyzed when data sets with distinct noise levels are classified and information about the best value for s is obtained in terms of the noise level of a data set. In this work, the bagging scheme using CC4.5 as base classifier will be presented, which obtains very good results when data sets with added noise are classified.

DTs are models with low bias and high variance. Normally, the variance and overfitting are reduced by using postpruning techniques. As we said, ensemble methods like bagging are also used to decrease the variance and overfitting. The procedures of the CDT and CC4.5 also represent other ways to reduce these two characteristics in a classification procedure. Hence, we have three methods to reduce variance and overfitting in a classification task which can be especially important when they are applied on noisy domains. We prove here that the combination of these three techniques (bagging + pruning + credal trees) represents a fusion of tools to be successful in noise domains. This assertion is shown in this work via a set of experiments where the bagging ensemble procedure is executed by using different models of trees (C4.5, CDT, and Credal C4.5) with and without postpruning process.

Experimentally, we show the performance of the CC4.5 model when it is inserted on the known ensemble scheme of bagging (called bagging CC4.5) and applied on data sets with different levels of label noise. This model obtains improvements with respect to other known ensembles of classifiers used in this type of setting: the bagging scheme with the C4.5 model and the known classifier Random Forest (RF). It is shown in the literature that the bagging scheme with the C4.5 model is normally the winning model in many studies about classification noise [23, 24].

A bagging scheme procedure, using CC4.5 as base classifier, has three important characteristics to be successful under noisy domains: (a) the different treatment of the imprecision, (b) the use of the bagging scheme, and (c) the production of medium-size trees (it is inherent to the model and related to (a)).

To reinforce the analysis of results, we will use a recent measure to quantify the degree of robustness of a classifier when it is applied on noisy data sets. This measure is the Equalized Loss of Accuracy (ELA) of Sáez et al. [25]. We will see that the bagging scheme using the CC4.5 attains the best values with this measure when the level of added noise is increased.

The rest of the paper is organized as follows. In Section 2, we begin with the necessary previous knowledge about decision trees, Credal Decision Trees, the Credal-C4.5 algorithm, and the ensemble schemes used. Section 4 contains the experimental results of the evaluation of the ensemble methods studied on a wide range of data sets varying the percentage of added noise. Section 5 describes and comments on the experimentation carried out. Finally, Section 6 is devoted to the conclusions.

2. Classic DTs versus DTs Based on Imprecise Probabilities

Decision trees are simple models that can be used as classifiers. In situations where elements are described by one or more *attribute variables* (also called *predictive attributes* or *features*) and by a single *class variable*, which is the variable under study, classification trees can be used to predict the class value of an element by considering its attribute values. In such a structure, each nonleaf node represents an attribute variable, the edges or branches between that node and its child nodes represent the values of that attribute variable, and each leaf node normally specifies an exact value of the class variable.

The process for inferring a decision tree is mainly determined by the followings aspects:

(1) The *split criterion*, that is, the method used to select the attribute to be inserted in a node and branching

(2) The criterion to stop the branching

(3) The method for assigning a class label or a probability distribution at the leaf nodes

An optional final step in the procedure to build DTs, which is used to reduce the overfitting of the model to the training set, is the following one:

(4) The postpruning process used to simplify the tree structure

In classic procedures for building DTs, where a measure of information based on PT is used, the criterion to stop the branching (above point (2)) normally is the following one: when the measure of information is not improved or when a threshold of gain in that measure is attained. With respect to the above point (3), the value of the class variable inserted in a leaf node is the one with more frequency in the partition of the data associated with that leaf node; its associated distribution of probabilities also can be inserted. Then the principal difference among all the procedures to build DTs is point (1), that is, the split criterion used to select the attribute variable to be inserted in a node.

Considering classic split criteria and split criteria based on imprecise probabilities, a basic point to differentiate them is how they obtain probabilities from data. We will compare a classical procedure using precise probabilities with the one based on the Imprecise Dirichlet Model (IDM) of Walley [14] based on imprecise probabilities:

(i) In classical split criteria, the probability associated with a state of the class variable, for a partition of the data, is the classical frequency of this state in that partition. Formally, let C be the class variable with states $\{c_1, \ldots, c_k\}$ and let $\mathscr{D}$ be a partition of the data set. The probability of c_j associated with the partition is

$$p\left(c_j\right) = \frac{n_{c_j}^D}{N}, \tag{1}$$

where $n_{c_j}^D$ is the number of pieces of data with the state $C = c_j$ in the partition set D; and N is the total number of pieces of data of that partition, $|D|$.

(ii) When we use the IDM, a model of imprecise probabilities (see Walley [14]), the probability of a state c_j of the class variable is obtained in a different way. Using the same notation, now the probability is obtained via an interval of probabilities:

$$p\left(c_j\right) \in \left[\frac{n_{c_j}^D}{N+s}, \frac{n_{c_j}^D + s}{N+s}\right], \tag{2}$$

where the parameter s is a hyperparameter belonging to the IDM. The value of parameter s regulates the convergence speed of the upper and lower probability when the sample size increases. Higher values of s produce an additional cautious inference. Walley [14] does not give a decisive recommendation for the value of the parameter s, but he proposed two candidates: $s = 1$ and $s = 2$; nevertheless, he recommend the value $s = 1$. It is easy to check that the size of the intervals increases when the value of s increases.

Table 1: Variable selected for branching (X_{sel}) by each split criterion.

IG	X_{sel}: $\operatorname{Arg\,min}_X \left\{H^{\mathscr{D}}(C \mid X)\right\}$
IIG	X_{sel}: $\operatorname{Arg\,min}_X \left\{H^*\left(K^{\mathscr{D}}(C \mid X)\right)\right\}$
IGR	X_{sel}: $\operatorname{Arg\,max}_X \left\{\frac{\text{IG}^{\mathscr{D}}(C, X)}{\text{SplitInfo}^{\mathscr{D}}(X)}\right\}$
IIGR	X_{sel}: $\operatorname{Arg\,max}_X \left\{\frac{\text{IIG}^{\mathscr{D}}(C, X)}{\text{SplitInfo}^{\mathscr{D}}(X)}\right\}$

In the following sections, we will explain the differences among the classic split criteria and the ones based on imprecise probabilities in a parallel way. We will compare the classic *Info-Gain* of Quinlan [6] with the *Imprecise Info-Gain* of Abellán and Moral [15] and the *Info-Gain Ratio* of Quinlan [3] with the *Imprecise Info-Gain Ratio* of Mantas and Abellán [21]. The final procedure to select the variable to be inserted in a node by each split criterion can be seen in Table 1.

The classical criteria use normally Shannon's measure as base measure of information, and the ones based on imprecise probabilities use the maximum entropy measure. This measure is based on the principle of maximum uncertainty [16] which is widely used in classic information theory, where it is known as maximum entropy principle [26]. This principle indicates that the probability distribution with the maximum entropy, compatible with available restrictions, must be chosen. The maximum entropy measure verifies an important set of properties on theories based on imprecise probabilities that are generalizations of the probability theory (see Klir [16]).

2.1. Info-Gain versus Imprecise Info-Gain. Following the above notation, let X be a general feature whose values belong to $\{x_1, \ldots, x_t\}$. Let $\mathscr{D}$ be a general partition of the data set. The Info-Gain (IG) criterion was introduced by Quinlan as the basis for his ID3 model [6], and it is explained as follows:

(i) The entropy of the class variable C for the data set $\mathscr{D}$ is Shannon's entropy [27] and it is defined as

$$H^{\mathscr{D}}(C) = \sum_i p\left(c_i\right) \log_2\left(\frac{1}{p\left(c_i\right)}\right), \tag{3}$$

where $p(c_i)$ represents the probability of the class i in $\mathscr{D}$.

(ii) The average entropy generated by the attribute X is

$$H^{\mathscr{D}}(C \mid X) = \sum_i P^{\mathscr{D}}(X = x_i) H^{\mathscr{D}_i}(C \mid X = x_i), \quad (4)$$

where $P^{\mathscr{D}}(X = x_i)$ represents the probability that $(X = x_i)$ in $\mathscr{D}$. $\mathscr{D}_i$ is the subset of $\mathscr{D}$ $(\mathscr{D}_i \subset \mathscr{D})$, where $(X = x_i)$.

Finally, we can define the *Info-Gain* as follows:

$$\mathrm{IG}(C, X)^{\mathscr{D}} = H^{\mathscr{D}}(C) - H^{\mathscr{D}}(C \mid X). \quad (5)$$

The Imprecise Info-Gain (IIG) [15] is based on imprecise probabilities and the utilization of uncertainty measures on credal sets (closed and convex sets of probability distributions). It was introduced to build the so-called *Credal Decision Tree* (CDT) model. Probability intervals are obtained from the data set using Walley's Imprecise Dirichlet Model (IDM) [14] (a special type of credal sets [28]). The mathematical basis applied is described below.

With the above notation, $p(c_j), j = 1, \ldots, k$, defined for each value c_j of the variable C, is obtained via the IDM:

$$p(c_j) \in \left[\frac{n_{c_j}}{N+s}, \frac{n_{c_j}+s}{N+s}\right], \quad j = 1, \ldots, k, \quad (6)$$

where n_{c_j} is the frequency of the case $(C = c_j)$ in the data set, N is the sample size, and s is the given hyperparameter belonging to the IDM.

That representation gives rise to a specific kind of credal set on the variable C, $K^{\mathscr{D}}(C)$ [28]. This set is defined as follows:

$$K^{\mathscr{D}}(C) = \left\{p \mid p(c_j) \in \left[\frac{n_{c_j}}{N+s}, \frac{n_{c_j}+s}{N+s}\right], \ j = 1, \ldots, k\right\}. \quad (7)$$

On this type of sets (really credal sets, [28]), uncertainty measures can be applied. The procedure to build CDTs uses the maximum of entropy function on the above defined credal set. This function, denoted as H^*, is defined in the following way:

$$H^*\left(K^{\mathscr{D}}(C)\right) = \max\left\{H^{\mathscr{D}}(p) \mid p \in K^{\mathscr{D}}(C)\right\}. \quad (8)$$

The procedure to obtain H^* for the special case of the IDM reaches its lowest computational cost for $s \leq 1$ (see Abellán [28] for more details).

The scheme to induce CDTs is like the one used by the classical ID3 algorithm [6], replacing its *Info-Gain* split criterion with the *Imprecise Info-Gain* (IIG) split criterion which can be defined in the following way:

$$\mathrm{IIG}^{\mathscr{D}}(C, X) = H^*\left(K^{\mathscr{D}}(C)\right) - H^*\left(K^{\mathscr{D}}(C \mid X)\right), \quad (9)$$

where $H^*(K^{\mathscr{D}}(C \mid X))$ is calculated via a similar way to $H^{\mathscr{D}}(C \mid X)$ in the IG criterion (for a more extended explanation, see Mantas and Abellán [21]).

It should be taken into account that, for a variable X and a data set $\mathscr{D}$, $\mathrm{IIG}^{\mathscr{D}}(C, X)$ can be negative. This situation does not occur with the Info-Gain criterion. This important characteristic implies that the IIG criterion can discard variables that worsen the information on the class variable. This is an important feature of the model which can be considered as an additional criterion to stop the branching of the tree, reducing the overfitting of the model.

As for IG and IIG, the first part of each criterion is a constant value for each attribute variable. Both criteria select the variable with lower value of uncertainty about the class variable when the attribute variable is known, which is expressed in the second parts in (5) and (9). This can be seen as a scheme in Table 1.

2.2. Info-Gain Ratio versus Imprecise Info-Gain Ratio. The *Info-Gain Ratio* (IGR) criterion was introduced for the C4.5 model [3] in order to improve the ID3 model. IGR penalizes variables with many states. It is defined as follows:

$$\mathrm{IGR}^{\mathscr{D}}(C, X) = \frac{\mathrm{IG}^{\mathscr{D}}(C, X)}{\mathrm{SplitInfo}^{\mathscr{D}}(X)}, \quad (10)$$

where

$$\mathrm{SplitInfo}^{\mathscr{D}}(X) = H^{\mathscr{D}}(X) = \sum_i P^{\mathscr{D}}(X = x_i) \log_2\left(\frac{1}{P^{\mathscr{D}}(X = x_i)}\right). \quad (11)$$

The method for building Credal C4.5 trees [21] is similar to Quinlan's C4.5 algorithm [3]. Credal C4.5 is created by replacing the *Info-Gain Ratio* split criterion from C4.5 with the *Imprecise Info-Gain Ratio* (IIGR) split criterion. The main difference is that Credal C4.5 estimates the values of the features and class variable by using imprecise probabilities. This criterion can be defined as follows:

$$\mathrm{IIGR}^{\mathscr{D}}(C, X) = \frac{\mathrm{IIG}^{\mathscr{D}}(C, X)}{\mathrm{SplitInfo}^{\mathscr{D}}(X)}, \quad (12)$$

where *SplitInfo* is defined in (11) and *Imprecise Info-Gain* (IIG) is

$$\mathrm{IIG}^{\mathscr{D}}(C, X) = H^*\left(K^{\mathscr{D}}(C)\right) - \sum_i P^{\mathscr{D}}(X = x_i) H^*\left(K^{\mathscr{D}}(C \mid X = x_i)\right), \quad (13)$$

where $K^{\mathscr{D}}(C)$ and $K^{\mathscr{D}}(C \mid X = x_i)$ are the credal sets obtained via the IDM for C and $(C \mid X = x_i)$ variables, respectively, for a partition $\mathscr{D}$ of the data set [15]; and $\{P^{\mathscr{D}}(X = x_i), i = 1, \ldots, n\}$ is a probability distribution that belongs to the credal set $K^{\mathscr{D}}(X)$.

We choose the probability distribution $P^{\mathscr{D}}$ from $K^{\mathscr{D}}(X)$ which maximizes the following expression:

$$\sum_i P(X = x_i) H(C \mid X = x_i). \quad (14)$$

It is simple to calculate this probability distribution. For more details, see Mantas and Abellán [21].

2.3. Bagging Decision Trees. In machine learning, the idea of taking into account several points of view before taking a decision is applied when several classifiers are combined. This is called by distinct names such as multiple classifier systems, committee of classifiers, mixture of experts, or ensemble-based systems. Normally, the ensemble of decision trees achieves a better performance than an individual classifier [10].

The usual strategy for the combination of decision trees is based on the creation of several decision trees aggregated with a majority vote criterion. If an unclassified instance appears, then each single classifier makes a prediction and the class value with the highest number of votes is assigned for the instance.

Breiman's *bagging* [7] (or Bootstrap Aggregating) is an intuitive and simple method that shows a good performance, reduces the variance, and avoids overfitting. Normally it is implemented with decision trees, but it can be applied with any type of classifier. Diversity in bagging is obtained by generating replicated bootstrap data sets of the original training data set: "different training data sets are randomly drawn with replacement from the original training set and, in consequence, the replicated training data sets have the same size as the original data, but some instances may not appear in it or may appear more than once." Afterwards, a single decision tree is built with each new instance of the training data set using the standard approach [29]. Thus, building each tree from a different data set, several decision trees are obtained, which are defined by a different set of variables, nodes, and leaves. Finally, the predictions of these trees are combined by a majority vote criterion.

3. Bagging Credal C4.5 and the Noise

Bagging Credal C4.5 consists of using the bagging scheme, presented in the previous section, with the Credal C4.5 algorithm as base classifier. The difference between CC4.5 and classic C4.5 is the split criterion. CC4.5 uses IIGR measure and C4.5 uses IGR. It can be shown that the measure IIGR is less sensitive to noise than the measure IGR. Hence, CC4.5 can perform a classification task on noisy data sets better than the classic C4.5, as it was experimentally demonstrated in [21].

The following example illustrates a case where the measure IIGR is more robust to noise than the measure IGR.

Example 1. Let us suppose a data set altered by noise and composed by 15 instances, 9 instances of class A and 6 instances of class B. It can be considered that there are two binary feature variables X_1 and X_2. According to the values of these variables, the instances are organized as follows:

$$\begin{aligned} X_1 = 0 &\longrightarrow (3 \text{ of class } A, 6 \text{ of class } B) \\ X_1 = 1 &\longrightarrow (6 \text{ of class } A, 0 \text{ of class } B) \\ X_2 = 0 &\longrightarrow (1 \text{ of class } A, 5 \text{ of class } B) \\ X_2 = 1 &\longrightarrow (8 \text{ of class } A, 1 \text{ of class } B)\,. \end{aligned} \tag{15}$$

If this data set appears in the node of a tree, then the C4.5 algorithm chooses the variable X_1 for splitting the node because

$$\mathrm{IGR}^{\mathscr{D}_n}(C, X_1) = 0.222 > \mathrm{IGR}^{\mathscr{D}_n}(C, X_2) = 0.13, \tag{16}$$

where $\mathscr{D}_n$ is the noisy data set composed by the 15 instances.

It can be supposed that the data set is noisy because it has an outlier point when $X_2 = 1$ and class is B. In this way, the clean distribution is composed by 10 instances of class A and 5 instances of class B, which are organized in the following way:

$$\begin{aligned} X_1 = 0 &\longrightarrow (4 \text{ of class } A, 5 \text{ of class } B) \\ X_1 = 1 &\longrightarrow (6 \text{ of class } A, 0 \text{ of class } B) \\ X_2 = 0 &\longrightarrow (1 \text{ of class } A, 5 \text{ of class } B) \\ X_2 = 1 &\longrightarrow (9 \text{ of class } A, 0 \text{ of class } B)\,. \end{aligned} \tag{17}$$

When this data set appears in the node of a tree, then the C4.5 algorithm chooses the variable X_2 for splitting the node because

$$\mathrm{IGR}^{\mathscr{D}}(C, X_1) = 0.497 < \mathrm{IGR}^{\mathscr{D}}(C, X_2) = 1.012, \tag{18}$$

where $\mathscr{D}$ is the clean data set composed by the 15 instances.

It can be observed that the C4.5 algorithm, by means of the IGR criterion, creates an incorrect subtree when noisy data are processed. However, a tree built with the IIGR criterion (and $s = 1$) selects the variable X_2 for splitting the node in both cases (noisy data set and clean data set). That is,

$$\mathrm{IIGR}^{\mathscr{D}_n}(C, X_1) = 0.053 < \mathrm{IIGR}^{\mathscr{D}_n}(C, X_2) = 0.123, \tag{19}$$

where $\mathscr{D}_n$ is the data set with noise, and

$$\mathrm{IIGR}^{\mathscr{D}}(C, X_1) = 0.164 < \mathrm{IIGR}^{\mathscr{D}}(C, X_2) = 0.481, \tag{20}$$

where $\mathscr{D}$ is the clean data set.

This example shows the difference with respect to the robustness. CC4.5 algorithm is more robust to noise than C4.5. For this reason, bagging Credal C4.5 is also more robust to noise than bagging C4.5. This fact will be shown with the experiments of this paper.

4. Experimentation

In this section, we shall describe the experiments carried out and comment on the results obtained. We have selected 50 well-known data sets in the field of machine learning, obtained from the *UCI repository of machine learning* [30]. The data sets chosen are very different in terms of their sample size, number and type of attribute variables, number of states of the class variable, and so forth. Table 2 gives a brief description of the characteristics of the data sets used.

We have performed a study where the bagging of Credal C4.5 on data with added noise is compared with the Random Forest algorithm [9] and the bagging of other tree based

Table 2: Data set description. Column "N" is the number of instances in the data sets, column "Feat" is the number of features or attribute variables, column "Num" is the number of numerical variables, column "Nom" is the number of nominal variables, column "k" is the number of cases or states of the class variable (always a nominal variable), and column "Range" is the range of states of the nominal variables of each data set.

Data set	N	Feat	Num	Nom	k	Range
anneal	898	38	6	32	6	2–10
arrhythmia	452	279	206	73	16	2
audiology	226	69	0	69	24	2–6
autos	205	25	15	10	7	2–22
balance-scale	625	4	4	0	3	—
breast-cancer	286	9	0	9	2	2–13
wisconsin-breast-cancer	699	9	9	0	2	—
car	1728	6	0	6	4	3-4
cmc	1473	9	2	7	3	2–4
horse-colic	368	22	7	15	2	2–6
credit-rating	690	15	6	9	2	2–14
german-credit	1000	20	7	13	2	2–11
dermatology	366	34	1	33	6	2–4
pima-diabetes	768	8	8	0	2	—
ecoli	366	7	7	0	7	—
Glass	214	9	9	0	7	—
haberman	306	3	2	1	2	12
cleveland-14-heart-disease	303	13	6	7	5	2–14
hungarian-14-heart-disease	294	13	6	7	5	2–14
heart-statlog	270	13	13	0	2	—
hepatitis	155	19	4	15	2	2
hypothyroid	3772	30	7	23	4	2–4
ionosphere	351	35	35	0	2	—
iris	150	4	4	0	3	—
kr-vs-kp	3196	36	0	36	2	2-3
letter	20000	16	16	0	26	—
liver-disorders	345	6	6	0	2	—
lymphography	146	18	3	15	4	2–8
mfeat-pixel	2000	240	0	240	10	4–6
nursery	12960	8	0	8	4	2–4
optdigits	5620	64	64	0	10	—
page-blocks	5473	10	10	0	5	—
pendigits	10992	16	16	0	10	—
primary-tumor	339	17	0	17	21	2-3
segment	2310	19	16	0	7	—
sick	3772	29	7	22	2	2
solar-flare2	1066	12	0	6	3	2–8
sonar	208	60	60	0	2	—
soybean	683	35	0	35	19	2–7
spambase	4601	57	57	0	2	—
spectrometer	531	101	100	1	48	4
splice	3190	60	0	60	3	4–6
Sponge	76	44	0	44	3	2–9
tae	151	5	3	2	3	2
vehicle	946	18	18	0	4	—
vote	435	16	0	16	2	2
vowel	990	11	10	1	11	2
waveform	5000	40	40	0	3	—
wine	178	13	13	0	3	—
zoo	101	16	1	16	7	2

models: C4.5 [10] and CDT [23]. We have used each model with and without a postpruning process. The pruning process of each model has been the one used by defect for each model. Hence, the algorithms considered are the following ones:

(i) Bagging C4.5 with unpruned tress (BA-C4.5-U)

(ii) Bagging CDTs with unpruned trees (BA-CDT-U)

(iii) Bagging Credal C4.5 with unpruned trees (BA-CC4.5-U)

(iv) Bagging C4.5 (BA-C4.5)

(v) Bagging CDTs (BA-CDT)

(vi) Bagging Credal C4.5 (BA-CC4.5)

(vii) Random Forest (RF)

The *Weka* software [31] has been used for the experimentation. The methods BA-CDT and BA-CC4.5 and their versions with unpruned trees were implemented using data structures of *Weka*. The implementation of C4.5 algorithm provided by *Weka* software, called *J48*, was employed with its default configuration. We added the necessary methods to build Credal C4.5 trees with the same experimental conditions. In CDTs and Credal C4.5, the parameter of the IDM was set to $s = 1$, that is, the value used in the original methods by [18, 21], respectively. The reasons to use this value were principally that it was the value recommended by Walley [14]; and the procedure to obtain the maximum entropy value reaches its lowest computational cost for this value (see [28]).

The implementation of bagging ensembles and Random Forest provided by *Weka* were used with their default configurations, except that the number of trees used for those methods was equal to 100 decision trees. Although the number of trees can strongly affect the ensemble performance, this is a reasonable number of trees for the low-to-medium size of the data sets used in this study, and moreover it was the number of trees used in related research, such as [8].

Using *Weka's* filters, we have added the following percentages of random noise to the class variable: 0%, 10%, 20%, 30%, and 40%, only in the training data set. The procedure to introduce noise was the following: a given percentage of instances of the training data set was randomly selected and, then, their current class values were randomly changed to other possible values. The instances belonging to the test data set were left unmodified.

We repeated 10 times a 10-fold cross validation procedure for each data set. It is a very known and used validation procedure. Tables 3, 4, 5, 6, and 7 show the accuracy of the methods with the different percentages of added noise. Table 8 presents a summary of the average accuracy results where the best algorithm for each added noise level is emphasized using bold fonts and the second best is marked with italic fonts.

Following the recommendation of Demšar [32], we used a series of tests to compare the methods using the *Keel* software [33]. We used the following tests to compare multiple classifiers on multiple data sets.

Friedman Test (Friedman [34, 35]). It is a nonparametric test that ranks the algorithms separately for each data set, with the best performing algorithm being assigned the rank of 1 and the second best being assigned the rank of 2 and so forth. The null hypothesis is that all the algorithms are equivalent. If the null hypothesis is rejected, we can compare all the algorithms with each other using the *Nemenyi test* [36].

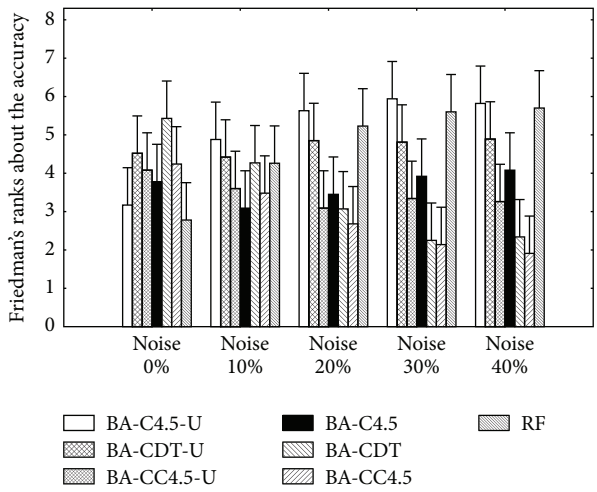

Figure 1: Values of Friedman's rank of the methods. The segment on the top expresses the size of the critical difference associated with the experiments and Nemenyi's test for the pairwise comparisons.

All the tests were carried out with a level of significance $\alpha = 0.05$. Hence, Table 9 shows Friedman's ranks about the accuracy of the methods when they are applied on data sets with different levels of added noise. The best algorithm for each noise level is emphasized using bold fonts and the second best one is marked with italic fonts. Tables 10, 11, 12, 13, and 14 show the p values of the Nemenyi test on the pairs of comparisons when they are applied on data sets with different percentage of added noise. In all cases, Nemenyi test rejects the hypotheses that the algorithms are equivalent if the corresponding p value is ≤ 0.002381. When there is a significant difference, the best algorithm is distinguished with bold fonts.

For the sake of clarity, the results of Nemenyi's test can be seen graphically in Figure 1. In this graph, the columns express the values of Friedman's ranks and the critical difference is expressed as a vertical segment. When the height of a segment on a column is lower than the one of the other column, the differences are statistically significant in favor of the algorithm represented with the lower rank (lower column).

To present the results of the average tree size (number of nodes) obtained by each method, we use Figure 2. In this figure, we can see in a quick way the average size of the trees built by each bagging method when they are applied on data sets with different levels of added noise.

We have extended the study of the results using a recent measure to quantify the degree of robustness of a classifier when it is applied on noisy data sets. This measure is the *Equalized Loss of Accuracy* (ELA) of Sáez et al. [25].

The *Equalized Loss of Accuracy (ELA)* measure is a new behavior-against-noise measure that allows us to characterize

Table 3: Accuracy results of the methods when they are used on data sets without added noise.

Data set	BA-C4.5-U	BA-CDT-U	BA-CC4.5-U	BA-C4.5	BA-CDT	BA-CC4.5	RF
anneal	98.90	98.89	98.65	98.79	98.59	98.78	99.68
arrhythmia	75.35	74.49	75.16	75.04	74.36	75.09	69.12
audiology	81.83	80.41	82.03	80.75	74.35	82.08	80.36
autos	85.45	80.27	79.28	84.39	72.65	78.98	84.29
balance-scale	81.56	82.41	82.65	82.39	83.82	82.65	80.30
breast-cancer	70.43	70.35	72.84	73.09	72.35	73.73	70.02
wisconsin-breast-cancer	96.45	96.14	96.31	96.32	95.85	96.14	96.58
car	94.33	93.55	93.30	93.65	91.24	93.04	94.70
cmc	52.19	53.21	53.92	53.12	56.02	54.09	50.69
horse-colic	85.51	84.91	85.40	85.21	85.21	85.48	85.59
credit-rating	85.68	86.07	86.84	86.14	86.16	86.43	86.14
german-credit	73.01	74.64	73.96	74.73	75.26	74.84	76.08
dermatology	97.13	94.18	96.77	96.61	93.63	96.23	96.91
pima-diabetes	76.14	75.80	75.94	76.17	75.92	75.90	76.01
ecoli	84.88	83.82	84.34	84.70	83.75	84.49	84.67
Glass	74.49	75.51	72.66	74.96	73.31	72.80	79.71
haberman	70.17	73.76	74.25	72.95	73.47	73.89	65.44
cleveland-14-heart-disease	80.23	78.68	80.13	79.90	80.39	80.20	81.56
hungarian-14-heart-disease	78.92	81.18	82.88	79.87	82.09	82.88	80.25
heart-statlog	80.96	81.41	82.26	81.19	82.33	82.19	82.26
hepatitis	81.76	80.99	82.09	81.37	81.57	81.90	83.58
hypothyroid	99.62	99.59	99.59	99.61	99.55	99.58	99.51
ionosphere	92.57	91.23	91.74	92.54	90.77	91.74	93.48
iris	94.47	95.07	94.40	94.47	94.80	94.40	94.53
kr-vs-kp	99.46	99.40	99.46	99.44	98.92	99.45	99.27
letter	94.03	92.44	93.48	93.86	90.80	93.45	96.60
liver-disorders	73.42	72.21	71.02	73.25	71.31	70.76	72.03
lymphography	79.96	76.24	79.47	79.69	77.51	79.74	83.42
mfeat-pixel	83.86	87.20	84.40	83.60	87.04	84.37	96.37
nursery	98.68	96.66	96.53	97.41	95.90	96.51	99.17
optdigits	95.84	95.55	95.84	95.79	94.74	95.83	98.30
page-blocks	97.36	97.32	97.33	97.37	97.29	97.38	97.46
pendigits	98.32	98.45	98.12	98.25	98.15	98.10	99.21
primary-tumor	44.22	43.93	44.11	44.93	41.98	44.52	43.45
segment	97.75	97.45	97.22	97.64	96.74	97.21	98.16
sick	98.97	98.97	98.91	98.85	98.54	98.84	98.43
solar-flare2	99.49	99.53	99.53	99.53	99.53	99.53	99.43
sonar	80.07	80.78	78.86	80.40	77.57	78.77	84.63
soybean	92.28	90.47	92.37	93.10	88.81	92.37	93.31
spambase	94.73	94.65	94.30	94.58	93.98	94.24	95.68
spectrometer	56.61	54.48	54.58	56.53	52.91	54.57	57.42
splice	94.70	94.40	94.84	94.68	94.06	94.71	95.88
Sponge	93.91	92.63	93.88	92.63	92.50	92.63	95.00
tae	60.88	60.88	54.85	59.43	57.56	55.18	68.25
vehicle	75.22	74.78	74.47	75.17	73.06	74.49	75.18
vote	96.78	96.34	96.62	96.69	95.52	96.69	96.43
vowel	94.04	92.17	90.66	92.64	88.96	90.68	98.16
waveform	83.40	83.51	83.08	83.35	83.31	83.08	85.20
wine	95.34	95.84	94.89	95.23	95.10	94.89	97.74
zoo	92.80	92.40	92.90	92.50	92.61	92.90	96.33
Average	85.28	84.90	84.98	85.29	84.24	84.97	86.24

Table 4: Accuracy results of the methods when they are used on data sets with a percentage of added noise equal to 10%.

Data set	BA-C4.5-U	BA-CDT-U	BA-CC4.5-U	BA-C4.5	BA-CDT	BA-CC4.5	RF
anneal	98.05	98.50	98.45	98.64	98.36	98.59	96.44
arrhythmia	74.29	73.88	74.85	74.40	73.28	74.93	67.74
audiology	80.84	79.28	81.27	81.01	75.68	81.13	75.72
autos	80.44	75.79	77.56	79.99	67.43	77.18	77.21
balance-scale	81.09	81.97	82.67	82.22	83.84	82.71	78.03
breast-cancer	67.17	69.87	70.93	72.04	71.44	72.74	66.77
wisconsin-breast-cancer	95.49	95.75	96.24	95.81	96.11	96.27	94.61
car	90.92	92.34	92.29	91.98	90.94	92.08	93.30
cmc	50.12	51.82	52.78	51.56	54.75	53.22	48.51
horse-colic	84.55	83.71	84.93	85.07	84.64	84.96	83.61
credit-rating	83.30	84.77	86.12	85.87	85.80	85.97	84.01
german-credit	72.67	73.43	73.09	73.92	74.66	73.78	74.79
dermatology	95.46	93.82	96.64	96.48	93.85	96.53	96.25
pima-diabetes	75.59	74.48	75.92	75.53	75.84	75.80	74.24
ecoli	84.82	84.70	84.79	85.09	84.11	84.67	83.87
Glass	73.33	74.37	72.48	73.10	72.32	72.30	76.82
haberman	69.05	70.44	73.40	71.68	73.99	73.44	62.66
cleveland-14-heart-disease	80.30	79.73	80.73	80.43	81.07	80.77	80.76
hungarian-14-heart-disease	78.96	79.46	82.13	79.54	80.86	82.29	79.56
heart-statlog	79.70	79.26	80.74	80.07	81.11	81.11	79.37
hepatitis	80.63	81.53	81.72	82.06	82.64	81.79	82.71
hypothyroid	99.30	99.48	99.48	99.50	99.47	99.48	99.24
ionosphere	91.80	90.58	91.40	91.71	91.35	91.40	92.31
iris	93.80	94.20	93.87	94.00	94.33	93.87	90.07
kr-vs-kp	98.02	98.72	98.81	99.17	98.79	99.09	96.57
letter	93.56	92.56	93.32	93.43	91.10	93.19	94.04
liver-disorders	70.47	69.43	68.66	70.09	69.97	68.37	69.38
lymphography	79.58	77.02	78.69	78.63	78.10	78.75	83.09
mfeat-pixel	83.09	86.71	83.89	83.14	87.27	84.01	95.82
nursery	96.27	97.11	96.95	97.12	96.08	96.69	97.55
optdigits	95.70	95.81	95.62	95.62	95.07	95.54	98.26
page-blocks	97.11	97.20	97.20	97.22	97.20	97.20	96.49
pendigits	98.43	98.43	98.27	98.35	97.99	98.19	99.08
primary-tumor	41.62	43.06	42.77	42.33	43.12	43.04	42.15
segment	96.75	97.08	97.10	97.14	96.49	97.04	95.92
sick	98.08	98.47	98.40	98.43	98.45	98.43	98.17
solar-flare2	98.58	99.47	99.48	99.53	99.53	99.53	97.56
sonar	77.45	79.47	78.02	77.60	76.99	77.97	81.61
soybean	91.22	90.25	92.61	92.72	88.45	92.62	90.41
spambase	93.23	93.32	93.55	93.53	93.55	93.54	93.13
spectrometer	55.42	51.85	54.16	55.67	50.58	54.16	56.39
splice	93.11	93.54	94.08	94.18	93.54	94.22	93.98
Sponge	91.39	92.68	93.00	92.34	92.50	92.57	92.98
tae	56.17	57.15	52.12	55.70	53.78	52.38	61.69
vehicle	73.88	73.54	72.96	74.15	72.42	72.98	74.48
vote	95.22	95.35	95.49	95.91	95.56	95.89	94.11
vowel	92.73	90.74	90.30	91.95	88.58	90.22	92.18
waveform	83.16	83.16	82.99	83.17	83.05	82.98	84.94
wine	94.44	94.50	94.77	94.60	94.54	94.77	96.86
zoo	93.66	93.37	93.27	93.77	93.77	93.27	92.97
Average	84.00	84.06	84.42	84.54	83.89	84.47	84.17

Table 5: Accuracy results of the methods when they are used on data sets with a percentage of added noise equal to 20%.

Data set	BA-C4.5-U	BA-CDT-U	BA-CC4.5-U	BA-C4.5	BA-CDT	BA-CC4.5	RF
anneal	95.34	97.42	97.41	98.04	98.10	98.05	91.16
arrhythmia	73.87	72.84	74.87	74.25	72.02	74.91	66.75
audiology	76.25	75.57	78.81	78.37	72.84	79.12	71.28
autos	73.34	69.80	74.52	73.88	63.51	74.61	70.63
balance-scale	79.26	80.97	82.09	81.37	83.25	82.08	75.28
breast-cancer	63.40	66.20	67.74	70.95	69.94	71.10	62.02
wisconsin-breast-cancer	93.41	94.00	95.91	94.91	96.22	95.98	90.83
car	85.43	89.72	89.76	89.82	89.76	90.02	90.48
cmc	48.38	50.14	50.93	50.39	53.21	51.54	46.58
horse-colic	81.46	80.73	83.93	83.96	83.44	84.42	80.70
credit-rating	79.41	82.67	84.03	83.58	85.42	85.03	80.00
german-credit	69.91	71.38	70.82	71.90	73.85	71.64	71.80
dermatology	92.73	93.52	95.46	95.39	94.01	95.63	94.86
pima-diabetes	74.62	72.60	75.41	74.76	75.30	75.64	71.85
ecoli	82.56	82.91	83.81	83.06	83.81	83.78	80.74
Glass	70.61	72.67	70.75	70.71	71.33	70.57	72.72
haberman	66.55	66.33	70.98	68.45	71.80	72.79	59.43
cleveland-14-heart-disease	79.02	79.15	80.39	79.71	81.04	79.93	79.48
hungarian-14-heart-disease	78.14	78.36	81.99	79.34	80.79	81.96	77.81
heart-statlog	76.93	76.81	79.00	78.11	79.04	79.48	76.93
hepatitis	79.35	79.95	80.88	80.63	81.38	81.20	79.69
hypothyroid	98.34	99.37	99.36	99.29	99.36	99.40	98.65
ionosphere	87.84	86.70	89.44	88.01	90.41	89.41	88.39
iris	90.07	90.93	92.73	92.27	93.80	92.67	82.80
kr-vs-kp	92.68	95.63	95.75	97.50	97.99	97.43	90.37
letter	92.57	92.32	93.01	92.97	91.28	92.92	90.57
liver-disorders	67.08	66.45	66.69	67.22	68.59	66.69	65.84
lymphography	75.99	76.00	78.17	77.49	77.44	78.65	78.08
mfeat-pixel	82.19	86.60	83.17	82.59	87.63	83.52	95.32
nursery	90.42	96.50	96.55	96.52	96.06	96.41	93.74
optdigits	95.73	96.07	95.74	95.75	95.41	95.68	98.01
page-blocks	96.33	96.79	97.12	96.80	97.10	97.10	94.68
pendigits	98.08	98.19	98.17	98.16	97.93	98.13	98.75
primary-tumor	40.20	41.03	41.71	41.26	42.80	42.39	40.53
segment	94.29	95.83	96.33	95.81	96.28	96.38	93.48
sick	96.14	97.87	97.99	97.29	98.29	98.10	96.82
solar-flare2	96.45	99.23	99.15	99.52	99.53	99.51	94.76
sonar	74.77	76.27	76.06	74.86	76.22	76.06	78.54
soybean	88.07	87.70	92.21	91.93	85.51	92.42	84.83
spambase	90.39	89.95	92.26	91.13	92.78	92.31	89.33
spectrometer	54.15	49.97	54.03	54.11	48.89	54.05	55.86
splice	90.87	91.50	92.12	92.87	92.87	92.92	91.52
Sponge	87.89	90.57	91.38	91.79	92.50	91.77	89.45
tae	53.13	54.80	51.21	53.27	50.48	51.02	54.87
vehicle	72.59	72.41	72.68	73.01	72.55	72.60	72.52
vote	92.59	93.93	93.93	95.17	95.49	95.24	90.55
vowel	88.88	84.42	88.31	89.17	84.80	88.26	84.23
waveform	82.70	82.80	82.82	82.71	83.08	82.82	84.46
wine	91.35	90.68	92.77	91.51	93.76	92.66	93.61
zoo	93.50	93.27	93.10	93.99	91.39	92.91	87.83
Average	81.50	82.15	83.27	83.11	83.01	83.58	80.99

Table 6: Accuracy results of the methods when they are used on data sets with a percentage of added noise equal to 30%.

Data set	BA-C4.5-U	BA-CDT-U	BA-CC4.5-U	BA-C4.5	BA-CDT	BA-CC4.5	RF
anneal	89.44	93.97	94.70	95.99	97.54	96.36	83.29
arrhythmia	72.86	71.64	73.41	73.14	70.64	73.88	65.58
audiology	73.37	71.51	75.62	77.21	69.01	76.72	66.02
autos	64.32	62.54	68.31	64.91	57.87	68.80	61.73
balance-scale	74.95	77.10	80.27	78.32	81.82	80.59	68.62
breast-cancer	59.83	61.24	63.19	64.31	64.62	66.34	59.10
wisconsin-breast-cancer	87.78	88.35	93.46	90.96	94.15	93.95	82.88
car	78.65	84.87	85.65	87.05	88.13	87.45	85.41
cmc	45.41	47.51	48.73	47.75	51.32	49.65	43.54
horse-colic	76.04	75.16	79.67	80.43	78.12	81.75	74.34
credit-rating	71.61	75.30	77.57	74.54	81.55	79.43	71.72
german-credit	65.07	67.19	66.37	67.27	70.65	67.52	66.93
dermatology	88.71	91.04	92.35	93.30	93.63	93.41	92.84
pima-diabetes	70.80	67.50	73.68	71.09	71.53	73.72	67.04
ecoli	79.88	80.86	83.58	81.04	83.58	83.46	77.34
Glass	66.90	68.39	68.46	67.18	69.32	68.46	67.69
haberman	62.34	60.46	66.06	62.95	66.79	70.26	56.03
cleveland-14-heart-disease	75.60	76.57	78.49	77.41	80.14	78.52	75.82
hungarian-14-heart-disease	75.96	76.09	81.93	78.63	80.18	81.52	74.10
heart-statlog	69.52	69.89	75.30	70.52	75.44	75.63	70.96
hepatitis	73.24	75.51	75.99	75.18	80.33	76.49	75.24
hypothyroid	95.90	98.82	98.77	97.43	99.15	99.06	97.31
ionosphere	79.86	78.38	83.57	80.06	84.37	83.94	81.01
iris	81.73	84.13	89.20	84.80	91.67	89.47	73.47
kr-vs-kp	82.68	86.36	86.56	88.91	95.14	89.77	79.88
letter	90.29	91.30	91.91	91.93	91.30	92.21	85.85
liver-disorders	61.66	61.44	61.71	61.89	63.61	62.01	60.26
lymphography	73.02	73.14	76.13	75.82	76.53	76.00	72.06
mfeat-pixel	81.81	87.03	83.18	82.61	88.30	83.56	94.35
nursery	81.74	93.42	94.23	95.32	95.64	95.60	87.09
optdigits	95.08	95.90	95.32	95.18	95.74	95.33	97.73
page-blocks	94.22	95.73	96.80	95.28	97.05	96.84	91.53
pendigits	97.39	97.76	97.87	97.62	97.86	97.87	98.04
primary-tumor	37.61	39.73	39.85	39.05	42.42	41.15	37.14
segment	90.50	93.15	94.84	92.33	95.99	95.09	90.13
sick	90.34	94.64	95.77	92.47	97.25	96.69	91.44
solar-flare2	92.24	97.11	97.28	99.39	99.50	99.40	90.19
sonar	69.52	71.75	71.05	69.47	73.22	70.99	72.75
soybean	83.45	81.65	90.73	90.82	81.61	91.21	79.31
spambase	86.06	83.51	89.57	86.91	89.57	89.69	83.02
spectrometer	51.62	47.97	52.28	51.89	47.05	52.28	53.58
splice	87.83	88.76	89.11	89.86	91.89	89.97	87.55
Sponge	77.45	84.05	82.27	89.16	91.95	86.88	81.07
tae	49.83	49.20	47.32	49.22	48.48	46.99	51.38
vehicle	70.13	70.36	71.25	70.45	72.02	71.45	69.86
vote	86.25	88.87	88.75	91.54	94.00	91.79	83.33
vowel	81.63	74.76	85.01	82.95	74.61	85.10	75.21
waveform	81.82	82.14	82.40	81.86	82.83	82.40	83.60
wine	85.63	85.69	89.12	85.97	92.91	89.06	88.90
zoo	89.71	90.71	90.50	91.31	90.53	91.41	80.50
Average	76.99	78.20	80.30	79.61	80.97	81.14	76.08

Table 7: Accuracy results of the methods when they are used on data sets with a percentage of added noise equal to 40%.

Data set	BA-C4.5-U	BA-CDT-U	BA-CC4.5-U	BA-C4.5	BA-CDT	BA-CC4.5	RF
anneal	80.69	87.46	88.69	89.11	96.65	92.29	74.12
arrhythmia	69.56	70.07	69.34	70.07	67.57	71.2	63.74
audiology	66.31	60.64	72.52	72.34	60.2	74.31	60.38
autos	53.96	53.59	59.65	54.36	50.98	59.99	52.03
balance-scale	65.7	69.1	75.04	69.8	78.24	75.86	59.94
breast-cancer	53.26	54.14	55.64	56.02	57.15	57.26	54.07
wisconsin-breast-cancer	75.72	73.16	84.16	79.23	83.39	85.48	68.71
car	68.86	76.19	77.37	83.32	85.42	83.63	77.93
cmc	42.41	44.33	44.45	44.19	49.14	45.84	41.06
horse-colic	64.64	63.42	69.32	68.48	65.78	71.58	63.08
credit-rating	60.59	61.03	63.65	62.96	64.65	65.58	61.55
german-credit	57.51	59.63	58.96	58.86	62.57	59.71	58.87
dermatology	83.33	85.41	87.62	88.38	91.74	89.62	88.02
pima-diabetes	66.12	60.25	68.4	66.05	64.37	68.49	60.12
ecoli	74.69	76.33	80.3	76.06	82.69	80.68	70.73
Glass	61.45	62.87	64.07	62.05	67.52	64.2	61.08
haberman	56.14	55.44	58.39	56.24	57.1	60.62	53.12
cleveland-14-heart-disease	69.76	72.58	75.2	71.6	78.52	75.83	70.26
hungarian-14-heart-disease	73.51	72.42	80.94	76.92	79.56	80.9	69.16
heart-statlog	62.07	61.56	64.59	63.3	65.96	64.67	62.33
hepatitis	62.77	63.98	66.71	65.3	69.43	68.1	64.67
hypothyroid	90.62	97.44	96.84	92.82	98.91	97.46	94.03
ionosphere	67.79	66.6	72.41	68.39	71.52	72.72	67.91
iris	72.6	74.87	86.13	75.2	87.73	86.07	65
kr-vs-kp	67.81	69.83	70.06	71.52	80.02	72.24	65.9
letter	85.81	88.67	89.1	89.21	90.94	90.23	79.34
liver-disorders	57.16	57.17	58.24	57.01	58.28	58.67	56.41
lymphography	65.98	66.58	70.64	69.15	75.07	71.51	65.66
mfeat-pixel	81.54	86.94	82.79	82.43	88.71	83.42	93.04
nursery	70.94	85.09	87.8	92.78	94.55	93.56	76.76
optdigits	93.99	95.56	94.46	94.23	95.82	94.52	96.85
page-blocks	89.84	92.85	96.05	91.33	96.75	96.15	86.03
pendigits	95.76	96.7	97.03	96.16	97.69	97.09	96.33
primary-tumor	34.53	36.51	37.22	35.86	40.53	38.61	33.68
segment	85.65	88.71	92.15	86.99	95.71	92.57	85.3
sick	76.76	80.85	85.46	78.64	87.27	87.38	76.52
solar-flare2	85.62	90.37	90.66	98.69	98.6	98.71	82.52
sonar	60.66	62.1	62.3	60.52	62.16	62.35	62.11
soybean	74.85	69.3	86.09	86.89	73.66	88.5	69.74
spambase	76.7	70.98	82.02	77.74	77.3	82.31	70.71
spectrometer	48.99	43.13	50.12	49.19	43.11	50.14	50.73
splice	82.91	84.34	84.4	84.97	89.2	85.23	81.45
Sponge	69.29	75.05	72.79	79.73	89.18	77.18	73.98
tae	46.17	45.71	43.48	45.9	43.17	43.22	46.67
vehicle	65	65.98	67.71	65.54	69.76	68.11	64.7
vote	73.54	75.62	76.2	79.39	83.92	79.85	70.73
vowel	71.84	65.65	78.47	73.27	63.14	78.54	65.86
waveform	79.59	80.21	80.84	79.65	81.78	80.84	81.54
wine	78.25	79.1	83.99	78.82	88.06	83.99	81.07
zoo	81.58	84.46	85.15	86.05	87.52	86.05	70.25
Average	70.02	71.2	74.51	73.25	75.77	75.86	68.92

Table 8: Average result of the accuracy of the different algorithms when they are built from data sets with added noise.

Algorithm	Noise 0%	Noise 10%	Noise 20%	Noise 30%	Noise 40%
BA-C4.5-U	85.28	84.00	81.50	76.99	70.02
BA-CDT-U	84.90	84.06	82.15	78.20	71.20
BA-CC4.5-U	84.98	84.42	*83.27*	80.30	74.51
BA-C4.5	*85.29*	**84.54**	83.11	79.61	73.25
BA-CDT	84.24	83.89	83.01	*80.97*	*75.77*
BA-CC4.5	84.97	*84.47*	**83.58**	**81.14**	**75.86**
RF	**86.24**	84.17	80.99	76.08	68.92

Table 9: Friedman's ranks about the accuracy of the algorithms when they are applied on data sets with different percentages of added noise.

Algorithm	Noise 0%	Noise 10%	Noise 20%	Noise 30%	Noise 40%
BA-C4.5-U	*3.17*	4.88	5.63	5.94	5.82
BA-CDT-U	4.52	4.42	4.85	4.81	4.89
BA-CC4.5-U	4.08	3.60	3.09	3.34	3.26
BA-C4.5	3.78	**3.09**	3.45	3.92	4.08
BA-CDT	5.43	4.27	*3.07*	*2.25*	*2.34*
BA-CC4.5	4.24	*3.48*	**2.68**	**2.14**	**1.91**
RF	**2.78**	4.26	5.23	5.60	5.70

Table 10: p values of the Nemenyi test about the accuracy on data sets without added noise.

i	Algorithms	p
21	BA-CDT versus **RF**	0
20	**BA-C4.5-U** versus BA-CDT	0
19	BA-CDT-U versus **RF**	0.000056
18	**BA-C4.5** versus BA-CDT	0.000134
17	BA-CC4.5 versus **RF**	0.000727
16	**BA-C4.5-U** versus BA-CDT-U	0.00178
15	**BA-CC4.5-U** versus BA-CDT	0.00178
14	BA-CC4.5-U versus RF	0.002622
13	BA-CDT versus BA-CC4.5	0.005882
12	BA-C4.5-U versus BA-CC4.5	0.013265
11	BA-C4.5 versus RF	0.020638
10	BA-C4.5-U versus BA-CC4.5-U	0.035183
9	BA-CDT-U versus BA-CDT	0.035183
8	BA-CDT-U versus BA-C4.5	0.086755
7	BA-C4.5-U versus BA-C4.5	0.157987
6	BA-C4.5 versus BA-CC4.5	0.287015
5	BA-CDT-U versus BA-CC4.5-U	0.308487
4	BA-C4.5-U versus RF	0.366699
3	BA-CC4.5-U versus BA-C4.5	0.487453
2	BA-CDT-U versus BA-CC4.5	0.516937
1	BA-CC4.5-U versus BA-CC4.5	0.711138

Table 11: p values of the Nemenyi test about the accuracy on data sets with 10% of added noise.

i	Algorithms	p
21	BA-C4.5-U versus **BA-C4.5**	0.000034
20	BA-C4.5-U versus **BA-CC4.5**	0.001194
19	BA-CDT-U versus **BA-C4.5**	0.002081
18	BA-C4.5-U versus BA-CC4.5-U	0.00305
17	BA-C4.5 versus BA-CDT	0.006311
16	BA-C4.5 versus RF	0.006769
15	BA-CDT-U versus BA-CC4.5	0.029579
14	BA-CDT-U versus BA-CC4.5-U	0.057705
13	BA-CDT versus BA-CC4.5	0.067475
12	BA-CC4.5 versus RF	0.07102
11	BA-CC4.5-U versus BA-CDT	0.120962
10	BA-CC4.5-U versus RF	0.126611
9	BA-C4.5-U versus RF	0.151281
8	BA-C4.5-U versus BA-CDT	0.157987
7	BA-CC4.5-U versus BA-C4.5	0.237833
6	BA-C4.5-U versus BA-CDT-U	0.287015
5	BA-C4.5 versus BA-CC4.5	0.366699
4	BA-CDT-U versus RF	0.711138
3	BA-CDT-U versus BA-CDT	0.728454
2	BA-CC4.5-U versus BA-CC4.5	0.781207
1	BA-CDT versus RF	0.981534

the behavior of a method with noisy data considering performance and robustness. *ELA* measure is expressed as follows:

$$\mathrm{ELA}_{x\%} = \frac{100 - A_{x\%}}{A_{0\%}}, \tag{21}$$

where $A_{0\%}$ is the accuracy of the classifier when it is applied on a data set without added noise and $A_{x\%}$ is the accuracy of the classifier with it is applied on a data set with level of added noise of $x\%$.

The *ELA* measure (there exists another similar measure named as the *Relative Loss of Accuracy* (RLA) of Sáez et al.

Table 12: p values of the Nemenyi test about the accuracy on data sets with 20% of added noise.

i	Algorithms	p
21	BA-C4.5-U versus **BA-CC4.5**	0
20	BA-C4.5-U versus **BA-CDT**	0
19	**BA-CC4.5** versus RF	0
18	BA-C4.5-U versus **BA-CC4.5-U**	0
17	BA-C4.5-U versus **BA-C4.5**	0
16	BA-CDT-U versus **BA-CC4.5**	0.000001
15	**BA-CDT** versus RF	0.000001
14	**BA-CC4.5-U** versus RF	0.000001
13	BA-CDT-U versus **BA-CDT**	0.000038
12	**BA-C4.5** versus RF	0.000038
11	BA-CDT-U versus **BA-CC4.5-U**	0.000046
10	BA-CDT-U versus **BA-C4.5**	0.001194
9	BA-C4.5-U versus BA-CDT-U	0.07102
8	BA-C4.5 versus BA-CC4.5	0.074716
7	BA-CC4.5-U versus BA-CC4.5	0.342638
6	BA-C4.5-U versus RF	0.354539
5	BA-CDT versus BA-CC4.5	0.366699
4	BA-C4.5 versus BA-CDT	0.379114
3	BA-CDT-U versus RF	0.379114
2	BA-CC4.5-U versus BA-C4.5	0.40471
1	BA-CC4.5-U versus BA-CDT	0.963078

Table 13: p values of the Nemenyi test about the accuracy on data sets with 30% of added noise.

i	Algorithms	p
21	BA-C4.5-U versus **BA-CC4.5**	0
20	BA-C4.5-U versus **BA-CDT**	0
19	**BA-CC4.5** versus RF	0
18	**BA-CDT** versus RF	0
17	BA-CDT-U versus **BA-CC4.5**	0
16	BA-C4.5-U versus **BA-CC4.5-U**	0
15	BA-CDT-U versus **BA-CDT**	0
14	**BA-CC4.5-U** versus RF	0
13	BA-C4.5-U versus **BA-C4.5**	0.000003
12	BA-C4.5 versus **BA-CC4.5**	0.000038
11	**BA-C4.5** versus RF	0.000101
10	BA-C4.5 versus **BA-CDT**	0.000111
9	BA-CDT-U versus **BA-CC4.5-U**	0.000668
8	BA-CC4.5-U versus BA-CC4.5	0.005479
7	BA-C4.5-U versus BA-CDT-U	0.008911
6	BA-CC4.5-U versus BA-CDT	0.01164
5	BA-CDT-U versus BA-C4.5	0.039403
4	BA-CDT-U versus RF	0.067475
3	BA-CC4.5-U versus BA-C4.5	0.179454
2	BA-C4.5-U versus RF	0.431313
1	BA-CDT versus BA-CC4.5	0.799032

[37]. We find that the *ELA* measure is more important than the *RLA* measure, because *ELA* takes into account higher levels of accuracy on data sets with added noise) considers the performance without noise as a value to normalize the degree of success. This characteristic makes it particularly useful when comparing two different classifiers over the same data set. The classifier with the lowest value for $ELA_{x\%}$ will be the most robust classifier.

Table 15 shows the values of the Equalized Loss of Accuracy *(ELA)* measures. The best algorithm for each level of added noise is identified using bold fonts and the second best one is represented with italic fonts.

Table 14: p values of the Nemenyi test about the accuracy on data sets with 40% of added noise.

i	Algorithms	p
21	BA-C4.5-U versus **BA-CC4.5**	0
20	**BA-CC4.5** versus RF	0
19	BA-C4.5-U versus **BA-CDT**	0
18	**BA-CDT** versus RF	0
17	BA-CDT-U versus **BA-CC4.5**	0
16	BA-C4.5-U versus **BA-CC4.5-U**	0
15	BA-CDT-U versus **BA-CDT**	0
14	**BA-CC4.5-U** versus RF	0
13	BA-C4.5 versus **BA-CC4.5**	0.000001
12	BA-C4.5 versus **BA-CDT**	0.000056
11	BA-C4.5-U versus **BA-C4.5**	0.000056
10	BA-CDT-U versus **BA-CC4.5-U**	0.000161
9	**BA-C4.5** versus RF	0.000177
8	BA-CC4.5-U versus **BA-CC4.5**	0.00178
7	BA-C4.5-U versus BA-CDT-U	0.031355
6	BA-CC4.5-U versus BA-CDT	0.033222
5	BA-CC4.5-U versus BA-C4.5	0.057705
4	BA-CDT-U versus BA-C4.5	0.060822
3	BA-CDT-U versus RF	0.060822
2	BA-CDT versus BA-CC4.5	0.319611
1	BA-C4.5-U versus RF	0.781207

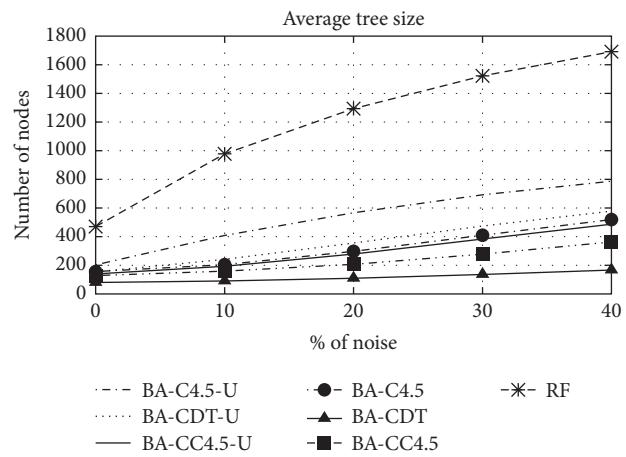

Figure 2: Average tree size for the bagging methods when they are applied on data sets with added noise.

Table 15: ELA measure for the algorithms when they are used on data sets with several percentages of added noise.

Method	10%	20%	30%	40%
BA-C4.5-U	0.1876	0.2169	0.2698	0.3515
BA-CDT-U	0.1878	0.2102	0.2568	0.3392
BA-CC4.5-U	0.1833	*0.1969*	0.2318	0.3000
BA-C4.5	**0.1813**	0.1980	0.2391	0.3136
BA-CDT	0.1912	0.2017	*0.2259*	*0.2876*
BA-CC4.5	*0.1828*	**0.1932**	**0.2220**	**0.2841**
RF	0.1836	0.2204	0.2774	0.3604

5. Comments on the Results

From a general point of view, we can state that bagging of credal trees (BA-CC4.5 and BA-CDT) has a better performance than the models used as reference (BA-C4.5 and RF) when the level of added noise is increased. This improvement is not only with respect to the accuracy, via the tests of Friedman and Nemenyi carried out, but also in terms of the measures of robustness.

An important characteristic of the results is that the bagging ensembles using credal trees built less complex models than the ones built by the bagging of classic C4.5, as can be seen in Figure 2. When the level of added noise is increased, the complexity of the bagging models using credal sets is notably smaller than the ones that use C4.5. That complexity is an important aspect of a classifier when it is applied on data set with noise, because when the model is larger, the overfitting on data with errors is larger too. Hence, the model can produce a worse performance. This is the case for RF according to Figure 2: the complexity of the random trees for RF is very large; therefore RF has a bad performance when it is applied on noisy data sets.

Next, we are going to analyze the results, on each level of added noise, taking into account principally the *accuracy* and *measures of robustness*. The following aspects must be remarked.

0%. According to accuracy and test of Friedman, without added noise, *RF* is the best model. We can observe in Table 9 (Friedman's ranking) that all the bagging models without pruning are better in accuracy than the same bagging models with pruning. Besides, BA-C4.5-U is the best model compared with the other bagging models. These results are coherent with the original bagging algorithm proposed in [7], where the trees are built without pruning for each bootstrap sample. In this way, the trees tend to be more different from each other than if they were pruned. This is a good characteristic of a model, for reducing variance, when it is used as base classifier in a bagging scheme. When we use unpruned trees, we are increasing the risk of overfitting; however, the aggregation of trees carried out by bagging offsets this risk. We remark that this assertion is right for data sets without added noise.

10%. With this low level of added noise, BA-C4.5 is now the best model but *RF* suffers notable deterioration in its performance about accuracy. Also BA-C4.5-U, which was excellent without added noise, is now the worse method. It must be remarked that it builds the largest trees. Here BA-CC4.5 begins to have excellent results in accuracy, being the second better classifier for this level of added noise. The *ELA* measure indicates that the best value is for BA-C4.5 followed by BA-CC4.5. According to Friedman's ranking about accuracy, we can observe that each bagging model with pruned trees is better than the same model with unpruned trees for this added noise level. With these results, we can conclude that the bagging algorithm needs to aggregate trees with pruning in order to manipulate data sets with low level of added noise. That is, using only a bagging scheme is insufficient in order to classify data sets with this level of added noise. Then, to prune the trees is also necessary here.

20%. With this medium-to-high level of added noise, the situation is notably different from the one with lowest level of added noise. Here BA-CC4.5 is the better procedure in terms of accuracy followed by BA-CDT. BA-C4.5 has still good performance but it is worse than the bagging credal trees. We cannot say the same for RF that has a very bad performance, getting worse when the level of noise increases. The Nemenyi test carried out presents significant differences in favor of bagging credal trees when they are compared with RF and some versions of the methods without pruning. The *ELA* measure has the best results for BA-CC4.5. BA-C4.5-U is again the worse method considering all the aspects analyzed. The size of the trees impairs seriously their performance. Hence, to obtain better results, the bagging scheme needs to use pruned credal trees when it is applied on data sets with a level of added noise greater than or equal to 20% (we will see similar conclusion for higher level of added noise).

30% and 40%. As with these levels of added noise the results are very similar, we will comment on their results together. For these levels of added noise, BA-CC4.5 is always the best procedure in terms of accuracy. The other model based on credal trees, BA-CDT, obtains the second better results. These comments are reinforced by the tests of Friedman and Nemenyi carried out. Here, even BA-C4.5 is significantly worse than the two bagging schemes of credal trees, via the test carried out. RF is now even worse than with medium level of added noise. It is remarkable that the method BA-CC4.5-U (without pruning) has better results than the pruned method BA-C4.5, although they have similar average tree sizes. Also the robustness measure confirms these assertions. Again BA-CC4.5 is the best model for the *ELA* measure. In all cases, BA-C4.5 has medium results but the same model without pruning, BA-C4.5-U, has now very bad results, being the worse method for these high levels of added noise. The second worse results are obtained by RF, which also is not a good procedure for high level of added noise, when it is compared with bagging schemes of credal trees. With these results, and considering the ones for 20% of added noise, we can say that the combination of bagging, pruning, and credal trees is necessary to obtain the best significant results when we want to apply the methods on data sets with levels of added noise greater than or equal to 20%.

With respect to the *average tree size*, we have the following comments. It can be observed that the model BA-CDT builds

always smaller trees. Perhaps this is one of the reasons why it works well with high level of added noise but not without added noise, when it is compared with the rest of models. When the level of added noise is increased, the percentage of increasing of the average size is the smallest one for BA-CDT. BA-CC4.5 has medium tree size compared with all the models with pruning; we remember that it has decent results in accuracy on data sets without added noise, and it is the best model in accuracy on data sets with medium and high levels of added noise. The following methods in tree size, with very similar sizes, are BA-C4.5 and BA-CC4.5-U, that is, a pruned method and an unpruned one; the second one is better in accuracy for level of added noise of 20–40%. At this point, we can argue that the size is not as important as the split criterion used; CC4.5 has a different treatment of the imprecision than C4.5, as was explained in previous sections. The rest of unpruned methods build larger trees, with BA-C4.5 being the one with larger results in tree size but the one with worse results in the rest of aspects, when it is compared with the other methods.

We can conclude that the method with a moderate or medium tree size, BA-CC4.5, has the best results in accuracy and measures of robustness, when the level of added noise is increased. Hence, we can think that the tree size is not a fundamental aspect of the performance of a model on noisy domains.

6. Conclusion

A very recent model called Credal C4.5 (CC4.5) is based on the classical C4.5 algorithm and imprecise probabilities. In a previous work, its excellent performance in noise domains has been shown. In this paper, we have used it in a bagging scheme on a large experimental study. We have compared it with other models that can be considered as very appropriate in this type of domains: bagging C4.5 and bagging Credal Decision Trees (CDTs). This last model, called CDT, represents other procedures based on imprecise probabilities, which was presented some years ago to be very suitable under noise.

With the results obtained in this paper, we show that bagging CC4.5 obtains excellent results when it is applied on data sets with label noise. Its performance is better than the ones of the other models used as benchmark here in two folds: accuracy and measures of robustness under noise. This improvement is even greater when the level of label noise increases.

Real data commonly have noise. This reason allows us to believe that the bagging of Credal C4.5 trees is an ideal candidate to use on data from real applications. It combines several resources to be successful in the treatment of noisy data: imprecise probabilities, bagging, and pruning. Hence, it could be considered as a powerful tool to apply in noise domains.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This work has been supported by the Spanish "Ministerio de Economía y Competitividad" and by "Fondo Europeo de Desarrollo Regional" (FEDER) under Project TEC2015-69496-R.

Supplementary Materials

Table 1. Accuracy results of the methods when they are used on data sets without added noise. Table 2. Accuracy results of the methods when they are used on data sets with a percentage of added noise equal to 10%. Table 3. Accuracy results of the methods when they are used on data sets with a percentage of added noise equal to 20%. Table 4. Accuracy results of the methods when they are used on data sets with a percentage of added noise equal to 30%. Table 5. Accuracy results of the methods when they are used on data sets with a percentage of added noise equal to 40%. Table 6. Average result of the accuracy of the different algorithms when they are built from data sets with added noise. Table 7. Friedman's ranks about the accuracy of the algorithms when they are applied on data sets with different percentages of added noise. Table 8. p values of the Nemenyi test about the accuracy on data sets without added noise. Nemenyi's procedure rejects those hypotheses that have an unadjusted p value -< 0:002381. Table 9. p values of the Nemenyi test about the accuracy on data sets with 10% of added noise. Nemenyi's procedure rejects those hypotheses that have an unadjusted p value -< 0:002381. Table 10. p values of the Nemenyi test about the accuracy on data sets with 20% of added noise. Nemenyi's procedure rejects those hypotheses that have an unadjusted p value -< 0:002381. Table 11. p values of the Nemenyi test about the accuracy on data sets with 30% of added noise. Nemenyi's procedure rejects those hypotheses that have an unadjusted p value -< 0:002381. Table 12. p values of the Nemenyi test about the accuracy on data sets with 40% of added noise. Nemenyi's procedure rejects those hypotheses that have an unadjusted p value -< 0:002381. Table 13. p values of the Bonferroni-Dunn test about the accuracy on data sets without added noise, where Random Forest is the best method in Friedman's rank. Table 14. p values of the Bonferroni-Dunn test about the accuracy on data sets with 10% of added noise, where bagging of C4.5 is the best method in Friedman's rank. Table 15. p values of the Bonferroni-Dunn test about the accuracy on data sets with 20% of added noise, where bagging of Credal C4.5 is the best method in Friedman's rank. Table 16. p values of the Bonferroni-Dunn test about the accuracy on data sets with 30% of added noise, where bagging of Credal C4.5 is the best method in Friedman's rank. Table 17. p values of the Bonferroni-Dunn test about the accuracy on data sets with 40% of added noise, where bagging of Credal C4.5 is the best method in Friedman's rank. *(Supplementary Materials)*

References

[1] D. J. Hand, *Construction and Assessment of Classification Rules*, John Wiley and Sons, New York, NY, USA, 1997.

[2] D. J. Hand, *Discrimination and Classification*, John Wiley, 1981.

[3] J. R. Quinlan, *C4.5: Programs for Machine Learning*, Morgan Kaufmann Publishers Inc, San Francisco, CA, USA, 1993.

[4] J. Pearl, *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*, Morgan Kaufmann, Boston, Mass, USA, 1988.

[5] E. B. Hunt, J. Marin, and P. Stone, in *Experiments in Induction*, Academic Press, 1966.

[6] J. R. Quinlan, "Induction of decision trees," *Machine Learning*, vol. 1, no. 1, pp. 81–106, 1986.

[7] L. Breiman, "Bagging predictors," *Machine Learning*, vol. 24, no. 2, pp. 123–140, 1996.

[8] Y. Freund and R. E. Schapire, "Experiments with a new boosting algorithm," in *Proceedings of the the Thirteenth International Conference on Machine Learning (ICML 1996)*, L. Saitta, Ed., pp. 148–156, Morgan Kaufmann, 1996.

[9] L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.

[10] T. G. Dietterich, "Experimental comparison of three methods for constructing ensembles of decision trees: bagging, boosting, and randomization," *Machine Learning*, vol. 40, no. 2, pp. 139–157, 2000.

[11] P. Melville and R. J. Mooney, "Constructing diverse classifier ensembles using artificial training examples," in *Proceedings of the 18th International Joint Conference on Artificial Intelligence, IJCAI'03*, pp. 505–510, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2003, http://dl.acm.org/citation.cfm?id=1630659.1630734.

[12] L.-Y. Dai, C.-M. Feng, J.-X. Liu, C.-H. Zheng, J. Yu, and M.-X. Hou, "Robust nonnegative matrix factorization via joint graph Laplacian and discriminative information for identifying differentially expressed genes," *Complexity*, Article ID 4216797, 11 pages, 2017.

[13] B. Frénay and M. Verleysen, "Classification in the presence of label noise: A survey," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 25, no. 5, pp. 845–869, 2014.

[14] P. Walley, "Inferences from multinomial data: learning about a bag of marbles," *ournal of the Royal Statistical Society. Series B (Methodological)*, vol. 58, no. 1, pp. 3–57, 1996.

[15] J. Abellán and S. Moral, "Building Classification Trees Using the Total Uncertainty Criterion," *International Journal of Intelligent Systems*, vol. 18, no. 12, pp. 1215–1225, 2003.

[16] G. J. Klir, *Uncertainty and Information, Foundations of Generalized Information Theory*, Wiley-Interscience, New York, NY, USA, 2006.

[17] J. Abellán, G. J. Klir, and S. Moral, "Disaggregated total uncertainty measure for credal sets," *International Journal of General Systems*, vol. 35, no. 1, pp. 29–44, 2006.

[18] J. Abellán and S. Moral, "Upper entropy of credal sets. Applications to credal classification," *International Journal of Approximate Reasoning*, vol. 39, no. 2-3, pp. 235–255, 2005.

[19] J. Abellán and A. R. Masegosa, "A filter-wrapper method to select variables for the naive bayes classifier based on credal decision trees," *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, vol. 17, no. 6, pp. 833–854, 2009.

[20] J. Abellán and J. G. Castellano, "A comparative study on base classifiers in ensemble methods for credit scoring," *Expert Systems with Applications*, vol. 73, pp. 1–10, 2017.

[21] C. J. Mantas and J. Abellán, "Credal-C4.5: Decision tree based on imprecise probabilities to classify noisy data," *Expert Systems with Applications*, vol. 41, no. 10, pp. 4625–4637, 2014.

[22] C. J. Mantas, J. Abellán, and J. G. Castellano, "Analysis of Credal-C4.5 for classification in noisy domains," *Expert Systems with Applications*, vol. 61, pp. 314–326, 2016.

[23] J. Abellán and A. R. Masegosa, "Bagging schemes on the presence of class noise in classification," *Expert Systems with Applications*, vol. 39, no. 8, pp. 6827–6837, 2012.

[24] S. Verbaeten and A. Van Assche, "Ensemble Methods for Noise Elimination in Classification Problems," in *Multiple Classifier Systems*, vol. 2709 of *Lecture Notes in Computer Science*, pp. 317–325, Springer Berlin Heidelberg, Berlin, Heidelberg, 2003.

[25] J. A. Sáez, J. Luengo, and F. Herrera, "Evaluating the classifier behavior with noisy data considering performance and robustness: The Equalized Loss of Accuracy measure," *Neurocomputing*, vol. 176, pp. 26–35, 2016.

[26] E. T. Jaynes, "On The Rationale of Maximum-Entropy Methods," *Proceedings of the IEEE*, vol. 70, no. 9, pp. 939–952, 1982.

[27] C. E. Shannon, "A mathematical theory of communication," *Bell Labs Technical Journal*, vol. 27, pp. 379–423, 623–656, 1948.

[28] J. Abellán, "Uncertainty measures on probability intervals from the imprecise Dirichlet model," *International Journal of General Systems*, vol. 35, no. 5, pp. 509–528, 2006.

[29] L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone, *Classification and Regression Trees*, Wadsworth, Belmont, Mass, USA, 1984.

[30] M. Lichman, *UCI Machine Learning Repository*, 2013, http://archive.ics.uci.edu/ml.

[31] I. H. Witten and E. Frank, *Data Mining: Practical Machine Learning Tools and Techniques*, Morgan Kaufmann Series in Data Management Systems, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2nd edition, 2005.

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

[33] J. Alcalá-Fdez, L. Sánchez, S. García et al., "KEEL: a software tool to assess evolutionary algorithms for data mining problems," *Soft Computing*, vol. 13, no. 3, pp. 307–318, 2009.

[34] M. Friedman, "The use of ranks to avoid the assumption of normality implicit in the analysis of variance," *Journal of the American Statistical Association*, vol. 32, no. 200, pp. 675–701, 1937.

[35] M. Friedman, "A comparison of alternative tests of significance for the problem of m rankings," *The Annals of Mathematical Statistics*, vol. 11, no. 1, pp. 86–92, 1940.

[36] P. Nemenyi, *Distribution-free multiple comparisons [Doctoral Dissertation]*, Princeton University, New Jersey, USA, 1963.

[37] J. A. Sáez, J. Luengo, and F. Herrera, "Fuzzy rule based classification systems versus crisp robust learners trained in presence of class noise's effects: A case of study," in *Proceedings of the 2011 11th International Conference on Intelligent Systems Design and Applications, ISDA'11*, pp. 1229–1234, Spain, November 2011.