entropy

MDPI

Article

Benchmarking Analysis of the Accuracy of Classification Methods Related to Entropy

Yolanda Orenes †, Alejandro Rabasa †, Jesus Javier Rodriguez-Sala † and Joaquin Sanchez-Soriano *,†,‡

I.U.I. Centro de Investigación Operativa (CIO), Universidad Miguel Hernandez de Elche, 03202 Elche, Spain; yolanda.orenes@alu.umh.es (Y.O.); a.rabasa@umh.es (A.R.); jesuja.rodriguez@umh.es (J.J.R.-S.)

* Correspondence: joaquin@umh.es

† These authors contributed equally to this work.

‡ Current address: Campus de Elche, Edificio Torretamarit, Avenida de la Universidad s/n, 03202 Elche, Spain.

Abstract: In the machine learning literature we can find numerous methods to solve classification problems. We propose two new performance measures to analyze such methods. These measures are defined by using the concept of proportional reduction of classification error with respect to three benchmark classifiers, the random and two intuitive classifiers which are based on how a non-expert person could realize classification simply by applying a frequentist approach. We show that these three simple methods are closely related to different aspects of the entropy of the dataset. Therefore, these measures account somewhat for entropy in the dataset when evaluating the performance of classifiers. This allows us to measure the improvement in the classification results compared to simple methods, and at the same time how entropy affects classification capacity. To illustrate how these new performance measures can be used to analyze classifiers taking into account the entropy of the dataset, we carry out an intensive experiment in which we use the well-known J48 algorithm, and a UCI repository dataset on which we have previously selected a subset of the most relevant attributes. Then we carry out an extensive experiment in which we consider four heuristic classifiers, and 11 datasets.

Keywords: entropy; classification methods; intuitive classification method; performance measures; benchmarking

Citation: Orenes, Y.; Rabasa, A.; Rodriguez-Sala, J.J.; Sanchez-Soriano, J. Benchmarking Analysis of the Accuracy of Classification Methods Related to Entropy. *Entropy* **2021**, *23*, 850. https://doi.org/10.3390/e23070850

Academic Editor: Kevin R. Moon

Received: 28 March 2021
Accepted: 24 June 2021
Published: 1 July 2021

1. Introduction

Classification is one of the most relevant topics in machine learning [1–4]. In general, the purpose of supervised classification is to predict the correct *class* , among a set of known *classes*, of a new observation given, based on the knowledge provided by a dataset, known as "*training data*". In addition, the classification problem is very important in decision-making in many different fields, so it is not difficult to find applications in fields such as medicine, biotechnology, marketing, security in communication networks, robotics, image and text recognition... Three issues in classification problems are the attribute subset selection, the design and implementation of classifiers, and the performance evaluation of classifiers [1–4]. In this paper, we will focus mainly on the latter.

On the other hand, entropy appears in statistics or information theory as a measure of diversity, uncertainty, randomness or even complexity. For this reason, we can find the use of entropy in the feature selection problem and the design of classifiers. Shannon [5] introduced entropy in the context of communication and information theory. This concept has been used frequently in information-based learning models [2]. Two extensions of the Shannon entropy measure, which are also frequently used, are the Renyi's entropy [6] and the Tsallis' entropy [7]. In [8], a review on generalized entropies can be found.

One of the most frequent difficulties found in the analysis of a dataset is that of high dimensionality, since when there are too many variables the analysis is more difficult and computationally expensive, there may be correlated variables, redundant variables or even

noisy variables. All of these problems can lead to poorer performance of classifiers. Thus, to solve these difficulties, one of two alternatives is commonly used: (1) reducing the dimension by transforming data, or (2) selecting a subset of characteristics while keeping most of the information in the dataset; this approach is known as feature selection. For example, in [9] the linear discriminant analysis and the RBS feature selection method are compared. An advantage of the feature selection approach is that the original meaning of the variables is kept. In classification problems, where there is a nominal target variable (the consequent), the selection of the most relevant variables is not a trivial matter. The issue of feature selection has already been addressed in many studies in the field of machine learning by using different approaches including information entropy [10–34]. Liu and Yu [35] reviewed feature selection algorithms for classification and clustering, and categorize them to facilitate the choice of the most suitable algorithm for the analysis of a particular dataset.

Many of the feature selection procedures incorporate the use of their own classifier to measure the quality of the selection, therefore, on many occasions it is possible to identify the feature selection method with the classifier itself, as can happen in wrapper and embedded methods of feature selection. There are different types of classification algorithms depending on its structure or the principles behind them. Thus, we can find classification algorithms (1) based on induction of decision tree algorithms such as ID3 [36] and its extension C4.5 [37], the classification and regression tree algorithm CART [38], and their extensions to random forest algorithms [39–41]; (2) based on similarities such as K-nearest neighbor algorithms [42,43] and their extensions to instance-based algorithms such as IBL [44]; (3) based on separation methods in vector spaces such as support vector machine algorithms [45,46]; or (4) based on probabilistic or statistical concepts and methods such as linear discriminant analysis [47], logistic regression or naïve Bayes algorithms [48,49]; among others. For details on classification and learning problems and their algorithms see [1]. Moreover, we can find in the machine learning literature many papers in which different concepts and methods from information entropy are used together with learning classification algorithms to design new classifiers to be applied in different contexts [50–60].

Given the same dataset, not all classifiers are equally accurate in their predictions. The accuracy achieved by a classification model depends on several factors such as the algorithm's own implementation, the heuristics of pruning and built-in boosting, the dataset used, and even the set of variables finally chosen for the construction of the model. Therefore, the analysis of the performance of classifiers is relevant in order to determine which works better. It is known that there is a lower bound on the error rate that can be achieved by classifiers: the Bayes error [61]. This error is associated with the Bayes classifier, which assigns an observation to the class with the highest posterior probability [61]. Therefore, this classifier and its associated error can be considered as benchmarks to evaluate the performance of a given classifier. However, the Bayes error can be computed only for a few number of problems. Therefore, different approximations and bounds of this error can be found in the literature (see, for example, Kumer and Ghosh [62] and the references herein). In the machine learning literature, there are different measures of the performance of a classifier and we can find various works that analyze the performance of different classifiers according to them. Costa et al. [63] showed that the most usual evaluation measures in practice were inadequate for hierarchical classifiers and reviewed the main evaluation measures for hierarchical classifiers. Sokolova and Lapalme [64] analyzed how different types of changes in the confusion matrix affected performance measures of classifiers. In particular, they studied the invariance properties of 24 performance measures for binary, multi-class, multi-labeled and hierarchical classifiers. Ferri et al. [65] carried out a experiment to analyze 18 different performance measures of classifiers. They also studied the relationships between the measures and their sensitivity from different approaches. Parker [66] analyzed the incoherences of seven performance measures for binary classifiers from both a theoretical and an empirical point of view in order to determine which measures were better. Labatut and Cherifi [67] studied

properties and the behavior of 12 performance measures for flat multi-class classifiers. Jiao and Du [68] reviewed the most common performance measures used in bioinformatics predictors for classifications. Valverde-Albacete and Peláez-Moreno [69–72] analyzed classification performance with information-theoretic methods. In particular, they proposed to analyze classifiers by means of entropic measures on their confusion matrices. To do this, they used the de Finetti entropy diagram or entropy triangle and a suitable decomposition of a Shannon-type entropy, and then defined two performance measures for classifiers: the entropy-modified accuracy (EMA) and the normalized information transfer (NIT) factor. The EMA is the expected proportion of times the classifier will guess the output class correctly, and the NIT factor is the proportion of available information transferred from input to output. The quotient of these two measures provides information on how much information is available for learning.

In this paper, we focus on the definition of performance measures. In particular, following the ideas on agreement coefficients from statistics, the Cohen's κ [73] and the Scott's π [74], which have also been used as performance measures of classifiers [75], we consider three performance measures closely related to them. Those statistics were originally defined to measure the concordance level between the classifications made by two evaluators. The mathematical formula is the following:

$$\text{Concordance level} = \frac{P_0 - P_e}{1 - P_e}, \tag{1}$$

where P_0 represents the observed proportion of classifications on which the two evaluators agree when classifying the same data independently; and P_e is the proportion of agreement to be expected on the basis of chance. Depending on how P_e is defined the Cohen's κ or the Scott's π are obtained. In machine learning, these statistics are used as performance measures by considering the classifier to be evaluated and a random classifier, where P_0 is the accuracy of the classifier. In this paper, we look at these performance measures from another point of view and define two new performance measures based on the Scott's π. In particular, we use the interpretation given in Goodman and Kruskal [76] for the λ statistics. Thus, we consider three benchmark classifiers, the random classifier and two intuitive classifiers. The three classifiers assign classes to new observations by using the information of the frequency distribution of all attributes in the training data. To be more specific, the random classifier, $\mathcal{X}$, predicts by random with the frequency distribution of the classes at hand, while the first intuitive classifier, $\mathcal{V}$, predicts the most likely outcome for each possible observation with the frequency distribution of the classes in the training data, and the second intuitive classifier, $\mathcal{I}$, predicts the most likely outcome for each possible observation with the joint frequency distribution of all attributes in the training data. The two described intuitive classifiers were postulated, built, and analyzed but rejected in favor of more modern classifier technologies before 2000. However, they could still be useful to define other performance measures in the style of the Cohen's κ or the Scott's π. Thus, in order to evaluate a classifier we determine the proportional reduction of classification error when we use the classifier to be evaluated with respect to using one of the benchmark classifiers. In this sense, P_0 is the accuracy of the classifier to be evaluated and P_e is the (expected) accuracy of the benchmark classifier. In the case where the benchmark classifier is the random classifier we obtain a performance measure like the Scott's π, but the interpretation given is different from the usual one in the machine learning literature. This is also an interesting approach of performance evaluation of classifiers because we can measure how advantageous a new classifier is with respect to three simple benchmark classifiers which can be seen as the best common sense options for non-expert (but sufficiently intelligent and with common sense) people, and whose error rates are simpler to determine than the Bayes error.

On the other hand, we analyze the relationship between the three benchmark classifiers and different aspects of the entropy of the dataset. Thus, the random classifier $\mathcal{X}$ and the intuitive classifier $\mathcal{V}$ are directly related to the entropy of the target attribute, while the

intuitive classifier $\mathcal{I}$ is closely related to the entropy of the target attribute when all dataset is considered, i.e., to the conditional entropy of the target attribute given the remaining variables in the dataset. With this relationships in mind, we can analyze the performance of classifiers taking into account the entropy of the dataset [77]. This is an interesting approach because it allows us to identify under what conditions of information uncertainty (measured by means of entropy) a classifier works better.

To the best of our knowledge, the main contributions of the paper to the machine learning literature are the following:

1. We consider the random classifier and two intuitive classifiers as benchmark classifiers. These classifiers can be considered as simple, intuitive and natural for common sense non-expert decision-makers.
2. We define three new performance measures of classifiers based on the Scott's π, the accuracy of classifiers, and the benchmark classifiers.
3. We interpret our performance measures of classifiers in terms of proportional reduction of classification error. Therefore, we measure how much a classifier improves the classification made by the benchmark classifiers. This interpretation is interesting because it is easy to understand and, at the same time, we determine the gain in accuracy related to three simple classifiers. In a sense, they provide information on whether the design of the classifier has been worth the effort.
4. The three performance measures of classifiers lie in the interval $[-1, 1]$, where -1 means that the classifier in evaluation worsens by 100% the correct classification made by the corresponding benchmark classifier, this corresponds to the classifier assigns incorrectly all observations, and 1 means that the classifier reduces by 100% the incorrect classification made by the corresponding benchmark classifier, this corresponds to the classifier assigns correctly all observations.
5. The benchmark classifiers catch the entropy of the dataset. The random classifier $\mathcal{X}$ and the intuitive classifier $\mathcal{V}$ measure the entropy of the target attribute, and the intuitive classifier $\mathcal{I}$ reflects the conditional entropy of the target attribute given the remaining variables in the dataset. Therefore, they allow us to analyze the performance of a classifier taking into account the entropy in the dataset. These measures, particularly that based on the intuitive classifiers, offer different information than other performance measures of the classifiers, which we consider to be interesting. The aim, therefore, is not to substitute for any known performance measure, but to provide a measure of a different aspect of the performance of a classifier.
6. We carry out an intensive experiment to illustrate how the proposed performance measures works and how the entropy can affect the performance of a classifier. For that we consider a particular dataset and the classification algorithm J48 [78–80], an implementation provided by Weka [75,81–83], of the classic C4.5 algorithm presented by Quinlan [36,37].
7. In order to validate what was observed in the previous experiment, we carried out an extensive experiment using four classifiers implemented in Weka and 11 datasets.

The rest of the paper is organized as follows. In Section 2, we provide the methodology and materials used in the paper. In particular, the method of feature selection, the algorithm of the intuitive classifier $\mathcal{I}$, the description of several heuristic classifiers implemented in Weka [75,81–83], and the definition and theoretical analysis of the performance measures introduced in this paper. In Section 3, we carry out the experiment to illustrate how the performance measures work and how they can be used to analyze the classifiers' performance in terms of entropy. In Section 4, we discuss the results obtained and conclude. Tables are included in Appendix A.

2. Materials and Methods

2.1. Method and Software Used for Feature Selection

The method used to perform the selection and ranking of the most influential variables is Gain Ratio Attribute Evaluation [25] (implemented in Weka [75,81–83]). This measure,

$GR(att)$ on Equation (2), provides an objective criterion for sorting explanatory variables by importance versus the target variable. Gain Ratio by its own design penalizes the proliferation of nodes and meliorates the variables that are distributed so uniformly. The gain ratio of each attribute is calculated using the following formula:

$$GR(att) = \frac{IG(att)}{H(att)}, \tag{2}$$

where (IG) is a measure to evaluate the informational gain provided by each attribute, which is considered to be a popular measure to evaluate attributes. In particular, it is the difference between the entropy of the consequent attribute and the entropy when *att* is known, $H(att)$. Thus, the feature selection method calculates the informational gain for each attribute *att* [25].

2.2. Methodology and Software for the Intuitive Classification Method $\mathcal{I}$

The basic idea of the intuitive classifier $\mathcal{I}$ is to generate classification rules from a dataset where all values are discrete (text tags). Dataset data will have C columns or attributes $(A_1, \ldots, A_C)$. One of the attributes (A_C in the Figure 1) is the target variable, used to classify instances. The remaining attributes $(A_1, \ldots, A_{C-1})$ are the explanatory variables of the problem or antecedents.

$$rule: \underbrace{< A_1 = V_1 >, \ldots, < A_{C-1} = V_{C-1} >}_{\text{left side}} \rightarrow \underbrace{< A_C = V_C >}_{\text{right side}} \tag{3}$$

A classification rule will consist of an antecedent (left side of the rule) and a consequent (right side of the rule), as illustrated in Equation (3). The antecedent will be composed of $C-1$ attribute/value pairs ($< A_i = V_i >$), where attributes are the explanatory variables. The consequent will consist of an attribute pair (target variable/value) in the form $< A_C = V_C >$.

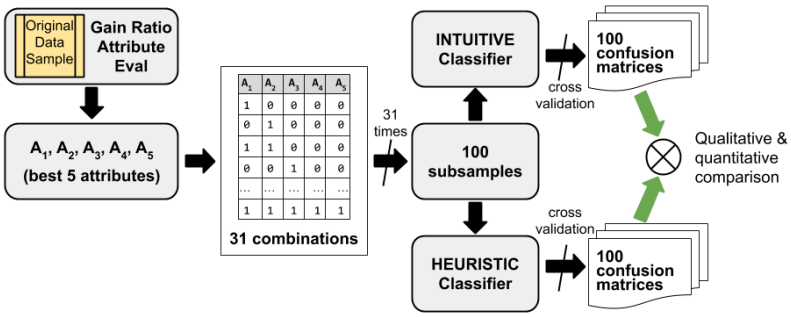

Figure 1. Experiment for each data scenario.

The intuitive classifier $\mathcal{I}$ counts the more repeated values within the data sample. In our opinion this could be what any non-expert person would do to try to identify the most likely patterns of a data sample by applying common sense. The algorithm of the intuitive classifier $\mathcal{I}$ (see Algorithm 1) performs a scan comprehensive by all records in the dataset and counts how many times each combination of values is given in the left side of the rule (antecedent), to that amount of what we will call *rule support* (R. supp). Analogously, given an antecedent, for each classification rule, the algorithm counts the number of times each of the its possible consequences or right part of the rule. We call it *rule confidence* (R. conf). (see Algorithm 1).

Algorithm 1 Pseudo-code of the algorithm of the intuitive classifier $\mathcal{I}$.

```
INPUT:
S: training data sample with C columns and N rows
C − 1 attributes are the antecedent
1 class variable is the consequent
START ALGORITHM
CRS ← ∅ {/*initialized as void set*/}
for each row in S do
  if there exists a rule Rj in CRS such that Antecedent(Rj) = Antecedent(row) and
  Consequent(Rj) = Consequent(row) then
    for all Ri in CRS such that Antecedent(Ri) = Antecedent(row) do
      Ri.supp ← Ri.supp + 1
    end for
    Rj.conf ← Rj.conf + 1
  else
    R ← New Rule
    R.antecedent ← Antecedent(row)
    R.consequent ← Consequent(row)
    R.supp ← 1
    R.conf ← 1
    for all Ri in CRS such that Antecedent(Ri) = Antecedent(row) do
      Ri.supp ← Ri.supp + 1
    end for
    CRS ← CRS + R {/*add R to CRS*/}
  end if
end for
return CRS: Classification Rule Set {/*OUTPUT*/}
END ALGORITHM
```

Note that each rule (R) of the set of rules (CRS), generated according to Algorithm 1, has associated both support and confidence values (R. supp, R. conf). These values are, as indicated above, the number of times the antecedent is repeated in the sample of data and, the number of times that, given a particular antecedent, its class of the consequent is repeated in the data sample. These two counters allow us to determine which patterns are the most repeated. This model, formed by the whole of CRS rules, predicts the class variable of an instance "s" by applying Algorithm 2.

Algorithm 2 infers the value of instance class "s", using the set rule CRS whose antecedent most closely resembles the antecedent of "s" (matching a greater number of attributes). In the case where there are multiple rules with the same number of matches, that which has a larger support is selected. If there are several rules with equal support, the most trusted is chosen. Once that rule is identified, the predicted class is the value of the consequent of the selected rule.

Algorithm 2 Pseudo-code of the algorithm to predict with a CRS model.

INPUT:
s: test row with $C-1$ antecedent attributes
CRS: Classification Rule Set
USE: RSS: Rule subset
START ALGORITHM
for $c = C-1$ **to** 1 **do**
  RSS $\leftarrow$ {Ri $\in$ CRS / c attributes of s are equal to c attributes of Ri}
  if RSS $\neq \varnothing$ **then**
    R $\leftarrow$ R1 {/* R1 is the first rule of RSS */}
    for $j = 2$ **to** $|RSS|$ **do**
      if R.supp < Rj.supp **then**
        R $\leftarrow$ Rj
      else if R.supp = Rj.supp **and** R.conf < Rj.conf **then**
        R $\leftarrow$ Rj
      end if
      return R.consequent
    end for
  end if
end for
return The resulting predicted class for row s (the consequent of a rule of CRS) {/*OUTPUT*/}
END ALGORITHM

2.3. Methodology and Software for the Heuristic Classifiers

For the generation of predictive models from the heuristic approach, we consider several heuristic classifiers: J48, Naïve Bayes, SMO, and Random Forest.

The decision tree learner J48 [78–80] is an implementation provided by Weka of the classic C4.5 algorithm [36,37]. J48 extends some of the functionalities of C4.5 such as allowing the post-pruning process of the tree to be carried out by a method based on error reduction or that the divisions over discrete variables are always binary, among others [75]. These decision trees are considered supervised classification methods. There is a dependent or class variable (variable of a discrete nature), and the classifier, from a training sample, determines the value of that class for new cases. The tree construction process begins with the root node, which has all training examples or cases associated. First, the variable or attribute from which to divide the original training sample (root node) is chosen, seeking that in the generated subsets there is minimal variability with respect to the class. This process is recursive, i.e., once the variable with the highest homogeneity is obtained with respect to the class in the child nodes, the analysis is performed again for each of the child nodes. This recursive process stops when all leaf nodes contain cases of the same class, and then over-adjustment should be avoided, for which the methods of pre-pruning and post-pruning of trees are implemented.

We also consider the Naïve Bayes algorithm implemented in Weka [75,81–83] which is a well-known classifier [48,49] based on the Bayes Theorem. Details on Naïve Bayes classifiers can be found almost in any data science or machine learning book. On the other hand, Ref. [81] is an excellent reference for the Weka software.

The SMO is an implementation in Weka [75,81–83] of the Platt's sequential minimal optimization algorithm [84–86] for training a support vector machine classifier [45]. SMO is a simple algorithm to quickly solve the support vector machine quadratic problems by means of the decomposition of the overall quadratic problem into smaller quadratic sub-problems which are easier and faster to be solved.

Finally, we will also use the random forest classifier implemented in the Weka software [75,81–83]. Random forests classifiers [41] consist of ensembles of decision trees which are built from randomly selected subset of training set, and the final classification is the result of the aggregation of the classification provided by each tree.

2.4. Evaluation Measures

The evaluation of classifiers or models to predict is very important because it allows us (1) to compare different classifiers or models to make the best choice, (2) to estimate how the classifier or model will perform in practice, and (3) to convince the decision maker that the classifier or model will be suitable for its purpose (see [1,2]). The simplest way to evaluate a classifier for a particular problem given by a dataset is to consider the ratio of correct classification. If we denote by $\mathcal{Z}$ the classifier and by $\mathcal{D}$ the dataset, then the performance of $\mathcal{Z}$ classifying a particular attribute (the consequent) in $\mathcal{D}$ is given by

$$acc(\mathcal{Z}(\mathcal{D})) = \frac{number\ of\ correct\ predictions}{total\ predictions}. \tag{4}$$

This measure is known as *accuracy*. There are other evaluation measures [1,2], but we focus in this paper on defining new measures based in some way on the concepts of proportional reduction of the classification error [76] and entropy [5].

Our approach for defining evaluation measures based on entropy is by considering simple classifiers that capture the entropy of the problem. These classifiers play the role of benchmark when evaluating other classifiers.

Let us consider a dataset $\mathcal{D}$ with N instances (rows) and C attributes (columns) such that attributes $A_1, A_2, \ldots, A_{C-1}$ are considered the explanatory variables (antecedents) and A_C is the attribute to be explained (consequent) or predicted. Let $a_{C1}, a_{C2}, \ldots, a_{CK}$ be the categories or classes of variable A_C, and let $p_{C1}, p_{C2}, \ldots, p_{CK}$ be the relative frequencies of those categories in $\mathcal{D}$. Associated with this problem, we can consider a random variable X from the sample space $\Omega = \{a_{C1}, a_{C2}, \ldots, a_{CK}\}$ to $\mathbb{R}$, such that $X(a_{Cj}) = j$, and $Prob(X = j) = p_{Cj}$. Therefore X has the non-uniform discrete distribution $D(p_{C1}, p_{C2}, \ldots, p_{CK})$, i.e., $X \sim D(p_{C1}, p_{C2}, \ldots, p_{CK})$. This X can be considered the *random classifier* for the consequent A_C in the dataset $\mathcal{D}$, defined as

$$\mathcal{X}(A_C, \mathcal{D})(i) = X(i), \tag{5}$$

where i is an observation or instance. Furthermore, we can define another simple and intuitive classifier for the consequent A_C in the dataset $\mathcal{D}$ as follows

$$\mathcal{V}(A_C, \mathcal{D})(i) = \arg\max\{p_{C1}, p_{C2}, \ldots, p_{CK}\}, \tag{6}$$

where i is an observation or instance, i.e., this intuitive classifier predicts the most likely outcome for each possible observation with the frequency distribution of the consequent A_C.

If we take the N instances of the dataset, then the classification of each instance i by the random classifier X has a categorical, generalized Bernoulli or multinoulli distribution with parameter p_i, where p_i is the frequency associated with the category that attribute A_C takes for the instance i, i.e., $X(i) \sim B(p_i)$. Therefore, the expected number of success in the classification of the N instances is given by

$$E(\sum_{i=1}^{N} X(i)) = \sum_{i=1}^{N} E(X(i)) = \sum_{i=1}^{N} p_i = \sum_{j=1}^{K} p_{Cj} N p_{Cj} = N \sum_{j=1}^{K} p_{Cj}^2. \tag{7}$$

Assuming that the classification of each instance is made independently, the variance of the number of success in the classification of the N instances is given by

$$V(\sum_{i=1}^{N} X(i)) = \sum_{i=1}^{N} V(X(i)) = \sum_{i=1}^{N} p_i(1-p_i) = \sum_{j=1}^{K} p_{Cj} N p_{Cj}(1-p_{Cj}) = N \sum_{j=1}^{K} p_{Cj}^2(1-p_{Cj}). \tag{8}$$

Note that if we consider a set of instances different from dataset $\mathcal{D}$ then Equations (7) and (8) would be given by

$$E(\sum_{i=1}^{N'} X(i)) = \sum_{j=1}^{K} N'_{Cj} p_{Cj} \quad \text{and} \quad V(\sum_{i=1}^{N'} X(i)) = \sum_{j=1}^{K} N'_{Cj} p_{Cj}(1 - p_{Cj}), \tag{9}$$

where N'_{Cj} is the number of instances for which attribute A_C takes the value a_{Cj}.

Likewise, if we are interested in the ratio of success in the classification, then Equation (7) simply becomes

$$E(\sum_{i=1}^{N} X(i)) = \sum_{j=1}^{K} p_{Cj}^2. \tag{10}$$

Thus, Equation (10) provides the expected accuracy of the random classifier $\mathcal{X}$, i.e.,

$$E(\sum_{i=1}^{N} X(i)) = E(acc(\mathcal{X}(A_C, \mathcal{D}))). \tag{11}$$

In the same way, we can arrive at the accuracy of the classifier $\mathcal{V}$ is

$$acc(\mathcal{V}(A_C, \mathcal{D})) = \max\{p_{C1}, p_{C2}, \ldots, p_{CK}\}. \tag{12}$$

On the other hand, the Shannon entropy [5] of attribute A_C in dataset $\mathcal{D}$ is given by

$$H^S(A_C, \mathcal{D}) = -\sum_{j=1}^{K} p_{Cj} \log_2 p_{Cj}. \tag{13}$$

Shannon entropy can be seen as a Renyi's entropy measure [6] or a Tsallis' entropy measure [7], which have the following mathematical expressions for attribute A_C in dataset $\mathcal{D}$,

$$H^{R,\alpha}(A_C, \mathcal{D}) = \frac{1}{1-\alpha} \log_2 \left(\sum_{j=1}^{K} p_{Cj}^{\alpha} \right), \text{ and} \tag{14}$$

$$H^{T,\alpha}(A_C, \mathcal{D}) = \frac{1}{\alpha - 1} \left(1 - \sum_{j=1}^{K} p_{Cj}^{\alpha} \right), \tag{15}$$

respectively.

Renyi's and Tsallis' entropy measures coincide with the Shannon entropy when α goes to 1, therefore Shannon's measure of entropy is seen as a Renyi's entropy measure or a Tsallis' entropy measure of order $\alpha = 1$. If we consider the Renyi's entropy measure and the Tsallis' entropy measure of order $\alpha = 2$, we obtain

$$H^{R,2}(A_C, \mathcal{D}) = -\log_2 \left(\sum_{j=1}^{K} p_{Cj}^2 \right), \text{ and} \tag{16}$$

$$H^{T,2}(A_C, \mathcal{D}) = \left(1 - \sum_{j=1}^{K} p_{Cj}^2 \right). \tag{17}$$

The entropy measures given in Equation (16), and Equation (17) are very closely related to Equation (10), which measures the expected ratio of success in the classification of the random classifier $\mathcal{X}$.

Now, we have the following result which relates the expected ratio of success of the random classifier $\mathcal{X}$ and the different entropy measures above of consequent A_C when it is binary.

Theorem 1. *Let $\mathcal{D}$, and $\mathcal{D}^*$ be two datasets with the same attributes and A_C a binary attribute which is considered the consequent. Then, the following statement holds*

1. $H^S(A_C, \mathcal{D}) > H^S(A_C, \mathcal{D}^*) \Leftrightarrow H^{R,2}(A_C, \mathcal{D}) > H^{R,2}(A_C, \mathcal{D}^*)$.

2. $H^S(A_C, \mathcal{D}) > H^S(A_C, \mathcal{D}^*) \Leftrightarrow H^{T,2}(A_C, \mathcal{D}) > H^{T,2}(A_C, \mathcal{D}^*)$.
3. $H^S(A_C, \mathcal{D}) > H^S(A_C, \mathcal{D}^*) \Leftrightarrow \sum_{j=1}^{2} p_{Cj}^2 < \sum_{j=1}^{2} p_{Cj}^{*2}$.

Proof of Theorem 1. In order to prove the theorem all you need is to prove statement 3, because the other two statements follow from the mathematical expressions of $H^{R,2}$, and $H^{T,2}$ and statement 3. Let p_{C1}, p_{C2} and p^*_{C1}, p^*_{C2} be two frequency distributions of A_C such that the entropy associated with the first is greater than the entropy associated with the second. Consider that $p_{C1} \neq p^*_{C1}$, then $p_{C2} \neq p^*_{C2}$. Otherwise, the result immediately follows. Since the entropy of the first frequency distribution is greater than the entropy of the second frequency distribution, we know that $|p_{C1} - p_{C2}| < |p^*_{C1} - p^*_{C2}|$. Let us suppose without loss of generality that $p_{C1} > p^*_{C1}$. Since $p_{C1} + p_{C2} = p^*_{C1} + p^*_{C2} = 1$, $p_{C2} < p^*_{C2}$.

On the other hand, we have that

$$p_{C1}^2 + p_{C2}^2 - \left(p_{C1}^{*2} + p_{C2}^{*2}\right) = p_{C1}^2 + (1 - p_{C1})^2 - \left(p_{C1}^{*2} + (1 - p^*_{C1})^2\right). \tag{18}$$

After some calculations, we have that

$$p_{C1}^2 + p_{C2}^2 - \left(p_{C1}^{*2} + p_{C2}^{*2}\right) = -2p_{C1} + 2p^*_{C1} < 0. \tag{19}$$

Therefore, $\sum_{j=1}^{2} p_{Cj}^2 < \sum_{j=1}^{2} p_{Cj}^{*2}$.

The proof of the converse follows similarly. □

Theorem 1 cannot be extended to attributes with more than 2 possible values, as the following example shows.

Example 1. *Consider two datasets $\mathcal{D}$ and $\mathcal{D}'$, and a common attribute A for both with three possible values $\{a, b, c\}$, such that $p_a = 0.54$, $p_b = 0.0.01$, $p_c = 0.45$, and $p'_a = 0.25$, $p'_b = 0.05$, $p'_c = 0.70$. In this situation, we have that $H^S(A, \mathcal{D}) = 1.065 < 1.076 = H^S(A, \mathcal{D}')$, but $H^{T,2}(A, \mathcal{D}) = 0.506 > 0.445 = H^{T,2}(A, \mathcal{D}')$.*

On the other hand, if we consider the Renyi's entropy measure when α goes to ∞, we obtain

$$H^{R,\infty}(A_C, \mathcal{D}) = -\log_2(\max\{p_{C1}, p_{C2}, \ldots, p_{CK}\}), \tag{20}$$

and results similar to the above can be proved.

However, all Renyi's entropy measures are correlated, therefore H^S, $H^{R,2}$, and $H^{R,\infty}$ are also correlated.

In view of the analysis above, the entropy of attribute A_C is somehow caught by the random classifier $\mathcal{X}$ and the intuitive classifier $\mathcal{V}$, in the sense that the higher the entropy, the lower the (expected) number of successes in the classification, and conversely. Therefore, the random classifier $\mathcal{X}$ and the intuitive classifier $\mathcal{V}$ can be used as benchmarks when evaluating other classifiers, taking into account the entropy of the consequent. Next we define an evaluation measure based on the analysis above.

Definition 1. *Let $\mathcal{Z}$ be a classifier. Given a dataset $\mathcal{D}$, and a consequent A_C, the performance of $\mathcal{Z}$ with respect to the random classifier X is given by*

$$\gamma^{\mathcal{X}}(\mathcal{Z}(\mathcal{D})) = \begin{cases} \frac{\mu(\mathcal{Z},\mathcal{D}) - \mu(\mathcal{X},\mathcal{D})}{1 - \mu(\mathcal{X},\mathcal{D})} & \text{if } \mu(\mathcal{Z},\mathcal{D}) - \mu(\mathcal{X},\mathcal{D}) \geq 0 \\ \\ \frac{\mu(\mathcal{Z},\mathcal{D}) - \mu(\mathcal{X},\mathcal{D})}{\mu(\mathcal{X},\mathcal{D})} & \text{if } \mu(\mathcal{Z},\mathcal{D}) - \mu(\mathcal{X},\mathcal{D}) < 0 \end{cases}, \tag{21}$$

where $\mu(X, \mathcal{D}) = \frac{E(\sum_{i=1}^{M} X(i))}{M}$, such that M is the total number of predictions, and $\mu(\mathcal{Z}, \mathcal{D})$ is the ratio of correct classifications using classifier (Z).

Note that the first case of the definition of the performance measure $\gamma^{\mathcal{X}}$ coincides with the Scott's π. If we use the intuitive classifier $\mathcal{V}$ instead of $\mathcal{X}$ as benchmark classifier, we obtain the performance measure $\gamma^{\mathcal{V}}$. The evaluation measure $\gamma^{\mathcal{X}}$ (resp. $\gamma^{\mathcal{V}}$) runs between -1 and 1, where -1 is the worst case, and is achieved when the classifier does not predict correctly any instance; 0 means that performance is as the random classifier $\mathcal{X}$ (resp. $\gamma^{\mathcal{V}}$); and 1 is the best case, and is achieved when the classifier correctly classifies all instances. The intermediate values measure in which proportion the classifier performs better (positive values) or worse (negative values) than the random classifier (resp. $\mathcal{V}$).

On the other hand, we can interpret the performance measure $\gamma^{\mathcal{X}}$ (resp. $\gamma^{\mathcal{V}}$) in terms of proportional reduction of classification error with respect to the random classifier (resp. $\mathcal{V}$). Indeed, if we predict M instances, we can write Equation (21) as follows:

$$\gamma^{\mathcal{X}}(\mathcal{Z}(\mathcal{D})) = \begin{cases} \frac{M\mu(\mathcal{Z},\mathcal{D})-M\mu(\mathcal{X},\mathcal{D})}{M-M\mu(\mathcal{X},\mathcal{D})} & \text{if } M\mu(\mathcal{Z},\mathcal{D}) - M\mu(\mathcal{X},\mathcal{D}) \geq 0 \\ \\ \frac{M\mu(\mathcal{Z},\mathcal{D})-M\mu(\mathcal{X},\mathcal{D})}{M\mu(\mathcal{X},\mathcal{D})} & \text{if } M\mu(\mathcal{Z},\mathcal{D}) - M\mu(\mathcal{X},\mathcal{D}) < 0 \end{cases}. \tag{22}$$

Now, we can write Equation (22) in the following way:

$$\gamma^{\mathcal{X}}(\mathcal{Z}(\mathcal{D})) = \begin{cases} \frac{(M-M\mu(\mathcal{X},\mathcal{D}))-(M-M\mu(\mathcal{Z},\mathcal{D}))}{M-M\mu(\mathcal{X},\mathcal{D})} & \text{if } M\mu(\mathcal{Z},\mathcal{D}) - M\mu(\mathcal{X},\mathcal{D}) \geq 0 \\ \\ \frac{M\mu(\mathcal{Z},\mathcal{D})-M\mu(\mathcal{X},\mathcal{D})}{M\mu(\mathcal{X},\mathcal{D})} & \text{if } M\mu(\mathcal{Z},\mathcal{D}) - M\mu(\mathcal{X},\mathcal{D}) < 0 \end{cases}. \tag{23}$$

Finally, Equation (23) can be interpreted as follows:

$$\gamma^{\mathcal{X}}(\mathcal{Z}(\mathcal{D})) = \begin{cases} \frac{\text{Expected \# errors by using } \mathcal{X} - \text{\# errors by using } \mathcal{Z}}{\text{Expected \# errors by using } \mathcal{X}}, & \text{if } M\mu(\mathcal{Z},\mathcal{D}) - M\mu(\mathcal{X},\mathcal{D}) \geq 0 \\ \\ \frac{\text{\# successes by using } \mathcal{Z} - \text{Expected \# successes by using } \mathcal{X}}{\text{Expected \# successes by using } \mathcal{X}} & \text{if } M\mu(\mathcal{Z},\mathcal{D}) - M\mu(\mathcal{X},\mathcal{D}) < 0 \end{cases}. \tag{24}$$

Thus, the first case of $\gamma^{\mathcal{X}}$ measures the proportional reduction of classification error when we use classifier $\mathcal{Z}$ with respect to using the random classifier $\mathcal{X}$. The second case of $\gamma^{\mathcal{X}}$ measures the proportional reduction of classification success when we use classifier $\mathcal{Z}$ with respect to using the random classifier $\mathcal{X}$. The same can be said when using the intutitive classifier $\mathcal{V}$ as benchmark.

Therefore, $\gamma^{\mathcal{X}}$ gives us information about how much a classifier $\mathcal{Z}$ improves or worsens the classification with respect to a classifier that decides the class randomly taking into account the frequency distribution of the classes. Furthermore, $\gamma^{\mathcal{X}}$ gives us information about how much a classifier $\mathcal{Z}$ improves or worsens the classification with respect to a classifier that simply predicts the most likely class according to the frequency distribution of the classes. Since the previous two classifiers only use information related to the classes, these two measures provide information on whether it is relevant to use more sophisticated classifiers that incorporate information from other attributes.

On the other hand, the measure $\gamma^{\mathcal{X}}$ and $\gamma^{\mathcal{V}}$ incorporate in a way the information on the entropy of the consequent to the evaluation of a classifier, but do not take into account the rest of the attributes (the antecedents). Nevertheless, a similar analysis can be carried out by considering all possible different strings of attributes, obtaining analogous results. On the other hand, the intuitive classification method described in Section 2.2 can be another way of taking into account all the attributes and the entropy of the dataset, since its definition is based on the repetition of instances which is related to the entropy of the dataset. In particular, it is related to the conditional entropy of the attribute A_C given the remaining variables in the dataset. Thus, another measure of evaluation of the classifiers related to entropy could be to use this intuitive classification method as a benchmark, its definition being analogous to those previously given. Below we formally outline the definition of this measure.

Definition 2. *Let $\mathcal{Z}$ be a classifier. Given a dataset $\mathcal{D}$, and a consequent A_C, the performance of $\mathcal{Z}$ with respect to the intuitive classifier $\mathcal{I}$ is given by*

$$\Gamma(\mathcal{Z}(\mathcal{D})) = \begin{cases} \frac{\mu(\mathcal{Z},\mathcal{D})-\mu(\mathcal{I},\mathcal{D})}{1-\mu(\mathcal{I},\mathcal{D})} & \text{if } \mu(\mathcal{Z},\mathcal{D}) - \mu(\mathcal{I},\mathcal{D}) \geq 0 \\ \\ \frac{\mu(\mathcal{Z},\mathcal{D})-\mu(\mathcal{I},\mathcal{D})}{\mu(\mathcal{I},\mathcal{D})} & \text{if } \mu(\mathcal{Z},\mathcal{D}) - \mu(\mathcal{I},\mathcal{D}) < 0 \end{cases}, \tag{25}$$

where $\mu(\mathcal{I},\mathcal{D})$ is the ratio of correct classifications using classifier (I), and $\mu(\mathcal{Z},\mathcal{D})$ is the ratio of correct classifications using classifier (Z).

The interpretation of Γ is completely analogous to that of γ above, only changing the random classifier $\mathcal{X}$ and the intuitive classifier $\mathcal{V}$ for the intuitive classifier $\mathcal{I}$. However, it gives some extra information about classifiers, in the sense that since it uses all information in the dataset, it provides information on how much relevant is to use more sophisticated classifiers.

3. Computer-Based Experiments: Design and Results

In this section, we illustrate how the evaluation measures introduced in Section 2 work. For that end, we design an experiment in which we consider five scenarios of entropy for a binary attribute (the consequent), and for each of those scenarios we study 31 combinations of explanatory attributes (the antecedents). Thus, we can give a better idea about how these evaluation measures work and how they measure the performance of classifiers in different entropy situations. We then go further and carry out an extensive comparison for four classifiers by using 11 different datasets whose results are concisely presented.

3.1. Datasets and Scenarios

We start from the hypothesis of working in a classification context where the target to be predicted is discrete and more specifically binary, but another multi-class target variable could be considered. A well-known dataset from UCI Machine Learning Repository [87] named "thyroid0387.data" [88] has been chosen for the most intensive experiment.

This dataset has been widely used in the literature in problems related to the field of classification. Since it is only used in this paper as an example and we are not interested in the clinical topic itself that the data collect, in order to facilitate the experiment of this study and make it exhaustive, that dataset has been minimally preprocessed as follows:

- Headers have been added and renamed.
- The numeric attributes have been removed and we have left only those which are nominal.
- The class variable has been recoded in positive and negative cases (the original sample has several types of positive instances).

Finally, the dataset used to perform the experiment has the following features:

- Number of rows: 9173
- Number of attributes/columns: 23 (all nominal)
  - 22 explanatory variables (antecedents)
  - 1 target variable (consequent)
    - 2401 positive cases
    - 6772 negative cases

The target variable used to classify which corresponds to a clinical diagnosis, is unbalanced, as it has a positive value in 2401 tuples and a negative value in 6772. From these data we will consider five types of possible scenarios with different ratios between positive and negative values (see Table 1).

Table 1. The five data scenarios.

Scenario	Positive	Negative	Total	Ratio Positive/Negative	Consequent's Entropy
S1	2400	800	3200	3:1	0.811
S2	2400	1200	3600	2:1	0.918
S3	2400	2400	4800	1:1	1.000
S4	2000	4000	6000	1:2	0.918
S5	2000	6000	8000	1:3	0.811

The remaining 10 datasets used in the most extensive experiment are also from UCI Machine Learning Repository [87]. The following modifications have been made, common to all of them.

1. In all the datasets that did not have a row with the header, it has been added, taking into account the specifications of the "Attribute Information" section of each of these UCI repository datasets.
2. The configuration in Weka to discretize has been with the parameter "bins" = 5 (to obtain 5 groups) and the parameter "UseEqualFrecuency" = true (so that the groups of data obtained were equitable).
3. When discretizing in Weka (filter→unsupervised→discretized) the results obtained were numerical intervals, so they were later renamed.

In particular, apart from the dataset already mentioned, we have used the following datasets:

- "Healthy_Older_People.data" [89,90];
- "Avila.data" [91,92];
- "Adult.data" [93];
- "nursery.data" [94];
- "Bank marketing" [95,96];
- "HTRU2.data" [97–99];
- "connect-4.data" [100];
- "tic-tac-toe.data" [101];
- "Credit approval.data" [102];
- "Mushroom.data" [103].

The main features of these datasets are summarized in Table 2.

Table 2. Main features of the datasets. **# rows** means the number of rows of the dataset; **# attributes** means the number of attributes including the consequent, and below the type of variables; **# classes** is the number of classes of the consequent; and **Distribution of the classes** is the number of cases of each class in the dataset.

Dataset	# Rows	# Attributes	# Classes	Distribution of the Classes
Thyroid	9173	23 2 categorical 21 binary	2	2401, 6772
Healthy	75128	10 8 real 1 binary 1 categorical	4	16406, 4911, 51,520, 2291
Avila	20867	11 10 real 1 categorical	12	8572, 10, 206, 705, 2190, 3923, 893, 1039, 1663, 89, 1044, 533
Adult	32561	12 3 real 1 integer 6 categorical 2 binary	2	7841, 24720
Nursery	12,960	9 8 categorical 1 binary	5	4320, 4266, 2 4044, 328
Bank	45,211	11 1 real 1 integer 5 categorical 4 binary	2	39,922, 5289
HTRU2	17,898	9 8 real 1 binary	2	16,259, 1639
Connect-4	67,557	43 43 categorical	3	6449, 16,635, 44,473
Tic-tac-toe	958	10 9 categorical 1 binary	2	332, 626
Credit	690	10 5 categorical 5 binary	2	383, 307
Mushroom	8124	23 17 categorical 6 binary	2	4208, 3916

In addition, some specific preprocessing of the data were carried out in the datasets "Adult.data" [93] and "Bank marketing" [95,96]. In "Adult.data", the rows with missing values were removed, and three attributes were discarded (capital-gain, capital-loss, native-country); and in "Bank marketing", the selected dataset was "bank-full.csv", and 6 attributes were discarded (balance, day, duration, campaign, pdays, and previous).

3.2. Experimental Design

The experiment consists of determining the accuracy of an heuristic classifier, the already mentioned J48, in comparison with three benchmark classifiers: the random classifier and two intuitive classifiers. These three classifiers to certain extent contain information about the entropy present in the dataset as explained in the previous section. Therefore, we provide evaluation measures of that classifier taking into account the entropy of the system. In this sense, we try to evaluate how this classifier performs in terms of the improvement (or deterioration) obtained with respect to three classifiers that can be considered as benchmarks and that are based on the simple distribution of data from the dataset, and then on the entropy of the data.

On the other hand, we are also interested in observing the differences between the three evaluation measures of the classifiers introduced in the previous section, and what effect, considering more or less information from the dataset, this has when making classifications of instances. To do this, we consider the five scenarios described in Table 1, which have different level of Shannon's entropy in the consequent. For each of these scenarios, we follow the process depicted in Figure 1.

First, starting from original sample of data and fixing the consequent variable (or target variable) A_C to be studied, the five variables (attributes) more correlated with the target variable are selected. Then they are sorted $(A_1, A_2, A_3, A_4, A_5)$, that is, we determine which is more correlated with the consequent and which less, for which we use the *gain ratio attribute method* described in Section 2.1. In Table 3, we show the gain ratio scores observed for each of the five scenarios (S1,S2, S3,S4, S5) considered.

Table 3. Results of gain ratio attribute evaluation in the five scenarios.

Attributes	S1	S2	S3	S4	S5
A_1	0.036	0.050	0.083	0.122	0.102
A_2	0.037	0.037	0.082	0.076	0.134
A_3	0.033	0.034	0.028	0.020	0.016
A_4	0.034	0.032	0.028	0.015	0.013
A_5	0.029	0.022	0.026	0.013	0.010

At this point, we would like to emphasize once again that it is not our purpose to analyze a particular problem, but only to use a dataset for analyzing the evaluation measures introduced in this paper and also show an analysis of heuristic classifiers when considering entropy characteristics of the dataset. For this reason, attributes $A_1, \ldots, A_5$ are not necessarily the same nor they are in the same order in the five scenarios. We simply call generically A_1 to the attribute best correlated with the target variable in each scenario, even if it is not the same variable in each of them. Accordingly, the other attributes occupy second to fifth positions in the correlation ranking with the consecutive attribute in each scenario, always according to the gain ratio attribute evaluation. In each of the scenarios, these five attributes will be used as predictor or explanatory variables (antecedents) to generate the classification models. It is not an objective of this work to delve into the different methods of features (attributes) subset selection, but we simply use one of them, always the same (gain ratio attribute), in order to work only with those attributes that in each case are really significant. Reducing the size of the problem from 22 to 5 explanatory variables will allow a comprehensive experiment with which to illustrate and analyze the two introduced evaluation measures, and to show a way to analyze the performance of an heuristic classifier when we consider different degrees of entropy in the dataset. In order to select the five best attributes, we use the software Weka [75,82,83], in particular, its *Select attributes* function, with *GainRatioAttributeEval* as the attribute evaluator, *ranker* as the search method, and *cross-validation* as attribution selection mode. Note that Weka gives two measures of the relevance of the (antecedent) attributes. The average merit and its standard deviation, and the average rank and its standard deviation. The first refers to the mean of

the correlations measured with GainRatioAttributeEval in 10 cycles (although with 5 cycles would have been sufficient, since only the first 5 attributes are wanted) of validation fold. The average rank refers to the average order in which each attribute remained in each of the ten cycles. See [75,82] for details about Weka.

Once the five best attributed are chosen, the next step is to establish the 31 possible combinations of the set of predictor variables. These 31 combinations will be the background to consider in a set of classification rules or in a decision tree. That is, 31 classification studies will be carried out to predict the consequent attribute A_C based on each of these combinations of explanatory variables (see Table 4).

For each of these attribute combinations we generate 100 subsamples to avoid possible biases in the selection of records.

Third, for each of the scenarios described (Table 1), for each of the 31 combinations of antecedent attributes (Table 4), and for each of the 100 random subsamples, classification models are generated, both with the two intuitive classifiers and with the heuristic method J48. Thus, we have carried out 15,500 heuristic classification models with the J48 method as well as with our own implementation of the intuitive classifier $\mathcal{I}$.

Finally, for both classifiers we calculate their accuracies, from their corresponding confusion matrices by using cross-validation. Therefore, to calculate the success ratio $\mu(\mathcal{X}, \mathcal{D})$ of the random classifier $\mathcal{X}$, we directly use the theoretical result given by Equation (7), and the same for the intuitive classifier $\mathcal{V}$ using Equation (12), while to calculate the success ratio $\mu(\mathcal{I}, \mathcal{D})$ of the intuitive classifier $\mathcal{I}$, we use the confusion matrix obtained by cross-validation. Likewise, the success ratio $\mu(\mathcal{Z}, \mathcal{D})$ of the heuristic classifier, in our case J48, is also calculated by the confusion matrix obtained by cross-validation. From these results, the evaluation measures introduced in Section 2.4 can already be calculated.

Table 4. The 31 combinations of the best five attributes A_1, A_2, A_3, A_4, and A_5 for predicting consequent A_C.

Comb.	Antecedents	Comb.	Antecedents	Comb.	Antecedents
#1	A_5	#12	A_2, A_3	#23	A_1, A_3, A_4, A_5
#2	A_4	#13	A_2, A_3, A_5	#24	A_1, A_2
#3	A_4, A_5	#14	A_2, A_3, A_4	#25	A_1, A_2, A_5
#4	A_3	#15	A_2, A_3, A_4, A_5	#26	A_1, A_2, A_4
#5	A_3, A_5	#16	A_1	#27	A_1, A_2, A_4, A_5
#6	A_3, A_4	#17	A_1, A_5	#28	A_1, A_2, A_3
#7	A_3, A_4, A_5	#18	A_1, A_4	#29	A_1, A_2, A_3, A_5
#8	A_2	#19	A_1, A_4, A_5	#30	A_1, A_2, A_3, A_4
#9	A_2, A_5	#20	A_1, A_3	#31	A_1, A_2, A_3, A_4, A_5
#10	A_2, A_4	#21	A_1, A_3, A_5		
#11	A_2, A_4, A_5	#22	A_1, A_3, A_4		

Therefore, we have an experimental design with two factors (entropy scenarios and attribute combinations) with 100 replications for each cross combination of factors. This allows us to analyze in depth how an heuristic classifier performs when we consider both the entropy of the consequent variable and the number of attributes used as antecedents.

Therefore, the experiment illustrates both how the evaluation measures work and how to analyze the effects of entropy and the number of selected attributes to predict the consequent variable in the performance of an heuristic classifier.

3.3. Results

After performing all the classification models described in the previous section for each of the five scenarios, each model is subjected to a cross-validation test, and confusion matrices are determined. With this information we can calculate some performance measures for the heuristic classifier J48. The simplest performance measure is accuracy,

which measures the success rate in the prediction. Table 5 shows the accuracy of J48 and the intuitive classifier $\mathcal{I}$ for each of the five scenarios considered.

Table 5. Accuracy measures for the random classifier $\mathcal{X}$, the intuitive classifier $\mathcal{V}$, J48 and the intuitive classifier $\mathcal{I}$ when using combination of attributes A31 for each scenario. The accuracy and the mean absolute error are calculated as the average accuracy and the average mean absolute error of the 100 subsamples. Results are presented as *accuracy*±*mean absolute error*.

Scenario	$E(acc(\mathcal{X}(\mathcal{D})))$	$acc(\mathcal{V}(\mathcal{D}))$	$acc(J48(\mathcal{D}))$	$acc(\mathcal{I}(\mathcal{D}))$
S1	0.6250	0.7500 ± 0.2500	0.7489 ± 0.3739	0.7481 ± 0.2519
S2	0.5556	0.6667 ± 0.3333	0.6724 ± 0.4358	0.6729 ± 0.3271
S3	0.5000	0.5000 ± 0.5000	0.5241 ± 0.4856	0.4835 ± 0.5165
S4	0.5556	0.6667 ± 0.3333	0.6751 ± 0.4366	0.6766 ± 0.3234
S5	0.6250	0.7465 ± 0.2535	0.7543 ± 0.3734	0.7537 ± 0.2487

In Table 5, we observe that, for this dataset, the performance of J48 is on average slightly better than the performance of the intuitive classifier $\mathcal{I}$, but the mean absolute errors for J48 are worse than the mean absolute errors of the intuitive classifier $\mathcal{I}$ except for S5. However, this comparison could be analyzed in more detail considering other aspects such as the number of times that one method beats the other or the entropy. Likewise, the improvements with respect to the intuitive classifier $\mathcal{V}$ are not too great, which would mean that either the model is not very good, or that in this specific case the use of information from other attributes and/or classifiers more sophisticated do not provide noticeable improvements over the intuitive classifier $\mathcal{V}$.

We now consider that a classifier beats another classifier each time that the first correctly classifies a number of items from the test set higher than the items correctly classified by second. When the reverse occurs, we will say that the second classifier beats the first. When the difference between the items well classified by both methods is 0, we will say that a draw has occurred. The number of times that J48 and the intuitive classifier $\mathcal{I}$ win for each scenario and each combination of the best five attributes are shown in Tables A1–A5 in Appendix A. Table 6 summarizes the percentage of times each method wins for each scenario.

Table 6. Percentage of times each method wins in each of the 3100 instances considered (100 subsamples for each of the 31 combinations of the best five attributes) for each scenario given in Table 1.

Scenario	J48 wins	$\mathcal{V}$ wins	J48 wins	$\mathcal{I}$ wins	$\mathcal{I}$ wins	$\mathcal{V}$ wins
S1	10.42	41.71	46.03	23.94	18.39	58.13
S2	67.65	15.65	24.48	37.39	74.90	13.29
S3	97.55	0.16	73.48	4.45	32.52	66.90
S4	78.13	0.26	16.52	42.19	84.52	0.00
S5	98.03	1.23	76.90	15.16	97.29	2.00
Average %	70.36	11.80	47.48	24.63	61.52	28.06

In Table 6, we observe that J48 classifies better than the intuitive method $\mathcal{I}$ in 47.48% of the instances, while the intuitive method $\mathcal{I}$ classifies better than J48 in 24.63% of the instances. J48 classifies particularly better in scenarios S5 and S3, while the intuitive method $\mathcal{I}$ classifies better in scenarios S2 and S4. Moreover, J48 clearly beats the intuitive classifier $\mathcal{V}$ in all scenarios except in S1, while the intuitive method $\mathcal{I}$ classifies better than the intuitive classifier $\mathcal{V}$ in scenarios S2, S4 and S5. Therefore, in absolute terms we can say that J48 performs reasonably well with respect to the dataset used. However, in addition to knowing whether one method classifies better than another, it is even more relevant to know how much better it classifies in relative terms as mentioned above. In this sense, having a benchmark is important to assess how much improvement there is when compared to

it. In Tables A1–A5 in Appendix A, we can find the evaluation measures introduced in Section 2.4 applied to the average of the results obtained for the 100 subsamples for each combination of the best attributes when J48 and the intuitive classifier are used. Table 7 summarizes these measures for each of the five scenarios considered.

Table 7. Intervals of values of evaluation measure $\gamma^{\mathcal{X}}$ for J48 and the intuitive method $\mathcal{I}$, and intervals of values of evaluation measure Γ for J48 for each scenario.

Scenario	$\gamma^{\mathcal{X}}(J48)$	$\gamma^{\mathcal{X}}(\mathcal{I})$	$\gamma^{\mathcal{V}}(J48)$	$\gamma^{\mathcal{V}}(\mathcal{I})$	$\Gamma(J48)$
S1	0.3303–0.3333	0.3282–0.3333	−0.0015–0.0000	−0.0025–0.0000	0.0000–0.0032
S2	0.2499–0.2636	0.2498–0.2650	−0.0001–0.0182	−0.0001–0.0200	−0.0009–0.0001
S3	0.0004–0.0492	−0.0903–0.0419	0.0004–0.0492	−0.0903–0.0419	0.0049–0.0844
S4	0.2500–0.2693	0.2500–0.2729	0.0000–0.0257	0.0000–0.0306	−0.0026–0.0004
S5	0.3134–0.3635	0.3125–0.3627	−0.0087–0.0524	−0.0091–0.0512	0.0010–0.0027

First note that in this case the measure $\gamma^{\mathcal{X}}$ coincides in all scenarios with the Scott's π. On the other hand, beyond that which was analyzed when we evaluate which method best classifies simply in terms of the number of successes, in Table 7 we observe that the performance of J48 and the intuitive classifier $\mathcal{I}$ are very similar when compared with the random classifier $\mathcal{X}$ and the intuitive classifier $\mathcal{V}$ for each of the scenarios (columns corresponding to evaluation measures $\gamma^{\mathcal{X}}$ and $\gamma^{\mathcal{V}}$). This is clearly reflected in the evaluation measure Γ of J48, which is the result of comparison with the intuitive method $\mathcal{I}$ (see Definition 2). We also observe that, for the dataset used in the experiment, the performance of the classifiers improves with the decrease in the entropy of the consequent, i.e., the lower the entropy, the higher the performance of both classifiers with respect to the random classifier $\mathcal{X}$.

Moreover, if we look, for example, at scenario S3, $\gamma^{\mathcal{V}}(J48)$ tells us that J48 improves the performance of the intuitive classifier $\mathcal{V}$, which only uses the information provided by the frequency distribution of the target attribute, by as much as 5% using the information provided by attributes other than the target attribute. Therefore, this percentage can be interpreted as the exploitation that J48 makes of this additional information. If we now look at $\Gamma(J48)$, then we see that this improvement reaches almost 8.5% with respect to the intuitive classifier $\mathcal{I}$. This percentage can be interpreted as the better exploitation that J48 makes of the information than the intuitive classifier $\mathcal{I}$. At this point, one could already assess, taking into account the practical implications of better performance, whether the use of a more sophisticated classifier than the two intuitive classifiers is worth it.

Therefore, comparison with a benchmark is important because performance measures often do not reflect what is actually gained with respect to a simple, even random, way of classifying. Therefore, the use of measures based on simple benchmark classifiers that somehow capture the entropy of the dataset seems appropriate and provides relevant information on the performance of the classifiers. In particular, the use of both intuitive classifiers as benchmark seems reasonable, because although as classifiers they have been discarded in favor of other classifiers that use more modern and elaborate technologies, they are still easy enough to understand and intuitive as to at least consider them as benchmark classifiers when measuring the performance of classifiers, as the random classifier is commonly used in machine learning.

3.4. Extensive experiment

In this subsection we present the results of an extensive experiment in which we consider four heuristic classifiers besides the intuitive classifier $\mathcal{I}$, and 11 datasets. In particular, we consider four classification algorithms implemented in Weka [75,81–83], J48, Naïve Bayes, SMO, and Random Forest, which have been briefly described in Section 2.3; and 11 datasets from UCI Machine Learning Repository [87] which have been described in Section 3.1.

The purpose of this extensive analysis is to check whether the results obtained in the previous experiment are repeated for other classifiers and other datasets. The first step in all cases is to select the 5 most relevant attributes by using the feature selection method described in Section 2.1. The results are shown in Table 8.

Table 8. The five most relevant attributes of each dataset according to Gain Ratio Attribute Evaluation (see Section 2.1).

#	Dataset	1st	2nd	3rd	4th	5th
1	Thyroid	hypopit.	pregnant	psych	goitre	referral_source
2	Healthy	C4	C3	C6	C5	C7
3	Avila	F5	F1	F9	F3	F7
4	Adult	Mar.Sta.	Relat.	Sex	Age	Educ
5	Nursery	F2	F1	F7	F5	F4
6	Bank	poutcome	contact	housing	month	loan
7	HTRU2	A3	A1	A4	A6	A5
8	Connect-4	g6	d3	f6	d2	b6
9	Tic-tac-toe	m-m-s	b-l-s	t-l-s	t-r-s	b-r-s
10	Credit	A9	A10	A4	A5	A6
11	Mushroom	odor	gill-size	stalk-surface-above-ring	spore-print-color	ring-type

Then the five classifiers are applied with the selection of attributes in Table 8. We calculate their accuracies, from their corresponding confusion matrices by using cross-validation. The resulting accuracies for each classifier and dataset are shown in Table 9.

Table 9. Accuracies and mean absolute errors for the five classifiers and the 11 datasets. Results are presented as *accuracy*±*mean absolute error*.

#	Dataset	$\mathcal{I}$	J48	SMO	Naïve Bayes	Random Forest
1	Thyroid	0.743 ± 0.257	0.744 ± 0.381	0.743 ± 0.257	0.741 ± 0.373	0.743 ± 0.374
2	Healthy	0.953 ± 0.024	0.963 ± 0.030	0.949 ± 0.255	0.935 ± 0.042	0.963 ± 0.028
3	Avila	0.653 ± 0.058	0.666 ± 0.074	0.600 ± 0.141	0.610 ± 0.087	0.657 ± 0.069
4	Adult	0.825 ± 0.175	0.824 ± 0.250	0.818 ± 0.182	0.763 ± 0.240	0.824 ± 0.236
5	Nursery	0.508 ± 0.197	0.548 ± 0.224	0.508 ± 0.265	0.531 ± 0.233	0.508 ± 0.224
6	Bank	0.885 ± 0.115	0.894 ± 0.186	0.893 ± 0.107	0.890 ± 0.175	0.893 ± 0.167
7	HTRU2	0.971 ± 0.029	0.971 ± 0.049	0.969 ± 0.031	0.969 ± 0.050	0.971 ± 0.048
8	Connect-4	0.658 ± 0.228	0.665 ± 0.313	0.665 ± 0.318	0.663 ± 0.318	0.665 ± 0.311
9	Tic-tac-toe	0.801 ± 0.193	0.794 ± 0.258	0.753 ± 0.247	0.753 ± 0.374	0.840 ± 0.209
10	Credit	0.859 ± 0.141	0.862 ± 0.220	0.858 ± 0.142	0.861 ± 0.193	0.848 ± 0.199
11	Mushroom	1.000 ± 0.000	1.000 ± 0.000	0.999 ± 0.001	0.999 ± 0.020	1.000 ± 0.000

In Tables 10 and 11, we present the results obtained when $\gamma^{\mathcal{X}}$ and $\gamma^{\mathcal{X}}$ are used as evaluation performance measure.

Table 10. Evaluation measure $\gamma^{\mathcal{X}}$ for the five classifiers and the 11 datasets, and $[0,1]$-normalized Shannon entropy of the consequent attribute for each dataset.

#	Dataset	Entropy	$\gamma^{\mathcal{X}}(\mathcal{I})$	$\gamma^{\mathcal{X}}(J48)$	$\gamma^{\mathcal{X}}(SMO)$	$\gamma^{\mathcal{X}}(NB)$	$\gamma^{\mathcal{X}}(RF)$
1	Thyroid	0.829	0.335	0.338	0.335	0.330	0.335
2	Healthy	0.632	0.901	0.922	0.893	0.864	0.922
3	Avila	0.737	0.549	0.566	0.480	0.493	0.554
4	Adult	0.796	0.521	0.519	0.502	0.352	0.519
5	Nursery	0.739	0.279	0.338	0.279	0.313	0.279
6	Bank	0.521	0.443	0.487	0.482	0.468	0.482
7	HTRU2	0.442	0.826	0.826	0.814	0.814	0.826
8	Connect-4	0.769	0.312	0.326	0.326	0.322	0.326
9	Tic-tac-toe	0.931	0.561	0.545	0.455	0.455	0.647
10	Credit	0.991	0.715	0.721	0.713	0.719	0.692
11	Mushroom	0.999	1.000	1.000	0.998	0.998	1.000

Table 11. Evaluation measure $\gamma^{\mathcal{V}}$ for the five classifiers and the 11 datasets, and the accuracy of the intuitive classifier $\mathcal{X}$.

#	Dataset	acc($\mathcal{V}$)	$\gamma^{\mathcal{V}}(\mathcal{I})$	$\gamma^{\mathcal{V}}(J48)$	$\gamma^{\mathcal{V}}(SMO)$	$\gamma^{\mathcal{V}}(NB)$	$\gamma^{\mathcal{V}}(RF)$
1	Thyroid	0.738	0.018	0.022	0.018	0.011	0.018
2	Healthy	0.686	0.850	0.882	0.838	0.793	0.882
3	Avila	0.411	0.411	0.433	0.321	0.338	0.418
4	Adult	0.759	0.273	0.269	0.244	0.016	0.269
5	Nursery	0.333	0.262	0.322	0.262	0.297	0.262
6	Bank	0.883	0.017	0.094	0.085	0.060	0.085
7	HTRU2	0.908	0.683	0.683	0.661	0.661	0.683
8	Connect-4	0.658	0.000	0.020	0.020	0.014	0.020
9	Tic-tac-toe	0.653	0.426	0.406	0.287	0.287	0.538
10	Credit	0.555	0.683	0.690	0.681	0.688	0.658
11	Mushroom	0.518	1.000	1.000	0.998	0.998	1.000

As we mentioned before, we know that the $\gamma^{\mathcal{X}}$ measure is close related to the κ and π measures. In Tables 10 and 11, we observe that a higher entropy in the consequent attribute does not mean a worse performance of the classifiers [70]. This is not surprising since all classifiers use not only the frequency distribution information of the consequent attribute, but also the information provided about it by the remaining attributes in the dataset. Therefore, it seems appropriate to use the entropy of the entire dataset as a reference when assessing the performance of the classifiers. This entropy is somehow captured by the intuitive classifier $\mathcal{I}$ as explained earlier. In Table 12, we present the results obtained when Γ is used as evaluation performance measure.

Table 12. Evaluation measure Γ for the four heuristic classifiers and the 11 datasets.

#	Dataset	$\Gamma(J48)$	$\Gamma(SMO)$	$\Gamma(NB)$	$\Gamma(RF)$
1	Thyroid	0.004	0.000	−0.003	0.000
2	Healthy	0.213	−0.004	−0.019	0.213
3	Avila	0.037	−0.081	−0.066	0.012
4	Adult	−0.001	−0.008	−0.075	−0.001
5	Nursery	0.081	0.000	0.047	0.000
6	Bank	0.078	0.070	0.043	0.070
7	HTRU2	0.000	−0.002	−0.002	0.000
8	Connect-4	0.020	0.020	0.015	0.020
9	Tic-tac-toe	−0.009	−0.060	−0.060	0.196
10	Credit	0.021	−0.001	0.014	−0.013
11	Mushroom	0.000	−0.001	−0.001	0.000

The intuitive classifier $\mathcal{I}$ will have better accuracy the lower the conditional entropy of the target attribute given the entire dataset (or the subset of selected attributes if a selection feature is previously carried out), therefore, it will be more difficult for a classifier to significantly improve the classification results of this intuitive classifier. On the other hand, it is necessary to emphasize that the selection of the best subset of attributes has been relevant throughout the classification process, since the method used is based on the reduction of entropy. In this sense, Γ would measure how much a classifier contributes to the complete classification procedure with respect to what is contributed by the attribute selection process. Therefore, Γ offers different information than other performance measures of the classifiers, which we consider to be interesting. The aim, therefore, is not to substitute for any known performance measure, but to provide a measure of a different aspect of the performance of a classifier.

Finally, in Tables 11 and 12, we observe that performance measures $\gamma^{\mathcal{V}}$ and Γ provide complementary information about classifiers. In Table 11, we can observe how each classifier takes advantage of the information provided by the attributes in the dataset to better classify the target attribute, while in Table 12 we can observe how much better than the intuitive classifier $\mathcal{I}$ are classifiers capable of using the information in the dataset to correctly predict the classes of the target attribute.

4. Discussion and Conclusions

In the experiment we have shown that both feature selection and the entropy of the consequent attribute may be relevant to the performance result of an algorithm of classification. Therefore, it would appear to be of interest to consider the diversity of the response variable or the dataset when evaluating a classifier. In addition, the effect of entropy is observed, in the sense that the lower the entropy, the higher the success rate in the classifications, which seems intuitively reasonable. On the other hand, we observe in the experiment that choosing a greater number of features does not always provide a better performance of the classification algorithm, so this kind of analysis is relevant when selecting an adequate number of features, above all when the feature selection algorithm has not used the classifier algorithm for optimal selection. A rigorous analysis of the latter can be found in [104].

The performance measures of classifiers which only use the results of the classification algorithm itself, such as the ratio of successes (accuracy), do not really provide information on how it is really capable of classifying correctly with respect to unsophisticated methods. For this reason, the use of relative measures when compared with simple benchmark classifiers is important, because they give us information about the relationship between the gain in the correct classification of instances and the effort made in the design of new classifiers with respect to the use of simple and intuitive classifiers, i.e., we can better assess the real improvement provided by the classification algorithm. Moreover, if the benchmark classifier incorporates some type of additional information, such as different aspects of

the entropy of all the dataset or the consequent attribute, the information provided by the performance measure will be even more relevant.

In this paper, three simple classifiers have been used, the random classifier $\mathcal{X}$, the intuitive classifier $\mathcal{V}$, and the intuitive classifier $\mathcal{I}$. The first two simply use the distribution of the consequent attribute to classify and we have shown that they are closely related to the entropy of that attribute, while the third uses the entire distribution of the whole data set to classify and its performance is close to the conditional entropy of the consequent attribute given the remaining attributes (or a subset of attributes if feature selection is previously applied) in the dataset . These three classifiers have been used as references to introduce three measures of the performance of classifiers. These measure how much a classifier improves (or worsens) over these simple classifiers that are related to certain aspects of the entropy of the consequent attribute within the dataset. Therefore, they are measures that reflect on the performance of the heuristic classifiers, taking into account entropy in some way, and this is important, because the greater the entropy, the greater the difficulty to classify correctly, as has been seen in the experiment, which gives a better idea of the true performance of a classifier. Likewise, the three performance measures of classifiers can be interpreted in terms of proportional reduction of the classification error, which makes these measures easily understandable. In particular, $\gamma^{\mathcal{X}}$ is closely related to the well-known κ and π measures, and provides information on how much a classifier improves the classification results relative to a random classifier that it only takes into account the information contained in the frequency distribution of the target attribute classes. $\gamma^{\mathcal{V}}$ gives information on how a classifier is capable to use the information contained in the whole dataset (or a subset of the dataset) to improve the classification results relative to a classifier that it only uses the information of the frequency distribution of the target attribute classes and always predicts the most likely class. Last, Γ provides information on how much a classifier improves the classification results when using a more elaborate technology of managing data than the intuitive classifier $\mathcal{I}$ which simply predicts the most likely class given a particular profile of attributes in the dataset.

To conclude, although the two intuitive classifiers used in this paper were already discarded in favor of more modern and sophisticated classifiers, we believe that they are still useful as benchmark classifiers, as the random classifier is commonly used in machine learning, and then to design performance measures based on them which we have shown throughout this work that provide relevant information about the performance of classifiers different from other performance measures.

Author Contributions: Conceptualization, Y.O., A.R., J.J.R.-S. and J.S.-S.; methodology, A.R. and J.S.-S.; software, Y.O. and J.J.R.-S.; validation, Y.O., A.R., J.J.R.-S. and J.S.-S.; formal analysis, A.R. and J.S.-S.; investigation, Y.O., A.R., J.J.R.-S. and J.S.-S.; resources, Y.O., A.R., J.J.R.-S. and J.S.-S.; data curation, Y.O., A.R. and J.J.R.-S.; writing—original draft prepA.R.ation, A.R. and J.S.-S.; writing—review and editing, A.R. and J.S.-S.; supervision, A.R. and J.S.-S. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The datasets used in the experiments can be found at UCI Machine Learning Repository: https://archive.ics.uci.edu/ml/index.php. (accessed on 23 April 2021)

Acknowledgments: We are most grateful to two anonymous Academic Editors, and two anonymous reviewers for their very helpful comments and suggestions for improving this paper.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

MDPI	Multidisciplinary Digital Publishing Institute
DOAJ	Directory of open access journals

Appendix A. Tables

Table A1. Scenario S1, 3.200 rows, 3:1 ratio of positive/negative values for target variable, 100 subsamples per combination, and the gain ratio attribute evaluations of the five best variables are 0.036, 0.037, 0.033, 0.034, and 0.029 (from most to least relevant).

Comb.	Antecedents	J48 wins	$\mathcal{I}$ wins	$\gamma^{\mathcal{X}}(J48)$	$\Gamma(J48)$	$\gamma^{\mathcal{X}}(\mathcal{I})$	$\gamma^{\mathcal{V}}(J48)$	$\gamma^{\mathcal{V}}(\mathcal{I})$
#1	5	45	10	0.3328	0.0019	0.3316	−0.0003	−0.0009
#2	4	39	33	0.3326	0.0003	0.3324	−0.0004	−0.0005
#3	45	56	28	0.3320	0.0022	0.3306	−0.0007	−0.0014
#4	3	39	16	0.3318	0.0024	0.3301	−0.0008	−0.0016
#5	35	55	33	0.3314	0.0031	0.3294	−0.0010	−0.0020
#6	34	57	30	0.3311	0.0029	0.3291	−0.0011	−0.0021
#7	345	56	34	0.3305	0.0032	0.3284	−0.0014	−0.0025
#8	2	0	0	0.3333	0.0000	0.3333	0.0000	0.0000
#9	25	45	10	0.3328	0.0019	0.3316	−0.0003	−0.0009
#10	24	44	28	0.3326	0.0005	0.3322	−0.0004	−0.0006
#11	245	61	26	0.3321	0.0025	0.3304	−0.0006	−0.0015
#12	23	47	24	0.3316	0.0021	0.3302	−0.0009	−0.0016
#13	235	55	36	0.3312	0.0027	0.3294	−0.0011	−0.0020
#14	231	56	32	0.3307	0.0024	0.3291	−0.0013	−0.0021
#15	2345	58	33	0.3303	0.0030	0.3282	−0.0015	−0.0025
#16	1	0	0	0.3333	0.0000	0.3333	0.0000	0.0000
#17	15	45	10	0.3328	0.0019	0.3316	−0.0002	−0.0009
#18	14	40	32	0.3326	0.0004	0.3324	−0.0004	−0.0005
#19	145	57	27	0.3321	0.0023	0.3306	−0.0006	−0.0014
#20	13	39	16	0.3318	0.0024	0.3301	−0.0008	−0.0016
#21	135	53	33	0.3312	0.0028	0.3294	−0.0011	−0.0020
#22	134	55	31	0.3310	0.0028	0.3291	−0.0012	−0.0021
#23	1345	58	33	0.3305	0.0032	0.3284	−0.0014	−0.0025
#24	12	0	0	0.3333	0.0000	0.3333	0.0000	0.0000
#25	125	45	10	0.3328	0.0019	0.3316	−0.0003	−0.0009
#26	124	44	28	0.3327	0.0007	0.3322	−0.0003	−0.0006
#27	1245	62	25	0.3321	0.0025	0.3304	−0.0006	−0.0015
#28	123	47	24	0.3316	0.0022	0.3302	−0.0009	−0.0016
#29	1235	55	35	0.3311	0.0026	0.3294	−0.0011	−0.0020
#30	1234	57	31	0.3308	0.0026	0.3291	−0.0013	−0.0021
#31	12345	57	34	0.3303	0.0031	0.3282	−0.0015	−0.0025
	Total	1427	742					
	%	46.03	23.94					

Table A2. Scenario S2, 3.600 rows, 2:1 ratio of positive/negative values for target variable, 100 subsamples per combination, and the gain ratio attribute evaluations of the five best variables are 0.050, 0.037, 0.034, 0.032, and 0.022 (from most to least relevant).

Comb.	Antecedents	J48 wins	$\mathcal{I}$ wins	$\gamma^{\mathcal{X}}(J48)$	$\Gamma(J48)$	$\gamma^{\mathcal{X}}(\mathcal{I})$	$\gamma^{\mathcal{V}}(J48)$	$\gamma^{\mathcal{V}}(\mathcal{I})$
#1	5	38	35	0.2499	0.0000	0.2498	−0.0001	−0.0001
#2	4	7	34	0.2526	−0.0004	0.2532	0.0034	0.0043
#3	45	35	45	0.2525	−0.0003	0.2530	0.0034	0.0041
#4	3	1	12	0.2611	−0.0004	0.2617	0.0148	0.0156
#5	35	43	34	0.2604	−0.0002	0.2607	0.0139	0.0143
#6	34	6	41	0.2636	−0.0008	0.2649	0.0182	0.0198
#7	345	33	54	0.2625	−0.0008	0.2638	0.0167	0.0184
#8	2	1	0	0.2500	0.0000	0.2500	0.0000	0.0000
#9	25	38	35	0.2499	0.0000	0.2498	−0.0001	−0.0001
#10	24	31	34	0.2524	−0.0003	0.2529	0.0032	0.0038
#11	245	41	46	0.2523	−0.0003	0.2527	0.0030	0.0036
#12	23	9	46	0.2612	−0.0007	0.2623	0.0150	0.0163
#13	235	37	48	0.2605	−0.0005	0.2612	0.0140	0.0150
#14	231	26	59	0.2636	−0.0009	0.2650	0.0181	0.0200
#15	2345	34	57	0.2628	−0.0008	0.2640	0.0171	0.0187
#16	1	0	0	0.2500	0.0000	0.2500	0.0000	0.0000
#17	15	38	35	0.2499	0.0000	0.2498	−0.0001	−0.0001
#18	14	7	34	0.2525	−0.0004	0.2532	0.0034	0.0043
#19	145	33	47	0.2524	−0.0004	0.2530	0.0032	0.0041
#20	13	1	12	0.2612	−0.0004	0.2617	0.0149	0.0156
#21	135	43	34	0.2605	−0.0002	0.2607	0.0140	0.0143
#22	134	6	41	0.2636	−0.0008	0.2649	0.0182	0.0198
#23	1345	36	51	0.2628	−0.0007	0.2638	0.0170	0.0184
#24	12	0	0	0.2500	0.0000	0.2500	0.0000	0.0000
#25	125	38	35	0.2499	0.0001	0.2498	−0.0001	−0.0001
#26	124	31	34	0.2524	−0.0003	0.2529	0.0032	0.0038
#27	1245	41	46	0.2523	−0.0002	0.2527	0.0031	0.0036
#28	123	9	45	0.2612	−0.0007	0.2623	0.0150	0.0163
#29	1235	37	48	0.2605	−0.0005	0.2612	0.0140	0.0150
#30	1234	25	60	0.2636	−0.0009	0.2650	0.0182	0.0200
#31	12345	34	57	0.2628	−0.0008	0.2640	0.0170	0.0187
	Total	759	1159					
	%	24.48	37.39					

Table A3. Scenario S3, 4.800 rows, 1:1 ratio of positive/negative values for target variable, 100 subsamples per combination, and the gain ratio attribute evaluations of the five best variables are 0.083, 0.082, 0.028, 0.028, and 0.026 (from most to least relevant).

Comb.	Antecedents	J48 wins	$\mathcal{I}$ wins	$\gamma^{\mathcal{X}}(J48)$	$\Gamma(J48)$	$\gamma^{\mathcal{X}}(\mathcal{I})$	$\gamma^{\mathcal{V}}(J48)$	$\gamma^{\mathcal{V}}(\mathcal{I})$
#1	5	72	1	0.0365	0.0390	−0.0026	0.0365	−0.0026
#2	4	100	0	0.0076	0.0839	−0.0833	0.0076	−0.0833
#3	45	33	0	0.0448	0.0173	0.0279	0.0448	0.0279
#4	3	100	0	0.0067	0.0842	−0.0846	0.0067	−0.0846
#5	35	49	4	0.0417	0.0259	0.0161	0.0417	0.0161
#6	34	100	0	0.0140	0.0833	−0.0756	0.0140	−0.0756
#7	345	18	5	0.0492	0.0076	0.0419	0.0492	0.0419
#8	2	18	0	0.0323	0.0070	0.0255	0.0323	0.0255
#9	25	100	0	0.0343	0.0797	−0.0493	0.0343	−0.0493
#10	24	60	11	0.0324	0.0253	0.0074	0.0324	0.0074

Table A3. *Cont.*

Comb.	Antecedents	J48 wins	$\mathcal{I}$ wins	$\gamma^{\mathcal{X}}(J48)$	$\Gamma(J48)$	$\gamma^{\mathcal{X}}(\mathcal{I})$	$\gamma^{\mathcal{V}}(J48)$	$\gamma^{\mathcal{V}}(\mathcal{I})$
#11	245	100	0	0.0436	0.0806	−0.0402	0.0436	−0.0402
#12	23	57	0	0.0323	0.0235	0.0090	0.0323	0.0090
#13	235	100	0	0.0399	0.0805	−0.0441	0.0399	−0.0441
#14	231	86	2	0.0324	0.0470	−0.0153	0.0324	−0.0153
#15	2345	99	0	0.0487	0.0781	−0.0319	0.0487	−0.0319
#16	1	100	0	0.0004	0.0832	−0.0903	0.0004	−0.0903
#17	15	77	0	0.0372	0.0433	−0.0064	0.0372	−0.0064
#18	14	100	0	0.0075	0.0838	−0.0832	0.0075	−0.0832
#19	145	37	0	0.0448	0.0190	0.0262	0.0448	0.0262
#20	13	100	0	0.0071	0.0844	−0.0845	0.0071	−0.0845
#21	135	50	4	0.0417	0.0274	0.0147	0.0417	0.0147
#22	134	100	0	0.0140	0.0832	−0.0755	0.0140	−0.0755
#23	1345	19	5	0.0492	0.0081	0.0414	0.0492	0.0414
#24	12	15	45	0.0325	0.0049	0.0277	0.0325	0.0277
#25	125	100	0	0.0345	0.0795	−0.0489	0.0345	−0.0489
#26	124	55	28	0.0328	0.0216	0.0115	0.0328	0.0115
#27	1245	100	0	0.0436	0.0811	−0.0407	0.0436	−0.0407
#28	123	49	23	0.0327	0.0192	0.0138	0.0327	0.0138
#29	1235	100	0	0.0399	0.0807	−0.0443	0.0399	−0.0443
#30	1234	84	10	0.0325	0.0437	−0.0117	0.0325	−0.0117
#31	12345	100	0	0.0482	0.0786	−0.0331	0.0482	−0.0331
	Total	2278	138					
	%	73.48	4.45					

Table A4. Scenario S4, 6.000 rows, 1:2 ratio of positive/negative values for target variable, 100 subsamples per combination, and the gain ratio attribute evaluations of the five best variables are 0.122, 0.076, 0.02, 0.015, and 0.013 (from most to least relevant).

Comb.	Antecedents	J48 wins	$\mathcal{I}$ wins	$\gamma^{\mathcal{X}}(J48)$	$\Gamma(J48)$	$\gamma^{\mathcal{X}}(\mathcal{I})$	$\gamma^{\mathcal{V}}(J48)$	$\gamma^{\mathcal{V}}(\mathcal{I})$
#1	5	10	19	0.2652	−0.0004	0.2659	0.0203	0.0212
#2	4	0	0	0.2500	0.0000	0.2500	0.0000	0.0000
#3	45	13	20	0.2653	−0.0004	0.2658	0.0204	0.0211
#4	3	0	0	0.2500	0.0000	0.2500	0.0000	0.0000
#5	35	28	18	0.2651	−0.0003	0.2655	0.0202	0.0207
#6	34	0	0	0.2500	0.0000	0.2500	0.0000	0.0000
#7	345	30	20	0.2651	−0.0002	0.2655	0.0201	0.0206
#8	2	0	0	0.2686	0.0000	0.2686	0.0248	0.0248
#9	25	23	76	0.2686	−0.0022	0.2720	0.0248	0.0293
#10	24	0	38	0.2692	−0.0002	0.2695	0.0256	0.0260
#11	245	21	78	0.2691	−0.0025	0.2728	0.0254	0.0305
#12	23	76	8	0.2686	0.0004	0.2684	0.0248	0.0245
#13	235	23	76	0.2685	−0.0020	0.2715	0.0247	0.0287
#14	231	46	40	0.2692	0.0000	0.2692	0.0256	0.0256
#15	2345	24	76	0.2690	−0.0022	0.2723	0.0254	0.0298
#16	1	0	47	0.2501	−0.0002	0.2505	0.0001	0.0006
#17	15	8	28	0.2653	−0.0004	0.2660	0.0204	0.0213
#18	14	0	47	0.2501	−0.0002	0.2505	0.0001	0.0006
#19	145	11	30	0.2653	−0.0004	0.2659	0.0204	0.0212
#20	13	0	47	0.2501	−0.0002	0.2505	0.0001	0.0006
#21	135	24	25	0.2651	−0.0003	0.2656	0.0202	0.0208
#22	134	0	47	0.2501	−0.0002	0.2505	0.0001	0.0006

Table A4. *Cont.*

Comb.	Antecedents	J48 wins	$\mathcal{I}$ wins	$\gamma^{\mathcal{X}}(J48)$	$\Gamma(J48)$	$\gamma^{\mathcal{X}}(\mathcal{I})$	$\gamma^{\mathcal{V}}(J48)$	$\gamma^{\mathcal{V}}(\mathcal{I})$
#23	1345	27	27	0.2651	−0.0003	0.2656	0.0202	0.0207
#24	12	0	47	0.2687	−0.0002	0.2691	0.0250	0.0254
#25	125	21	76	0.2686	−0.0023	0.2721	0.0247	0.0295
#26	124	0	69	0.2693	−0.0004	0.2699	0.0257	0.0266
#27	1245	18	80	0.2689	−0.0026	0.2729	0.0253	0.0306
#28	123	41	47	0.2687	−0.0001	0.2688	0.0250	0.0251
#29	1235	22	76	0.2685	−0.0020	0.2716	0.0247	0.0288
#30	1234	24	69	0.2693	−0.0002	0.2697	0.0257	0.0262
#31	12345	22	77	0.2691	−0.0022	0.2724	0.0254	0.0299
	Total	512	1308					
	%	16.52	42.19					

Table A5. Scenario S5, between 7820 and 7940 rows, 1:3 ratio of positive/negative values for target variable, 100 subsamples per combination, and the gain ratio attribute evaluations of the five best variables are 0.102, 0.134, 0.016, 0.013, and 0.010 (from most to least relevant).

Comb.	Antecedents	J48 wins	$\mathcal{I}$ wins	$\gamma^{\mathcal{X}}(J48)$	$\Gamma(J48)$	$\gamma^{\mathcal{X}}(\mathcal{I})$	$\gamma^{\mathcal{V}}(J48)$	$\gamma^{\mathcal{V}}(\mathcal{I})$
#1	5	59	41	0.3336	0.0013	0.3327	0.0191	0.0179
#2	4	81	0	0.3339	0.0016	0.3328	0.0181	0.0165
#3	45	59	41	0.3234	0.0012	0.3226	−0.0003	−0.0007
#4	3	81	0	0.3288	0.0016	0.3277	0.0090	0.0074
#5	35	59	41	0.3209	0.0012	0.3201	−0.0022	−0.0026
#6	34	81	0	0.3187	0.0016	0.3176	−0.0037	−0.0043
#7	345	59	41	0.3310	0.0012	0.3302	0.0141	0.0129
#8	2	73	10	0.3187	0.0012	0.3179	−0.0039	−0.0043
#9	25	59	41	0.3234	0.0012	0.3226	−0.0003	−0.0007
#10	24	73	10	0.3338	0.0012	0.3330	0.0190	0.0178
#11	245	59	41	0.3209	0.0012	0.3201	−0.0020	−0.0024
#12	23	74	9	0.3263	0.0012	0.3254	0.0041	0.0028
#13	235	59	41	0.3109	0.0012	0.3101	−0.0087	−0.0091
#14	231	74	9	0.3313	0.0013	0.3305	0.0146	0.0133
#15	2345	59	41	0.3234	0.0012	0.3226	−0.0006	−0.0010
#16	1	80	1	0.3462	0.0015	0.3452	0.0377	0.0363
#17	15	89	10	0.3331	0.0023	0.3316	0.0112	0.0089
#18	14	93	1	0.3438	0.0020	0.3425	0.0319	0.0300
#19	145	89	9	0.3384	0.0026	0.3366	0.0220	0.0194
#20	13	75	4	0.3424	0.0014	0.3414	0.0291	0.0277
#21	135	93	5	0.3397	0.0026	0.3380	0.0239	0.0214
#22	134	89	4	0.3246	0.0016	0.3235	−0.0018	−0.0024
#23	1345	95	3	0.3398	0.0027	0.3380	0.0240	0.0214
#24	12	74	9	0.3541	0.0013	0.3532	0.0524	0.0512
#25	125	89	8	0.3232	0.0024	0.3215	−0.0026	−0.0034
#26	124	84	9	0.3336	0.0015	0.3326	0.0124	0.0109
#27	1245	89	8	0.3308	0.0026	0.3290	0.0070	0.0044
#28	123	70	13	0.3347	0.0010	0.3341	0.0141	0.0131
#29	1235	91	5	0.3346	0.0025	0.3330	0.0144	0.0119
#30	1234	81	11	0.3398	0.0013	0.3390	0.0236	0.0223
#31	12345	94	4	0.3447	0.0025	0.3431	0.0343	0.0319
	Total	2384	470					
	%	76.90	15.16					

References

1. Aggarwal, C.C. *Data Mining: The Textbook*; Springer: Berlin/Heidelberg, Germany, 2015.
2. Kelleher, J.D.; Namee, B.M.; D'Arcy, A. *Machine Learning for Predictive Data Analytics: Algorithms, Worked Examples, and Case Studies*; The MIT Press: Cambridge, MA, USA, 2015.
3. Kubat, M. *An Introduction to Machine Learning*, 2nd ed.; Springer: Berlin/Heidelberg, Germany, 2017.
4. Skiena, S.S. *The Data Science Design Manual*; Springer: Berlin/Heidelberg, Germany, 2017.
5. Shannon, C.E. A mathematical theory of communication. *Bell Syst. Tech. J.* **1948**, *27*, 379–423.
6. Rényi, A. On measures of entropy and information. In Proceedings of the 4th Berkeley Symposium on Mathematics, Statistics and Probability, Berkeley, CA, USA, 20 June–30 July 1960; Neyman, J., Ed.; University of California Press: Berkeley, CA, USA, 1961; pp. 547–561.
7. Tsallis, C. Possible generalization of Boltzmann—Gibbs statistics, *J. Stat. Phys.* **1988**, *52*, 479–487.
8. Amigó, J.M.; Balogh, S.G.; Hernández, S. A Brief Review of Generalized Entropies. *Entropy* **2018**, *20*, 813.
9. Orenes, Y.; Rabasa, A.; Pérez-Martín, A.; Rodríguez-Sala, J.J.; Sánchez-Soriano, J. A Computational Experience For Automatic Feature Selection On Big Data Frameworks. *Int. J. Des. Nat. Ecodynamics* **2016**, *11*, 168–177.
10. Fu, K.S.; Cardillo, G.P. An Optimum Finite Sequential Procedure For Feature Selection Furthermore, Pattern Classification. *IEEE Trans. Autom. Control* **1967**, *AC12*, 588.
11. Cardillo, G.P.; Fu, K.S. Divergence Furthermore, Linear Classifiers For Feature Selection. *IEEE Trans. Autom. Control.* **1967**, *AC12*, 780.
12. Chien, Y.T. Adaptive strategies of selecting feature subsets in pattern recognition. In Proceedings of the IEEE Symposium on Adaptive Processes (8th) Decision and Control, University Park, PA, USA, 17–19 November 1969; p. 36.
13. Jurs, P.C.; Kowalski, B.R.; Isenhour, T.L.; Reilley, C.N. Computerized learning machines applied to chemical problems. Convergence rate and predictive ability of adaptive binary pattern classifiers. *Anal. Chem.* **1969**, *41*, 690–695.
14. Jurs, P.C. Mass spectral Feature Selection and structural correlations using computerized learning machines. *Anal. Chem.* **1970**, *42*, 1633–1638.
15. Narendra, P.; Fukunaga, K. Branch and bound algorithm for Feature subset Selection. *IEEE Trans. Comput.* **1977**, *26*, 917–922.
16. Pudil, P.; Novovicova, J.; Kittler, J. Floating Search Methods in Feature-Selection. *Pattern Recognit. Lett.* **1994**, *15*, 1119–1125.
17. Siedlecki, W.; Sklansky, J. A note on genetic algorithms for largescale Feature-Selection. *Pattern Recognit. Lett.* **1989**, *10*, 335–347.
18. Leardi, R.; Boggia, R.; Terrile, M. Genetic algorithms as a strategy for Feature-Selection. *J. Chemom.* **1992**, *6*, 267–281.
19. Yang, J.H.; Honavar, V. Feature subset Selection using a genetic algorithm. *IEEE Intell. Syst. Appl.* **1998**, *13*, 44–49.
20. John, G.; Kohavi, R.; Pfleger, K. Irrelevant features and the subset selection problem. In Proceedings of the Fifth International Conference on Machine Learning, New Brunswick, NJ, USA, 10–13 July 1994; pp. 121–129.
21. Kohavi, R.; John, G.H. Wrappers for Feature subset Selection. *Artif. Intell.* **1997**, *97*, 273–324.
22. Mitra, P.; Murthy, C.A.; Pal, S.K. Unsupervised feature selection using feature similarity. *IEEE Trans. Pattern Anal. Mach. Intell.* **2002**, *24*, 301–312.
23. Yu, L.; Liu, H. Efficient Feature Selection via analysis of relevance and redundancy. *J. Mach. Learn. Res.* **2004**, *5*, 1205–1224.
24. Peng, H.C.; Long, F.H.; Ding, C. Feature Selection based on mutual information: Criteria of max-dependency, max-relevance, and minredundancy. *IEEE Trans. Pattern Anal. Mach. Intell.* **2005**, *27*, 1226–1238.
25. Trabelsia, M.; Meddouria, N.; Maddourib, M. A New Feature Selection Method for Nominal Classifier based on Formal Concept Analysis. *Procedia Comput. Sci.* **2017**, *112*, 186–194.
26. Meddouri, N.; Khoufi, H.; Maddouri, M. Parallel learning and classification for rules based on formal concepts. *Procedia Comput. Sci.* **2014**, *35*, 358–367.
27. Cohen, S.; Dror, G.; Ruppin, G. Feature Selection via Coalitional Game Theory. *Neural Comput.* **2007**, *19*, 1939–1961.
28. Afghah, F.; Razi, A.; Soroushmehr, R.; Ghanbari, H.; Najarian, K. Game Theoretic Approach for Systematic Feature Selection; Application in False Alarm Detection in Intensive Care Units. *Entropy* **2018**, *20*, 190.
29. Duch, W.; Wieczorek, T.; Biesiada, J.; Blachnik, M. Comparison of feature ranking methods based on information entropy. In Proceedings of the 2004 IEEE International Joint Conference on Neural Networks (IEEE Cat. No.04CH37541), Budapest, Hungary, 25–29 July 2004; Volume 2, pp. 1415–1419.
30. Aremu, O.O.; Cody, R.A.; Hyland-Wood, D.; McAree, P.R. A relative entropy based feature selection framework for asset data in predictive maintenance, *Comput. Ind. Eng.* **2020**, *145*, 106536.
31. Bai, L.; Han, Z.; Ren, J.; Qin, X. Research on feature selection for rotating machinery based on Supervision Kernel Entropy Component Analysis with Whale Optimization Algorithm, *Appl. Soft Comput.* **2020**, *92*, 106245.
32. Qu, Y.; Li, R.; Deng, A.; Shang, C.; Shen, Q. Non-unique decision differential entropy-based feature selection, *Neurocomputing* **2020**, *393*, 187–193.
33. Revanasiddappa, M.B.; Harish, B.S. A New Feature Selection Method based on Intuitionistic Fuzzy Entropy to Categorize Text Documents, *Int. J. Interact. Multimed. Artif. Intell.* **2018**, *5*, 106–117.
34. Zhao, J.; Liang, J.; Dong, Z.; Tang, D.; Liu, Z. Accelerating information entropy-based feature selection using rough set theory with classified nested equivalence classes, *Pattern Recognit.* **2020**, *107*, 107517.
35. Liu, H.; Yu, L. Toward integrating Feature Selection algorithms for classification and clustering. *IEEE Trans. Knowl. Data Eng.* **2005**, *17*, 491–502.

36. Quinlan, J.R. Induction of decision tree. *Mach. Learn.* **1986**, *1*, 81–106.
37. Quinlan, J.R. *C4.5: Programs for Machine Learning*, 1st ed.; Morgan Kaufmann Publishers: San Mateo, CA, USA, 1992.
38. Breiman, L.; Friedman, J.H.; Olshen, R.A.; Stone, C.J. *Classification and Regression Trees*; Wadsworth International Group: Belmont, CA, USA 1984.
39. Ho, T.K. Random Decision Forests. In Proceedings of the 3rd International Conference on Document Analysis and Recognition, Montreal, QC, USA, 14–16 August 1995; pp. 278–282.
40. Ho, T.K. The Random Subspace Method for Constructing Decision Forests'. *IEEE Trans. Pattern Anal. Mach. Intell.* **1998**, *20*, 832–844.
41. Breiman, L. Random Forests. *Mach. Learn.* **2001**, *45*, 5–32.
42. Cover, T.M.; Hart, P.E. Nearest neighbor pattern classification. *Inst. Electr. Electron. Eng. Trans. Inf. Theory* **1967**, *13*, 21–27.
43. Dasarathy, B.V. *Nearest-Neighbor Classification Techniques*; IEEE Computer Society Press: Los Alomitos, CA, USA, 1991.
44. Aha, D.W.; Kibler, D.; Albert, M.K. Instance-based learning algorithms. *Mach. Learn.* **1991**, *6*, 37–66.
45. Cortes, C.; Vapnik, V. Support vector networks. *Mach. Learn.* **1995**, *20*, 273–297.
46. Ben-Hur, A.; Horn, D.; Siegelmann, H.; Vapnik, V.N. Support vector clustering. *J. Mach. Learn. Res.* **2001**, *2*, 125–137.
47. McLachlan, G.J. *Discriminant Analysis and Statistical Pattern Recognition*; Wiley Interscience: New York, NY USA 2004.
48. Langley, W.I.; Thompson, K. An analysis of Bayesian classifiers. In Proceedings of the AAAI-94, Seattle, WA, USA, 1–4 August 1994; MIT Press: Cambridge, MA, USA, 1994; pp. 223–228.
49. John, G.H.; Langley, P. Estimating Continuous Distributions in Bayesian Classifiers. In Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence, San Mateo, CA, USA, 18–20 August 1995; pp. 338–345.
50. Ramírez-Gallego, S.; García, S.; Herrera, F. Online entropy-based discretization for data streaming classification. *Future Gener. Comput. Syst.* **2018**, *86*, 59–70.
51. Rahman, M.A.; Khanam, F.; Ahmad, M. Multiclass EEG signal classification utilizing Rényi min-entropy-based feature selection from wavelet packet transformation. *Brain Inform.* **2020**, *7*, 7.
52. Wang, J.; Xu, S.; Duan, B.; Liu, C.; Liang, J. An Ensemble Classification Algorithm Based on Information Entropy for Data Streams. *Neural Process. Lett.* **2019**, *50*, 2101–2117.
53. Mannor, S.; Peleg, D.; Rubinstein, R. The cross entropy method for classification. In Proceedings of the 22nd International Conference on Machine Learning (ICML '05), Association for Computing Machinery, New York, NY, USA, 11–13 August 2005; pp. 561–568.
54. Lee, H.M.; Chen, C.M.; Chen, J.M.; Jou, Y.L. An efficient fuzzy classifier with feature selection based on fuzzy entropy. *IEEE Trans. Syst. Man Cybern. Part B Cybern.* **2001**, *31*, 426–432.
55. Cleary, J.G.; Trigg, L.E. K*: An Instance-based Learner Using an Entropic Distance Measure. In *Machine Learning Proceedings 1995*; Prieditis, A., Russell, S., Eds.; Morgan Kaufmann, Burlington, MA, USA, 1995; pp. 108–114.
56. Holub, A.; Perona, P.; Burl, M.C. Entropy-based active learning for object recognition. In Proceedings of the 2008 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, Anchorage, AK, USA, 23–28 June 2008; pp. 1–8.
57. Fujino, A.; Ueda, N.; Saito, K. Semisupervised Learning for a Hybrid Generative/Discriminative Classifier based on the Maximum Entropy Principle. *IEEE Trans. Pattern Anal. Mach. Intell.* **2008**, *30*, 424–437.
58. Fan, Q.; Wang, Z.; Li, D.; Gao, D.; Zha, H. Entropy-based fuzzy support vector machine for imbalanced datasets. *Knowl. Based Syst.* **2017**, *115*, 87–99.
59. Ramos, D.; Franco-Pedroso, J.; Lozano-Diez, A.; Gonzalez-Rodriguez, J. Deconstructing Cross-Entropy for Probabilistic Binary Classifiers. *Entropy* **2018**, *20*, 208.
60. Berezinski, P.; Jasiul, B.; Szpyrka, M. An Entropy-Based Network Anomaly Detection Method. *Entropy* **2015**, *17*, 2367–2408.
61. Fukunaga, K. *Introduction to Statistic al Pattern Recognition*, 2nd ed.; Academic Press: Cambridge, MA, USA, 1990.
62. Tumer, K.; Ghosh, J. Estimating the Bayes error rate through classifier combining. In Proceedings of the 13th International Conference on Pattern Recognition, Vienna, Austria, 25–29 August 1996; Volume 2, pp. 695–699.
63. Costa, E.P.; Lorena, A.C.; Carvalho, A.C.; Freitas, A.A. A Review of Performance Evaluation Measures for Hierarchical Classifiers. In Proceedings of the AAAI-07 Workshop Evaluation Methods for Machine Learning II, Vancouver, BC, Canada, 22–23 July 2007.
64. Sokolova, M.; Lapalme, G. A systematic analysis of performance measures for classification tasks. *Inf. Process. Manag.* **2009**, *45*, 427–437.
65. Ferri, C.; Hernández-Orallo, J.; Modroiu, R. An experimental comparison of performance measures for classification. *Pattern Recognit. Lett.* **2009**, *30*, 27–38.
66. Parker, C. An Analysis of Performance Measures for Binary Classifiers. In Proceedings of the 2011 IEEE 11th International Conference on Data Mining, Vancouver, BC, Canada, 11–14 December 2011; pp. 517–526.
67. Labatut, V.; Cherifi, H. Evaluation of Performance Measures for Classifiers Comparison, Computer Science, Machine Learning. *arXiv* **2011**, arXiv:1112.4133.
68. Jiao, Y., Du, P. Performance measures in evaluating machine learning based bioinformatics predictors for classifications. *Quant. Biol.* **2016**, *4*, 320–330.
69. Valverde-Albacete, F.J.; Peláez-Moreno, C. Two information-theoretic tools to assess the performance of multi-class classifiers. *Pattern Recognit. Lett.* **2010**, *31*, 1665–1671.

70. Valverde-Albacete, F.J.; Peláez-Moreno, C. 100% Classification Accuracy Considered Harmful: The Normalized Information Transfer Factor Explains the Accuracy Paradox. *PLoS ONE* **2014**, *9*, e84217.
71. Valverde-Albacete, F.J.; Peláez-Moreno, C. The evaluation of data sources using multivariate entropy tools. *Expert Syst. Appl.* **2017**, *78*, 145–157.
72. Valverde-Albacete, F.J.; Peláez-Moreno, C. A Framework for Supervised Classification Performance Analysis with Information-Theoretic Methods. *IEEE Trans. Knowl. Data Eng.* **2020**, *32*, 2075–2087.
73. Cohen, J. A Coefficient of Agreement for Nominal Scales. *Educ. Psychol. Meas.* **1960**, *20*, 37–46.
74. Scott, W.A. Reliability of Content Analysis: The Case of Nominal Scale Coding. *Public Opin. Q.* **1955**, *19*, 321–325.
75. Witten, I.H., Frank, E. *Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations*; Elsevier: Amsterdam, The Netherlands, 2005.
76. Goodman, L.A.; Kruskal, W.H. Measures of Association for Cross Classifications. *J. Am. Stat. Assoc.* **1954**, *XLIX*, 732–764.
77. Williams, P.L.; Beer, R.D. Nonnegative Decomposition of Multivariate Information. *arXiv* **2010**, arXiv:1004.2515v1.
78. Yadav, A.K.; Chandel, S.S. Solar energy potential assessment of western Himalayan Indian state of Himachal Pradesh using J48 algorithm of WEKA in ANN based prediction model. *Renew. Energy* **2015**, *75*, 675–693.
79. Alloghani, M.; Aljaaf, A.; Hussain, A.; Baker, T.; Mustafina, J.; Al-Jumeily, D.; Khalaf, M. Implementation of machine learning algorithms to create diabetic patient re-admission profiles. *BMC Med. Inform. Decis. Mak.* **2019**, *19*, 253.
80. Romeo, V.; Cuocolo, R.; Ricciardi, C.; Ugga, L.; Cocozza, S.; Verde, F.; Stanzione, A.; Napolitano, V.; Russo, D.; Improta, G.; et al. Prediction of Tumor Grade and Nodal Status in Oropharyngeal and Oral Cavity Squamous-cell Carcinoma Using a Radiomic Approach. *Anticancer. Res.* **2020**, *40*, 271–280.
81. Frank, E.; Hall, M.A.; Witten, I.H. "The WEKA Workbench," Online Appendix for "Data Mining: Practical Machine Learning Tools and Techniques" Morgan Kaufmann, Fourth Edition, 2016. Available online: (accessed on 23 April 2021).
82. Weka. Available online: http://ocw.uc3m.es/ingenieria-informatica/herramientas-de-la-inteligencia-artificial/contenidos/transparencias/TutorialWeka.pdf (accessed on 9 March 2020).
83. Waikato Environment for Knowledge Analysis (Weka). Available online: http://www.cs.waikato.ac.nz/ml/weka (accessed on 15 June 2021).
84. Platt, J. Fast Training of Support Vector Machines using Sequential Minimal Optimization. In *Advances in Kernel Methods—Support Vector Learning*; Schoelkopf, B., Burges, C., Smola, A., Eds.; MIT Press: Cambridge, MA, USA, 1998
85. Keerthi, S.S.; Shevade, S.K.; Bhattacharyya, C.; Murthy, K.R.K. Improvements to Platt's SMO Algorithm for SVM Classifier Design. *Neural Comput.* **2001**, *13*, 637–649.
86. Hastie, T.; Tibshirani, R. Classification by Pairwise Coupling, In *Advances in Neural Information Processing Systems*; MIT Press: Cambridge, MA, USA, 1998.
87. Dua, D.; Graff, C. UCI Machine Learning Repository. University of California, Irvine, School of Information and Computer Sciences. 2019. Available online: http://archive.ics.uci.edu/ml (accessed on 23 April 2021).
88. Available online: http://archive.ics.uci.edu/ml/datasets/Thyroid+disease (accessed on 23 April 2021).
89. Shinmoto Torres, R.L.; Ranasinghe, D.C.; Shi, Q.; Sample, A.P. Sensor enabled wearable RFID technology for mitigating the risk of falls near beds. In Proceedings of the 2013 IEEE International Conference on RFID, Johor Bahru, Malaysia, 30 April–2 May 2013; pp. 191–198.
90. Available online: https://archive.ics.uci.edu/ml/datasets/Activity+recognition+with+healthy+older+people+using+a+batteryless+wearable+sensor (accessed on 23 April 2021).
91. de Stefano, C.; Maniaci, M.; Fontanella, F.; di Freca, A.S. Reliable writer identification in medieval manuscripts through page layout features: The "Avila" Bible case. *Eng. Appl. Artif. Intell.* **2018**, *72*, 99–110.
92. Available online: https://archive.ics.uci.edu/ml/datasets/Avila (accessed on 23 April 2021).
93. Available online: https://archive.ics.uci.edu/ml/datasets/adult (accessed on 23 April 2021).
94. Available online: https://archive.ics.uci.edu/ml/datasets/nursery (accessed on 23 April 2021).
95. Moro, S.; Cortez, P.; Rita, P. A Data-Driven Approach to Predict the Success of Bank Telemarketing. *Decis. Support Syst.* **2014**, *62*, 22–31.
96. Available online: https://archive.ics.uci.edu/ml/datasets/Bank+Marketing (accessed on 23 April 2021).
97. RLyon, J.; Stappers, B.W.; Cooper, S.; Brooke, J.M.; Knowles, J.D. Fifty Years of Pulsar Candidate Selection: From simple filters to a new principled real-time classification approach. *Mon. Not. R. Astron. Soc.* **2016**, *459*, 1104–1123.
98. Lyon, R.J. HTRU2. Available online: https://doi.org/10.6084/m9.figshare.3080389.v1 (accessed on 23 April 2021).
99. Available online: https://archive.ics.uci.edu/ml/datasets/HTRU2 (accessed on 23 April 2021).
100. Available online: https://archive.ics.uci.edu/ml/datasets/Connect-4 (accessed on 23 April 2021).
101. Available online: https://archive.ics.uci.edu/ml/datasets/Tic-Tac-Toe+Endgame (accessed on 23 April 2021).
102. Available online: https://archive.ics.uci.edu/ml/datasets/Credit+Approval (accessed on 23 April 2021).
103. Available online: https://archive.ics.uci.edu/ml/datasets/mushroom (accessed on 23 April 2021).
104. Brown, G.; Pocock, A.; Zhao, M.-J.; Luján, M. Conditional Likelihood Maximisation: A Unifying Framework for Information Theoretic Feature Selection. *J. Mach. Learn. Res.* **2012**, *13*, 27–66.