Available online at www.sciencedirect.com
SCIENCE DIRECT®
International Journal of Approximate Reasoning
39 (2005) 235–255
INTERNATIONAL JOURNAL OF APPROXIMATE REASONING
www.elsevier.com/locate/ijar

Upper entropy of credal sets. Applications to credal classification

Joaquín Abellán, Serafín Moral *

Dpto. Ciencias de la Computación, Universidad de Granada, Granada 18071, Spain

Received 1 February 2004; accepted 1 October 2004
Available online 25 November 2004

Abstract

We present an application of the measure of entropy for credal sets: as a branching criterion for constructing classification trees based on imprecise probabilities which are determined with the imprecise Dirichlet model. We also justify the use of upper entropy as a global uncertainty measure for credal sets and present a deduction of this measure. We have carried out several experiments in which credal classification trees are built taking a global uncertainty measure as a basis. The results show how the introduced methodology improves the performance of traditional methods (Naive Bayes and C4.5), by providing a much lower error rate.
© 2004 Elsevier Inc. All rights reserved.

Keywords: Imprecise probabilities; Uncertainty; Upper entropy; Imprecision; Non-specificity; Classification; Classification trees; Credal sets

1. Introduction

Classification is an important problem in the area of machine learning in which traditional probability theory has been extensively used. Basically, we have an incoming set of observations, called the training set, and, generally, we want to

* Corresponding author. Tel.: +34 9 58 242819; fax: +34 9 58 243317.
E-mail addresses: jabemu@teleline.es (J. Abellán), smc@decsai.ugr.es (S. Moral).

0888-613X/$ - see front matter © 2004 Elsevier Inc. All rights reserved.
doi:10.1016/j.ijar.2004.10.001

obtain a model to assign a value of the class variables to any new observation. The set of observations used to assess the quality of this model is also called the test set. Classification has notable applications in medicine, recognition of hand-written characters, astronomy, banking, etc. The learned classifier can be represented as a Bayesian network, a neural network, a classification tree, etc. These methods normally use the theory of probability to estimate the parameters with a stopping criterion to limit the complexity of the classifier and to avoid overfitting.

In some previous papers [4–6], we have introduced a new procedure to build *classification trees* based on the use of *imprecise probabilities*. Classification trees have their origin in Quinlan's ID3 algorithm [24], and a basic reference is the book by Breiman et al. [8]. In this paper, we also apply decision trees for classification, but as in [32], the *imprecise Dirichlet model* [29] is used to learn the model and to decide among the possible classes.

In classical probabilistic approaches, *information gain* [24] is used to build the tree, but then other procedures must subsequently be used to prune it, since information gain tends to build structures which are too complex. We have shown that if imprecise probabilities are used and the information gain is computed by measuring the total amount of uncertainty of the associated *credal set* (a closed and convex set of probability distributions), then the problem of overfitting disappears and results improve.

In [1–3], we studied how to measure the uncertainty of a credal set by generalizing the measures used in the *theory of evidence*, [11,26]. We considered two main sources of uncertainty: *entropy* and *non-specificity*. We proved that the proposed functions satisfy the most basic desiderata of these types of measures [2,14,20].

We previously proved that by using a global uncertainty measure which is the result of adding an entropy measure and a non-specificity measure, classification results are better than those obtained by the C4.5 classification method, based on Quinlan's ID3 algorithm. In this paper, we have carried out some experiments in which the upper entropy of the probability distributions of a credal set is used to measure its uncertainty, and we find that the results obtained are even better.

We consider two methods of building classification trees. In the first method, [4], we start with an empty tree and in each step, the variable which produces the largest decrease in the entropy of the class variable is selected for branching. The second method quantifies the uncertainty of each individual variable in each node in the same way, but also considers the results of adding two variables at the same time. In this way, we aim to discover relationships involving more than two variables that were not seen when investigating the relationships of a single variable with the class variable.

In traditional probability, adding a new branch always produces a decrease in entropy. It is necessary to include an additional criterion so as not to create models which are too complex and therefore overfit the data. With credal sets, adding a branch can produce smaller entropy but, at the same time, it will always give rise to greater non-specificity. Under these conditions, we follow the same procedure as in probability theory, but measuring the total uncertainty of adding a branch.

The stopping criterion is very simple: we stop when every possible addition of a branch produces an increase in total uncertainty.

Finally, in order to carry out the classification given a set of observations, we consider *credal classification* with a concept of *non-dominance* and assign to the class variable the set of non-dominated classes. We also use a *maximum frequency criterion* when we want to classify all the cases in a single class value, for comparison with traditional classification procedures.

In Section 2, we present the necessary previous concepts of uncertainty for credal sets. We place special emphasis on upper entropy as a global uncertainty measure. In Section 3, we introduce the necessary notation and definitions for our procedure of building classification trees. In Section 4, we describe the methods based on imprecise probabilities. In Section 5, we test our procedure with known data sets used in classification, in order to compare the two measures of global uncertainty.

2. Total uncertainty for credal sets

We will consider a variable X which takes values on a finite set U. A credal set concerning X is a convex set of probability distributions, $\mathscr{P}$. A credal set will represent the available information concerning the unknown value of the variable.

Dempster–Shafer's theory of evidence is based on the concept of basic probability assignment and it defines a special type of convex set of probability distributions [11,26]. A basic probability assignment is a mapping, $m{:}\wp(U) \to [0,1]$, such that $m(\emptyset) = 0$ and $\sum_{A\subseteq U} m(A) = 1$, where $\wp(U)$ is the power set of U.

A basic probability assignment defines a lower probability, usually called *belief*, and an upper probability, called *plausibility*, given respectively by,

$$Bel(A) = \sum_{B\subseteq A} m(B), \quad Pl(A) = \sum_{B\cap A\neq\emptyset} m(B).$$

The associated credal set for a pair of belief-plausibility measures can be written as:

$$\mathscr{P} = \{P | Bel(A) \leqslant P(A) \leqslant Pl(A), \quad \forall A \in \wp(U)\}.$$

In this theory, Yager [30] distinguished two types of uncertainty that he called *randomness* and *non-specificity*. Randomness is similar to probabilistic entropy and measures the contradictory nature of the information; i.e. is high when mass m is divided uniformly among a large number of disjoint sets. Non-specificity measures the imprecision of information; i.e. is high when mass m is large for sets with a large number of elements.

In [6] we show that a general convex set of probability distributions may contain the same types of uncertainty, and we consider similar randomness and non-specificity measures.

The classical non-specificity measure in the theory of evidence is given by: $IE(m) = \sum_{A\in\wp(U)} m(A)\cdot \ln(|A|)$, where $|A|$ stands for the cardinality of A. One of the difficulties of extending this measure to general credal sets is that it is defined

in terms of the mass m and there is not a direct expression in terms of the associated credal set. In [2], we define a measure of non-specificity for convex sets that generalizes Dubois and Prade's measure of non-specificity in the theory of evidence [13]. Consider the following definitions:

Definition 1. Let $\mathscr{P}$ be a credal set on a finite set U. We define the lower probability function associated to $\mathscr{P}$,

$$f_{\mathscr{P}}(A) = \inf_{P \in \mathscr{P}} P(A), \quad \forall A \in \wp(U).$$

Definition 2. (Shafer [26]) For any mapping $f_{\mathscr{P}} : \wp(U) \to \mathbb{R}$ another mapping $m_{\mathscr{P}} : \wp(U) \to \mathbb{R}$ can be associated by

$$m_{\mathscr{P}}(A) = \sum_{B \subseteq A} (-1)^{|A-B|} f_{\mathscr{P}}(B), \quad \forall A \in \wp(U).$$

This correspondence is one-to-one, since conversely we can obtain

$$f_{\mathscr{P}}(A) = \sum_{B \subseteq A} m_{\mathscr{P}}(B), \quad \forall A \in \wp(U).$$

These functions, $f_{\mathscr{P}}$ and $m_{\mathscr{P}}$, are called Möbius inverses [10].

Definition 3. Let $\mathscr{P}$ be a credal set defined on set U, $f_{\mathscr{P}}$ its lower probability as in Definition 1, and let $m_{\mathscr{P}}$ be its Möbius inverse. We say that function $m_{\mathscr{P}}$ is an *assignment of masses* on $\mathscr{P}$. Any $A \subseteq U$ such that $m_{\mathscr{P}}(A) \neq 0$ is called a focal element of $m_{\mathscr{P}}$.

We can now define a general measure of non-specificity.

Definition 4. Let $\mathscr{P}$ be a credal set on set U. Let $m_{\mathscr{P}}$ be its associated assignment of masses on $\mathscr{P}$. We define the following measure of non-specificity of $\mathscr{P}$:

$$IG(\mathscr{P}) = \sum_{A \subseteq U} m_{\mathscr{P}}(A) \ln(|A|).$$

This function can be considered as a general Hartley measure [17]. Hartley measure has already been extended to restricted types of convex sets of probability distributions, as the case of the theory of possibility [18] and the case of the theory of evidence [13], our function being a generalization of these extensions.

In [3], we proposed the following measure of randomness for general credal sets:

$$G^*(\mathscr{P}) = \max_{P \in \mathscr{P}} \left\{ -\sum_{x \in U} P(x) \ln(P(x)) \right\},$$

where $\mathscr{P}$ is a general credal set. This measure generalizes the classical Shannon entropy [27] and satisfies similar properties. We call this value the *upper entropy* of the credal set. In the theory of evidence, it can be used either as one of the components of a measure of total uncertainty, or as a total uncertainty measure [16]. We have proved that this function is also a good randomness measure for credal sets and possesses all the basic properties [3].

This function is the solution of a nonlinear optimization problem and its usefulness was initially questioned because of the difficulty of computations. However, Meyerowitz et al. [23] proposed a general and efficient algorithm to compute this measure in the theory of evidence. We proposed a similar algorithm, [3], to compute this measure for probability intervals, the type of credal sets that we obtain in our classification methods.

Adding two uncertainty measures for credal sets, we can define a measure of total uncertainty as $TU1(\mathscr{P}) = G^*(\mathscr{P}) + IG(\mathscr{P})$. In some previous work, we thought that this function was intuitively correct, quantifying the total uncertainty contained in a credal set in a similar way to Shannon entropy in the theory of probability [6]. But our understanding of the situation has changed and now we consider that the upper entropy, $TU2(\mathscr{P}) = G^*(\mathscr{P})$, is a measure of total uncertainty, as this measure also increases with imprecision. So adding imprecision to it, gives rise to overweight imprecision.

In the particular case of belief functions, Harmanec and Klir [16] have already considered that upper entropy is a measure of total uncertainty. They justify it by using an axiomatic approach. However, uniqueness is not proved. But perhaps the most compelling reason is given in [28]. We start by explaining the case of a single probability distribution, P. It is based on the logarithmic scoring rule. To be subject to this rule means that we are forced to select a probability distribution Q on U, and if the true value is x then we must pay $-\ln(Q(x))$. For example, if we say that $Q(x)$ is very small and x is found to be the true value, we must pay a lot. If $Q(x)$ is close to one, then we must pay a small amount. If our information about X is represented by a subjective probability P, then we should choose Q so that $E_P[-\ln(Q(X))]$ is minimum, where E_P is the mathematical expectation with respect to P. This minimum is obtained when $Q = P$ and the value of $E_P[-\ln(P(x))]$ is the entropy of P: the expected loss or the minimum amount that we would require to be subject to the logarithmic scoring rule. This rule is widely used in statistics. The entropy is the negative of the expected logarithm of the likelihood under distribution P. The reason for taking logarithms is that if we do the prediction in two independent experiments at the same time, then the payment should be the addition of the payments in the two experiments.

In the case of a credal set, $\mathscr{P}$, we can also apply the logarithmic scoring rule, but now we choose Q in such a way that the upper expected loss $\overline{E}_{\mathscr{P}}[-\ln(Q(x))]$ (the supremum of the expectations with respect to the probabilities in $\mathscr{P}$) is minimum. Under fixed Q, $\overline{E}_{\mathscr{P}}[-\ln(Q(x))]$ is the maximum loss we can have (the minimum we should be given to accept this gamble). As we have freedom to choose Q, we should select it, so that this amount $\overline{E}_{\mathscr{P}}[-\ln(Q(x))]$ is minimized.

Walley shows that this minimum is obtained for the distribution $P_0 \in \mathscr{P}$ with maximum entropy. [1] Furthermore, $\overline{E}_{\mathscr{P}}[-\ln(P_0(x))]$ is equal to $G^*(\mathscr{P})$, the upper entropy in $\mathscr{P}$. This is the minimum payment that we should require before being subject to the logarithmic scoring rule. This argument is completely analogous with the probabilistic one, except that we change expectation to upper expectation. This is really a measure of uncertainty, as the better we know the true value of X, then the less we should need to be paid to accept the logarithmic scoring rule (lower value of $G^*(\mathscr{P})$).

Our approach is different of what it is called *principle of maximum entropy* [19]. This principle always considers an unique probability distribution: the one with maximum entropy compatible with available restrictions. But, here we are not saying that $\mathscr{P}$ can be replaced by the probability distribution distribution of maximum entropy. We continue using the credal set to represent uncertainty. We only say that the uncertainty of the credal set can be measured by its upper entropy.

3. Obtaining conditional probability intervals with the IDM

The *imprecise Dirichlet model* (IDM) was introduced by Walley [29] to make inference about the probability distribution of a categorical variable. Assume that X is a variable taking values on a finite set U and that we have a sample of size N of independent and identically distributed outcomes of X. If we want to make inferences about the probabilities, $\theta_x = p(x)$, with which X takes its values, a common Bayesian procedure consists in assuming an 'a priori' Dirichlet distribution for the parameter vector $(\theta_x)_{x\in U}$, and then taking the 'a posteriori' expectation of the parameters given the sample. The Dirichlet distribution depends on the parameters s, a positive real value, and $\mathbf{t}$, a vector of positive real numbers $\mathbf{t} = (t_x)_{x\in U}$, verifying $\sum_{x\in U} t_x = 1$. The density is of the form:

$$f((\theta_x)_{x\in U}) = \frac{\Gamma(s)}{\prod_{x\in U}\Gamma(s\cdot t_x)}\prod_{x\in U}\theta_x^{s\cdot t_x - 1}$$

where Γ is the gamma function.

If $n(x)$ is the number of occurrences of value x in the sample, the expected 'a posteriori' value of parameter θ_x is $\frac{n(x)+s\cdot t_x}{N+s}$, which is the predictive probability for the event $[X = x]$ conditioned to the sample, under the hypothesis that we have a new value for the variable with the same distribution and conditionally independent of the sample given the parameter vector, $(\theta_x)_{x\in U}$.

The imprecise Dirichlet model [29] only depends on the parameter s and assumes all the possible values of $\mathbf{t}$. This defines a non-closed convex set of 'a priori' distribu-

[1] The proof is based on the Minimax theorem which can be found in Appendix E of Walley's book [28].

tions. It represents a much weaker assumption than a precise 'a priori' model, but it is possible to make useful inferences. In our particular case, in which we apply the IDM to only one variable, we obtain for this variable X a credal set that can be represented by a system of probability intervals. We obtain for each parameter, θ_x, a probability interval given by the lower and upper 'a posteriori' expected values of the parameter given the sample. These intervals can be easily computed and are given by $\left[\frac{n(x)}{N+s}, \frac{n(x)+s}{N+s}\right]$. The associated credal set for variable X is given by all the probability distributions P' on U, such that $P'(x) \in \left[\frac{n(x)}{N+s}, \frac{n(x)+s}{N+s}\right], \forall x$. The intervals are coherent in the sense that if we compute the intervals by taking infimum and supremum in the credal set, we get again the same set of intervals.

The parameter s determines how quickly the lower and upper probabilities converge as more data become available; larger values of s produce more cautious inferences. Walley [29] does not give a definitive recommendation, but he advocates values between $s = 1$ and $s = 2$.

Now, let us consider the case of a classification problem with a variable to classify, C, which has to be predicted as a function of a family of categorical attributes, $\mathbf{X} = \{X_1, \ldots, X_m\}$. A generic value of variable X_i will be denoted by x_i. C will have values in U_C. If $\mathbf{Y}$ is a subset of all the variables $\mathbf{X}$, then $\mathbf{y}$ will denote a generic value of it (a value for each one of the variables in the subset).

Our application of the IDM will be local in the following sense. The values of the attribute variables will be used to select a subset of the original sample with which to estimate a credal set only for variable C, according to above procedure. When classifying a new case, the values of attribute variables will be used to select the appropriate credal set for C. A global application of this model was proposed by Walley and used in [32]. The global IDM assumes an 'a priori' imprecise information about all the variables, $\mathbf{X}$ and C, and then inferences are done by conditioning this 'a priori' credal set to the available observations of attribute variables. It is not always the case that the credal set about C obtained in this way can be represented by a set of probability intervals without losing information. The main difference between the two methods is that in the global method IDM is applied jointly to the attributes and class variables, whereas in the local procedure IDM is applied a repeated number of times to class variable C (the attribute variables are used to select one application among the different ones).

We assume that we have a data set $\mathscr{D}$ with examples in which we have values for the variables in $\mathbf{X}$ and the class variable C. The objective will be to build a model (here a classification tree) allowing to assign a value for the class variable in a new example in which we have values for variables $\mathbf{X}$.

Definition 5. A configuration, σ, about $\mathbf{X}$ is an assignment of values for a subset of variables: $\mathbf{Y} = \mathbf{y}$, where $\mathbf{Y} \subseteq \mathbf{X}$.

If $\mathscr{D}$ is a data set and σ is a configuration, then $\mathscr{D}[\sigma]$ will denote the subset of $\mathscr{D}$ given by the cases which are compatible with configuration σ (cases in which the variables in σ have the same values as the ones assigned in the configuration).

Definition 6. Given a data set and a configuration σ, we consider the credal set $\mathscr{P}^\sigma$ for variable C with respect to σ defined by the set of probability distributions, P, such that

$$P(c) \in \left[\frac{n_c^\sigma}{N^\sigma + s}, \frac{n_c^\sigma + s}{N^\sigma + s}\right], \quad \forall c \in U_C,$$

where n_c^σ is the number of occurrences of $(C = c)$ in $\mathscr{D}[\sigma]$, N^σ is the number of cases in $\mathscr{D}[\sigma]$, and $s > 0$ is a parameter.

We denote this interval as

$$[\underline{P}^\sigma(c), \overline{P}^\sigma(c)].$$

This credal set is the one obtained with the imprecise Dirichlet model, [29], applied to the subsample $\mathscr{D}[\sigma]$.

Example 7. Assume that we have a class variable with 3 possible values: $U_C = \{c_1, c_2, c_3\}$.

Suppose that we have a database and a configuration σ such that:

$$n_{c_1}^\sigma = 4, \quad n_{c_2}^\sigma = 0, \quad n_{c_3}^\sigma = 0.$$

With $s = 1$, we have the following vector of probability intervals (for the three values of C), using the IDM:

$$\left(\left[\frac{4}{5}, 1\right]; \left[0, \frac{1}{5}\right]; \left[0, \frac{1}{5}\right]\right).$$

Credal set $\mathscr{P}^\sigma$ has three vertices:

$$\left\{(1,0,0); \left(\frac{4}{5}, \frac{1}{5}, 0\right); \left(\frac{4}{5}, 0, \frac{1}{5}\right)\right\}.$$

This credal set is represented in Fig. 1. Each point of the triangle represents a probability distribution in which the probability of c_i is the distance to the edge opposite to vertex c_i. The credal set is represented by the shadowed triangle. A general algorithm to find all the extreme points of a credal set that is defined by a system of intervals is given by Walley [28] and Campos et al. [9].

It is simple to compute the upper entropy of this credal set (a general algorithm applicable to probability intervals can be found in [3]). In general, its basic idea is to obtain a probability distribution satisfying the bounds given by the intervals and distributing the probability as evenly as possible. We obtain: $G^*(\mathscr{P}^\sigma) = H(\frac{4}{5}, \frac{1}{10}, \frac{1}{10}) = 0.639$, where H is the classical Shannon entropy.

To measure its non-specificity, we should compute its associated lower capacity $f_{\mathscr{P}^\sigma}$ and its Möbius inverse $m_{\mathscr{P}^\sigma}$, obtaining

$$m_{\mathscr{P}^\sigma}(\{c_1\}) = \frac{4}{5}, \quad m_{\mathscr{P}^\sigma}(\{c_2\}) = m_{\mathscr{P}^\sigma}(\{c_3\}) = 0,$$

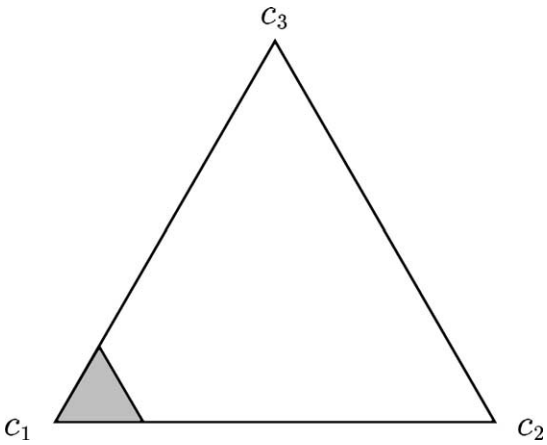

Fig. 1. Simplex representation of the credal set in Example 7.

$$m_{\mathscr{P}^\sigma}(\{c_1, c_2\}) = m_{\mathscr{P}^\sigma}(\{c_1, c_3\}) = 0, \quad m_{\mathscr{P}^\sigma}(\{c_2, c_3\}) = 0,$$

$$m_{\mathscr{P}^\sigma}(\{c_1, c_2, c_3\}) = \frac{1}{5}$$

and

$$IG(\mathscr{P}^\sigma) = \frac{1}{5}\ln(3) = 0.220.$$

If we have a different database with the same relative frequencies for the values of C, but different sample size, then the credal set changes. If the sample size is smaller then the intervals are wider and if the sample size is larger then the intervals are more precise.

Example 8. With the assumptions of the previous example, consider now that the absolute frequencies in the database are

$$n^\sigma_{c_1} = 9, \quad n^\sigma_{c_2} = 0, \quad n^\sigma_{c_3} = 0.$$

Observe as before in all the cases we have $C = c_1$, but with a higher sample size.

Considering again $s = 1$, we have the following vector of probability intervals, using the IDM:

$$\left(\left[\frac{9}{10}, 1\right]; \left[0, \frac{1}{10}\right]; \left[0, \frac{1}{10}\right]\right).$$

It is represented by the credal set $\mathscr{P}^\sigma$ with vertices:

$$\left\{(1,0,0); \left(\frac{9}{10}, \frac{1}{10}, 0\right); \left(\frac{9}{10}, 0, \frac{1}{10}\right)\right\}.$$

This is a convex set that is a proper subset of the credal set shown in Fig. 1.

The values of G^* and IG are smaller than the ones obtained with a sample size of 4. The probability distribution with maximum entropy is now: $P = (\frac{9}{10}, \frac{1}{20}, \frac{1}{20})$, which is more concentrated at c_1 than the one in former example. Therefore we have a smaller upper entropy: $G^*(\mathscr{P}^\sigma) = H(P) = 0.394$.

To compute its non-specificity, we obtain again the mass assignment associated to this system of intervals:

$$m_{\mathscr{P}^\sigma}(\{c_1\}) = \frac{9}{10}, \quad m_{\mathscr{P}^\sigma}(\{c_2\}) = m_{\mathscr{P}^\sigma}(\{c_3\}) = 0,$$

$$m_{\mathscr{P}^\sigma}(\{c_1, c_2\}) = m_{\mathscr{P}^\sigma}(\{c_1, c_3\}) = 0, \quad m_{\mathscr{P}^\sigma}(\{c_2, c_3\}) = 0,$$

$$m_{\mathscr{P}^\sigma}(\{c_1, c_2, c_3\}) = \frac{1}{10}$$

Note as we have the same focal sets than in the case of a shorter sample, but now the complete set has a smaller mass. As a consequence, we have a lower non-specificity: $IG(\mathscr{P}^\sigma) = \frac{1}{10}\ln(3) = 0.110$.

4. Classification procedure

We have proposed two methods for building a classification tree: the simple method [4] and the extended method [5]. Here we describe the extended procedure and give the simple one as a particular case.

A classification tree is a tree where each interior node is labelled with a variable of the data set $X_i \in \mathbf{X}$, with a child for each one of its possible values: $X_i = x_i \in U_i$. Each leaf will have a decision rule to assign a value of the class variable C. In traditional classification trees, the decision rule assigns a single value of C.

In a classification tree there is a correspondence between nodes and configurations. Each node defines a configuration: the set of variables that can be found in the path to that node from the root, with the values associated with the children that lie in this path. A complete configuration (a value for each one of the variables in $\mathbf{X}$) defines a leaf: we start at the root, and at each inner node with label X_i, we select the child corresponding to the value of X_i in the configuration.

In Fig. 2 we give an example of a classification tree, involving three variables with two possible values (0,1) for each of them. The root node corresponds to the empty configuration (no value for any variable). Its two children are two nodes corresponding to configurations ($X_1 = 0$) and ($X_1 = 1$) respectively. The leaf labelled with c_3 corresponds to the configuration $\sigma = (X_1 = 1, X_3 = 0)$. In each leaf of this tree we have a single value of the class variable.

Given a data set $\mathscr{D}$, each node of the tree defines a credal set for C in the following way: we first consider the configuration σ associated to it, and then the credal set, $\mathscr{P}^\sigma$, as in Definition 6. For example, we have seen in Fig. 2 that the node with label c_3

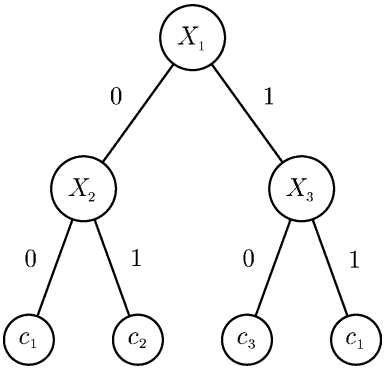

Fig. 2. Example of a Classification Tree.

determines a configuration $\sigma = (X_1 = 1, X_3 = 0)$. This configuration has an associated data set, $\mathscr{D}[\sigma]$, which is the subset of the original $\mathscr{D}$ given by those cases for which $X_1 = 1$ and $X_3 = 0$. $\mathscr{P}^{\sigma}$ is the credal set built from this restricted data set, using the imprecise Dirichlet model, as explained in Definition 6.

Our method for building classification trees is based on measuring the total uncertainty of the credal set associated with each leaf. In the following we shall describe how to build the structure of the tree. The decision rules will be considered later.

The method starts with a tree with a single node. We shall describe it as a recursive algorithm, which is started with the root node with no label associated to it. Each node will have a list $\mathscr{L}^*$ of possible labels of variables which can be associated to it. The procedure will initially be started with the complete list of variables.

We will consider that we have two functions implemented: `Infl`(σ, X_i) and `Inf2`(σ, X_i, X_j), computing respectively the values:

$$\texttt{Infl}(\sigma, X_i) = \left(\sum_{x_i \in U_i} r^{\sigma}_{x_i} TU\left(\mathscr{P}^{\sigma \cup (X_i = x_i)}\right) \right)$$

$$\texttt{Inf2}(\sigma, X_i, X_j) = \left(\sum_{x_i \in U_i, x_j \in U_j} r^{\sigma}_{x_i, x_j} TU\left(\mathscr{P}^{\sigma \cup (X_i = x_i, X_j = x_j)}\right) \right)$$

where $r^{\sigma}_{x_i}$ is the relative frequency with which X_i takes value x_i in $\mathscr{D}[\sigma]$, $r^{\sigma}_{x_i, x_j}$ is the relative frequency with which X_i and X_j take values x_i and x_j, respectively, in $\mathscr{D}[\sigma]$, $\sigma \cup (X_i = x_i)$ is the result of adding the value $X_i = x_i$ to configuration σ (analogously for $\sigma \cup (X_i = x_i, X_j = x_j)$), and TU is any total uncertainty measure ($TU1$ or $TU2$).

If *No* is a node and σ a configuration associated with it, `Infl` tries to measure the weighted average total uncertainty of the credal sets associated with the children of this node if variable X_i is added to it (and there is a child for each one of the possible values of this node). The average is weighted by the relative frequency of each one of the children in the data set. `Inf2` is similar, but considers adding two variables in one step: assigning X_i to the first node and then assigning X_j to all the children of

the first node. It measures the average of the total uncertainty of the credal sets associated to the grandchildren (the result of this function does not depend on the order).

In the following we describe the extended method. The basic idea is very simple and it is applied recursively to each one of the nodes we obtain. For each one of these nodes, we consider whether the total uncertainty of the credal set at this node can be decreased by adding one or two nodes. If this is the case, then we add a node with a maximum decrease of uncertainty. If the uncertainty cannot be decreased, then this node is not expanded and it is transformed into a leaf of the resulting tree.

Procedure `BuilTree`($No, \mathscr{L}^*$)

1. If $\mathscr{L}^* = \emptyset$, then `Exit`
2. Let σ be the configuration associated with node No
3. Compute the credal set associated with σ and compute its total uncertainty $TU(\mathscr{P}^\sigma)$
4. Compute the values
   $\alpha = \min_{X_i \in \mathscr{L}^*} \mathtt{Infl}(\sigma, X_i)$
   $\beta = \min_{X_i, X_j \in \mathscr{L}^*} \mathtt{Inf2}(\sigma, X_i, X_j)$
5. If the minimum of $\{\alpha,\beta\}$ is greater than or equal to $TU(\mathscr{P}^\sigma)$ then
   6. `Exit`
7. If the minimum of $\{\alpha,\beta\}$ is smaller than $TU(\mathscr{P}^\sigma)$, then
   8. If $\alpha \leqslant \beta$, then
      9. Let X_k be the variable for which the minimum α is attained
   10. Else
      11. Let X_i, X_j be the variables for which the minimum β is attained,
      12. Let X_k be the variable X_i or X_j with minimum `Infl`(σ,X_i)
13. Remove X_k from $\mathscr{L}^*$
14. Assign X_k to node No
15. For each possible value x_k of X_k
   16. Add a node No_k
   17. Make No_k a child of No
   18. Call `BuilTree`(No_k,$\mathscr{L}^*$)

In the above algorithm, X_k is the branching variable of node No. The intuitive idea is that when we assign this variable to No, we divide the database associated with this node among its different children. In each one of the children, we can have more precise average knowledge about C but based on a smaller sample. We consider that the total uncertainty of the associated credal sets can be a good measure of the appropriate trade-off between the precision gained by dividing the database according to the different values of X_k and the precision lost by estimating the probability distribution of C from a smaller database.

The simple method is analogous to the extended method, but it does not compute β, [4]. It only considers α and in steps 5 and 7, the minimum of $\{\alpha,\beta\}$ is α. Steps 8–12 which are devoted to selecting the branching variable, are simplified to step 9 only. The rest of the algorithm is the same.

In most cases, the simple method should be enough. But, as we shall see in the experiments, there are relationships between groups of variables that cannot be captured by pairwise relationships; i.e., it is possible that none of the variables X_i and X_j adds information about C, but the two variables together provide substantial information about C. This situation is the one in which the extended method does better. Of course, we could add three or more variables in a single step, but then the complexity of the algorithm will increase exponentially in the number of included variables, and we feel that there is not a corresponding gain in performance.

Traditional probabilistic classification trees are built in a similar way, but with the difference that we have precise estimations of probability values and the uncertainty measure is Shannon entropy. The quantity that is used to decide what variable to use to add a branch to a node is called *information gain* and it is similar to $TU(\mathscr{P}^\sigma) - \texttt{Infl}(\sigma, X_i)$, which is what we compute to decide the branching variable. The only difference is that information gain is applied to precise probabilities. If P^σ is a precise probability estimation of probabilities about C in $\mathscr{D}[\sigma]$ (maximum likelihood is the usual estimation method), then the information gain is given by

$$\texttt{InfGain}(\sigma, X_i) = H(P^\sigma) - \left(\sum_{x_i \in U_i} r^\sigma_{x_i} H\left(P^{\sigma \cup (X_i = x_i)}\right) \right)$$

The information gain is also called the *mutual information* between X_i and C in sample $\mathscr{D}[\sigma]$ and it is always a non-negative number. It is important to remark that the quantity we use, $TU(\mathscr{P}^\sigma) - \texttt{Infl}(\sigma, X_i)$, is not the supremum, nor the infimum of the mutual information, as in fact we are computing a difference of two upper values. If we were computing the supremum or the infimum of mutual information, we would always obtain a non-negative value. On the other hand, $TU(\mathscr{P}^\sigma) - \texttt{Infl}(\sigma, X_i)$ can be negative and this is important as it is our criterion to stop branching [2]. So, our procedure is not based on a sensitivity analysis of mutual information under imprecision. It can better understood as a method to choose between models: by comparing the information they give about the class variable.

4.1. Complexity

We are going to estimate the complexity as a function of the sample size, N, and the number of variables. As we never add a branch to a node which is compatible with no cases of the data, then the number of leaves is of order $O(N)$. The total number of nodes of the tree is of the same order. So we call procedure `BuilTree` a maximum of $O(N)$ times. Each time we call the procedure, we have to evaluate the weighted average of total uncertainty. To compute frequencies, we revise all the cases of the data set (N cases). In the simple method, for each call, we have to compute `Infl` for each variable. So, taking into account that m is the number of variables, we have a complexity for a call (without the recursive part) of $O(N \cdot m)$. Finally,

[2] In the simple method. In the double a similar value is used but considering the possibility of adding two variables at the same time.

the total complexity of building the complete tree in the simple method is $O(N^2 \cdot m)$. For the double method, a single call is of order $O(N \cdot m^2)$ and the complete procedure of order $O(N^2 \cdot m^2)$.

We have considered that the number of possible values of a variable is constant. In fact, this is not an important factor, as the upper entropy for our interval probabilities can be found in linear time as a function of the number of possible classes by using algorithm in [3]. Also the non-specificity is very simple to compute in this case: for intervals coming from the imprecise Dirichlet model, it is equal to $\ln(s/(N^\sigma + s))$, where N^σ is the sample size associated with σ and s is the IDM parameter. Only the computation of `Inf2` is cubic in the maximum number of cases of a variable (total uncertainty is computed a quadratic number of times).

4.2. Decision at the leaves

In order to classify a new example, with observations of all the variables except the class variable C, we obtain the leaf corresponding to the observed configuration, i.e. we start at the root of the tree and follow the path corresponding to the observed values of the variables in the interior nodes of the tree, i.e. if we are at a node with variable X_i which takes the value x_i, then we choose the child corresponding to this value. We then use the associated credal set for C, $\mathscr{P}^\sigma$, to classify the new example.

To do that, we first follow *dominance* under the *strict preference* ordering induced by a credal set as considered by Walley [28]. Strict preference does not determine a total order in the set of possible classes, and as a consequence the decision rule will not select a single value, but a set of possible values. We say that class c_1 is dominated under the strict preference ordering induced by $\mathscr{P}^\sigma$ if and only if for every P in $\mathscr{P}^\sigma$, there is a class value c_2 such that $P(c_2) > P(c_1)$. In this particular application of the IDM, dominance under the strict preference ordering is equivalent to *interval dominance*: c_1 is dominated if and only if there is a class value c_2 such that $\overline{P}^\sigma(c_1) < \underline{P}^\sigma(c_2)$.

The decision rule is to assign to every leaf with credal set $\mathscr{P}^\sigma$, the set of non-dominated class values corresponding to $\mathscr{P}^\sigma$. In this way, we obtain what Zaffalon [33] calls a *credal* classifier, in which, for a set of observations, we obtain a set of predicted values for class variable, non-dominated cases, instead of a unique prediction.

5. Experimentation

We have applied this method to some known data sets, obtained from the *UCI repository of machine learning databases*, which can be found on the following website: http://www.sgi.com/Technology/mlc/db. We use the less conservative parameter $s = 1$, since with larger values of s, we obtained a high degree of non-classified data in some databases (although with a greater percentage of correct classifications).

We compared the behavior of the two total uncertainty measures we have previously defined:

- $TU1 = G^* + IG$,
- $TU2 = G^*$.

The reason for using $TU1$ is that this measure was the first that was used to build classification trees by computing a total uncertainty measure for the credal sets at the leaves [4]. The results were good when compared with traditional methods, but the results for $TU2$ in the experiments are even better, as we shall see later. This pointed our attention to upper entropy as a measure of the total uncertainty associated with a credal set. We consider both measures, to compare the new procedure with the previous one.

The data sets are: *Breast*, *Breast Cancer*, *Heart*, *Hepatitis*, *Cleveland*, *Cleveland nominal* and *Pima* (medical); *Australian* (banking); *Monks1* (artificial) and *Soybean-small* (botanical).

These databases were studied by Acid [7]. We will use the same training and test sets as in Acid [7]. Some of the original data sets have observations with missing values and in some cases, some of the variables are not discrete. The cases with missing values were removed (both from training and test sets) and the continuous variables were discretized using MLC++ software, available at the website http://www.sgi.com/Technology/mlc. The measure used to discretize them is the entropy. The number of intervals is not fixed and it is obtained following the Fayyad and Irani procedure [15]. Only the training part of the database is used to determine the discretization procedure. In Table 1 there is a brief description of these databases (the column $N{\cdot}Tr$ contains the number of cases in the training set, the column $N{\cdot}Ts$ is the number of cases in the test set, the column $N{\cdot}var$ is the number of variables in the database and the column $N{\cdot}cl$ is the number of different values of the class variable).

In these experiments, when the set of non-dominated classes has more than one element we simply do not classify, without giving the set of non-dominated classes. We only classify when the set of non-dominated classes has only one element. But, we recognize that this implies a loss of some valuable information in certain

Table 1
Description of the databases

Data set	$N{\cdot}Tr$	$N{\cdot}Ts$	$N{\cdot}var$	$N{\cdot}cl$	$NB(Tr\|Ts)$	C4.5($Tr\|Ts$)
Breast Cancer	184	93	9	2	78.2|74.2	81.5|**75.3**
Breast	457	226	10	2	97.8|**97.3**	97.6|95.1
Cleveland nominal	202	99	7	5	63.9|**57.6**	69.3|51.5
Cleveland	200	97	13	5	78.0|50.5	73.5|**54.6**
Pima	512	256	8	2	76.4|74.6	79.9|**75.0**
Heart	180	90	13	2	87.8|**82.2**	83.3|75.6
Hepatitis	59	21	19	2	96.2|81.5	96.2|**85.2**
Australian	460	230	14	2	87.6|**86.1**	89.3|83.0
Vote1	300	135	15	2	87.6|**88.9**	94.5|88.3
Monks1	124	432	6	2	79.8|71.3	83.9|**75.7**
Soybean-small	31	16	21	4	100|93.8	100|**100**

Table 2
The measured experimental percentages for the simple method and $TU1$–$TU2$

Data set	Training	$UC(Tr)$	Test	$UC(Ts)$
Breast Cancer	75.5–89.0	0.0–16.3	81.7–93.5	0.0–17.2
Breast	98.0–99.1	1.3–2.6	96.9–98.6	0.9–2.6
Cleveland nominal	62.7–73.6	4.4–21.2	66.0–74.4	5.0–13.1
Cleveland	72.8–82.6	21.0–34.0	69.9–80.3	24.7–31.9
Pima	79.7–86.6	0.2–15.6	80.5–86.2	0.0–15.2
Heart	92.2–93.9	7.2–8.8	95.2–93.8	6.7–10.0
Hepatitis	96.4–96.4	5.0–5.0	94.7–94.7	9.5–9.5
Australian	92.3–95.3	3.4–6.5	91.0–94.4	3.4–6.5
Vote1	96.1–98.2	6.6–5.3	96.9–98.4	5.9–4.4
Soybean-small	100.0–100.0	0.0–0.0	100.0–100.0	0.0–0.0

situations (if for example we have a set with two non-dominated classes when the number of possible classes is 5).

Algorithms were implemented using Java language version 1.1.8. In order to obtain the value of G^* for probability intervals, we used the algorithm proposed in [3].

The percentages of correct classifications obtained from the simple model and $TU1$ and $TU2$ can be seen in Table 2 (the columns $UC(Tr)$ and $UC(Ts)$ are the percentages of the rejected cases obtained with the training and the test set respectively, rejected due to the fact that there is not a unique non-dominated class).

In these results we can see that [4] there is no overfitting (one of the most common problems of learning procedures): the success rates in the training set and the test set are very similar.

In general, we have few cases that are not classified. Only the *Cleveland* database has a high rate of non-classified data. This is the database with the highest number of cases of the class variable and then it is more difficult to obtain a class dominating all the other classes. We would have obtained more information by changing the output to a set of non-dominated classes. In most of the other databases, the variable to be classified has two possible states and in this situation the classification in the experiments is equivalent to the set of non-dominated values.

In Table 1, we can see the performance of other well known methods on the same databases, using the same discretization procedure, and the same partition for test-training as given by Acid [7]. The NB-column corresponds to the results of the *Naive Bayes* classifier [12] on the training set and the test set ($Tr|Ts$). Similarly, *C4.5* column correspond to Quinlan's method [25], based on the ID3 algorithm [24], where a classification tree with classical probabilities is used. We report the results obtained by Acid [7]. We can see that there is overfitting in these methods, principally in C4.5, being especially notable in certain data sets (*Cleveland nominal*, *Cleveland*, *Hepatitis*).

With $TU2$ we have a higher percentage of success and a much higher percentage of unclassified cases than with $TU1$ (see Table 2). $TU2$ also produces larger trees than $TU1$, as shown by the number of leaves presented in Table 3.

In Table 4 we can see the results of the extended method with TU1 and TU2. We find that the percentages of non-classified cases are higher if we use the function

Table 3
Number of leaves in the trees obtained from the simple method and $TU1$ or $TU2$

Data set	$TU1$	$TU2$	N of possible leaves
Breast	10	17	512
Cleveland	17	112	635904

Table 4
Double method with $TU1$-$TU2$

Data set	Training	$UC(Tr)$	Test	$UC(Ts)$
Breast Cancer	75.5–90.3	0.0–16.1	81.7–93.5	0.0–15.0
Breast	98.0–99.1	1.3–2.1	96.9–98.6	0.9–2.2
Cleveland nominal	64.6–75.7	5.0–24.4	68.8–74.4	6.1–17.1
Cleveland	72.8–83.1	21.0–32.0	69.9–81.2	24.7–28.9
Pima	79.7–86.8	0.2–14.4	80.5–87.8	0.0–16.0
Heart	91.7–96.3	6.1–10.5	94.1–96.4	5.6–7.7
Hepatitis	96.4–96.6	5.0–0.0	94.7–95.2	9.5–0.0
Australian	90.8–94.9	0.6–6.3	89.0–93.9	0.9–7.3
Vote1	96.1–99.0	6.6–4.6	96.9–99.2	5.9–4.4
Soybean-small	100.0–100.0	0.0–0.0	100.0–100.0	0.0–0.0

TU2. The reason comes from the fact that TU1 is equal to TU2 plus a factor measuring the non-specificity. So, TU1 penalizes imprecision in comparison to TU2. This produces that the trees built with TU2 are bigger and with more imprecise credal sets at the leaves. The final consequence is that we have less cases in which there is a unique non-dominated class (more cases in which no classification is produced).

We want also to compare our methods with existing classification methods. These methods classify all the cases of the training and test sets, predicting always a single class value. If with credal classification we reject to classify some difficult cases, then it can be expected that our procedures provide higher rates of correct classifications. So, in order to carry out a fair comparison with such complete procedures, we should also use a decision rule that classifies all the cases. For this purpose, we use the *maximum frequency criterion* based on frequency of the data, i.e., we will choose the case with maximum frequency in $\mathscr{D}[\sigma]$ as the value of the class variable (if there is more than one with maximum frequency we make an arbitrary selection among them). This criterion is the usual one in C4.5 and it is equivalent in this case (credal sets estimated with the imprecise Dirichlet model to the restricted database on the leaf) to consider the class value with maximum lower probability interval; and also equivalent to the class value with the maximum upper probability interval.

The success rates of the simple method when the frequency criterion is used to classify all the otherwise non-classified cases, are presented in Table 5 for the test set, to compare it with the models C4.5 and Naive Bayes. In the same table we show the results of similar experiments with the extended method. We can see the high percentages of correct classifications with $TU2$. These are a little higher than those obtained with $TU1$ and substantially higher than the other methods (C4.5 and Naive

Table 5
Percentages of correct classifications using the frequency criterion on the test set with functions $TU1$ and $TU2$ using the simple procedure (TUi^s) and the extended procedure (TUi^a)

Data set	$TU1^s$	$TU2^s$	$TU1^a$	$TU2^a$	NB	C4.5
Breast Cancer	81.7	90.3	81.7	**91.4**	74.2	75.3
Breast	96.9	97.8	96.9	**98.7**	97.3	95.1
Cleveland nominal	65.7	**75.8**	68.7	74.7	57.6	51.5
Cleveland	67.0	**80.4**	67.0	**80.4**	50.5	54.6
Pima	80.5	80.9	80.5	**82.4**	74.6	75.0
Heart	93.3	92.2	93.3	**94.4**	82.2	75.6
Hepatitis	**95.2**	**95.2**	**95.2**	**95.2**	81.5	85.2
Australian	90.9	**93.5**	89.1	91.7	86.1	83.0
Vote1	94.8	97.8	94.8	**98.5**	88.9	88.3
Soybean-small	**100.0**	**100.0**	**100.0**	**100.0**	93.8	**100.0**
AVERAGE	86.6	89.5	86.7	**90.7**	78.7	78.4

Bayes). Considering the percentages obtained for the different databases, we have carried out several t-tests to evaluate the significance of the average differences. Our methods (with TU1 and TU2) are significantly better than C4.5 and NB (p-value < 0.01) in both cases: with the simple and extended procedures. The differences of averages between TU1 and TU2 are significant at the 0.05 level, but not at the 0.01 level (the p-value is 0.042 for the simple method and 0.018 for the extended procedure).

The results of the simple and extended methods are similar (slightly better in the extended method). In order to see the potential of the extended method we studied an artificial database: *Monks1*.

Monks1 is a database with six variables. The class variable has two possible states: $a0$ and $a1$, being $a1$ when the first and the second attribute variables are equal or the fourth variable has the first of its four possible states. This type of dependence is very difficult to find with some classification methods, as it is a deterministic relationship involving more than two variables. The extended method should work much better than the simple one on that data.

Table 6 shows the success rate of the methods C4.5 and Naive Bayes on Monks1. Table 7 shows the rate of success of the simple and extended method when all cases are classified (frequency criterion).

We can observe some interesting facts. There is appreciable overfitting with C4.5 and Naive Bayes but not with our methods. The success rate obtained in the test set is better for the extended method than for the simple method, and there is a difference of 23.1% between the extended method with $TU2$ and Naive Bayes.

Table 6
C4.5 and Naive Bayes on Monks1 data

Data set	$NB(Tr)$	$NB(Ts)$	C4.5(Tr)	C4.5(Ts)
Monks1	79.8	71.3	83.9	75.7

Table 7
Results on *Monks1* using TU1 or TU2 and the frequency criterion

	Simple method		Double method	
Function	*Tr*	*Ts*	*Tr*	*Ts*
*TU*1	81.5	80.6	94.4	91.7
*TU*2	89.5	80.6	96.7	94.4

6. Conclusions

In this paper, we have discussed the role of upper entropy as a total uncertainty measure for credal sets. First, we have revised some decision theoretic justification based on the logarithmic scoring rule. Second, we have applied it to the construction of classification trees. We have carried out a series of experiments in which we compared this measure with the one we had previously proposed. The main conclusion is that, in general, the performance of the classifier is always at least as good when only upper entropy is used ($TU2$) than when a non-specificity value is added to it ($TU1$). And, in some examples, the percentages of correct classifications are substantially better with upper entropy. The reason we give is that $TU2$ is by itself a total uncertainty measure and that adding the non-specificity to it does not make sense as $TU2$ already measures the amount of imprecision. If $TU1$ is used then imprecision is over weighted, giving rise to smaller trees with more precise credal sets than when using $TU2$.

Other conclusions from the experiments can be summarized in the following points:

- The use of imprecise probability methods to build precise classification trees can improve the performance of standard procedures such as C4.5 and Naive Bayes. Furthermore, we have also the option of not classifying difficult cases.
- In general, the extended method produces slightly better results than the simple one, but in some particular cases the differences can be remarkable.
- Upper entropy ($TU2$) produces larger trees than the other uncertainty measure ($TU1$), but even this classifier does not suffer from overfitting.
- As $TU1$ produces smaller trees than $TU2$, this measure can be appropriate in situations in which the space is a limited resource.

The methods in the paper are designed for complete data. The case of incomplete data (either with the missing at random assumption or in the general case) will be considered in the future.

Acknowledgments

A previous version of this paper was presented at the *3rd International Symposium on Imprecise Probabilities and Their Applications (ISIPTA '03)*. We are very grateful

to Peter Walley, Marco Zaffalon, and to the anonymous referees for their valuable comments and suggestions. This work has been supported by the Spanish Ministry of Science and Technology, project Elvira II (TIC2001-2973-C05-01).

References

[1] J. Abellán, S. Moral, Completing a total uncertainty measure in Dempster–Shafer theory, Int. J. General Systems 28 (1999) 299–314.
[2] J. Abellán, S. Moral, A non-specificity measure for convex sets of probability distributions, International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems 8 (2000) 357–367.
[3] J. Abellán, S. Moral, Maximum entropy for credal sets, International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems 11 (2003) 587–597.
[4] J. Abellán and S. Moral, Using the total uncertainty criterion for building classification trees, in: Proceedings of the Second International Symposium on Imprecise Probabilities and Their Applications (ISIPTA '01), Ithaca, 2001, pp. 1–8.
[5] J. Abellán, S. Moral, Construcción de árboles de clasificación con probabilidades imprecisas, Actas de la Conferencia de la Asociación Española para la Inteligencia Artificial 2 (2001) 1035–1044.
[6] J. Abellán, Medidas de entropía y distancia en conjuntos convexos de probabilidad: definiciones y aplicaciones., PhD thesis, Universidad de Granada, 2003.
[7] S. Acid, Métodos de aprendizaje de redes de creencia. Aplicación a la clasificación, PhD thesis, Universidad de Granada, 1999.
[8] L. Breiman, J.H. Friedman, R.A. Olshen, and C.J. Stone, Classification and regression trees. Wadsworth Statistics, Probability Series, Belmont, 1984.
[9] L.M. de Campos, J.F. Huete, S. Moral, Probability intervals: a tool for uncertainty reasoning, International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems 2 (1994) 167–196.
[10] G. Choquet, Théorie des capacités, Ann. Inst. Fourier 5 (1953/54) 131–292.
[11] A.P. Dempster, Upper and lower probabilities induced by a multivalued mapping, Ann. Math. Stat. 38 (1967) 325–339.
[12] P. Domingos, M. Pazzani, On the optimality of the simple Bayesian classifier under zero-one loss, Machine Learning 29 (1997) 103–130.
[13] D. Dubois, H. Prade, A note on the measure of specificity for fuzzy sets, BUSEFAL 19 (1984) 83–89.
[14] D. Dubois, H. Prade, Properties and measures of information in evidence and possibility theories, Fuzzy Sets and Systems 24 (1987) 183–196.
[15] U.M. Fayyad and K.B. Irani, Multi-valued interval discretization of continuous-valued attributes for classification learning, in: Proceedings of the 13th International Joint Conference on Artificial Intelligence, Morgan Kaufmann, San Mateo, 1993, pp. 1022–1027.
[16] D. Harmanec, G.J. Klir, Measuring total uncertainty in Dempster–Shafer theory: a novel approach, Int. J. General Systems 22 (1994) 405–419.
[17] R.V.L. Hartley, Transmission of information, The Bell Systems Technical Journal 7 (1928) 535–563.
[18] M. Higashi, G.J. Klir, Measures of uncertainty and information based on possibility distributions, Int. J. General Systems 9 (1983) 43–58.
[19] E.T. Jaynes, Information theory and statistical mechanics, in: K. Ford (Ed.), Statistical Physics, Benjamin, New York, 1963, pp. 182–218.
[20] G.J. Klir, M.J. Wierman, Uncertainty-Based Information, Physica-Verlag, 1998.
[23] A. Meyerowitz, F. Richman, E.A. Walker, Calculating maximum-entropy probability densities for belief functions, Int. J. of Uncertainty* Fuzziness and Knowledge-Based Systems 2 (1994) 377–389.
[24] J.R. Quinlan, Induction of decision trees, Machine Learning 1 (1986) 81–106.
[25] J.R. Quinlan, Programs for Machine Learning, Morgan Kaufmann Series in Machine Learning (1993).
[26] G. Shafer, A Mathematical Theory of Evidence, Princeton University Press, Princeton, 1976.

[27] C.E. Shannon, A mathematical theory of communication, The Bell System Technical Journal 27 (1948) 379–423.
[28] P. Walley, Statistical Reasoning with Imprecise Probabilities, Chapman and Hall, London, 1991.
[29] P. Walley, Inferences from multinomial data: learning about a bag of marbles, J. Roy. Statist. Soc. B 58 (1996) 3–57.
[30] R.R. Yager, entropy and specificity in a mathematical theory of evidence, Int. J. General Systems 9 (1983) 249–260.
[32] M. Zaffalon, Statistical inference of the naive credal classifier, in: Proceedings of the Second International Symposium on Imprecise Probabilities and their Applications, ISIPTA '01, Ithaca, 2001, pp. 384–393.
[33] M. Zaffalon, The naive credal classifier, Journal of Statistical Planning and Inference 105 (2002) 5–21.