Building Classification Trees Using the Total Uncertainty Criterion

Joaquín Abellán and Serafín Moral
Dpto. Ciencias de la Computación e IA
ETSI Informática
Universidad de Granada
18071 - Granada - Spain
e-mails: jabemu@teleline.es, smc@decsai.ugr.es

Abstract

We present an application of the measure of total uncertainty on convex sets of probability distributions, also called credal sets, to the construction of classification trees. In these classification trees the probabilities of the classes in each one of its leaves is estimated by using the imprecise Dirichlet model. In this way, smaller samples give rise to wider probability intervals. Branching a classification tree can decrease the entropy associated to the classes but, at the same time, as the sample is divided among the branches the non-specificity increases. We use a total uncertainty measure (entropy + non-specificity) as branching criterion. The stopping rule is not to increase the total uncertainty. The good behavior of this procedure for standard classification problems is shown. It is important to remark that it does not suffer of overfitting, with similar results in the training and test samples.

Keywords. Imprecise probabilities, uncertainty, imprecision, non-specificity, classification, classification trees, credal sets.

1 Introduction

Classification is an important problem of machine learning in which classical probability theory has been extensively used. Basically, we have an incoming set of observations, called the training set, and we want to obtain a set of rules to assign to any new set of observations one value of the variable to classify. The set used to assess the quality of this set of rules is also called the test set. It has notable applications in medicine, recognition of hand-written characters, astronomy, banks, etc... The learned classifier can be represented as a Bayesian network, a neural network, a classification tree, etc... Normally, these methods 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 this paper, we will use the theory of Imprecise Probabilities to build a classification tree. A classification tree is a structure easy to understand and is an efficient classifier. It has its origin in ID3 algorithm by Quinlan [15]. A basic reference is the book by Breiman et al. [5]. Here, we also apply decision trees for classification, but as in Zaffalon [22], the imprecise Dirichlet model is used to estimate the probabilities of belonging to the respective classes defined by the variable to be classified.

In Abellán and Moral [2, 3], we have studied how to measure the uncertainty of a credal set generalizing the measures used in the Theory of Evidence, Dempster [7] and Shafer [17]. We consider two main sources of uncertainty: entropy and non-specificity. We have proved that the proposed functions verify the most basic properties of this type of measures (Abellán and Moral [3], Dubois and Prade [9], Klir and Wierman [12]).

Building a classification tree is usually done by starting with an empty tree and selecting, in each step, a node and a variable to branch with a greater decreasing in the final entropy of the variable to be classified. In classical probability a branching always implies a decreasing of the entropy. So, it is necessary to include an additional criterion to not create too complex models with overfitting to the data. With credal sets, a branching will produce a lower entropy but, at the same time, a greater non-specificity. In these conditions, we will follow the same procedure as in probability theory, but measuring the total uncertainty of a branching. The stopping criterion will be very simple: when every possible branching produces an increment of the total uncertainty (the entropy decrement does not compensate the increment of non-specificity).

Finally to carry out the classification in front of a set of observations, we will use a strong dominance criterion to obtain the value of the variable to classify. In some cases, the output will be a set of possible classes instead of a precise classification, i.e. we will have what is called a credal classifier (Zaffalon [23]).

In Section 2 we present necessary previous concepts on uncertainty of credal sets. In Section 3 we introduce the notation and prvious definitions. In Section 4 we describe in detail our classification procedure. In Section 5 we will carry out several experiments with some known data sets used in classification.

2 Total Uncertainty on Credal Sets

Dempster-Shafer's theory is based on the concept of basic probability assignment, and this defines a special type of credal set [7, 17]. In this theory, Yager [21] distinguishes two types of uncertainty. One is associated with cases where the information is focused in sets with empty intersections and the other is associated with cases where the information is focused in sets with cardinality over one. We call them *randomness* and *non-specificity* respectively. Since we consider that a general convex set of probability distributions (a credal set) may contain the same type of uncertainty as a b.p.a., we consider similar randomness and non-specificity measures on it.

In Abellán and Moral [3], we have defined a measure of non-specificity for convex sets that generalizes Dubois and Prade's measure of non-specificity in the theory of evidence [8]. Using the Möbius inverse function for monotonic capacities [6], we can define:

Definition 1 *Let $\mathcal{P}$ be a credal set on a frame X. We define the following capacity function,*

$$f_{\mathcal{P}}(A) = \inf_{P \in \mathcal{P}} P(A), \quad \forall A \in \wp(X),$$

where $\wp(X)$ is the power set of X.

Definition 2 *For any mapping $f_{\mathcal{P}} : \wp(X) \to I\!R$ another mapping $m_{\mathcal{P}} : \wp(X) \to I\!R$ can be associated by*

$$m_{\mathcal{P}}(A) = \sum_{B \subset A} (-1)^{|A-B|} f_{\mathcal{P}}(B), \forall A \in \wp(X),$$

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

$$f_{\mathcal{P}}(A) = \sum_{B \subset A} m_{\mathcal{P}}(B), \ \forall A \in \wp(X),$$

as we can see in Shafer [17], who calls the correspondence Möbius inversion.

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

Now, we can define a general function of non-specificity.

Definition 4 *Let $\mathcal{P}$ be a credal set on a frame X. Let $m_{\mathcal{P}}$ be its associated assignment of masses on $\mathcal{P}$. We define on $\mathcal{P}$ the following function of non-specificity:*

$$IG(\mathcal{P}) = \sum_{A \subset X} m_{\mathcal{P}}(A) \ln(|A|).$$

In Abellán and Moral [4], we have proposed the following measure of randomness for general credal sets

$$GG(\mathcal{P}) = Max \left\{ - \sum_{x \in X} p_x \ln p_x \right\}$$

where the maximum is taken over all probability distributions on $\mathcal{P}$, and $\mathcal{P}$ is a general credal set. This measure generalizes the classical Shannon's measure [18], for Dempster-Shafer's theory verifying similar properties. It can be used as one of the components of measure of total uncertainty, Harmanec and Klir [11]. We have proved that this function is also a good randomness measure for credal sets and verifies all the basic properties that were verified in Dempster-Shafer's theory [4].

We define a measure of total uncertainty as $TU(\mathcal{P}) = IG(\mathcal{P}) + GG(\mathcal{P})$. This measure could be modified by the factor introduced in Abellán and Moral [2], but this will not be considered here, due to its computational difficulties (it is a supremum that is not easy to compute). The properties of this measure are studied in Abellán and Moral [3, 4] and these are similar to the properties verified by total uncertainty measures in Dempster-Shafer's theory [14].

3 Notation and previous definitions

For a classification problem we shall consider that we have a data set $\mathcal{D}$ with values of a set $\mathcal{L}$ of discrete variables $\{X_i\}_1^n$. Each variable will take values or states on a finite set $\Omega_{X_i} = \{x_i^1, x_i^2, ..., x_i^{|\Omega_{X_i}|}\}$. Our aim will be to create a classification tree on the data set $\mathcal{D}$ of one target variable C, with values or attributes in $\Omega_C = \{c^1, c^2, ..., c^{|\Omega_C|}\}$.

Definition 5 *Let $\{X_i\}_1^n$ be a set of discrete variables with values in the finite sets Ω_{X_i}, respectively. We call a configuration of $\{X_i\}_1^n$ any m-tuple*

$$(X_{r_1} = x_{r_1}^{t_{r_1}}, X_{r_2} = x_{r_2}^{t_{r_2}}, ..., X_{r_m} = x_{r_m}^{t_{r_m}}),$$

where $x_{r_j}^{t_{r_j}} \in \Omega_{r_j}$, $j \in \{1, ..., m\}$, $r_j \in \{1, ..., n\}$ and $r_j \neq r_h$ with $j \neq h$. That is, a configuration is an assignment of values for some of the variables in $\{X_i\}_1^n$.

Definition 6 *Given a data set and a configuration σ of set $\{X_i\}_1^n$ we consider the credal set $\mathcal{P}_C^\sigma$ for variable C with respect to σ defined by the set of probability distributions, p, such that*

$$p_j \in \left[\frac{n^\sigma_{c^j}}{N+s}, \frac{n^\sigma_{c^j}+s}{N+s}\right],$$

for every $j \in \{1, ..., |\Omega_C|\}$, obtained on the basis of the imprecise Dirichlet model, Walley [20], for a generic state $c^j \in \Omega_C$. Here $n^\sigma_{c^j}$ is the number of occurrences of the configuration $\{C = c^j\} \cup \sigma$ in the data set, N is the number of observations compatible with configuration σ and $s > 0$ is a hyperparameter.

We denote this interval as

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

This parameter s determines how quickly the lower an upper probabilities converge as more data become available, larger values of s produce more cautious inferences. Walley [20] suggests a candidate value for s between $s = 1$ and $s = 2$, but no definitive statement is given.

4 Exposition of the method

4.1 Classification procedure

Using the above notation, we give the procedure to build a classification tree. A classification tree is a tree where each interior node is labeled with a variable of the data set X_j with a child for each one of its possible values: $X_j = x_j^r \in \Omega_{X_j}$. In each leaf node, we shall have a credal set for the variable to be classified, $\mathcal{P}_C^\sigma$, as defined above, where σ is the configuration with all the variables in the path from the root to this leaf node, with each variable assigned to the value corresponding to the child followed in the path. This method can be described using the following points:

I. We start with an empty tree. The root node will be a variable that attains the following value:

$$\min_{X_i \in \mathcal{L}} \left(\sum_{j \in \{1,..,|\Omega_{X_i}|\}} \rho_{x_i^j} TU(\mathcal{P}_C^{x_i^j}) \right),$$

with $\rho_{x_i^j}$ the relative frequency of x_i^j and $\mathcal{L}$ the list of variables in the database. This value should be less than $TU(\mathcal{P}_C^\emptyset)$. In other case, the classification tree will have an only node with $\mathcal{P}_C^\emptyset$ and the classification will take into account only the frequency of the states of the variable to be classified, and not the values of the rest of the variables.

II. For each node already generated, we compute the total uncertainty of the credal set associated to the configuration, σ, of the path from the root node to that node: $TU(\mathcal{P}_C^\sigma)$. Then we find the variable X_{i_0} with the value:

$$\min_{X_i \in \mathcal{L}^*} \left(\sum_{j \in \{1,..,|\Omega_{X_i}|\}} \rho_{\sigma \cup (X_i = x_i^j)} TU(\mathcal{P}_C^{\sigma \cup (X_i = x_i^j)}) \right),$$

where $\mathcal{L}^*$ is the set of variables of the data set that are not included in configuration σ.

That is, in a node with a configuration σ, we compute for each variable the weighted average of the total uncertainty of the leaves associated to the branching for this variable, where the weights are the frequencies of occurrence of the different values of the variable under configuration σ. Then, we take the variable X_{i_0}, with minimum value of total uncertainty after branching. If this value is lower than the total uncertainty before branching $TU(\mathcal{P}_C^\sigma)$, this node is labeled with X_{i_0} and a branch is added for each one of its children. The process will be repeated for each one of them.

III. If there is no variable that reduces the uncertainty or $\mathcal{L}^*$ is empty, then this node will be a leaf and will contain the credal set associated to the configuration with the values of the variables in the path from the root node to this leaf.

4.2 Decision in the leaves

To classify a new case with observations in all the variables except in the variable to be classified C, then we start in 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 in a node with variable X_i and this variable takes the value x_i^j in the set of observations, then we choose the child corresponding to this value. This process is followed till we arrive to a leaf node. Then, we use the associated credal set about C to obtain a value for this variable.

We will use a strong dominance criterion on C. This criterion generally implies only a partial order, and in some situations, no possible precise classification can be done. We will choose an attribute of the variable $C = c^h$ if it verifies that $\forall i \neq h$

$$\overline{P}(c^i|\sigma) < \underline{P}(c^h|\sigma)$$

When there is no value dominating all other possible values of C, the output can be the set of non-dominated cases (cases c^i for which there is not another case c^h verifying above inequality). In this way, we obtain what Zaffalon [23] calls a *credal* classifier, in which for a set of observations we obtain a set of possible values for the variable to classify instead of an unique prediction.[1]

An alternative criterion for classification is credal dominance [23] called also strict preference [19]. This criterion is based in comparing the probability of the two cases for each one of the probabilities of the credal set. Strong dominance implies credal dominance, but the converse is not true: there are situations in which there is credal dominance but not strong dominance. However, in this particular case in which we have credal sets that are defined by reachable intervals for the values of the variable to classify it is easy to prove that both criteria are equivalent.

4.3 Complexity

The complexity of the procedure to build credal classification trees is similar than the complexity of building probabilistic trees [15]. In general this depends of the number of nodes and the complexity of adding a node. In our case, the number of nodes is limited by the size of the database: each leaf will need a minimum number of cases of the database because if a branching is that the number of individuals in each leave is too low, then the credal sets associated to them will be very unspecific and the branching will not take place. In general, in the probabilistic case an almost complete tree is created and this is pruned afterward. Here we use a simpler approach by adding only nodes when they give some information and without a pruning phase. So, for final trees of the same size, we add less nodes than with classical probabilistic procedures.

The critical point here is the complexity of adding a node. As stated in Section 2, the computation of total uncertainty associated to a credal set can be too high (the non specificity is exponential in the number of values of the variable and for the randomness we should compute a maximum on a convex set). But due to the special characteristics of the credal sets we have (defined by reachable intervals in the possible cases of the variables), these computations can be done more efficiently [4]. In fact randomness is linear in the number of elements of the variable. In our implementation non specificity is exponential in the number of cases of the variable, but this number is not very high in general. We believe that it is also possible to take advantage of the special structure of these convex sets to compute it more efficiently. Another alternative is to consider an alternative measure based on the amplitude of intervals, which would be linear in the number of cases of the variable.

We have also to take into account that the learning time is not of great importance as soon as learning is feasible, i.e. the practical implications of learning in some minutes or several hours are minimum. However, the classification time here is similar than in the probabilistic case: linear (or quadratic if we want to compute all non dominated values) in the number of cases of the variable to classify plus a linear factor linear in the number of variables (to traverse the path from the root to a leaf).

5 Experimentation

We have applied this method to some known data sets, obtained from the Uci repository of machine Learning databases, http://www.sgi.com/Technology/mlc/db, with the parameter less conservative $s = 1$, since with $s > 1$ we obtained a high degree of non-classified data in some databases (though with a greater percentage of correct classifications).

The data sets *Breast*, *Breast Cancer*, *Heart*, *Hepatitis* and *Cleveland* (medical); *German* and *Australian* (financial); *Vote* (political); and *Soybean-small* (botanical).

These databases were used by Acid [1]. 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 and the continuous variables have been discretized using MLC++ software, available in http://www.sgi.com/Technology/mlc. The measure used to discretize them have been the entropy. The number of intervals is not fixed, and it is obtained following Fayyad and Irani [10] procedure. Only the training part of the database was used to determine the discretization procedure. In Table 1 there is a brief description of these databases. We can see the number of cases of the training set (N. Tr), of the test set (N. Ts), number of variables in the database (N. variables) and the number of different values of the variable to be classified (N. classes).

In general, when there is not a case dominating all the other possible values of the variable to classify, we simply does not classify this individual.

Algorithms have been implemented using Java language version 1.1.8.

[1] In the experiments, when there is not a dominant value, we simply do not classify, without calculating the set of non-dominated attributes. As it was pointed out by Zaffalon to us this implies to lose some valuable information in some situations.

Data base	N. Tr	N. Ts	N. variables	N. classes
Breast Cancer	184	93	9	2
Breast	457	226	10	2
Cleveland nominal	202	99	7	5
Cleveland	200	97	13	5
Pima	512	256	8	2
Heart	180	90	13	2
Hepatitis	59	21	19	2
German	670	330	24	2
Australian	460	230	14	2
Vote1	290	148	15	2
Soybean-small	31	16	24	4

Table 1: Description of the databases

The obtained percentages of correct classifications can be seen in Table 2.

The Training column is the percentage of correct classifications in the data set that was used for learning. In $UC(Tr)$ column we have the percentage of rejected cases, i.e., the observations that were not classified by the method due to the fact that no value verifies the strong dominance criterion, and in $UC(Ts)$ column we have the rejected cases in the test set. The success rate can be improved if we had used $s = 2$, though the percentage of rejected observations will be also increased.

As we can see in Table 2 there is no overfitting (one of the most common problems of learning procedures): the success of the training set and the test set are very similar.

Only database *Cleveland* has a high rate of non-classified data. This is the case with the highest number of cases of the variable to classify and then it is more difficult to obtain a class dominating all the other values. In this case, we would have obtained more information by changing the output to a set of non-dominated cases. In most of the other databases, the variable to be classified has two possible states and in this situation our classification is equivalent to the set of non-dominated values.

In Table 3 we compare the results with another known methods with good behavior on the same databases, Acid [1].

The NB-columns correspond to results of the Naive Bayesian classifier on the Training set and the Test set. This known method is based on the conditional independence of the variables given the variable to be classified. Similarly, the C4.5-columns correspond to the Quinlan's method [16], based on ID3 [15], where a classification tree with classical precise probabilities is used. It is possible to obtain a implementation of this method in http://www.sgi.com/Technology/mlc. We report the results obtained by Acid [1].

We can see that there is overfitting in these methods, principally in C4.5 being specially notable in some data sets (*Cleveland nominal*, *Cleveland*, *Hepatitis* or *German*).

Another method based on credal set, with excellent success, is the one presents by Zaffalon [22]. He used two databases that we have studied here, *Breast* and *German*, but he used another criterion of dominance, credal dominance. In Table 4 we give the results of Zaffalon's procedure in these two databases[2].

The NCCi-columns is the percentages of Zaffalon's method with s parameter equal to $i = \{1, 2\}$. It is the same parameter that we use in our method with the same significance. The UC-columns are the percentages of rejected observations.

6 Conclusions

We have introduced an interesting application of the study of the measure of uncertainty on credal sets to the constructions of credal classification trees.

We could take a more complete function, to measure the uncertainty, adding to IG and GG, the function given in Abellán and Moral [2] applying Kullback distance [13]. We believe that it could improve the degree of success as it is a more accurate measure of the degree of information of a credal set, but it is not easy to compute, Abellán and Moral [2]. Anyway, we would investigate if the computation can be effectively

[2] Comparisons with our procedure should take into account the differences due to the discretization procedures used in each one of the implementations

Data base	Training	UC(Tr)	Test	UC(Ts)
Breast Cancer	75.5	0.0	81.7	0.0
Breast	98.0	1.3	96.9	0.9
Cleveland nominal	62.7	4.4	66.0	5.0
Cleveland	70.9	17.5	68.0	19.2
Pima	79.7	0.2	80.5	0.0
Heart	92.2	7.2	95.2	6.7
Hepatitis	96.4	5.0	94.7	9.5
German	86.5	8.5	88.2	7.9
Australian	92.5	4.3	91.4	4.3
Vote1	92.3	1.0	92.5	1.6
Soybean-small	100.0	0.0	100.0	0.0

Table 2: The measured experimental percentages

Data set	NB(Tr)	NB(Ts)	C4.5(Tr)	C4.5(Ts)
Breast Cancer	78.2	74.2	81.5	75.3
Breast	97.8	97.3	97.6	95.1
Cleveland nominal	63.9	57.6	69.3	51.5
Cleveland	78.0	50.5	73.5	54.6
Pima	76.4	74.6	79.9	75.0
Heart	87.8	82.2	83.3	75.6
Hepatitis	96.2	81.5	96.2	85.2
German	77.0	75.1	98.4	74.8
Australian	87.6	86.1	89.3	83.0
Vote1	87.6	88.9	94.5	88.3
Soybean-small	100	93.8	100	100

Table 3: Percentages of another methods

Data set	NCC1	UC(NCC1)	NCC2	UC(NCC2)
Breast	97.8	1.7	97.9	1.9
German	76.6	5.9	77.8	11.6

Table 4: Results of the Naive classifiers

carried out in this particular case of credal sets.

The simple method we present here have an excellent behavior as compared with existing procedures. We plan to extend the experiments allowing general sets of possible values as output for the classifier. We could measure the degree of success by randomly assigning one of the possible values in the obtained set, or better by giving a loss function which depends of the presence of the correct value in the set and the number of elements in this set.

In our experiments we have rejected the missing data, however as pointed out in [24] credal sets can be an appropriate tool to deal with missing data and so they can be naturally incorporated to credal classification trees. We plan to investigate this possibility in the future.

We think that it is possible to improve our method. In the future, we plan to introduce a mixture of our method and the Naive Bayesian classifier. We also want to test the possibility of introducing more than one variable in each branching step, with the aim to discover more complex relationships without marginal significance.

Acknowledgments

This work has been supported by CICYT under project Elvira (TIC97-1135-C04-01).

We would like to thank Marco Zaffalon for his interesting and insightful comments and suggestions.

References

[1] S. Acid. *Métodos de aprendizaje de Redes de Creencia. Aplicación a la Clasificación.* PhD thesis, Universidad de Granada, 1999.

[2] J. Abellán and S. Moral. Completing a Total Uncertainty Measure in Dempster-Shafer Theory. *Int. J. General Systems*, 28:299–314, 1999.

[3] J. Abellán and S. Moral. A Non-specificity Measure for Convex Sets of Probability Distributions. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 8:357–367, 2000.

[4] J. Abellán and S. Moral. Randomness Measures for Credal Sets. Unpublished manuscript, 2000.

[5] L. Breiman, J.H. Friedman, R.A. Olshen, and C.J. Stone. *Classification and Regression Trees.* Wadsworth Statistics, Probability Series, Belmont, 1984.

[6] G. Choquet. Théorie des Capacités. *Ann. Inst. Fourier*, 5:131–292, 1953/54.

[7] A.P. Dempster. Upper and Lower Probabilities Induced by a Multivaluated Mapping, *Ann. Math. Statistic*, 38:325–339, 1967.

[8] D. Dubois and H. Prade. A Note on Measure of Specificity for Fuzzy Sets. *BUSEFAL*, 19:83–89, 1984.

[9] D. Dubois and H. Prade. Properties and Measures of Information in Evidence and Possibility Theories. *Fuzzy Sets and Systems*, 24:183–196, 1987.

[10] U.M. Fayyad and K.B. Irani. Multi-valued Interval Discretization of Continuous-valued Attributes for Classification Learning. *Proceeding of the 13th International Joint Conference on Artificial Intelligence*, Morgan Kaufmann, San Mateo, 1022-1027, 1993.

[11] D. Harmanec and G.J. Klir. Measuring Total Uncertainty in Dempster-Shafer Theory: a Novel Approach, *Int. J. General System*, 22:405–419, 1994.

[12] G.J. Klir and M.J. Wierman. *Uncertainty-Based Information*, Phisica-Velag, 1998.

[13] S. Kullback. *Information Theory and Statistics*, Dover, 1968.

[14] Y. Maeda and H. Ichihashi. A Uncertainty Measure with Monotonicity under the Random Set Inclusion, *Int. J. General Systems* 21:379–392, 1993.

[15] J.R. Quinlan. Induction of decision trees, *Machine Learning*, 1:81–106, 1986.

[16] J.R. Quinlan. *Programs for Machine Learning.* Morgan Kaufmann series in Machine Learning, 1993.

[17] G. Shafer. *A Mathematical Theory of Evidence.* Princeton University Press, Princeton, 1976.

[18] C.E. Shannon. A mathematical theory of communication. *The Bell System Technical Journal*, 27:379–423,623–656, 1948.

[19] P. Walley. *Statistical Reasoning with Imprecise Probabilities.* Chapman and Hall, New York, 1991.

[20] P. Walley. Inferences from Multinomial Data: Learning about a Bag of Marbles. *J.R. Statist. Soc. B*, 58:3–57, 1996.

[21] R.R. Yager. Entropy and Specificity in a Mathematical Theory of Evidence. *Int. J. General Systems*, 9:249–260, 1983.

[22] M. Zaffalon. A Credal Approach to Naive Classification. *Proceedings of the First International Symposium on Imprecise Probabilities and their Applications*, 405-414, 1999.

[23] M. Zaffalon. The Naive Credal Classifier. To appear in: *Journal of Statistical Planning and Inference*, 2001.

[24] M. Zaffalon. Exact Credal Treatment of Missing Data. To appear in: *Journal of Statistical Planning and Inference*, 2001.