Handling uncertain labels in multiclass problems using belief decision trees

P. Vannoorenberghe and T. Denœux
HEUDIASYC, UMR 6599 CNRS
Université de Technologie de Compiègne,
BP 20529 F-60205 Compiègne Cedex, France
Patrick.Vannoorenberghe,tdenoeux@hds.utc.fr

Abstract

This paper investigates the induction of decision trees based on the theory of belief functions. This framework allows to handle training examples whose labeling is uncertain or imprecise. A former proposal to build decision trees for two-class problems is extended to multiple classes. The method consists in combining trees obtained from various two-class coarsenings of the initial frame.

Keywords: Evidential reasoning, Dempster-Shafer theory, classification, learning.

1 Introduction

In recent years, the decision tree (DT) approach has become increasingly popular in the Machine Learning community [2, 14, 13]. A decision tree is a direct and acyclic graph in which each node is either a decision node or a leaf node. To each decision node is associated a test based on attribute values, and a node has two or more successors (depending on the number of possible outcomes of the test). The most commonly used decision tree classifiers are binary trees which use a single feature at each node with two outcomes. This results in decision boundaries that are parallel to the feature axes. Consequently, such classification rules are suboptimal, but give the possibility to interpret each decision rule in terms of individual features.

Given a learning set $\mathcal{L}$ composed of n patterns $\mathbf{x}_i$ with known classification ω_i, the DT induction mechanism follows a top-down strategy for splitting nodes, based on an impurity measure derived from the examples reaching the node. When the tree classifies all learning examples in $\mathcal{L}$, the process is stopped. This procedure would obviously lead to overfitting without a procedure to limit the complexity of the tree using, i.e., a pruning approach [3]. After the construction of the tree, each leaf is labeled with a class, and the tree is used to classify new patterns.

More recently, several DT induction methods based on the Dempster-Shafer Theory (DST) of belief functions [4, 8] have been introduced, giving rise to the notion of Belief Decision Tree (BDT). In this paper, we only consider the approach introduced by Denœux and Skarstein-Bjanger [4]. Thanks to the greater flexibility of DST to represent different kinds of knowledge (from total ignorance to full knowledge), BDT's allow to process training sets whose labeling has been specified with belief functions (which can include probabilistic, possibilistic or imprecise labels). An impurity measure, based on a total uncertainty criterion, is used to grow the tree and has the advantage of defining simultaneously the pruning strategy. The main objectives of this paper are to extend the method described in [4], which was originally restricted to two-class problems, and to show how uncertain class labels can be handled by this approach.

Different solutions have been proposed to decompose a K-class problem into several 2-

class problems [9]. The "one-against-one" approach consists in considering each pair of classes [10], with the drawback that the number of classifiers to train increases rapidely with the number K of classes. More sophisticated schemes, based on error-correcting codes, have been proposed by Dietterich [6]. In this paper, we focus on the simple "one-against-all" approach in which one class is selected at a time, and the K other classes are aggregated to form a new class [12]. In that way, a K-class learning task is decomposed into K two-class problems, corresponding to K *coarsenings* of the initial set of classes. We thus consider a strategy similar to the one proposed by Marsala and Bouchon-Meunier [12] in the case of fuzzy DT's, and study its application in the context of DST.

This paper is organized as follows. The basic concepts of belief function theory are first briefly introduced, including uncertainty measures and credal inference (Section 2). The proposed methodology based on the induction of BDT's and the way to handle uncertain labels in this framework are described in Section 3. Finally, Section 4 presents some experimental results.

2 Background

2.1 Belief functions

Let $\Omega = \{\omega_1, \cdots, \omega_k, \cdots, \omega_K\}$ be a finite space called the *frame of discernment*. A belief function *bel* is a function from 2^Ω to $[0, 1]$ defined as:

$$bel(A) = \sum_{\emptyset \neq B \subseteq A} m(B) \qquad \forall\, A \subseteq \Omega \qquad (1)$$

where m, called basic belief assignment (bba), is a function from 2^Ω to $[0, 1]$ verifying

$$\sum_{A \subseteq \Omega} m(A) = 1\ .$$

Each subset $A \subseteq \Omega$ such as $m(A) > 0$ is called a focal element of m. Among the functions derived from m introduced in Shafer's book [15], the commonality function q is defined as:

$$q(A) = \sum_{B \supseteq A} m(B) \qquad \forall\, A \subseteq \Omega. \qquad (2)$$

Functions m, *bel* and q are in one-to-one correspondence [15] and can be seen as three facets of the same piece of information[1]. Let q_1 and q_2 denote the commonality functions related to two bba's m_1 and m_2 induced by *distinct* items of evidence. The conjunctive combination of these two pieces of evidence ($m = m_1 \cap m_2$) can be computed from q_1 and q_2 as:

$$q(A) = q_1(A) q_2(A) \qquad \forall\, A \subseteq \Omega. \qquad (3)$$

This rule is sometimes referred to as the (unnormalized) Dempster's rule of combination. Based on rationality arguments developed in the TBM (**T**ransferable **B**elief **M**odel), Smets [17] proposes to transform m into a probability function p_m on Ω (called the *pignistic* probability function) defined for all $\omega_k \in \Omega$ as:

$$p_m(\omega_k) = \sum_{A \ni \omega_k} \frac{m(A)}{|A|} \qquad (4)$$

where $|A|$ denotes the cardinality of $A \subseteq \Omega$. In this transformation, the mass of belief $m(A)$ is distributed equally among the elements of A. This pignistic probability function is used in the TBM for decision making.

2.2 Uncertainty in DST

Because a belief function can represent several kinds of knowledge, it constitutes a rich and flexible way to represent uncertainty. As remarked by Klir [11], a belief function can model two different kinds of uncertainty: nonspecificity and conflict. A measure of nonspecificity, which generalizes the Hartley measure to belief functions, was introduced by Dubois and Prade [7]. It is defined as:

$$N(m) = \sum_{A \subseteq \Omega} m(A) \log_2 |A|. \qquad (5)$$

Since focal elements of probability measures are singletons, nonspecificity is null for probability functions, and it is maximal ($\log_2 |\Omega|$)

[1] In the sequel and when necessary, the notation $m^\Omega[data]$ will be used to denote a bba on domain Ω based on observed $[data]$.

for the vacuous belief function. Several measures of conflict, viewed as generalized Shannon entropy measures, have also been introduced [11]. One such measure is *discord*, defined as:

$$D(m) = -\sum_{A \subseteq \Omega} m(A) \log_2 p_m(A) \qquad (6)$$

which is maximal ($\log_2 |\Omega|$) for the uniform probability distribution on Ω. Finally, a measure U_λ of total uncertainty can be defined using a linear combination of N and D:

$$U(m) = (1-\lambda)N(m) + \lambda D(m) \qquad (7)$$

where $\lambda \in [0,1]$ is a coefficient. The choice of λ is not theoretically justified (Klir recommends to take $\lambda = 0.5$). In the sequel, we shall see that it can be used as a regularization parameter and determined from learning data.

2.3 Refinement and Coarsening

Part of the flexibility of DST is due to the existence of justified mechanisms allowing to change the level of detail, or *granularity* of the frame of discernement. In this section, we briefly recall the concepts of refinement and coarsening of a frame of discernment ([15], p.115), which play a key role in the theory. Let Ω and Θ be two finite sets. A mapping ρ from 2^Θ to 2^Ω is called a refining if and only if it verifies:

$$\rho(\{\theta\}) \neq \emptyset \qquad \forall\, \theta \in \Theta,$$
$$\rho(\{\theta\}) \cap \rho(\{\theta'\}) = \emptyset \qquad \forall \theta \neq \theta',$$
$$\bigcup_{\theta \in \Theta} \rho(\{\theta\}) = \Omega.$$

In other terms, the sets $\rho(\{\theta\})$, $\theta \in \Theta$ constitute a partition of Ω. Θ is then called a coarsening of Ω, and Ω is called a refinement of Θ. Given a bba m^Θ defined on Θ, we can define its *vacuous extension* (see [15] p. 146) m^Ω on Ω by transferring each mass $m^\Theta(A)$ to $\rho(A)$, for all subset A of Θ:

$$m^\Omega(\rho(A)) = m^\Theta(A) \qquad \forall\, A \subseteq \Theta. \qquad (8)$$

Conversely, let m^Ω be a bba on Ω. Transferring m^Ω to Θ is not so easy because, for some $B \subset \Omega$, there may exist no subset A of Θ such that $\rho(A) = B$. However, the restriction (or outer reduction) of m^Ω may still be defined as:

$$m^\Theta(A) = \sum_{\{B \subseteq \Omega \;|\; \rho(A) \cap B \neq \emptyset\}} m^\Omega(B) \qquad (9)$$

for all $A \subseteq \Theta$.

2.4 Credal inference

The aim of this section is not to give a detailed account of credal inference (i.e., statistical inference based on belief functions), but only to summarize a theoretical result obtained by Smets [16]. The problem considered here is to derive the belief function concerning the outcome of a Bernoulli trial, having observed a sequence of past outcomes. For example, let us consider a coin toss game with the two events H (head) and T (tail) leading to the set $\Theta = \{H, T\}$. The available information consists in observed outcomes from n independent trials. Given that you have observed n_h heads and n_t tails (so, $n_h + n_t = n$), the belief function derived by Smets in [16] is defined as follows:

$$m^\Theta[n_h, n_t](\{H\}) = \frac{n_h}{n+1} \qquad (10)$$
$$m^\Theta[n_h, n_t](\{T\}) = \frac{n_t}{n+1} \qquad (11)$$
$$m^\Theta[n_h, n_t](\Theta) = \frac{1}{n+1}. \qquad (12)$$

This belief function converges to the true probability when n tends to infinity. The method described in [16] can, in principle, be generalized to more than 2 outcomes. However, the calculations become quite cumbersome, and the counterparts of (10)-(12) in the general case are not available to date.

3 Induction of belief decision trees

In this section, we briefly summarize the method presented in [4], and consider extensions to multi-class problems for both precise and uncertain labels.

3.1 Two classes, certain labels

Let us first consider a learning set $\mathcal{L} = \{(\mathbf{x}_i, \omega_i), i = 1, \cdots, n\}$. Note that, in this section, we only consider the problem where the true class $\omega_i \in \Omega$ is exactly known for each example in $\mathcal{L}$, and $|\Omega| = 2$. At each node t, a belief function quantifying the degree of belief about the class of an example reaching t is first build. This belief function, denoted $m^{\Omega}[t]$, is defined from examples belonging to node t as:

$$m^{\Omega}[t](\{\omega_k\}) = \frac{n_k(t)}{n(t)+1},\ k \in \{1,2\} \quad (13)$$

$$m^{\Omega}[t](\Omega) = \frac{1}{n(t)+1}, \quad (14)$$

where $n_k(t)$ is the total number of examples with label ω_k reaching t, and $n(t) = n_1(t) + n_2(t)$. Note that equations (13) and (14) are just (10)-(12) with different notations.

For each node t, an impurity measure is computed from the belief function $m^{\Omega}[t]$ using the total uncertainty measure:

$$U_\lambda(t) = (1-\lambda)N(m^{\Omega}[t]) + \lambda D(m^{\Omega}[t])\ . \quad (15)$$

In this equation, the term corresponding to the non-specificity $N(m^{\Omega}[t])$ is a decreasing function of $n(t)$. It is given by:

$$N(m^{\Omega}[t]) = \frac{1}{n(t)+1}. \quad (16)$$

The discord measure $D(m^{\Omega}[t])$ depends on the representation of examples belonging to node t. It is expressed as follows:

$$D(m^{\Omega}[t]) = -\sum_{k=1}^{2} \frac{n_k(t)}{n(t)+1} \log_2 \left(\frac{n_k(t)+1/2}{n(t)+1} \right). \quad (17)$$

Finally, this impurity measure is used at node t to choose a candidate split s which divides t into two nodes t_L and t_R. The goodness of a split s is defined as a decrease in impurity by:

$$\Delta U_\lambda(s,t) = U_\lambda(t) - (p_L U_\lambda(t_L) + p_R U_\lambda(t_R)) \quad (18)$$

where p_L and p_R are, respectively, the proportions of examples reaching t_L and t_R. The best split $\hat{s}$ is chosen by testing all possible splits for each attribute.

One of the advantages of this technique is that the tree growing can be controlled using parameter λ. In fact, according to the value of λ, it is possible to give more importance to the non-specificity term which penalizes small nodes. Optimizing this parameter by cross-validation allows to build smaller trees, thus avoiding overtraining.

3.2 Two classes, uncertain labels

In [4], the problem of handling uncertain labels is solved for two-class problems. In this context, the available learning set is given by: $\mathcal{L} = \{(\mathbf{x}_i, m_i^{\Omega}), i = 1, \cdots, n\}$ where m_i^{Ω} is defined on $\Omega = \{\omega_1, \omega_2\}$ and represents the knowledge on the label of the i^{th} example. This belief function can represent different forms of label including probabilistic, possibilistic or imprecise labels. Equations (10)-(12) can be extended to take into account uncertain labels. The belief function $m^{\Omega}[t]$ at node t is then derived from the $n(t)$ belief functions m_i^{Ω}. The expression of this belief function is given in Appendix A. According to the belief function $m^{\Omega}[t]$, the impurity measure is computed in the same manner as previously and leads to a similar tree growing strategy.

In the following section, we introduce a multi-class generalization of the method developed in [4], which allows to handle the most general case in which each example is labeled by a general belief function.

3.3 K classes, general case

A standard way of handling a K-class problem is to decompose it into several 2-class subproblems. One way to do this is to train K binary classifiers, each classifier attempting to discriminate between one class ω_k and all other classes. When the learning set is of the form

$$\mathcal{L} = \{(\mathbf{x}_i, m_i^{\Omega}), i = 1, \cdots, n\}, \quad (19)$$

where m_i^Ω is a bba defined on Ω, this approach implies transforming each bba m_i^Ω originally defined on Ω into a bba defined on the 2-class coarsened frame. For each coarsening, a tree is grown, and the resulting K trees are combined using the averaging operator.

More precisely, let us denote by Θ_k the following coarsening of Ω:

$$\Theta_k = \{\theta_{k,1}, \theta_{k,2}\} \tag{20}$$

with $\theta_{k,1} = \{\omega_k\}$ and $\theta_{k,2} = \overline{\{\omega_k\}}$ (the complement of $\{\omega_k\}$ in Ω).

Each bba m_i^Ω defined on Ω may be transformed into a bba $m_i^{\Theta_k}$ on Θ_k using (9) which leads to the the following transformation:

$$\begin{aligned} m_i^{\Theta_k}(\{\theta_{k,1}\}) &= m_i^\Omega(\{\omega_k\}) \\ m_i^{\Theta_k}(\{\theta_{k,2}\}) &= \sum_{\{A\subseteq\Omega \mid \omega_k\notin A\}} m_i^\Omega(A) \\ m_i^{\Theta_k}(\Theta_k) &= \sum_{\{A\subseteq\Omega \mid |A|>1, \omega_k\in A\}} m_i^\Omega(A). \end{aligned}$$

Each of the K coarsenings thus leads to a training set $\mathcal{L}_k = \{(\mathbf{x}_i, m_i^{\Theta_k}), i = 1, \cdots, n\}$, which is used to build a decision tree using the approach described in Section 3.2.

At the testing step, we obtain, for each input vector $\mathbf{x}$, K bba's $\widehat{m}^{\Theta_k}$, each defined on a distinct coarsening Θ_k. These pieces of evidence can be expressed in a common frame by taking their vacuous extensions in Ω, using (8):

$$\widehat{m}_k^\Omega(\{\omega_k\}) = \widehat{m}^{\Theta_k}(\{\theta_{k,1}\}) \tag{21}$$
$$\widehat{m}_k^\Omega(\Omega \setminus \{\omega_k\}) = \widehat{m}^{\Theta_k}(\{\theta_{k,2}\}) \tag{22}$$
$$\widehat{m}_k^\Omega(\Omega) = \widehat{m}^{\Theta_k}(\Theta_k). \tag{23}$$

Because information sources are not independent, Dempster's rule of combination cannot be used to combine the bba's $\widehat{m}_k^\Omega$, $k = 1, \ldots, K$. An alternative is to use the averaging operator, which leads to

$$\widehat{m}^\Omega = \frac{1}{K}\sum_{k=1}^{K} \widehat{m}_k^\Omega. \tag{24}$$

3.4 Evaluation

Performance assessment is an important issue in the design of a classifier. In a decision-theoretic setting, this problem is formalized by considering a set of actions $\mathcal{A}$, and a loss function $L : \mathcal{A} \times \Omega \mapsto \mathbb{R}$, where $L(\alpha, \omega)$ is the loss incurred if one selects action α and the true state of nature is ω. Typically, each action in $\mathcal{A}$ corresponds to the choice of a class, and the loss is one for misclassification, and 0 for correct classification. The performance of a classifier $c : \mathbb{R}^d \mapsto \mathcal{A}$ can then be measured by taking the expectation of $L(c(\mathbf{x}), \omega)$ over both $\mathbf{x}$ and ω. This expectation is usually estimated by a sample average over test data.

In our case, this framework needs to be extended in two directions:

- the output of a BDT classifier is a belief function: the set of actions is thus a set of belief functions; we then need to define the loss associated to an output bba $\widehat{m}$ when the true state of nature is ω;
- the test set may be of the form defined in (19), i.e., the class of test pattern may be only partially known.

A first solution was proposed in [4],[5]. This solution postulates the following loss function:

$$L(\widehat{m}, m) = 1 - \sum_{A\subseteq\Omega} m(A) p_{\widehat{m}}(A) \tag{25}$$

where $\widehat{m}$ is the output bba produced by the classifier, and m is a bba that quantifies the uncertainty concerning the true state of nature ω. A nice property of this loss function is that, when $m(\Omega) = 1$, $L(\widehat{m}, m) = 0$ whatever $\widehat{m}$, which seems reasonable. Deeper understanding of this loss function can be gained by observing that:

$$\begin{aligned} L(\widehat{m}, m) &= 1 - \sum_{A\subseteq\Omega} m(A) \sum_{B\subseteq\Omega} \widehat{m}(B) \frac{|B\cap A|}{|B|} \\ &= 1 - \sum_{A,B\subseteq\Omega} m(A)\widehat{m}(B)\mathrm{Incl}(B, A) \end{aligned}$$

where $\mathrm{Incl}(B, A) = |B \cap A|/|B|$ is the degree of inclusion of B in A. An alternative form of

$L(\widehat{m}, m)$ is given by

$$L(\widehat{m}, m) = 1 - \sum_{A \subseteq \Omega} m(A) \sum_{\omega \in A} p_{\widehat{m}}(\omega) \quad (26)$$

$$= 1 - \sum_{\omega \in \Omega} p_{\widehat{m}}(\omega) \sum_{A \ni \omega} m(A) \quad (27)$$

$$= 1 - \sum_{\omega \in \Omega} p_{\widehat{m}}(\omega) q(\{\omega\}) \ . \quad (28)$$

An alternative solution is to assume that the loss of providing bba $\widehat{m}$ when the true state of nature is ω is:

$$L(\widehat{m}, \omega) = 1 - p_{\widehat{m}}(\omega), \quad (29)$$

which generalizes 0-1 losses. If the true state of nature ω is unknown but we have a bba m on Ω, it is then natural to take the expectation of $L(\widehat{m}, \omega)$ with respect to the pignistic probability measure associated to m. The expected loss is then

$$C(\widehat{m}, m) = \sum_{\omega \in \Omega} p_m(\omega) \left(1 - p_{\widehat{m}}(\omega)\right) \quad (30)$$

$$= 1 - \sum_{\omega \in \Omega} p_m(\omega) p_{\widehat{m}}(\omega) \quad (31)$$

which can be compared to (28).

We can therefore propose two criteria to evaluate the performance of a BDT on a test set of n' examples $(\mathbf{x}_i, m_i)$, $i = 1, \ldots, n'$:

$$C_1 = 1 - \frac{1}{n'} \sum_{i=1}^{n'} \sum_{\omega \in \Omega} p_{\widehat{m}_i}(\omega) q_i(\{\omega\})$$

$$C_2 = 1 - \frac{1}{n'} \sum_{i=1}^{n'} \sum_{\omega \in \Omega} p_{\widehat{m}_i}(\omega) p_{m_i}(\{\omega\})$$

where q_i is the commonality function associated to m_i, and $\widehat{m}_i$ is the output bba for example i.

4 Simulations

The method described in this paper was applied to real data concerning acoustic emission testing of pressure vessels[2]. The data consists in 37 examples described by 27 features.

[2]These data were collected by the Centre Technique des Industries Mécaniques (CETIM) in Senlis, France.

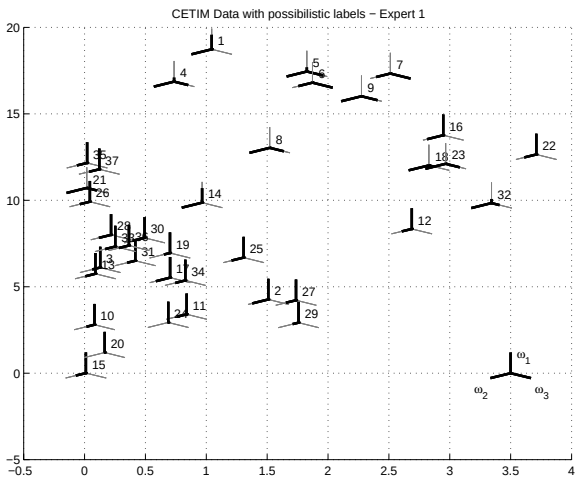

Figure 1: The data in a two-dimensional subspace of the feature space. For each example, the three bars correspond to the three classes, and their lengths are proportional to the degrees of possibility given by one expert.

Each training pattern corresponds to a cluster of acoustic emission signals, and belongs to one of three classes: minor, major, or critical source. Two different experts were asked to assess, for each training example, the degree of possibility that this example belongs to each class, resulting in 2 different possibility distributions for each example. Figure 1 displays the data in a two-dimensional subspace of the feature space, together with possibilistic labels provided by one of the experts.

A possibility measure is known to be formally equivalent to a consonant belief function, i.e., a belief function with nested focal elements [5]. Hence, possibilistic labels are a special case of evidential labels considered in this paper.

Three training sets were considered: the data labeled by each of the two experts, and the data labeled by a conjunctive combination of the labels provided by the two experts (by taking the minimum of the possibility distributions, and normalizing). For each training set, we considered two learning strategies. In the first one, the possibilistic labels were transformed into hard labels by selecting only, for each example, the class with the highest possibility. In the second strategy, the possibilistic labels were used as explained in Sec-

Data set	E	C_1	C_2
expert 1	0.32	0.38	0.58
expert 2	0.35	0.40	0.59
1+2 (labels)	0.32	0.43	0.59
1+2 (decisions)	0.33	0.42	0.56

Table 1: Results with crisp labels

Data set	E	C_1	C_2
expert 1	0.32	0.39	0.59
expert 2	0.29	0.41	0.60
1+2 (labels)	0.29	0.45	0.59
1+2 (decisions)	0.30	0.37	0.57

Table 2: Results with possibilistic labels

tions 3.2 and 3.3.

Additionally, we tested another way of combining the information provided by the two experts: the classifiers training with the data from each of the two experts were combined at the decision level, using the Dempster's rule of combination.

For these experiments, we used 10-fold cross-validation to optimize the value of the parameter λ and to evaluate the performance of the proposed method. For strategy 1 (training with hard labels), the misclassification error rate was used as a performance criterion. For strategy 2 (training with possibilistic labels), the use of both criteria C_1 and C_2 described in Section 3.4 was investigated. Both criteria proved equivalent, and only the results with C_2 were retained. The results are summarized in Tables 1 (strategy 1) and 2 (strategy 2).

We can see that:

- training with possibilistic labels tends to decrease the error rate, which is an indication that our method succeeds in using more refined information than just hard labels (similar results were reported in [4] and [5] with different data sets);
- combining the expert information improves the results, whatever the method used (possibilistic combination of the labels, or conjunctive combination of the output bba's). This shows that collecting information from several experts may be useful when the class of training patterns can only be assessed subjectively.

5 Conclusion

In this paper, a method for handling uncertain labels using belief decision trees has been introduced. The method allows to process training sets whose labelling has been specified with a belief function. A method to grow and aggregate trees learnt from 2-class versions of the training set has been developed. The use of error correcting output codes could also be envisaged to cope with more complex problems involving a higher number of classes. Bagged and randomized versions of belief decision trees can also be introduced to reduce the variance of these classification rules.

6 Acknowledgements

The authors thank Catherine Hervé and Cristel Rigault from CETIM for providing the acoustic emission data.

A Appendix

Suppose that we have performed n independent Bernoulli experiments as in Section 2.4, but that the outcomes could only be partially observed (for example, the urn experiment was observed at a distance, so that the results of some trials could only be partially observed). Let m_i^Θ be the bba describing one's belief concerning the result of experiment i, and m^Θ the bba quantitying one's belief regarding the outcome of the next experiment. Based on the results in [16], it was shown in [1][4] that m^Θ is given by:

$$m^\Theta(\{H\}) = \sum_{j+k\leq n(t)} \alpha_{jk} \frac{j}{j+k+1},$$

$$m^\Theta(\{T\}) = \sum_{j+k\leq n(t)} \alpha_{jk} \frac{k}{j+k+1},$$

$$m^\Theta(\Theta) = \sum_{j+k\leq n(t)} \alpha_{jk} \frac{1}{j+k+1}.$$

where α_{jk} is defined as:

$$\alpha_{jk} = \sum_{\{I_1,I_2,I_3\}} \left(\prod_{i_1 \in I_1} m^{\Theta}_{i_1}(\{H\}) \times \prod_{i_2 \in I_2} m^{\Theta}_{i_2}(\{T\}) \times \prod_{i_3 \in I_3} m^{\Theta}_{i_3}(\Theta) \right)$$

where $\{I_1, I_2, I_3\}$ ranges over all partitions of $\{1, \ldots, n\}$ such that $|I_1| = j$ and $|I_2| = k$. Note that these expressions are similar to equations (10)-(12) when the bba's m^{Θ}_i are derived from precise observations.

References

[1] M. Skarstein Bjanger. Induction of decision trees from partially classified data using belief functions. Master's thesis, Norwegian Univ. of Science and Technology, Dpt of Computer and Information Science, Feb. 2000. Available at `http://www.hds.utc.fr/~tdenoeux`.

[2] L. Breiman, J.H. Friedman, R. Olshen, and C.J. Stone. *Classification and regression trees*. Wadsworth and Brooks/Cole, Monterey, CA, 1984.

[3] L. A. Breslow and D. W. Aha. Simplifying decision trees: A survey. *Knowledge Engineering Review*, 12(1):1–40, 1997.

[4] T. Denœux and M. Skarstein Bjanger. Induction of decision trees from partially classified data using belief functions. In *Proceedings of SMC'2000*, pages 2923–2928, Nashville, USA, 2000. IEEE.

[5] T. Denœux and L. M. Zouhal. Handling possibilistic labels in pattern classification using evidential reasoning. *Fuzzy Sets and Systems*, 122(3):47–62, 2001.

[6] T.G. Dietterich and G. Bakiri. Solving multiclass learning problems via error-correcting output codes. *Journal of Artificial Intelligence Research*, 2:263–286, 1995.

[7] D. Dubois and H. Prade. A note on measures of specificity for fuzzy sets. *International Journal of General Systems*, 10:279–283, 1985.

[8] Z. Elouedi, K. Mellouli, and P. Smets. Belief decision trees: Theoretical foundations. *International Journal of Approximate Reasoning*, 28:91–124, 2001.

[9] J.H. Friedman. Another approach to polychotomous classification. Technical report, Department of Statistics, Stanford University, 1996.

[10] T. Hastie and R. Tibshirani. Classification by pairwise coupling. In Michael I. Jordan, Michael J. Kearns, and Sara A. Solla, editors, *Advances in Neural Information Processing Systems*, volume 10. The MIT Press, 1998.

[11] G.J. Klir and M.J. Wierman. *Uncertainty-Based Information*. Physica-Verlag, Heidelberg, Germany, 1998.

[12] Ch. Marsala and B. Bouchon-Meunier. Forests of fuzzy decision trees. In *IFSA'97 World Congress*, Prague, 1997.

[13] S.K. Murthy. Automatic construction of decision trees from data: A multi-disciplinary survey. *Data Mining and Knowledge Discovery*, 2(4):345–389, 1998.

[14] J.R. Quinlan. *C4.5 : Programs for Machine Learning*. Morgan Kaufmann, San Mateo, CA, 1993.

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

[16] P. Smets. What is Dempster-Shafer's model? In R.R. Yager, M. Fedrizzi, and J. Kacprzyk, editors, *Advances in the Dempster-Shafer Theory of Evidence*, pages 5–34. Wiley, 1994.

[17] Ph. Smets and R. Kennes. The Transferable Belief Model. *Artificial Intelligence*, 66(2):191–234, 1994.