Pairwise Classifier Combination using Belief Functions

Benjamin Quost, Thierry Denœux and Marie-Hélène Masson

UMR CNRS 6599 Heudiasyc
Université de Technologie de Compiègne
BP 20529 - F-60205 Compiègne cedex - France

November 6, 2006

Abstract

In the so-called pairwise approach to polychotomous classification, a multi-class problem is solved by combining classifiers trained to discriminate between each pair of classes. In this paper, this approach is revisited in the framework of the Dempster-Shafer theory of belief functions, a non-probabilistic framework for quantifying and manipulating partial knowledge. It is proposed to interpret the output of each pairwise classifiers by a conditional belief function. The problem of classifier combination then amounts to computing the non-conditional belief function which is the most consistent, according to some criterion, with the conditional belief functions provided by the classifiers. Experiments with various datasets demonstrate the good performances of this method as compared to previous approaches to the same problem.

Keywords: Polychotomous classification, Dempster-Shafer theory, Evidence Theory, Classification, classifier fusion.

1 Introduction

In pattern classification problems, we typically have a training set composed of n p-dimensional feature vectors $\mathbf{x}_i$, $i = 1, \ldots, n$ with associated class labels y_i taking values in a set of classes $\Omega = \{\omega_1, \ldots, \omega_K\}$. A classifier has to be trained to map the input space $\mathbb{R}^p$ to the label space Ω, on the basis of the training set; it should then be able to predict the label of any previously unseen pattern $\mathbf{x}$. The complexity of a classifier has to fit that of the classification task: problems with many overlapping classes and non linear class boundaries require complex classifiers, whose training can be very time-consuming.

Polychotomous classification problems, i.e., problems involving more than two classes, can be solved directly using a multiclass classifier, or by combining several binary classifiers [7]. In the latter approach, complexity is reduced by decomposing a difficult task into simpler subtasks. Furthermore, some classifiers such as logistic regression or support vector machines are more adapted to two-class problems.

In this article, we propose a new approach to pairwise classifier combination based on the Transferable Belief Model (TBM) [16], an interpretation of the theory of belief functions. This framework allows the represention of partial knowledge, and thus provides an adequate theoretical basis for combining classifiers trained using parts of the training set.

First, pairwise classifier combination as well as other approaches to polychotomous classification are reviewed in Section 2. The TBM is then summarized in Section 3, and pairwise classifier combination is formalized within this framework in Section 4. Experimental results are presented in Section 5, and Section 6 concludes the paper.

2 Pairwise Classifier Combination

2.1 Different Decompositions of a Multiclass Problem

When reducing a polychotomous classification problems into two-class problems, dichotomies can be obtained in various ways. Each class can be opposed to all others (one-against-all decomposition): K binary classifiers are trained, each one using the whole set of training patterns. Alternatively, each class can be opposed to each other (one-against-one or pairwise decomposition): in that case, $K(K-1)/2$ pairwise classifiers are trained using

patterns from each pair (ω_k, ω_ℓ) of classes. Fewer classifiers are used in the former case, whereas the training cost of a single classifier is lower in the latter.

In [12], Ω is decomposed using the one-versus-one scheme, but $\mathbf{x}$ is classified sequentially by $K-1$ classifiers. Let $\mathbf{x}$ be evaluated, at some iteration, by a pairwise classifier trained to separate ω_i from ω_j; if ω_j is rejected, then $\mathbf{x}$ is evaluated by the classifier trained to separate ω_i from ω_k $(k \neq i, j)$, otherwise it is evaluated by the classifier trained to separate ω_j from ω_l $(l \neq i, j)$; and so on, until its actual class is found, after $K-1$ evaluation-rejection steps.

Error-Correcting Output Codes [5] provide yet another framework for binary decomposition. Binary classifiers $\mathcal{E}_i$ $(i = 1 \ldots N)$ are trained to separate two sets of classes A_i and B_i. An N-dimensional codeword vector $\mathbf{c}_k$ is defined for each class ω_k, as

$$c_{ki} = \begin{cases} +1, & \omega_k \in A_i \\ -1, & \omega_k \in B_i \\ 0, & \omega_k \notin A_i \cup B_i. \end{cases}$$

When evaluating $\mathbf{x}$, an output vector is computed, and $\mathbf{x}$ is assigned to the class with the nearest $\mathbf{c}_k$, according to some distance measure.

In the rest of this article, we focus on the pairwise approach.

2.2 Different Methods for Combining Pairwise Probabilistic Classifiers

Various pairwise classifier combination schemes have been proposed, according to the nature of the classifier outputs. If a pairwise classifier, trained to separate class ω_i from class ω_j, provides an estimate r_{ij} of the conditional posterior probability $\mu_{ij} = \mathbb{P}(\omega_i | \{\omega_i, \omega_j\}, \mathbf{x})$, the posterior probabilities $p_i = \mathbb{P}(\omega_i | \mathbf{x})$ may be estimated, by exploiting the relations $\mu_{ij} = p_i/(p_i + p_j)$, for all $j > i$.

Computing the p_i, satisfying $0 \leq p_i \leq 1$, $\sum p_i = 1$, such that $p_i/(p_i + p_j) = r_{ij}$ is an overdetermined problem involving $K-1$ variables and $(K-1)K/2$ equality constraints. Consequently, this problem generally does not admit an exact solution. However, the p_i can then be determined so that the μ_{ij} are close to the r_{ij} according to an error criterion.

Let $\mathbf{p}$ and $\widehat{\mathbf{p}}$ be the respective vectors of p_i and $\widehat{p}_i$. Let n_k be the number of patterns in class ω_k, and $n_{ij} = n_i + n_j$. Define $r_{ji} = 1 - r_{ij}$ and $\mu_{ji} = 1 - \mu_{ij}$. It was proposed in [8] to compute the estimates $\widehat{p}_i$ of the p_i by iteratively minimizing the negative weighted

Kullback-Leibler distance between the μ_{ij} and the r_{ij}, using gradient descent:

$$\widehat{\mathbf{p}} = \arg\min \mathcal{L}(\mathbf{p}), \tag{1}$$

$$\mathcal{L}(\mathbf{p}) = \sum_{i<j} n_{ij} \left(r_{ij} \log \frac{r_{ij}}{\mu_{ij}} + r_{ji} \log \frac{r_{ji}}{\mu_{ji}} \right), \tag{2}$$

under the constraints $\sum_i p_i = 1$, $p_i \geq 0$ for all $i = 1 \dots K$. This method will be noted as the PCpl method.

Hastie and Tibshirani [8] also remarked that each p_i may be written as:

$$p_i = \sum_{j \neq i} \left(\frac{p_i + p_j}{K-1} \right) \left(\frac{p_i}{p_i + p_j} \right). \tag{3}$$

By replacing $p_i + p_j$ by $2/K$ in the first ratio and each of the second ratios by the corresponding r_{ij}, we obtain simple estimates of the p_i:

$$\tilde{p}_i = \frac{2}{K(K-1)} \sum_{j \neq i} r_{ij}. \tag{4}$$

Although crude estimates, the $\tilde{p}_i$ are shown in [8] to have the same ordering as the $\widehat{p}_i$. They can be used as starting values in the iterative procedure, or for decision making.

In [17], two alternative non-iterative methods for estimating the p_i from the r_{ij} were introduced: they will be referred to as PEst1 and PEst2. Similar to the approach detailed in [8], the estimation of posterior probabilities was formalized as the resolution of optimization problems. It was shown that the solutions of these problems implicitly satisfy the positivity constraints, which can be omitted. The solutions can be retrieved by solving linear systems of equations.

As pointed out in [8], each pairwise classifier was trained to separate two classes only. Hence, a classifier evaluating $\mathbf{x}$ may provide an erroneous estimate r_{ij} if it was not trained to recognize the actual class of $\mathbf{x}$. Therefore, it was proposed in [11] to train additional correcting classifiers, separating classes $\{\omega_i, \omega_j\}$ from the others, to provide estimates q_{ij} of the probabilities $\mathbb{P}(\{\omega_i, \omega_j\}|\mathbf{x}) = p_i + p_j$. By replacing $p_i + p_j$ by q_{ij} in the first ratios of (3), we obtain new estimates of the p_i:

$$\tilde{p}'_i = \frac{1}{K-1} \sum_{j \neq i} r_{ij} q_{ij}. \tag{5}$$

Although this method (hereafter referred to as PCorr) may improve classification accuracy, it implies training $(K-1)K/2$ additional pairwise classifiers using the whole training set, which may be very time-consuming for large K.

3 The Transferable Belief Model

The Dempter-Shafer theory of belief functions [15] is a generalization of probability theory allowing the represention and manipulation of various forms of partial knowledge, from certainty to complete ignorance. The Transferable Belief Model (TBM) [16] is a subjectivist and non probabilistic interpretation of this theory in which a belief function is interpreted as modeling an agent's state of belief regarding the value of an unknown quantity. This theoretical framework seems well-suited to our problem, in which each pairwise classifier provides partial information about the class of the object under consideration.

The TBM aims at modelling the knowledge of the actual value of a variable y with domain $\Omega = \{\omega_1, \ldots, \omega_K\}$ refered to as the frame of discernment. In a classification problem, y represents the class of a pattern $\mathbf{x}$ to be classified. The beliefs held by an agent concerning the value of y may be quantified by a basic belief assignment (bba) m^Ω, defined as a function $m^\Omega : 2^\Omega \rightarrow [0, 1]$ verifying

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

The quantity $m^\Omega(A)$ is interpreted as the fraction of a unit mass of belief assigned to subset A given some evidence, and which cannot be assigned to any strict subset of A for lack of more specific evidence. Any subset $A \subseteq \Omega$ such that $m^\Omega(A) > 0$ is called a focal set of m^Ω. A Bayesian bba is a bba whose focal sets are singletons. The superscript Ω may sometimes be omitted when no confusion is possible.

The empty set $\emptyset$ may be a focal set. The mass $m(\emptyset)$ is then interpreted as the belief that the actual value of y lies outside the frame Ω (open-world assumption). If the frame is considered to be exhaustive (closed-world assumption), the normality condition $m(\emptyset)$ is usually assumed. A bba m such that $m(\emptyset) = 0$ is said to be normal, otherwise it is subnormal. A subnormal bba m may be normalized by dividing each mass $m(A)$ with $A \neq \emptyset$ by $1 - m(\emptyset)$. The resulting normal bba will be noted m^*.

Given a bba m^Ω, a belief function and a plausibility function can be computed as, respectively:

$$bel^\Omega(A) = \sum_{\emptyset \neq B \subseteq A} m^\Omega(B), \tag{6}$$

and

$$pl^{\Omega}(A) = \sum_{B \cap A \neq \emptyset} m^{\Omega}(B), \tag{7}$$

for all $A \subseteq \Omega$. The quantity $bel^{\Omega}(A)$ represents the degree of belief in A given the available evidence, whereas $pl^{\Omega}(A)$ is interpreted as an upper bound on the degree of belief that could be assigned to A if further evidence became available.

Two bbas m_1^{Ω} and m_2^{Ω} on the same frame Ω induced by two distinct items of evidence can be combined using the unnormalized Dempster's rule of combination, denoted by $\textcircled{\cap}$:

$$m_1^{\Omega} \textcircled{\cap} m_2^{\Omega}(A) = \sum_{B \cap C = A} m_1^{\Omega}(B) m_2^{\Omega}(C), \quad \forall A \subseteq \Omega.$$

This rule is the fundamental belief updating mechanism in the TBM. Given a bba m^{Ω} on Ω and a subset B of Ω, conditioning m^{Ω} on B is defined as the $\textcircled{\cap}$ combination of m^{Ω} with the bba m_B^{Ω} such that $m_B^{\Omega}(B) = 1$. The conditional bba given B, noted $m^{\Omega}[B]$, can be computed as follows:

$$m^{\Omega}[B](A) = \begin{cases} \sum_{C \cap B = A} m^{\Omega}(C) & \text{if } A \subseteq B, \\ 0 & \text{otherwise.} \end{cases} \tag{8}$$

Hence, any mass of belief initially assigned to $C \subseteq \Omega$ is transfered to $C \cap B$. The bba $m^{\Omega}[B]$ quantifies the agent beliefs regarding the value of y, assuming that $y \in B$. The mass $m^{\Omega}[B](\emptyset)$ quantifies the belief given by m^{Ω} to hypotheses incompatible with B or, equivalently, the belief that the actual value of y lies outside B.

The operation defined by (8) is referred to as the unnormalized Dempster's rule of conditioning. Its normalized version (corresponding to the closed-world assumption) is obtained by adding a normalization step:

$$m^{\Omega}[B]^{*}(A) = \begin{cases} \frac{m^{\Omega}[B](A)}{1 - m^{\Omega}[B](\emptyset)} & \text{if } A \subseteq \Omega, A \neq \emptyset, \\ 0 & \text{if } A = \emptyset. \end{cases} \tag{9}$$

The normalized Dempster's rule of conditioning generalizes Bayesian conditioning (it yields the same result when applied to a Bayesian bba).

In the TBM, two levels are distinguished: the *credal level* where beliefs are represented and combined using belief functions, and the *decision level* where decision making takes place. Once a decision has to be made, pignistic probabilities are computed from belief

functions. The pignistic transformation [16] equally distributes each normalized mass $m^{\Omega*}(A)$:

$$BetP^*(\omega) = \sum_{A \subseteq \Omega : \omega \in A} \frac{m^{\Omega*}(A)}{|A|}, \quad \forall \omega \in \Omega. \tag{10}$$

4 Combining Pairwise Classifiers within the TBM

We present in this section a method for combining pairwise classifiers within the TBM. The generality of this framework will allow to use for each two class problem not only probabilistic classifiers, but also credal classifiers, i.e., classifiers whose outputs are belief functions (see, e.g., [2, 1, 3, 6] for descriptions of algorithms generating credal classifiers from data).

4.1 Classifier Outputs as Conditional bbas

Let $\{E_{ij}\}_{j>i}$ be a set of classifiers evaluating the class y of a pattern $\mathbf{x}$, taking values in $\Omega = \{\omega_1, \ldots, \omega_K\}$. As each classifier E_{ij} was trained using learning examples from classes ω_i and ω_j only, the information it provides is conditional on the object belong to class ω_i or class ω_j.

Let us assume that classifier E_{ij} delivers a normal bba m^*_{ij} on the frame of discernment $\Omega_{ij} = \{\omega_i, \omega_j\}$ (it is a Bayesian bba in the case of a probabilistic classifier). This bba may be considered as the result of conditioning an unknown bba m^Ω on Ω_{ij}, using the normalized rule of conditioning (8)-(9): $m^*_{ij} = m^\Omega[\Omega_{ij}]^*$. Equivalently,

$$m^\Omega[\Omega_{ij}](A) = m^*_{ij}(A)(1 - m^\Omega[\Omega_{ij}](\emptyset)), \quad \forall A \in 2^{\Omega_{ij}} \setminus \emptyset, \forall i > j, \tag{11}$$

where $m^\Omega[\Omega_{ij}]$ is the unnnormalized conditional bba computed from m^Ω using (8). As each number $m^\Omega[\Omega_{ij}](A)$ is a linear combination of masses $m^\Omega(C)$, the above system has $3K(K-1)/2$ linear equations with $2^K - 1$ unknowns. If the normality condition $m^\Omega(\emptyset) = 0$ is not imposed, then this system has a trivial solution: the bba such that $m^\Omega(\emptyset) = 1$ verifies $m^\Omega[\Omega_{ij}](\emptyset) = 1$ and $m^\Omega[\Omega_{ij}](A) = 0$ for all $A \neq \emptyset$, for all $i > j$. To discard this solution, the normality condition could be imposed:

$$m^\Omega(\emptyset) = 0 \tag{12}$$

However, the system (11)-(12) often has no solution because the m^*_{ij}, being computed by different pairwise classifiers, are generally not consistent: there is usually no normal bba m^Ω on Ω whose normalized conditioning with respect to Ω_{ij} yields exactly m^*_{ij}, for all $i > j$. An approximate solution to system (11)-(12) may, however, be computed by solving the following quadratic problem

$$\min_{m^\Omega} \sum_{\Omega_{ij} \subseteq \Omega} \sum_{A \in 2^{\Omega_{ij}} \setminus \emptyset} \left(m^\Omega[\Omega_{ij}](A) - m^*_{ij}(A) \left(1 - m^\Omega[\Omega_{ij}](\emptyset)\right)\right)^2, \tag{13}$$

under the constraints:

$$m^\Omega(A) \geq 0, \quad \forall A \subseteq \Omega, A \neq \emptyset \tag{14}$$

$$m^\Omega(\emptyset) = 0 \tag{15}$$

$$\sum_{A \subseteq \Omega} m^\Omega(A) = 1. \tag{16}$$

This problem can be solved using any classical optimization solver.

Example 1 Let us consider a problem with three classes $\Omega = \{\omega_1, \omega_2, \omega_3\}$. We have three evidential pairwise classifiers E_{12}, E_{13} and E_{21}. Assume that, for a given pattern $\mathbf{x}$, these classifiers have provided the following bbas:

- Classifier E_{12}: $m^*_{12}(\{\omega_1\}) = 0.439$, $m^*_{12}(\{\omega_2\}) = 0.498$, $m^*_{12}(\Omega_{12}) = 0.063$;
- Classifier E_{13}: $m^*_{13}(\{\omega_1\}) = 0.849$, $m^*_{13}(\{\omega_3\}) = 0.014$, $m^*_{13}(\Omega_{13}) = 0.137$;
- Classifier E_{23}: $m^*_{23}(\{\omega_2\}) = 0.666$, $m^*_{23}(\{\omega_3\}) = 0.010$, $m^*_{23}(\Omega_{23}) = 0.324$.

Using the above method, we obtain the following bba on Ω:

$$m^\Omega(\{\omega_1\}) = 0.448, \qquad m^\Omega(\{\omega_2\}) = 0.414,$$
$$m^\Omega(\{\omega_2, \omega_3\}) = 0.056, \quad m^\Omega(\Omega) = 0.082.$$

When conditioning m^Ω on each Ω_{ij} using the normalized rule of conditioning (9), we find

- $m^\Omega[\Omega_{12}]^*(\{\omega_1\}) = 0.448$, $m^\Omega[\Omega_{12}]^*(\{\omega_2\}) = 0.470$, $m^\Omega[\Omega_{12}]^*(\Omega_{12}) = 0.082$;
- $m[\Omega_{13}]^*(\{\omega_1\}) = 0.766$, $m[\Omega_{13}]^*(\{\omega_3\}) = 0.095$, $m[\Omega_{13}]^*(\Omega_{13}) = 0.139$;
- $m[\Omega_{23}]^*(\{\omega_2\}) = 0.751$, $m[\Omega_{23}]^*(\{\omega_3\}) = 0.000$, $m[\Omega_{23}]^*(\Omega_{23}) = 0.249$,

which are the best approximations to m^*_{12}, m^*_{13} and m^*_{23} according to criterion (13). $\square$

The above approach does provide a means to recover a bba on m^Ω which is as consistent as possible with the conditional bbas provided by pairwise classifiers. However, it suffers from the same drawback as probabilistic methods PCpl, PEst1 and PEst2 described in Section 2.2: since each classifier E_{ij} was trained using examples from classes ω_i and ω_j only, its output is irrelevant when the pattern under consideration does not belong to any of these two classes. To tackle this problem, we propose to introduce additional information regarding the relevance of each classifier E_{ij}, in the form of an estimate of the mass $m^\Omega[\Omega_{ij}](\emptyset)$. A high value of $m^\Omega[\Omega_{ij}](\emptyset)$ indicates that the pattern is not likely to belong to ω_i and ω_j and, consequently, classifier E_{ij} is not relevant to classify that pattern. If none of the classifier is relevant, then this is an indication that the pattern might belong to none of the K classes, and the solution will verify $m^\Omega(\emptyset) \approx 1$. The normality condition (12) should thus be dropped. This additional information will play a role similar to that of the "correcting classifiers" introduced in [11] in a pure probabilistic setting.

4.2 Modeling Classifier Relevance

As most classification methods do not have "novelty detection" capabilities, we need additional information concerning the relevance of classifier E_{ij}, i.e., the plausibility that the current pattern belongs to class ω_i or ω_j. In Moreira and Mayoraz's method [11], a correcting classifier was trained to discriminate each pair $\{\omega_i, \omega_j\}$ from all other classes, by estimating the probability that the current pattern belongs to ω_i or ω_j. In the belief function framework, we propose to determine the *plausibility* that the pattern belongs to ω_i or ω_j. This information is directly related to the conditional bba $m^\Omega[\Omega_{ij}]$, since

$$
\begin{aligned}
m^\Omega[\Omega_{ij}](\emptyset) &= \sum_{A \cap \Omega_{ij} = \emptyset} m^\Omega(A) && (17)\\
&= 1 - \sum_{A \cap \Omega_{ij} \neq \emptyset} m^\Omega(A) && (18)\\
&= 1 - pl^\Omega(\Omega_{ij}). && (19)
\end{aligned}
$$

Consequently, the knowledge of $pl^\Omega(\Omega_{ij})$ is equivalent to that of $m^\Omega[\Omega_{ij}](\emptyset)$. Let $pl_{ij} = pl^\Omega(\Omega_{ij})$, and let m_{ij} be the bba obtained by "denormalizing" bba m^*_{ij} provided by classifier

E_{ij}:

$$m_{ij}(A) = pl_{ij} m^*_{ij}(A), \quad \forall A \subseteq \Omega_{ij}, A \neq \emptyset \tag{20}$$

$$m_{ij}(\emptyset) = 1 - pl_{ij}. \tag{21}$$

By substituting $m^*_{ij}(A)\left(1 - m^{\Omega}[\Omega_{ij}](\emptyset)\right)$ with $m_{ij}(A)$ in (13), and dropping the normality constraint, the optimization problem (13)-(16) can be replaced by

$$\min_{m^{\Omega}} \sum_{\Omega_{ij} \subseteq \Omega} \sum_{A \in 2^{\Omega_{ij}}} \left(m^{\Omega}[\Omega_{ij}](A) - m_{ij}(A)\right)^2, \tag{22}$$

under the constraints:

$$m^{\Omega}(A) \geq 0, \quad \forall A \subseteq \Omega, \tag{23}$$

$$\sum_{A \subseteq \Omega} m^{\Omega}(A) = 1. \tag{24}$$

The plausibilities $pl^{\Omega}(\Omega_{ij})$ could be determined in a manner similar to that proposed in [11], using correcting classifiers discriminating Ω_{ij} against all other classes. This, however, requires the training of $K(K-1)/2$ additional classifiers, each one with n training vectors. To decrease computing time, we propose instead to determine the pl_{ij} using one-class classifiers. For each class $\omega_i \in \Omega$, a one-class classifier may be trained to evaluate the plausibility $pl^{\Omega}(\{\omega_i\})$ that the pattern belongs to ω_i (see Sections 5.1 for an explanation of how this can be done using one-class support vector machines). The plausibility of Ω_{ij} is constrained by the following inequalities

$$\max(pl^{\Omega}(\{\omega_i\}), pl^{\Omega}(\{\omega_j\})) \leq pl_{ij} \leq pl^{\Omega}(\{\omega_i\}) + pl^{\Omega}(\{\omega_j\}). \tag{25}$$

Heuristically, we propose to estimate this quantity using a triangular conorm such as, e.g., the probabilistic t-conorm:

$$pl_{ij} = pl^{\Omega}(\{\omega_i\}) + pl^{\Omega}(\{\omega_j\}) - pl^{\Omega}(\{\omega_i\}) pl^{\Omega}(\{\omega_j\}). \tag{26}$$

Example 2 Coming back to the problem of Example 1, let us assume that one-class classifiers are now available in addition to pairwise classifiers, and that they produce the following class plausibilities:

$$pl^{\Omega}(\{\omega_1\}) = 0.555, \quad pl^{\Omega}(\{\omega_2\}) = 0.418, \quad pl^{\Omega}(\{\omega_3\}) = 0.$$

Using (26), we obtain the following plausibilities for each pair of classes $\{\omega_i, \omega_j\}$:

$$pl_{12} = 0.741, \quad pl_{13} = 0.555, \quad pl_{23} = 0.418.$$

Denormalizing the bbas m^*_{ij} provided by the pairwise classifiers using (20)-(21), we get:

- $m_{12}(\emptyset) = 0.259$, $m_{12}(\{\omega_1\}) = 0.325$, $m_{12}(\{\omega_2\}) = 0.369$, $m_{12}(\Omega_{12}) = 0.047$;
- $m_{13}(\emptyset) = 0.445$, $m_{13}(\{\omega_1\}) = 0.471$, $m_{13}(\{\omega_3\}) = 0.008$, $m_{13}(\Omega_{13}) = 0.076$;
- $m_{23}(\emptyset) = 0.582$, $m_{23}(\{\omega_2\}) = 0.279$, $m_{23}(\{\omega_3\}) = 0.004$, $m_{23}(\Omega_{23}) = 0.135$.

The minimization of (22) yields the following bba on Ω:

$$\begin{array}{lll} m^\Omega(\emptyset) = 0.218 & m^\Omega(\{\omega_1\}) = 0.372, & m^\Omega(\{\omega_2\}) = 0.254, \\ m^\Omega(\{\omega_1, \omega_2\}) = 0.039, & m^\Omega(\{\omega_2, \omega_3\}) = 0.068, & m^\Omega(\Omega) = 0.049, \end{array}$$

other masses being equal to zero. When conditioning m^Ω on each Ω_{ij} using the unnormalized rule of conditioning (8), we find

- $m^\Omega[\Omega_{12}](\emptyset) = 0.218$, $m^\Omega[\Omega_{12}](\{\omega_1\}) = 0.372$, $m^\Omega[\Omega_{12}](\{\omega_2\}) = 0.322$, $m^\Omega[\Omega_{12}](\Omega_{12}) = 0.088$;
- $m^\Omega[\Omega_{13}](\emptyset) = 0.472$, $m[\Omega_{13}](\{\omega_1\}) = 0.411$, $m[\Omega_{13}](\{\omega_3\}) = 0.068$, $m[\Omega_{13}](\Omega_{13}) = 0.049$;
- $m^\Omega[\Omega_{23}](\emptyset) = 0.590$, $m[\Omega_{23}](\{\omega_2\}) = 0.293$, $m[\Omega_{23}](\{\omega_3\}) = 0$, $m[\Omega_{23}](\Omega_{23}) = 0.117$,

which are the best approximations to m_{12}, m_{13} and m_{23} according to criterion (22). □

4.3 Variant of the method in the case of probabilistic classifiers

When the pairwise classifiers are probabilistic, they produce a probability distribution on each subset Ω_{ij}. This probability distribution can be seen either as a Bayesian bba approximating the conditional bba $m^\Omega[\Omega_{ij}]$, or as an approximation to the pignistic distribution $BetP^*_{ij}$ associated to $m^\Omega[\Omega_{ij}]$, defined using (10). Alternatively, $BetP^*_{ij}$ may be computed as

$$BetP^*_{ij}(\omega_i) = \frac{BetP_{ij}(\omega_i)}{1 - m^\Omega[\Omega_{ij}](\emptyset)} \tag{27}$$

$$BetP^*_{ij}(\omega_j) = \frac{BetP_{ij}(\omega_j)}{1 - m^\Omega[\Omega_{ij}](\emptyset)}, \tag{28}$$

where $BetP_{ij}$ is the "unnormalized" pignistic probability distribution computed from the unnormalized conditional bba $m^\Omega[\Omega_{ij}]$ as

$$BetP_{ij}(\omega_i) = m^\Omega[\Omega_{ij}](\{\omega_i\}) + \frac{m^\Omega[\Omega_{ij}](\Omega_{ij})}{2} \quad (29)$$

$$BetP_{ij}(\omega_j) = m^\Omega[\Omega_{ij}](\{\omega_j\}) + \frac{m^\Omega[\Omega_{ij}](\Omega_{ij})}{2}. \quad (30)$$

Let r^*_{ij} denote the estimated probability of class ω_i produced by classifier E_{ij}, and $r^*_{ji} = 1 - r^*_{ij}$ (the notation of Section 2.2 has been adapted to emphasize that the outputs from a probabilistic classifier are normalized, i.e., $r^*_{ij} + r^*_{ji} = 1$). The underlying bba m^Ω may then be expected to verify the following conditions

$$BetP^*_{ij}(\omega_i) \approx r^*_{ij} \quad (31)$$

$$BetP^*_{ij}(\omega_j) \approx r^*_{ji}, \quad (32)$$

or, equivalently,

$$BetP_{ij}(\omega_i) \approx r^*_{ij}\left(1 - m^\Omega[\Omega_{ij}](\emptyset)\right) \quad (33)$$

$$BetP_{ij}(\omega_j) \approx r^*_{ji}\left(1 - m^\Omega[\Omega_{ij}](\emptyset)\right), \quad (34)$$

Using the same method as in Section 4.2, let us substitute, in the above equations, $1 - m^\Omega[\Omega_{ij}](\emptyset)$ with pl_{ij} provided by one-class classifiers, and let us denote $r_{ij} = r^*_{ij} pl_{ij}$ and $r_{ji} = r^*_{ij} pl_{ij}$. We can find a bba m^Ω maximally consistent with the available information as the solution to the following quadratic optimization problem:

$$\min_{m^\Omega} \sum_{\Omega_{ij} \subseteq \Omega} \left(BetP_{ij}(\omega_i) - r_{ij}\right)^2 + \left(BetP_{ij}(\omega_j) - r_{ji}\right)^2 + \left(m^\Omega[\Omega_{ij}](\emptyset) - 1 + pl_{ij}\right)^2, \quad (35)$$

under constraints (23) and (24).

Example 3 For the same problem as in Example 1, assume that we now have three probabilistic pairwise classifiers, which deliver the following pignistic probabilities: $r^*_{12} = 0.843$, $r^*_{13} = 0.810$ and $r^*_{23} = 0.665$.

Additionally, assume that one-class classifiers have produced the following class plausibilities:

$$pl^\Omega(\{\omega_1\}) = 1, \quad pl^\Omega(\{\omega_2\}) = 0.444, \quad pl^\Omega(\{\omega_3\}) = 0.624.$$

Using (26), we obtain the following plausibilities for each pair of classes $\{\omega_i, \omega_j\}$:

$$pl_{12} = 1, \quad pl_{13} = 1, \quad pl_{23} = 0.791,$$

from which we deduce $r_{12} = r^*_{12}$, $r_{21} = 1 - r^*_{12}$, $r_{13} = r^*_{13}$, $r_{31} = 1 - r^*_{13}$, and

$$r_{23} = r^*_{23} pl_{23} = 0.526, \quad r_{32} = (1 - r^*_{23}) pl_{23} = 0.265.$$

The minimization of (35) yields the following bba on Ω:

$$m^\Omega(\{\omega_1\}) = 0.223,\ m^\Omega(\{\omega_1, \omega_2\}) = 0.464,\ m^\Omega(\{\omega_1, \omega_3\}) = 0.313,$$

all other masses being equal to zero. When conditioning m^Ω on each Ω_{ij} using the unnormalized rule of conditioning (8), and computing the corresponding pignistic probabilities, we get $BetP^*_{12}(\{\omega_1\}) = 0.768$, $BetP^*_{13}(\{\omega_1\}) = 0.843$, and $BetP^*_{23}(\{\omega_2\}) = 0.597$, which approximates the pairwise classifier outputs. □

4.4 Complexity Reduction

As the number of subsets of Ω grows exponentially with $K = |\Omega|$, problems with a large number of classes may quickly become intractable: for example, with $K = 26$ the number of focal sets may attain $2^{26} = 67108864$. We propose to reduce this complexity by limiting the number of focal sets of m^Ω.

This may be tackled by identifying the $L \leq K$ classes ω_i for which $pl^\Omega(\{\omega_i\})$ exceeds some threshold α, and treating the $K - L$ other classes as a single class. More formally, let $\omega_{(1)}, \ldots, \omega_{(K)}$ denote the classes ordered by decreasing plausibility, i.e.,

$$pl^\Omega(\{\omega_{(1)}\}) \geq \ldots \geq pl^\Omega(\{\omega_{(K)}\}). \tag{36}$$

Let us denote $\theta_i = \{\omega_{(i)}\}$, $i = 1, \ldots, L$, and $\theta_{L+1} = \{\omega_{(L+1)}, \ldots, \omega_{(K)}\}$. The set $\Theta = \{\theta_1, \ldots, \theta_{L+1}\}$ constitutes a *coarsening* [15] (i.e., a partition) of Ω. Since the classes in θ_{L+1} have low plausibility, they need not be discerned. We thus consider as possible focal sets of m^Ω the subsets of Ω of the form $\bigcup_{k \in I} \theta_k$ for some $I \subset \{1, \ldots, L+1\}$. The number of variables in the above optimization problems is thus reduced from 2^K to 2^{L+1}.

Example 4 Let $pl^\Omega(\{\omega_1\}) = 0$, $pl^\Omega(\{\omega_2\}) = 0.1$, $pl^\Omega(\{\omega_3\}) = 0.7$, $pl^\Omega(\{\omega_4\}) = 0.3$ and $pl^\Omega(\{\omega_5\}) = 0.05$ be the plausibilities computed when evaluating a test pattern $\mathbf{x}$,

and assume that $\alpha = 0.15$. We have $\theta_1 = \{\omega_3\}$, $\theta_2 = \{\omega_4\}$ and $\theta_3 = \{\omega_1, \omega_2, \omega_5\}$. We then have the following eight allowed focal sets: $\emptyset$, θ_1, θ_2, θ_3, $\theta_1 \cup \theta_2 = \{\omega_3, \omega_4\}$, $\theta_1 \cup \theta_3 = \{\omega_1, \omega_2, \omega_3, \omega_5\}$, $\theta_2 \cup \theta_3 = \{\omega_1, \omega_2, \omega_4, \omega_5\}$, and Ω. □

5 Experiments

5.1 Implementation of the methods

Five combination methods were compared experimentally: the method developped in this article (referred to as TBM), and the methods presented in Section 2.2: PCpl, PEst1, PEst2 and PCorr.

Two binary classification methods were used: *logistic regression* [9] , which computes a linear boundary between a pair of classes and provides conditional probability estimates, and the *evidential neural network* method[1] introduced in [3], which computes non-Bayesian conditional bbas m^*_{ij}. The method described in Section 4.2 was used to combine the evidential neural network outputs, whereas the variant described in Section 4.3 was used to combine the logistic regression outputs.

In the evidential neural network, three prototypes were used to characterize each class and the other parameters were set to their default value. The pignistic probabilities derived from the pairwise bbas m^*_{ij} were used for evaluating the PCpl, PEst1 and PEst2, and PCorr methods.

In the TBM method, the plausibilities $pl^\Omega(\{\omega_k\})$ were computed with *one-class support-vector machines* (1-SVM) [14]. A 1-SVM estimates the support of a class ω_k, and describes it with selected patterns of ω_k, called the support vectors. A signed distance $f_k(\mathbf{x})$ of a test pattern $\mathbf{x}$ to the support of ω_k may then be computed. A user-defined parameter ν corresponds to a lower bound on the fraction of support vectors and an upper bound on the fraction of outliers; here, ν was set to 0.2. The plausibility $pl^\Omega(\{\omega_k\})$ was computed by rescaling $f_k(\mathbf{x})$:

$$pl^\Omega(\{\omega_k\}) = \frac{f_k(\mathbf{x}) + \rho}{\rho}, \tag{37}$$

where ρ is a parameter obtained during the training of the SVM (see [14] for more details). In the PCorr method, the correcting probabilities q_{ij} were computed from kernel estimates

[1] Matlab code for this method is available at `http://www.hds.utc.fr/~tdenoeux/software.htm`.

g_i of the probability density of each class ω_i as:

$$q_{ij} = \frac{n_i g_i + n_j g_j}{\sum_{k=1\ldots K} n_k g_k}.$$

For both methods (1-SVM and kernel density estimation), estimates $\widehat{\sigma}_k$ of the kernel bandwidth for each class ω_k were computed using the method proposed in [10] and averaged. The kernel bandwidth for which class was then set to:

$$\widehat{\sigma}_{opt} = 1.5 \left(\frac{1}{K} \sum_{k=1}^{K} \widehat{\sigma}_k \right).$$

For the datasets with more than $K = 6$ classes, the complexity was reduced using the method described in Section 4.4. The threshold α was heuristically defined as:

$$\alpha = \frac{1}{2} \left(\min_{k \in \{1,\ldots,K\}} (u_{k,\,1/3}) \right),$$

with $u_{k,\,1/3}$ such that 1/3 of the training patterns in class ω_k satisfy $pl^{\Omega}(\{\omega_k\}) < u_{k,\,1/3}$.

5.2 Qualitative example

Figures 1 to 4 illustrate the results of our method applied to a three class, two-dimensional artificial problem. As it was not possible to show all the results here for lack of space, we focused on the results related to classes ω_1 and ω_2.

Figure 1 shows the basic ingredients of the method: the conditional probability estimates r_{12} computed using logistic regression, and the plausibilities pl_{12} computed using one-class SVM and equation (26). We can see that logistic regression estimates the location of the boundary between classes ω_1 and ω_2, whereas the 1-SVM method determines a region of feature space where these two classes are plausible. By combining these two pieces of information, we obtain the unnormalized conditional bba m_{12} shown in Figure 2. Note that we have $m_{12}(\{\omega_1, \omega_2\}) = 0$, as classifier E_{12} is probabilistic, and $m_{12}(\emptyset) = 1 - pl_{12}$. The masses $m^{\Omega}(\emptyset)$, $m^{\Omega}(\{\omega_1\})$, $m^{\Omega}(\{\omega_2\})$ and $m^{\Omega}(\{\omega_1\omega_2\})$ obtained by combining the outputs of the three pairwise classifiers and the three one-class classifiers using the method described in Section 4.3 are shown in Figure 3. As expected, the mass $m^{\Omega}(\{\omega_1\omega_2\})$ is maximum around the boundary of classes ω_1 and ω_2, whereas the mass $m^{\Omega}(\emptyset)$ is concentrated in regions of low data density. The corresponding pignistic probabilities $BetP^*(\omega_1)$ and $BetP^*(\omega_2)$ are shown in Figure 4.

5.3 Quantitative results

Table 1 presents the characteristics of the datasets used in these experiments. All these datasets were downloaded from the UCI Machine Learning database repository[2], except for the two-dimensional dataset Synth dataset (similar to the one used in Section 5.2, with an additional class), which was generated from a Gaussian mixture.

Tables 2 and 3 present the recognition rates obtained using logistic regression and evidential networks as pairwise classifiers, respectively. The significance of the differences between results was evaluated, by comparing the rates using a *Mc Nemar test* [4] at level 5%. For each dataset, the best result is underlined, and those that are not significantly lower are printed in bold.

Out of the six datasets, the TBM scheme yielded the best results for five datasets when using logistic regression as pairwise classifiers, and for four datasets when using evidential networks. The other results are not significantly worse than the best ones. Despite its simplicity, logistic regression yields good results, although the evidential neural networks performed generally slightly better.

Here, the relevance of pairwise classifiers was assessed using kernel methods. It is known that tuning the kernel bandwidth becomes harder as the dimension grows. However, the TBM scheme seems to be robust to the choice of the bandwidth. This may be due to the subadditivity of the plausibility measure, which gives the plausibilities pl_{ij} a minor role in the computation of the boundary, in the regions where ω_i and ω_j overlap.

6 Conclusion

Pairwise classifier combination is an interesting approach for solving complex multiclass problems using simple pairwise classifiers. In this paper, a technique based on the theory of belief function has been introduced. The proposed approach uses two kinds of classifiers: pairwise probabilistic or credal classifiers provide normalized conditional bbas m^*_{ij}, or corresponding pignistic probabilities, whereas one-class classifiers provide estimates of the plausibilities $pl^{\Omega}(\{\omega_k\})$ that the pattern belongs to each class ω_k. For each pattern to be classified, a bba on Ω is constructed such that its conditioning with respect to each

[2]Available at `http://www.ics.uci.edu/~mlearn`.

pair of classes ω_i and ω_j recovers the output conditional bba m^*_{ij}, or the associated pignistic probabilities. This method was shown to perform well as compared probabilistic approaches to the same problem on various datasets. It was also shown to provide easily interpretable results in the form of belief functions, which can be used for decision making or combined with other information sources.

A similar approach applied to the combination of one-against-all binary classifiers is presented in a companion paper [13]. The extension of this method to error-correcting output code classifier combination is under way.

References

[1] A. Appriou. Uncertain data aggregation in classification and tracking processes. In B. Bouchon-Meunier, editor, *Aggregation and Fusion of imperfect information*, pages 231–260. Physica-Verlag, Heidelberg, 1998.

[2] T. Denœux. A k-nearest neighbor classification rule based on Dempster-Shafer theory. *IEEE Trans. on Systems, Man and Cybernetics*, 25(05):804–813, 1995.

[3] T. Denœux. A neural network classifier based on Dempster-Shafer theory. *IEEE Trans. on Systems, Man and Cybernetics A*, 30(2):131–150, 2000.

[4] T. G. Dietterich. Approximate statistical tests for comparing supervised classification learning algorithms. *Neural Computation*, 10(7):1895–1923, 1998.

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

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

[7] J. Friedman. Another approach to polychotomous classification. Technical report, Department of Statistics and Stanford Linear Accelerator Center, Stanford University, 1996.

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

[9] T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning: Data Mining, Inference and Prediction*. Springer Verlag, New York, 2001.

[10] W. Koontz and K. Fukunaga. Asymptotic analysis of a nonparametric clustering technique. *IEEE Transactions on Computers*, C-21(9):967–974, 1972.

[11] M. Moreira and E. Mayoraz. Improved pairwise coupling classification with correcting classifiers. In *Tenth European Conference on Machine Learning (ECML'98)*, pages 160–171, Chemnitz, Germany, 1998.

[12] J. Platt, N. Cristianini, and J. Shawe-Taylor. Large margin DAGs for multiclass classification. In S. Solla, T. Leen, and K.-R. Mueller, editors, *Advances in Neural Information Processing Systems 12*, pages 547–553, 2000.

[13] B. Quost, T. Denœux, and M.-H. Masson. One-against-all classifier combination in the framework of belief functions. In *Proceedings of Information Processing and Management of Uncertainty in Knowledge-Based Systems (IPMU'2006)*, Vol. I, pages 356–363, Paris, France, July 2006.

[14] B. Schölkopf, J. Platt, J. Shawe-Taylor, A. Smola, and R. Williamson. Estimating the support of a high-dimensional distribution. Technical report, Microsoft Research, 1999.

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

[16] P. Smets and R. Kennes. The transferable belief model. *Artificial Intelligence*, 66:191–234, 1994.

[17] T.-F. Wu, C.-J. Lin, and R. C. Weng. Probability estimates for multi-class classification by pairwise coupling. *Journal of Machine Learning Research*, 5:975–1005, 2004.

Figures

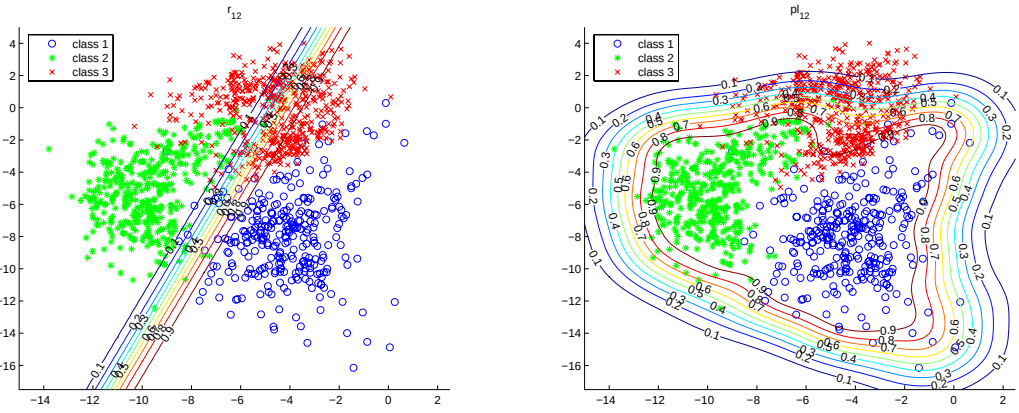

Figure 1: Conditional probablity estimate r_{12} obtained using logistic regression (left), and plausibility pl_{12} obtained using a one-class SVM (right).

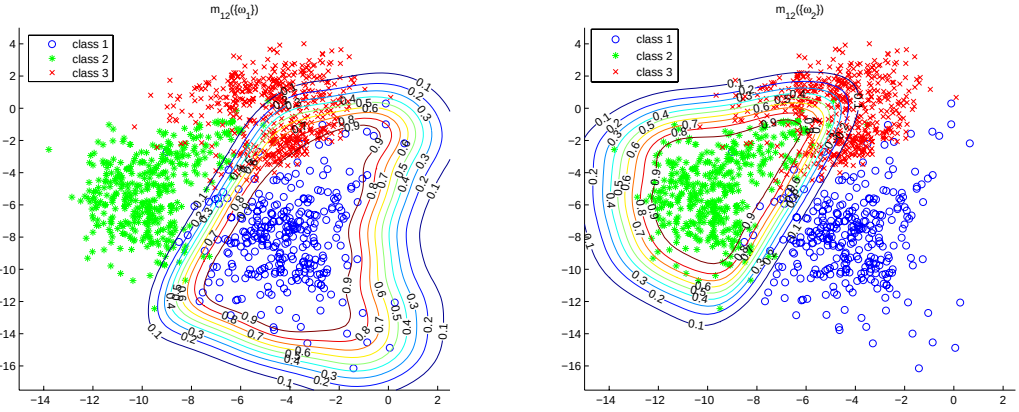

Figure 2: Unormalized conditional basic belief masses $m_{12}(\{\omega_1\})$ (left) and $m_{12}(\{\omega_2\})$ (right) obtained by combining r_{12} and pl_{12}.

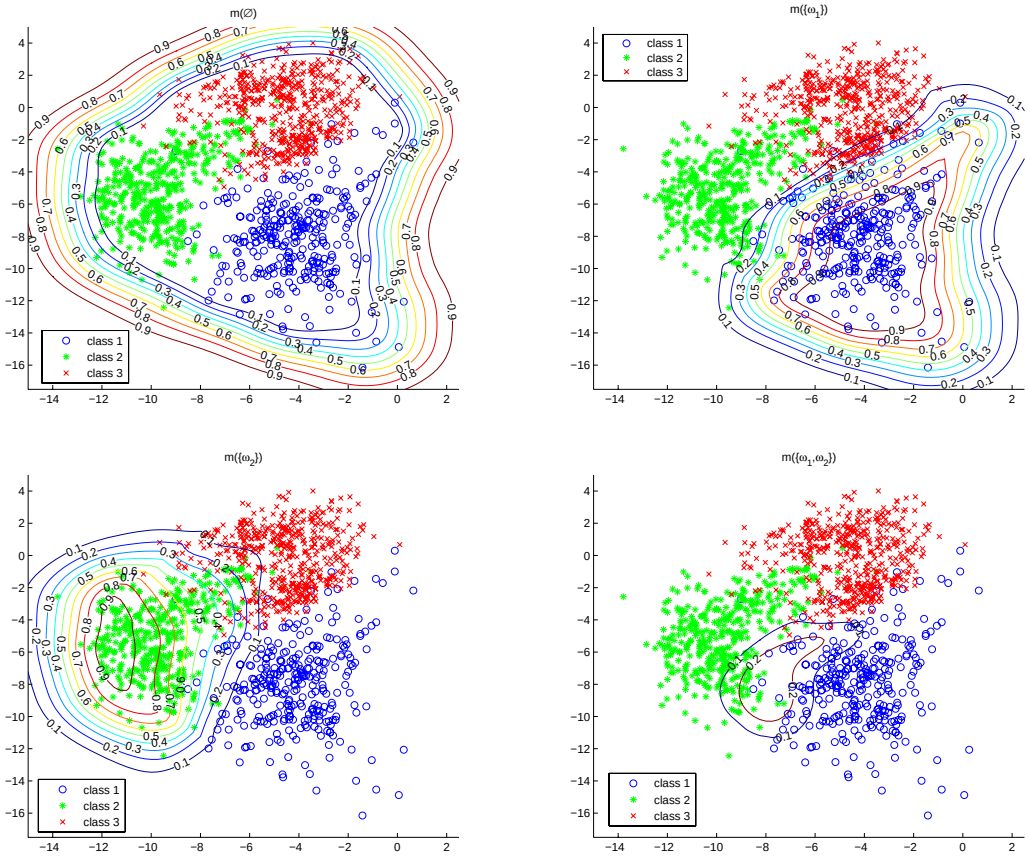

Figure 3: Basic belief masses $m^{\Omega}(\emptyset)$ (upper left), $m^{\Omega}(\{\omega_1\})$ (upper right), $m^{\Omega}(\{\omega_2\})$ (lower left) and $m^{\Omega}(\{\omega_1, \omega_2\})$ (lower right) obtained after combining all pairwise and one-class classifier outputs.

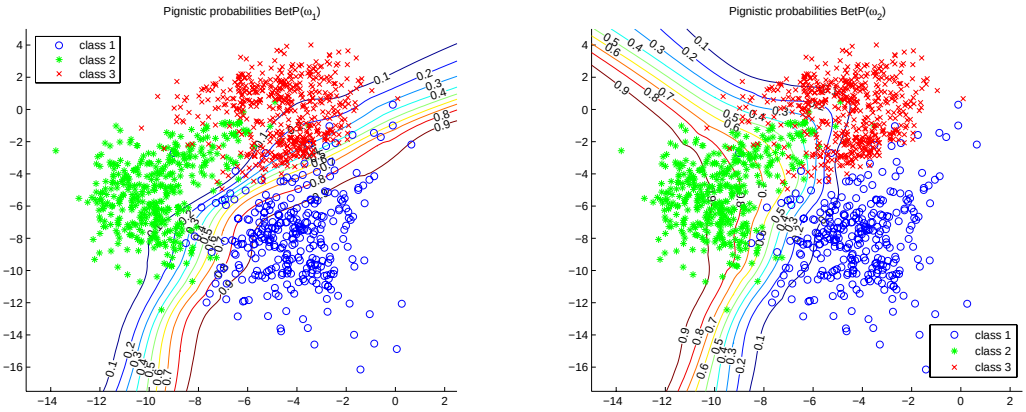

Figure 4: Pignistic probabilities $BetP^*(\omega_1)$ (left) and $BetP^*(\omega_2)$ (right) computed from m^{Ω}.

Tables

Table 1: Datasets.

dataset	dimension	nb. classes	nb. training patterns	nb. test patterns
Glass	9	6	139	75
Satimage	36	6	2573	3862
Segment	19	7	1400	910
Synth	2	4	1700	340
Vowel	10	11	528	462
Waveform	21	3	1500	3500

Table 2: Recognition rates (%), logistic regression.

Method	Glass	Satimage	Segment	Synth	Vowel	Waveform
TBM	**56.0**	**93.1**	**96.0**	**95.6**	**65.4**	**85.3**
PCpl	**58.7**	87.1	**96.0**	**94.4**	51.3	**85.1**
PEst1	**58.7**	86.9	**95.6**	**94.4**	50.9	**85.1**
PEst2	**60.0**	86.9	**95.6**	**94.4**	52.6	**85.1**
PCorr	**60.0**	90.8	90.3	**95.6**	60.6	**85.2**

Table 3: Recognition rates (%), evidential networks.

Method	Glass	Satimage	Segment	Synth	Vowel	Waveform
TBM	**58.7**	**93.2**	**89.7**	**96.2**	**66.7**	**85.9**
PCpl	**52.0**	85.5	84.7	**95.0**	64.1	**86.5**
PEst1	**56.0**	85.6	85.7	**95.9**	65.4	**86.3**
PEst2	**57.3**	86.0	86.2	**95.9**	**67.3**	**86.2**
PCorr	**62.7**	89.9	82.4	**95.9**	61.7	**85.6**