Pairwise Classifier Combination in the Transferable Belief Model

Benjamin Quost
UMR 6599 CNRS Heudiasyc
Université de Technologie
BP 20529
60205 Compiègne Cedex
France
bquost@hds.utc.fr

Thierry Denœux
UMR 6599 CNRS Heudiasyc
Université de Technologie
BP 20529
60205 Compiègne Cedex
France
tdenoeux@hds.utc.fr

Mylène Masson
UMR 6599 CNRS Heudiasyc
Université de Picardie Jules Verne
Chemin du Thil
80 025 Amiens
France
mmasson@hds.utc.fr

Abstract – *Classifier combination constitutes an interesting approach when solving multi-class classification problems. We propose to carry out this combination in the belief functions framework. Our approach, similar to a method proposed by Hastie and Tibshirani in a probabilistic framework, is first presented. The performances obtained on various datasets are then analyzed, showing a gain of classification accuracy using the belief functions approach.*

Keywords: Belief functions, Dempster-Shafer theory, Transferable Belief Model, Pattern Recognition, Classification.

1 Introduction

A generic pattern recognition task can be formalized as follows. Let a *training set* $\mathcal{T}$ be composed of a set $\mathcal{X} = \{\mathbf{x_1}, \ldots, \mathbf{x_n}\}$ of *patterns* $\mathbf{x_i} \in \mathbb{R}^p$. Each pattern $\mathbf{x}_i$ is associated with a *label* y_i, which represents its actual class $\omega_k \in \Omega = \{\omega_1, \ldots, \omega_K\}$.

A *classifier* can be trained to identify the relationships between the input space $\mathbb{R}^p$ and the label space Ω, on the basis of the training set; *generalizing* them to new – and unknown – data should then enable to predict the label of a test pattern $\mathbf{x}$. The architecture of a designed classifier has to fit the complexity of the problem: the more complex the situation to deal with, the more complex the classifier. Its *training cost* can therefore become arbitrarily large in terms of required time and training data.

In this article, we propose an approach to design classifiers of well suited complexity, in a rich and flexible knowledge-representation framework, which allows to represent various types of imprecision and uncertainty. This framework provides an adequate theoretical basis for classifier combination; moreover, it appears to increase the accuracy and robustness of the classification process.

We first present the problem of classifier combination, along with a short review of existing methods. We then describe the Transferable Belief Model, and more precisely the tools used in our method. Motivations for its choice are given, from which we derive and formalize our approach. Results obtained by various methods are then presented and analyzed. We eventually conclude on perspectives concerning future work.

2 Pairwise classifier combination

2.1 Motivations of classifier combination

The case of *polychotomous* classification of a test pattern $\mathbf{x}$ is considered here: the actual class of $\mathbf{x}$ has to be chosen among $K > 2$ different classes.

2.1.1 Direct multiclass classification

A single classifier can be taught to recognize all the classes: computing *decision boundaries* between them enables to evaluate the actual class of $\mathbf{x}$. According to the training cost of the classifier used, this *direct multiclass approach* can be burdensome.

Furthermore, some classifiers are best-suited to handle two-class problems. Another approach, involving classifier combination, has therefore been proposed to handle multiple classes.

2.1.2 Decomposition into simpler problems

This alternative approach consists in decomposing Ω into subsets; a classifier then learns to separate the classes of each subset. Pattern $\mathbf{x}$ is evaluated by each classifier, and assigned to a class according to a decision rule combining these evaluations. In the case of *binary decomposition*, *binary classifiers* separate two classes; the complexity of the problem, and its computational training cost, may then be reduced.

Classifier combination aiming at improving the performances of multiclass classifiers by taking advantage of their complementarity [1], will not be considered here.

2.2 Different decompositions of a multiclass problem

Dichotomous classification problems can be built by various ways. The *one-against-all* decomposition consists in opposing each class to all the others: K binary classifiers are trained from the whole set of training patterns. Alternatively, each class can be opposed to each other one (*one-against-one* or *pairwise* decomposition): $K(K-1)/2$ *pairwise classifiers* are trained from training patterns corresponding to two classes. Fewer classifiers are used in the former case, but the global training cost is lower in the latter [2].

The decomposition of Ω in pairs of classes can also be hierarchical [3]. A binary classifier learns to separate class ω_i from the set of classes $\Omega \setminus \{\omega_i\}$; another one, class ω_j ($\omega_j \neq \omega_i$) from the set $\Omega \setminus \{\omega_i, \omega_j\}$; and so on. A test pattern is successively evaluated by the classifiers, until its actual class is found. This evaluation must be here sequential, whereas it can be parallelized in the previous cases.

Error-Correcting Output Codes (ECOC) [4] provide another framework for binary decomposition. N *binary functions* $\mathrm{f}_1, \ldots, \mathrm{f}_N : \mathrm{f}_i : \mathbb{R}^p \rightarrow \{0,1\}, \forall i \in \{1, \ldots, N\}$, are defined. These functions are learnt, such that they take the same value for all the patterns of a same class: the concanetation of these values, for each class, defines then a N-bit long *codeword*. The codeword associated with $\mathbf{x}$ is computed by concatenating the $\mathrm{f}_i(\mathbf{x})$, and $\mathbf{x}$ is assigned to the class whose codeword is the nearest, according to some measure of similarity.

This article presents a way of combining pairwise classifiers.

2.3 Different ways of combining pairwise classifiers

Let the pairwise classifiers evaluate the actual class of a test pattern $\mathbf{x}$, among two classes ω_i and ω_j.

2.3.1 Voting rule [5]

Each classifier outputs a decision about the actual class of $\mathbf{x}$, for example a class label; $\mathbf{x}$ is assigned to the class receiving the largest number of votes.

2.3.2 Iterative pairwise coupling of posterior probabilities [6]

Let $p_i = \mathbb{P}(\omega_i|\mathbf{x})$ be the posterior probability of class ω_i, $\mu_{ij} = \mathbb{P}(\omega_i|\omega_i \text{ or } \omega_j, \mathbf{x})$ the conditional probability of class ω_i given $\{\omega_i, \omega_j\}$ ($\mu_{ij} = p_i/(p_i + p_j)$), n_{ij} the number of training patterns in classes $\{\omega_i, \omega_j\}$. The pairwise classifiers are assumed to provide estimates r_{ij} of μ_{ij}.

Trying to determine the probabilities p_i such that $\mu_{ij} = r_{ij}$, under the constraints $0 \leq p_i \leq 1$, $\sum p_i = 1$, is an overdetermined problem with $K-1$ unknown variables and $(K-1)K/2$ constraints; it has usually no exact solution. The p_i can then be estimated, such that the μ_{ij} are close to the r_{ij}.

It is proposed in [6] to use the negative weighted Kullback-Leibler divergence between the μ_{ij} and the r_{ij} (Equation (2)) to compute iteratively the estimates $\hat{p_i}$ of the p_i (Equation (1)). Let $\mathbf{p}$ and $\hat{\mathbf{p}}$ be the respective vectors of p_i and $\hat{p_i}$, and n_{ij} be the total number of patterns in classes ω_i and ω_j:

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

$$\mathcal{L}(\mathbf{p}) = \sum_{i<j} n_{ij} \left(r_{ij} \log \frac{r_{ij}}{\mu_{ij}} + (1 - r_{ij}) \log \frac{1 - r_{ij}}{1 - \mu_{ij}} \right). \tag{2}$$

Another non iterative method for computing estimates $\tilde{p_i}$ was also proposed:

$$\tilde{p_i} = \frac{2}{K} \frac{\sum_{j \neq i} r_{ij}}{(K-1)}. \tag{3}$$

Although crude estimates, the $\tilde{p_i}$ have the same ordering as the $\hat{p_i}$; they can be used as starting values in the iterative procedure, or for decision purposes.

2.3.3 Non-iterative pairwise coupling of posterior probabilities [7]

In [7], two non-iterative methods for estimating the p_i, such that the conditional probabilities μ_{ij} are close to the estimates r_{ij}, are proposed.

Considering that:

$$\begin{aligned} p_i &= \sum_{j:j \neq i} \left(\frac{p_i + p_j}{K-1} \right) \left(\frac{p_i}{p_i + p_j} \right), \quad \forall i \\ &= \sum_{j:j \neq i} \frac{p_i + p_j}{K-1} \mu_{ij}, \quad \forall i, \end{aligned}$$

it is proposed to estimate the p_i by solving:

$$p_i = \sum_{j:j \neq i} \frac{p_i + p_j}{K-1} r_{ij}, \quad \forall i \tag{4}$$

under constraints $\sum_{i=1}^{K} p_i = 1, p_i \geq 0, \forall i$. It can be shown that the unique solution to this problem is the same, with or without the positivity constraints. It can therefore be obtained by solving a simple linear system:

$$Q\,\mathbf{p} = \mathbf{p}, \tag{5}$$

$$\text{subject to: } \sum_{i=1}^{K} p_i = 1, \tag{6}$$

$$\text{where } Q(i,j) = \begin{cases} r_{ji}/(K-1) & \text{if } i \neq j \\ \sum_{s:s \neq i} r_{is}/(K-1) & \text{if } i = j \end{cases}.$$

Alternatively, Equation (4) may be rewritten as:

$$\sum_{j:j \neq i} r_{ji} p_i - \sum_{j:j \neq i} r_{ij} p_j = 0, \quad i = 1 \ldots K.$$

Motivated by this formulation, the authors propose another method to compute the p_i, by solving:

$$\mathbf{p}^* = \arg\min_{\mathbf{p}} \frac{1}{2} \sum_{i=1}^{K} \sum_{j:j \neq i} (r_{ji} p_i - r_{ij} p_j)^2,$$

under constraints $\sum_{i=1}^{K} p_i = 1, p_i \geq 0, \forall i$. It can be shown again that the positivity constraints are redundant; therefore, the problem may be rewritten as a quadratic convex problem with linearity constraints:

$$\min_{\mathbf{p}} \frac{1}{2}\,\mathbf{p}^T\,Q\,\mathbf{p},$$

$$\text{subject to: } \sum_{i=1}^{K} p_i = 1,$$

$$\text{where } Q(i,j) = \begin{cases} -r_{ji} r_{ij} & \text{if } i \neq j \\ \sum_{s:s \neq i} r_{si}^2 & \text{if } i = j \end{cases}.$$

Using the Karush-Kuhn-Tucker optimality conditions enables to find the solution by solving a linear system:

$$\begin{pmatrix} Q & \mathbf{e} \\ \mathbf{e}^T & 0 \end{pmatrix} \begin{pmatrix} \mathbf{p} \\ b \end{pmatrix} = \begin{pmatrix} \mathbf{0} \\ 1 \end{pmatrix}, \quad (7)$$

where $\mathbf{e}$ is a $(K \times 1)$-vector of ones, $\mathbf{0}$ is a $(K \times 1)$-vector of zeros, and b is the Lagrange multiplier associated with the equality constraint $\sum_{i=1}^{K} p_i = 1$.

2.3.4 Improved pairwise coupling classification with correcting classifiers [8]

The above methods propose to estimate the p_i, such that the μ_{ij} are close to the r_{ij}. However, as pointed out in [6], the actual class of $\mathbf{x}$ is unknown to most of the classifiers, which were trained to separate two classes only, and ignore all the others; their estimates r_{ij} of the μ_{ij} might hence be erroneous.

It is therefore proposed in [8] to train additional correcting classifiers, which evaluate whether $\mathbf{x}$ belongs either to ω_i or ω_j, or not: they separate $\{\omega_i, \omega_j\}$ from $\Omega \setminus \{\omega_i, \omega_j\}$. The probabilities $q_{ij} = \hat{\mathbb{P}}(\{\omega_i, \omega_j\}|\mathbf{x})$ are estimated using Equation (3), and the probabilities quantifying the knowledge of the actual class of $\mathbf{x}$ may be computed:

$$\hat{\mathbb{P}}(\{\omega_i\}|\mathbf{x}) = r_{ij}\, \hat{\mathbb{P}}(\{\omega_i, \omega_j\}|\mathbf{x}). \quad (8)$$

Although correcting the estimated conditional probabilities can give more robust probabilities estimates, this approach involves training $(K-1)K/2$ additional pairwise classifiers *on the whole training set*. It is therefore much more complex than both the one-versus-one and one-versus-all approaches.

3 The Transferable Belief Model

3.1 A flexible framework of knowledge representation

The need to manage various types of ignorance has led to define new knowledge representation frameworks. One of them, the *theory of evidence* [9] or *theory of belief functions*, has been declined into several approaches, among which the *Transferable Belief Model* (TBM) [10].

Stating the actual class of a test pattern $\mathbf{x}$ in a precise way may not always be possible; instead of compelling classifiers to give precise information, modelling the ignorance or imprecision of their statements is likely to improve the results of their combination. Hence the use of the TBM, which is particularly well-suited for representing, manipulating and combining imprecise knowledge.

3.2 Representing knowledge

3.2.1 Assessing the knowledge of the value taken by a variable

The TBM aims at modelling the knowledge of the actual value of a variable y, which belongs to a set of *atoms* $\Omega = \{\omega_1, \ldots, \omega_K\}$ refered to as the *frame of discernment* or *frame*. In a classification problem, y depicts the class of a test pattern $\mathbf{x}$.

3.2.2 Quantifying knowledge with belief functions

Knowledge is quantified by *basic belief assignments* (bba). A bba m satisfies:

$$\begin{cases} \sum_{A \subseteq \Omega} \mathrm{m}(A) = 1, \\ 0 \leq \mathrm{m}(A) \leq 1, \quad \forall A \subseteq \Omega. \end{cases}$$

The certainty that $y \in A$ is quantified by $\mathrm{m}(A)$; any subset $A \subseteq \Omega$ to which m gives belief is called a *focal element* of m. A bba m defined on a frame Ω is written m^{Ω}.

3.2.3 Particular belief functions

Categorical belief functions quantify total support given to a proposition $A \subseteq \Omega$:

$$\mathrm{m}^{\Omega}(A) \quad = \quad 1.$$

The *vacuous belief function*, defined by $\mathrm{m}^{\Omega}(\Omega) = 1$, is the categorical belief function expressing total lack of knowledge.

Bayesian belief functions quantify precise knowledge; the focal elements are atoms only:

$$\mathrm{m}^{\Omega}(A) \neq 0 \quad \Rightarrow \quad |A| = 1.$$

3.2.4 Exhaustiveness of the frame

The empty set $\emptyset$ can be given some belief. The mass $\mathrm{m}(\emptyset)$ can be interpreted as the belief that the actual value of y lies outside the frame (open-world assumption).

If the frame is considered to be exhaustive (closed-world assumption), the actual value of y has to be in Ω, and the belief functions are systematically normalized:

$$\begin{aligned} \mathrm{m}^*(A) &= \frac{\mathrm{m}(A)}{1 - \mathrm{m}(\emptyset)}, \quad \forall A \subseteq \Omega, A \neq \emptyset; \\ \mathrm{m}^*(\emptyset) &= 0. \end{aligned}$$

Clearly, m^* is not defined whenever $\mathrm{m}(\emptyset) = 1$. Another normalization procedure defined by Yager [11] consists in transfering $\mathrm{m}(\emptyset)$ to Ω:

$$\mathrm{m}'(A) = \begin{cases} \mathrm{m}(A) & \text{if } A \neq \emptyset, A \neq \Omega \\ \mathrm{m}(A) + \mathrm{m}(\emptyset) & \text{if } A = \Omega \\ 0 & \text{if } A = \emptyset \end{cases}. \quad (9)$$

3.3 Manipulating knowledge

3.3.1 Combining belief functions

Two belief functions m_1 and m_2 can be combined, using a suitable operator; the most common is the *conjunctive rule of combination* (CRC), symbolized by $\textcircled{\cap}$:

$$\mathrm{m}_1 \textcircled{\cap} \mathrm{m}_2(A) = \sum_{X \cap Y = A} \mathrm{m}_1(X)\, \mathrm{m}_2(Y), \forall A \subseteq \Omega.$$

3.3.2 *Conditioning belief functions*

Conditional belief functions quantify knowledge which are valid provided that an hypothesis is satisfied. Let m be a bba, $B \subseteq \Omega$ an hypothesis and m_B the categorical bba defined by $\mathrm{m}_B(B) = 1$; the conditioning $\mathrm{m}[B]$ of the bba m on B can be obtained by combining m with m_B:

$$\mathrm{m}[B] = \mathrm{m} \textcircled{\cap} \mathrm{m}_B \,.$$

Hence, any belief formerly assigned to $A \subseteq \Omega$ is transfered to $A \cap B$. The mass $\mathrm{m}[B](\emptyset)$ quantifies the belief given by m to hypotheses incompatible with B, or equivalently the belief that the actual value of y lies outside the new frame B.

3.4 Decision making

In the TBM framework, *pignistic probabilities* are computed from belief functions when a decision has to be made. The *pignistic transformation* [10] consists in equally redistributing the amount of belief, formerly given to a focal element $A \subseteq \Omega$, to the atoms $\omega_k \in A$, after a normalization step:

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

4 Combining pairwise classifiers within the TBM

4.1 Evaluating the actual class of $\mathbf{x}$ in Ω

Let a pattern $\mathbf{x}$ be observed, whose actual class is $\omega_k \in \Omega$; the information about the class of $\mathbf{x}$ may be quantified by a bba m^Ω. Conditioning m^Ω on *restricted frames* $\Omega_{ij} = \{\omega_i, \omega_j\}$ would give conditional bbas $\mathrm{m}[\Omega_{ij}]$, which quantify the knowledge of the actual class of $\mathbf{x}$ assuming that it is in Ω_{ij}.

The pairwise classifiers are assumed to estimate the actual class of $\mathbf{x}$ in the restricted frames Ω_{ij}. The outputs of these classifiers, that will be refered to as *pairwise estimates*, quantify partial knowledge of this class; they may be seen as estimates of the conditional bbas $\mathrm{m}[\Omega_{ij}]$.

An estimate of the bba m^Ω, quantifying the knowledge of the actual class of $\mathbf{x}$ in the general frame Ω, may then be obtained by combining the estimates provided by the pairwise classifiers, as will be detailed in Section 4.1.4.

4.1.1 *Estimating the actual class of $\mathbf{x}$ in various restricted frames Ω_{ij}*

Pairwise classifiers are trained on two classes only, and hence ignore the others; thus, the pairwise estimates are considered to be computed *under a closed-world assumption*: they can be seen as *normalized estimates of the conditionings* $\mathrm{m}[\Omega_{ij}]$. For all $A \subseteq \Omega_{ij}$, $A \neq \emptyset$:

$$\mathrm{m}^*_{ij}(A) = \frac{\hat{\mathrm{m}}[\Omega_{ij}](A)}{1 - \hat{\mathrm{m}}[\Omega_{ij}](\emptyset)}. \qquad (11)$$

If the pairwise classifiers compute probabilities, the pairwise estimates are then Bayesian belief functions:

$$\begin{cases} \mathrm{m}^*_{ij}(\{\omega_i\}) &= r_{ij} \\ \mathrm{m}^*_{ij}(\{\omega_j\}) &= 1 - r_{ij}. \end{cases}$$

Figure 1 shows the contour plot of posterior probabilities computed with logistic regression, when separating two classes drawn from a synthetic two-dimensional dataset of 4 classes.

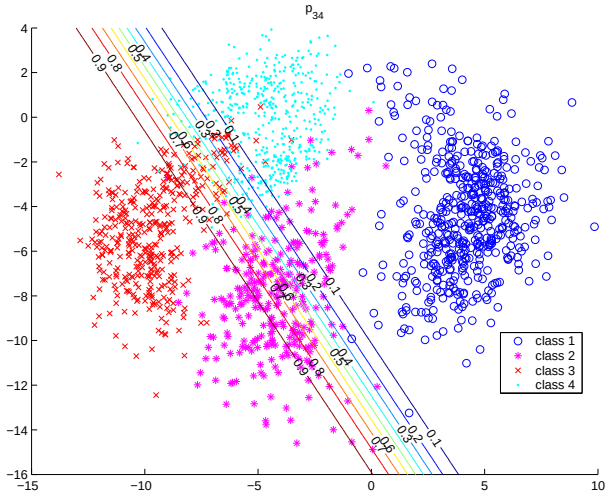

Fig. 1: Contour plot of the posterior probabilities r_{34}, obtained when separating class 3 (crosses) from class 4 (points).

4.1.2 *Estimating the validity of the pairwise estimates*

As pointed out in Section 2.3.4, a classifier compelled to assess the actual class of $\mathbf{x}$ can give erroneous information. Hence, we propose to assess the *validity* of such an assessment. A classifier, which evaluates a test pattern $\mathbf{x}$, provides valid information if the actual class of $\mathbf{x}$ may be one of the classes of its training set.

Let $\mathrm{f}_i(\mathbf{x})$ be the density of class ω_i at point $\mathbf{x}$; we propose to evaluate the possibility that $\mathbf{x}$ belongs to ω_i. A *possibility distribution* representing this knowledge may be built:

$$\widehat{\mathrm{f}}_i(\mathbf{x}) = \frac{\mathrm{f}_i(\mathbf{x})}{\max_{\mathbf{x}_k \in \omega_i} \mathrm{f}_i(\mathbf{x}_k)}. \qquad (12)$$

The possibility that $\mathbf{x}$ belongs to the frame Ω_{ij} can be obtained by combining $\widehat{\mathrm{f}}_i(\mathbf{x})$ and $\widehat{\mathrm{f}}_j(\mathbf{x})$ with the $\max$ *rule*:

$$\widehat{\mathrm{f}}_{ij}(\mathbf{x}) = \max\left(\widehat{\mathrm{f}}_i(\mathbf{x}), \widehat{\mathrm{f}}_j(\mathbf{x})\right). \qquad (13)$$

The belief that the actual class of $\mathbf{x}$ may lie outside the restricted frame Ω_{ij} is then estimated:

$$\mathrm{m}_{ij}(\emptyset) = \hat{\mathrm{m}}[\Omega_{ij}](\emptyset) = 1 - \widehat{\mathrm{f}}_{ij}(\mathbf{x}). \qquad (14)$$

Figure 2 shows the contour plots of the estimated possibilities $\widehat{\mathrm{f}}_3$, $\widehat{\mathrm{f}}_4$ to belong to the classes ω_3 and ω_4 of the synthetic dataset, respectively.

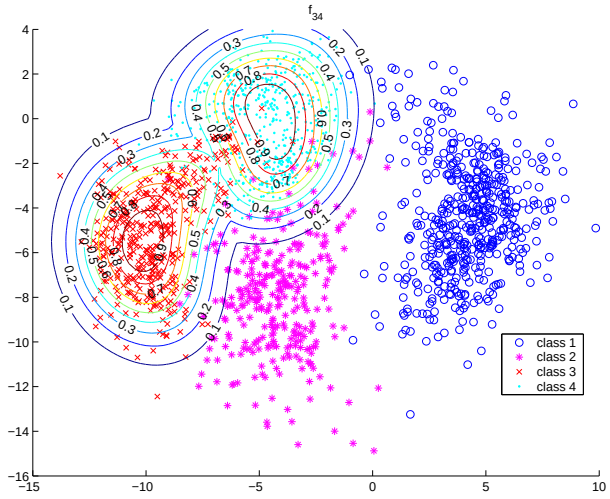

Fig. 2: Contour plot of the estimated possibilities to belong to the classes 3 and 4 of the training set.

4.1.3 *Processing the unnormalized pairwise estimates*

The knowledge of the validity of the pairwise classifiers may then be used in Equation (11) to process the *unnormalized pairwise estimates* m_{ij}. For all $A \subseteq \Omega_{ij}$, $A \neq \emptyset$:

$$\hat{\mathrm{m}}[\Omega_{ij}]^*(A) = \frac{\hat{\mathrm{m}}[\Omega_{ij}](A)}{1 - \hat{\mathrm{m}}[\Omega_{ij}](\emptyset)};$$

therefore, for all $A \subseteq \Omega_{ij}$, $A \neq \emptyset$:

$$\begin{aligned} \mathrm{m}_{ij}(A) &= \hat{\mathrm{m}}[\Omega_{ij}](A) \\ &= (1 - \hat{\mathrm{m}}[\Omega_{ij}](\emptyset))\, \hat{\mathrm{m}}[\Omega_{ij}]^*(A), \end{aligned}$$

which leads to:

$$\mathrm{m}_{ij}(\{\omega_i\}) = \widehat{\mathrm{f}}_{ij} r_{ij}, \quad (15)$$

$$\mathrm{m}_{ij}(\{\omega_j\}) = \widehat{\mathrm{f}}_{ij}(1 - r_{ij}), \quad (16)$$

$$\mathrm{m}_{ij}(\emptyset) = 1 - \widehat{\mathrm{f}}_{ij}. \quad (17)$$

4.1.4 *Retrieving original information from conditional information*

The bba m^Ω, quantifying the knowledge of the actual class of $\mathbf{x}$ in Ω, has to be retrieved from the unnormalized pairwise estimates. We propose, in a manner somewhat similar to that used in [6], to compute an estimate $\hat{\mathrm{m}}^\Omega$ of m^Ω, such that its conditionings $\hat{\mathrm{m}}[\Omega_{ij}]$ are as close as possible to the unnormalized pairwise estimates.

Let the vectors $\mathbf{m}$, $\mathbf{m_{ij}}$ ($\forall i, j$) correspond to $\hat{\mathrm{m}}$ and m_{ij} ($\forall i, j$), respectively; let their elements be put in binary order [12]. Let the $2^{|\Omega|} \times 2^{|\Omega|}$ matrix Γ_{ij} of conditioning on Ω_{ij} be defined by:

$$\mathbf{m}[\Omega_{ij}] = \Gamma_{ij} \cdot \mathbf{m}.$$

Let k and l be the indices corresponding to elements B and C, respectively. The coefficients $\Gamma_{ij}(k, l)$ are defined by:

$$\begin{aligned} \Gamma_{ij}(k,l) &= 1 \quad \text{if } C \cap \Omega_{ij} = B, \\ &= 0 \quad \text{otherwise.} \end{aligned}$$

We propose to retrieve $\hat{\mathbf{m}}^\Omega$ by solving:

$$\hat{\mathbf{m}}^\Omega = \arg\min_{\mathbf{m}} \sum_{j>i} \|\Gamma_{ij} \cdot \mathbf{m} - \mathbf{m_{ij}}\|^2, \quad (18)$$

$$\text{subject to: } \begin{cases} \sum_{A \subseteq \Omega} \mathrm{m}(A) = 1, \\ 0 \leq \mathrm{m}(A) \leq 1, \quad \forall A \subseteq \Omega; \end{cases} \quad (19)$$

where $\|\cdot\|$ denotes the euclidean norm.

As proposed in [6], the unnormalized pairwise estimates may be weighted according to the number n_{ij} of patterns in classes ω_i and ω_j; Equation (18) then becomes:

$$\hat{\mathbf{m}}^\Omega = \arg\min_{\mathbf{m}^\Omega} \sum_{j>i} n_{ij} \|\Gamma_{ij} \cdot \mathbf{m} - \mathbf{m_{ij}}\|^2. \quad (20)$$

Figures 3 and 4 show the pignistic probabilities of observing respectively classes 3 and 4, computed from the combined bbas. The masses were normalized using Yager's procedure (Equation (9)).

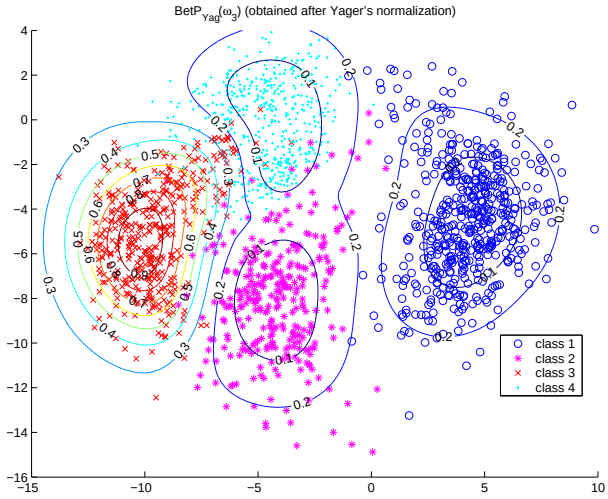

Fig. 3: Contour plot of the pignistic probabilities of observing class 3.

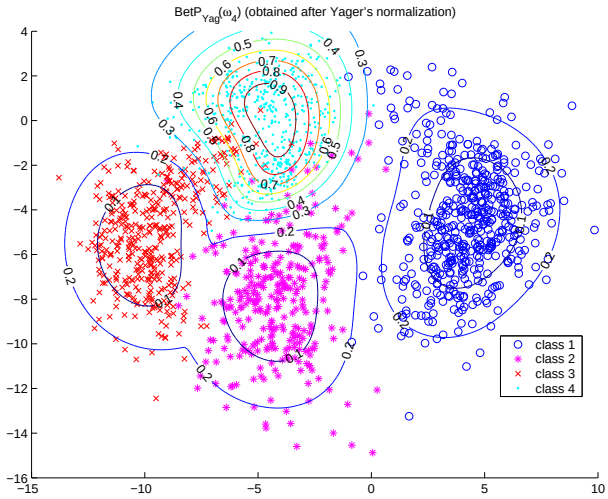

Fig. 4: Contour plot of the pignistic probabilities of observing class 4.

4.2 Reducing the complexity of the method

The number of subsets of Ω grows exponentially with K. Problems with high K are hence intractable: for example, the bbas computed when solving a letter recognition problem with 26 classes may have $2^{26} = 67108864$ focal elements. We propose to reduce this complexity by restricting the set of possible focal elements when computing the bba m^{Ω}.

4.2.1 Cardinal restriction of the set of possible elements

Analyzing the estimates $\hat{\mathrm{m}}^{\Omega}$ leads to notice that subsets of high cardinality are usually given few belief. This is not particularly surprising: the pairwise estimates m_{ij}, whose combination gives $\hat{\mathrm{m}}^{\Omega}$, have $\{\omega_i\}$, $\{\omega_j\}$ and $\emptyset$ as focal elements.

The size of the set of focal elements may therefore be restricted, by removing the focal elements, whose cardinality exceeds a threshold, from the set of possible elements of the bba m^{Ω}.

4.2.2 Coarsened restriction of the set of possible elements

The membership of $\mathbf{x}$ to the supports of the classes of Ω (Equations (12,13)) allows to compute the validity of the classifiers in the frames Ω_{ij}. It can also be used to *restrict the frame* in which the membership of $\mathbf{x}$ is evaluated.

A new frame Θ may be obtained, by aggregating the classes, to which $\mathbf{x}$ most likely does not belong, into a single one. The frame Θ is by definition a *coarsening* of Ω: each $\theta_l \in \Theta$ corresponds to a subset $A \subseteq \Omega$. The new set 2^{Θ} of possible focal elements in Θ is then of cardinality 2^L, with $L = |\Theta|$.

This coarsening can be used to select the focal elements to which some belief could be given, and therefore to restrict the set of possible elements of m^{Ω} to these focal elements.

5 Experiments

5.1 Procedure

5.1.1 Classification methods and parameterization

Five combination methods were compared:

- the evidential method developped in this article, with and without weighting (method TBMW, Equation (20), and method TBM, Equation (18), respectively);
- the method presented in Section 2.3.2, with and without weighting (methods PCPLW and PCPL, respectively);
- the methods presented in Section 2.3.3 (methods PEST1 and PEST2, respectively);
- the method presented in Section 2.3.4 (PCORR).

The same conditional posterior probabilities estimates were used to compare the methods. The performance criterion was the *recognition rate*.

The posterior probability estimates were processed using a *logistic regression* method [13]. The densities of the classes (methods TBM and TBMW) were evaluated with Parzen windows; kernel bandwidths were learnt by computing the bandwidths with a method proposed by Koontz and Fukunaga [14], averaging them and multiplying this average by a factor. The correcting probabilities (method PCORR) were computed by combining the estimates of the densities with the Bayes' rule.

5.1.2 Datasets description

Table 1 presents the features (dimension, number of classes, total number of patterns for training and testing) of the datasets with which the methods were tested. The Satimage dataset was used partially: training and test sets were chosen randomly, the training set in the same proportions as in the original dataset. The Glass, Vowel and Satimage datasets are used with courtesy of the UCI Machine Learning database repository (`http://www.ics.uci.edu/~mlearn/`).

Table 1: Datasets features

dataset	dim.	nb. cl.	nb. pat. / train.	nb. pat. / test
Iris	4	3	90	60
Glass	9	6	139	75
Satimage	36	6	2850	1200
Vowel	10	11	528	462

In the Iris dataset, the training set numbers 30 patterns per class, the test set 20 patterns per class. In the Vowel dataset, the training set numbers 48 patterns per class, the test set 42 patterns per class. The composition of the training and test sets, for the Glass and the Satimage datasets, are presented in Tables 2 and 3, respectively.

Table 2: Glass dataset – composition of the training and test sets (number of patterns)

	cl. 1	cl. 2	cl. 3	cl. 4	cl. 5	cl. 6
training set	46	49	11	8	6	19
test set	24	27	6	5	3	10

It can be seen in Table 2 that the Glass dataset contains few examples; the classifiers must therefore be trained with very few patterns. The classes of the Vowel dataset are intrinsically hard to discriminate; moreover, patterns come from different sources, depending on whether they are used for training or testing the classifiers. These datasets may therefore be difficult to process.

The complexity of the method was reduced: the set of possible focal elements was restricted to elements composed of 4 atoms at most.

Table 3: Satimage dataset – composition of the training and test sets (number of patterns)

	cl. 1	cl. 2	cl. 3	cl. 4	cl. 5	cl. 6
training set	700	300	600	250	300	700
test set	200	200	200	200	200	200

5.2 Results and interpretations

5.2.1 Results

Table 4 summarizes the recognition rates performed by the methods tested. The significance of the differences between the methods TBM and TBMW and each of the others was evaluated, by comparing the rates using a *Mc Nemar test* [15] at level 5%. Significantly better results are printed in bold in Table 4.

Table 4: Recognition rates (%)

Method	Iris	Glass	Satimage	Vowel
TBM	96.7	62.7	**87.1**	**65.2**
TBMW	96.7	65.3	**87.0**	**65.2**
PCPL	96.7	58.7	80.2	51.3
PCPLW	96.7	60	80.2	51.3
PEST1	96.7	58.7	80.8	50.9
PEST2	96.7	60	80.6	52.6
PCORR	96.7	60	**85.9**	60.6

5.2.2 Importance of the validity of the classifiers

Results show that methods TBM, TBMW and PCORR outperform the others for the Glass, Satimage, and Vowel datasets. The significance of the results obtained with methods TBM and TBMW can be assessed for the Satimage, and Vowel datasets; the method PCORR also outperforms the methods PCPL, PCPLW, PEST1 and PEST2 for the Satimage dataset.

These results highlight the importance of assessing the validity of the classifiers to be combined. However, the way of assessing the validity of a classifier, as well as combining this information with those provided by this classifier, seems to have an influence on the results. Indeed, the methods TBM and TBMW produce significantly better results than the method PCORR for the Vowel dataset, whereas this difference cannot be assessed for the Satimage dataset.

This difference may be related to the way of evaluating the validity of the pairwise classifiers. In the case of the method PCORR, correcting probabilities leads to define mutually exclusive domains of validity; whereas in the case of the methods TBM and TBMW, the domains of relevance of various classifiers may overlap, these domains being evaluated by taking into account examples of the corresponding classes only.

5.2.3 Weighting the conditional information according to the size of the classes

Out the four datasets on which the combination methods were tested, two of them (Glass and Satimage) have classes with different sizes.

The ratios between these sizes are much more important in the case of the Glass dataset than in the case of the Satimage dataset; moreover, the training set of the former is much smaller than that of the latter.

Although the recognition rates obtained for the methods involving weighting are higher for the Glass dataset, this difference is not significant at the 5% level (obviously due to the low number of test patterns); as for the Satimage dataset, no difference between weighted and non-weighted methods could be observed.

6 Conclusion and prospects

In this article, several methods for combining pairwise classifiers and computing estimates of posterior probabilities were reviewed. The Transferable Belief Model appears to be well suited to formalize pairwise classifier combination, and particularly the underlying problems such as assessing the validity of the classifiers to combine. The classification results obtained definitely assess the relevance of this framework to combine classifiers.

The flexibility of the TBM enables to adapt this combination method to a wide range of problems. Posterior conditional probabilities may be combined without estimating the validity of the classifiers (the results obtained are similar to those of the method PCPL). Classifiers computing estimates of posterior conditional belief functions in restricted frames might also be combined, whatever the number of classes in each restricted frame.

There are many directions of future research. First, further work has to be done to assess the validity of a classifier, in particular evaluating this validity without computing the density of the classes. The *supports* of the distributions underlying the classes could be estimated; *one-class SVMs* [16] seem to provide powerful tools for such processings.

Reducing the complexity of the method by determining a coarsening Θ of the original frame Ω also seems to be a very promising approach, which needs to be further investigated. New cost functions for computing a belief function, whose conditionings are close to the pairwise estimates, may be determined, and their impact on the accuracy of the result may be studied. A non-iterative combination method might then be deduced from this new iterative method, in a same manner as that proposed in [7].

References

[1] Ludmila Kuncheva. *Combining Pattern Classifiers: Methods and Algorithms*. Wiley, Chichester, 2004.

[2] Johannes Fürnkranz. Round robin rule learning. In C. E. Brodley and A. P. Danyluk, editors, *Proceedings of the 18th International Conference on Machine Learning (ICML-01)*, pages 146–153, Williamstown, MA, 2001. Morgan Kaufmann Publishers.

[3] John Platt, Nello Cristianini, and John Shawe-Taylor. Large margin dags for multiclass classification.

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

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

[6] Trevor Hastie and Robert 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.

[7] Ting-Fan Wu, Chih-Jen Lin, and Ruby C. Weng. Probability estimates for multi-class classification by pairwise coupling. *Journal of Machine Learning Research*, 5:975–1005, 2004.

[8] Miguel Moreira and Eddy Mayoraz. Improved pairwise coupling classification with correcting classifiers. In *European Conference on Machine Learning*, pages 160–171, 1998.

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

[10] Philippe Smets and Robert Kennes. The transferable belief model. *Artificial Intelligence*, 66:191–234, 1994.

[11] Ronald Yager. On the dempster-shafer framework and new combination rules. *Information Sciences*, 41:93–137, 1987.

[12] Philippe Smets. The application of the matrix calculus to belief functions. *International Journal of Approximate Reasoning*, 31:1–30, 2002.

[13] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning: Data Mining, Inference and Prediction*. Springer Verlag, New York, 2001.

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

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

[16] Bernhard Schölkopf and Alexander Smola. *Learning with Kernels*. MIT Press, 2002.