One-against-all Classifier Combination in the Framework of Belief Functions

Benjamin Quost
laboratoire HeuDiaSyC
UTC, Compiègne
bquost@hds.utc.fr

Thierry Denœux
laboratoire HeuDiaSyC
UTC, Compiègne
tdenoeux@hds.utc.fr

Mylène Masson
laboratoire HeuDiaSyC
UPJV, Amiens
mmasson@hds.utc.fr

Abstract

Classifier combination is an interesting approach for solving multi-class classification problems. We study here the combination of one-against-all binary classifiers, in the framework of belief functions. Our approach is first formalized; its performances are then compared to that of two other methods, on various datasets. We conclude with perspectives on future work.

Keywords: Classifier combination, belief functions, coarsening.

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 address the problem of designing a complex classifier by combining binary classifiers (trained to separate two sets of patterns). We propose to carry on this combination in the framework of the theory of belief functions. First we recall some material about classifier combination ; then we present the Transferable Belief Model, and more particularly the concept of *coarsening*. Then we present and formalize our approach. Results on synthetic and real datasets are provided and discussed, before we conclude on perspectives left opened by this work.

2 Binary Classifier Combination

We address here the case of multiclass classifier combination, where a test pattern $\mathbf{x}$ has to be assigned to a single class among $K > 2$ different ones.

A natural approach is to train a single classifier to separate the set of classes; however, the training cost of such a classifier can be burdensome. Moreover, some classifiers are naturally designed to handle binary classification problems. Thus, a multiclass classification task can be handled by decomposing it into binary subproblems, then solved with binary classifiers, and combining the results.

2.1 Different decompositions of a multiclass problem

Various decomposition-combination schemes can be used. In the *One-versus-one (OVO) decomposition* ([5, 6]), each class is opposed to each other, thus giving C_K^2 subproblems;

in the *One-versus-all (OVA) decomposition*, each class is opposed to all the others, thus giving K subproblems. Both decomposition schemes can be seen as instances of *Error-Correcting Output Codes* (ECOC) ([3, 1]).

We studied OVO-classifier combination in the framework of belief functions in [8]. We address in this article the combination of OVA classifiers; ECOC-combination of binary classifiers is beyond the scope of this paper.

2.2 Combination of OVA classifiers

Let $\mathcal{E}_k$ be the classifier trained to separate class ω_k from the others; its output $f_k(\mathbf{x})$, provided when evaluating test pattern $\mathbf{x}$, can be a posterior probability, a belief function, or simply a score. In the latter case, the voting rule is generally applied, and $\mathbf{x}$ is assigned to the class with the greatest score. A method for estimating the posterior probabilities $\Pr(\omega_k|f)$ by combining probabilistic classifiers is proposed in [7] (in the ECOC framework).

In this article, we present a method for combining binary classifiers in the Transferable Belief Model. The outputs of the classifiers are interpretated as belief functions: thus, probabilistic as well as possibilistic or credal classifiers may be combined by this approach.

3 The Transferable Belief Model

The Transferable Belief Model (TBM) [10] is an interpretation of the Dempster-Shafer theory of belief functions; in this formalism, belief functions quantify weighted opinions, irrespective of any underlying probability distributions. Thus, it is particularly well-suited to represent and manipulate partial or subjective knowledge.

3.1 Basic concepts

The knowledge of the actual class $\omega_0 \in \Omega$ of a pattern $\mathbf{x}$ can be quantified by a basic belief assignment (bba) $m^\Omega : 2^\Omega \to [0;1]$, verifying:

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

A subset $A \subseteq \Omega$ is a focal element of m^Ω if $m^\Omega(A) > 1$. In the TBM, the bbas need not be normalized: $m^\Omega(\emptyset)$ can be strictly positive.

Once a decision regarding the actual class of $\mathbf{x}$ has to be taken, pignistic probabilities can be processed from the bba m^Ω: for all $\omega_k \in \Omega$,

$$BetP(\omega_k) = \frac{1}{1 - m(\emptyset)} \sum_{A \subseteq \Omega : \omega_k \in A} \frac{m(A)}{|A|}.$$

3.2 Coarsenings, refinements

Let Θ be a partition of Ω; let a mapping $\rho : 2^\Theta \to 2^\Omega$ verify the following properties:

1. the set $\{\rho(\{\theta\}), \theta \in \Theta\} \subseteq 2^\Omega$ is a partition of Ω ;
2. for each $A \subseteq \Theta$, $\rho(A) = \bigcup_{\theta \in A} \rho(\{\theta\})$;

then, ρ is called a *refining* of Θ to Ω; by extension, Ω is called a *refinement* of Θ, and Θ a *coarsening* of Ω.

A refining is generally not onto: there may exist subsets $B \subseteq \Omega$ that are not images by ρ of any $A \subseteq \Theta$. Then, B could be associated with the largest element $A_1 \subseteq \Theta$ whose image by ρ is included in B. Formally,

$$A_1 = \bigcup_l \theta_l \in \Theta : \rho(\theta_l) \subseteq B.$$

The element A_1 is defined as the *inner reduction* of B on Θ, written $A_1 = \underline{\theta}(B)$. The inner reduction of B on Ω is $B_1 = \rho(A_1) = \rho(\underline{\theta}(B))$.

Alternatively, B could be associated with the smallest element $A_2 \subseteq \Theta$ whose image by ρ includes B. Formally,

$$A_2 = \bigcup_l \theta_l \in \Theta : \rho(\theta_l) \cap B \neq \emptyset.$$

The element A_2 is defined as the *outer reduction* of B, written $A_2 = \overline{\theta}(B)$. The outer reduction of B on Ω is $B_2 = \rho(A_2) = \rho(\overline{\theta}(B))$.

Example 1 Let $\Omega = \{\omega_1, \omega_2, \omega_3, \omega_4, \omega_5\}$; let $\Theta = \{\theta_1, \theta_2\}$ be a coarsening of Ω defined by $\rho(\{\theta_1\}) = \{\omega_1\}$, $\rho(\{\theta_2\}) = \{\omega_2, \omega_3, \omega_4, \omega_5\}$.

Tables 1 and 2 show the inner and outer reductions of some $A \subseteq \Omega$, on Θ and on Ω.

Table 1: Inner reductions of some $A \subseteq \Omega$

$A \subseteq \Omega$	$\{\omega_1\}$	$\{\omega_1, \omega_3\}$	$\{\omega_2, \omega_4\}$
$\underline{\theta}(A)$	$\{\theta_1\}$	$\{\theta_1\}$	$\emptyset$
$\rho(\underline{\theta}(A))$	$\{\omega_1\}$	$\{\omega_1\}$	$\emptyset$
$A \subseteq \Omega$	$\emptyset$	$\{\omega_2, \dots, \omega_5\}$	Ω
$\underline{\theta}(A)$	$\emptyset$	$\{\theta_2\}$	Θ
$\rho(\underline{\theta}(A))$	$\emptyset$	$\{\omega_2, \dots, \omega_5\}$	Ω

Table 2: Outer reductions of some $A \subseteq \Omega$

$A \subseteq \Omega$	$\{\omega_1\}$	$\{\omega_1, \omega_3\}$	$\{\omega_2, \omega_4\}$
$\overline{\theta}(A)$	$\{\theta_1\}$	Θ	$\{\theta_2\}$
$\rho(\overline{\theta}(A))$	$\{\omega_1\}$	Ω	$\{\omega_2, \dots, \omega_5\}$
$A \subseteq \Omega$	$\emptyset$	$\{\omega_2, \dots, \omega_5\}$	Ω
$\overline{\theta}(A)$	$\emptyset$	$\{\theta_2\}$	Θ
$\rho(\overline{\theta}(A))$	$\emptyset$	$\{\omega_2, \dots, \omega_5\}$	Ω

These definitions can easily be extended to bbas: the inner and outer reductions of a bba m^Ω on Θ, written respectively $\underline{m}^\Theta$ and $\overline{m}^\Theta$, can be defined by:

$$\underline{m}^\Theta(A) = \sum_{B \subseteq \Omega, \underline{\theta}(B) = A} m^\Omega(B), \quad \forall A \subseteq \Theta;$$

$$\overline{m}^\Theta(A) = \sum_{B \subseteq \Omega, \overline{\theta}(B) = A} m^\Omega(B), \quad \forall A \subseteq \Theta.$$

Thus, although a bba m_1^Θ can be exactly *extended* on Ω, *reducing* a bba m_2^Ω on Θ will generally imply a loss of information. The extension by ρ of a bba m^Θ onto Ω corresponds to the *vacuous extension* $m^{\Theta\uparrow\Omega}$ ([9]): for all $B \subseteq \Omega$,

$$m^{\Theta\uparrow\Omega}(B) = \begin{cases} m^\Theta(A) & \text{if } B = \rho(A),\ A \subseteq \Theta \\ 0 & \text{otherwise.} \end{cases} \tag{1}$$

For convenience, the vacuous extension $\rho(m^\Theta)$ of a bba m^Θ onto Ω will be written m^Ω.

Example 2 Let Ω and Θ be defined as in Example 1. Let m^Ω be defined by:

$m^\Omega(\emptyset) = 0.1$ $\quad m^\Omega(\{\omega_1\}) = 0.3$
$m^\Omega(\{\omega_1, \omega_3\}) = 0.2$ $\quad m^\Omega(\{\omega_2, \omega_4\}) = 0.1$
$m^\Omega(\{\omega_2, \dots, \omega_5\}) = 0.2$ $\quad m^\Omega(\Omega) = 0.1$

Table 3 shows the inner and outer reductions of m^Ω on Θ. The inner and outer reductions of m^Ω on Ω are directly obtained by replacing any $A \subseteq \Theta$ by $B \subseteq \Omega : B = \rho(A)$.

Table 3: Inner and outer reductions of m^Ω

$A \subseteq \Theta$	$\emptyset$	$\{\theta_1\}$	$\{\theta_2\}$	Θ
$\underline{m}^\Theta(A)$	0.2	0.5	0.2	0.1
$\overline{m}^\Theta(A)$	0.1	0.3	0.3	0.3

4 OVA classifier combination using belief functions

4.1 Principle

Let $\mathcal{E}_k$ be a classifier trained to separate class ω_k from the set of remaining classes $\Omega \setminus \{\omega_k\}$. The output provided by $\mathcal{E}_k$ when evaluating a test pattern $\mathbf{x}$ can be expressed as a bba $m_k^{\Theta_k}$. The frame $\Theta_k = \{\theta_{k1}, \theta_{k2}\}$ is obviously a coarsening of Ω, defined by $\rho(\{\theta_{k1}\}) = \{\omega_k\}$, $\rho(\{\theta_{k2}\}) = \Omega \setminus \{\omega_k\}$.

Let the knowledge of the actual class of $\mathbf{x}$ be quantified by a bba m^Ω. Let an approximation of m^Ω be a bba obtained by reducing m^Ω on a frame Θ and extending the result back to Ω. Each $m_k^{\Theta_k}$ can be seen as an estimate of a reduction of m^Ω on Θ_k, and its extension m_k^Ω as an estimate of the corresponding approximation of m^Ω.

We propose to combine the outputs of the $\mathcal{E}_k$ by computing an estimate $\widehat{m}^\Omega$ of m^Ω that is consistent with the m_k^Ω.

4.2 Formalization

When computing the inner reduction $\underline{m}^\Omega$:

- the bbm given by m^Ω to any A such that $\omega_k \notin A$ and $\overline{\omega_k} \not\subseteq A$, is transfered to $\emptyset$: $\underline{m}^\Omega(\emptyset) = \sum_{A \subset \overline{\omega_k}} m^\Omega(A)$;
- the bbm given by m^Ω to any A such that $\omega_k \in A$ and $\overline{\omega_k} \not\subseteq A$, is transfered to ω_k: $\underline{m}^\Omega(\{\omega_k\}) = \sum_{A \subset \Omega, \{\omega_k\} \in A} m^\Omega(A)$;
- the bbm given by m^Ω to any A such that $\omega_k \notin A$ and $\overline{\omega_k} \subseteq A$, is transfered to $\overline{\omega_k}$: $\underline{m}^\Omega(\overline{\{\omega_k\}}) = m^\Omega(\overline{\{\omega_k\}})$;

- the bbm given by m^Ω to any A such that $\omega_k \in A$ and $\overline{\omega_k} \subseteq A$, is transfered to Ω: $\underline{m}^\Omega(\Omega) = m^\Omega(\Omega)$.

Similarly, when computing $\overline{m}^\Omega$:

- we transfer to $\emptyset$ the bbm given by m^Ω to any A such that $\omega_k \notin A$ and $\overline{\omega_k} \cap A = \emptyset$: $\overline{m}^\Omega(\emptyset) = m^\Omega(\emptyset)$;
- we transfer to ω_k the bbm given by m^Ω to any A such that $\omega_k \in A$ and $\overline{\omega_k} \cap A = \emptyset$: $\overline{m}^\Omega(\{\omega_k\}) = m^\Omega(\{\omega_k\})$;
- we transfer to $\overline{\omega_k}$ the bbm given by m^Ω to any A such that $\omega_k \notin A$ and $\overline{\omega_k} \cap A \neq \emptyset$: $\overline{m}^\Omega(\overline{\{\omega_k\}}) = \sum_{A \subseteq \overline{\{\omega_k\}}} m^\Omega(A)$;
- we transfer to Ω the bbm given by m^Ω to any A such that $\omega_k \in A$ and $\overline{\omega_k} \cap A \neq \emptyset$: $\overline{m}^\Omega(\Omega) = \sum_{\{\omega_k\} \in A, \overline{\{\omega_k\}} \cap A \neq \emptyset} m^\Omega(A)$.

Hence, it can be checked that the bbas m^Ω, $\underline{m}^\Omega$ and $\underline{m}^\Omega$ verify the following inequalities:

$$\begin{cases} \overline{m}^\Omega(\emptyset) & \leq & m^\Omega(\emptyset) & \leq & \underline{m}^\Omega(\emptyset) \\ \overline{m}^\Omega(\{\omega_k\}) & \leq & m^\Omega(\{\omega_k\}) & \leq & \underline{m}^\Omega(\{\omega_k\}) \\ \underline{m}^\Omega(\overline{\{\omega_k\}}) & \leq & m^\Omega(\overline{\{\omega_k\}}) & \leq & \overline{m}^\Omega(\overline{\{\omega_k\}}) \\ \underline{m}^\Omega(\Omega) & \leq & m^\Omega(\Omega) & \leq & \overline{m}^\Omega(\Omega) \end{cases}$$

It seems reasonable to require that approximations of m^Ω (and hence the estimates m_k^Ω) also satisfy these constraints.

4.2.1 Minimizing the errors

If the m_k^Ω are crude estimates of the corresponding approximations of m^Ω, they may violate these constraints for some k. Slack variables may be introduced to quantify the errors (Equation (4)). Then, an estimate $\widehat{m}^\Omega$ of the bba m^Ω, that is as consistent with the m_k^Ω as possible, may be retrieved by minimizing these errors:

$$\widehat{m}^\Omega = \arg\min_{m^\Omega} \sum_{k=1}^{K} \left(\sum_{i=1}^{4} \underline{\epsilon_k}^i + \overline{\epsilon_k}^i \right), \qquad (2)$$

satisfying:

$$\begin{cases} m^\Omega(B) & \geq & 0 & , \forall B \subseteq \Omega, \\ \sum_{B \subseteq \Omega} m^\Omega(B) & = & 1 & ; \end{cases} \qquad (3)$$

and, for all k:

$$\begin{cases} \overline{m}^\Omega(\emptyset) - \overline{\epsilon_k}^1 & \leq & m_k^\Omega(\emptyset), \\ m_k^\Omega(\emptyset) - \underline{\epsilon_k}^1 & \leq & \underline{m}^\Omega(\emptyset), \\ \overline{m}^\Omega(\{\omega_k\}) - \overline{\epsilon_k}^2 & \leq & m_k^\Omega(\{\omega_k\}), \\ m_k^\Omega(\{\omega_k\}) - \underline{\epsilon_k}^2 & \leq & \underline{m}^\Omega(\{\omega_k\}), \\ \underline{m}^\Omega(\overline{\{\omega_k\}}) - \underline{\epsilon_k}^3 & \leq & m_k^\Omega(\overline{\{\omega_k\}}), \\ m_k^\Omega(\overline{\{\omega_k\}}) - \overline{\epsilon_k}^3 & \leq & \overline{m}^\Omega(\overline{\{\omega_k\}}), \\ \underline{m}^\Omega(\Omega) - \underline{\epsilon_k}^4 & \leq & m_k^\Omega(\Omega), \\ m_k^\Omega(\Omega) - \overline{\epsilon_k}^4 & \leq & \overline{m}^\Omega(\Omega), \\ \underline{\epsilon_k}^i, \overline{\epsilon_k}^i & \geq & 0, \quad \text{for all } i. \end{cases} \qquad (4)$$

4.3 Complexity reduction

The number of subsets of Ω grows exponentially with its size K: for example, when solving a letter recognition problem numbering 26 classes, a bba can have up to $2^{26} = 67\,108\,864$ focal elements. We propose to reduce this complexity, by restricting the set of possible focal elements when computing the bba $\widehat{m}^\Omega$.

The bbas $m_k^{\Theta_k}(\{\omega_k\})$ may be used to identify the classes to which $\mathbf{x}$ likely belongs, and to aggregate the others into a single class. Thus, a coarsening Ω' of Ω is defined; computing the bba $\widehat{m}$ in Ω' enables to restrain the set of its focal elements, by focusing on the most likely ones.

5 Experiments

5.1 Procedure

Three combination methods were evaluated: the evidential approach presented in this article (TBM scheme), the voting rule (VOTE scheme), and the probabilistic method presented in [7] (PROB scheme). Classification decision trees, and evidential neural networks ([4]), were used as binary classifiers.

The classification trees were implemented using the CART algorithm ([2]). The trees were pruned by computing a sequence of trees of increasing size, for which the error and the error variance were estimated using 10-fold cross-validation. Then, the smallest tree which error did not exceed one standard above the minimal error was selected as the best one.

The evidential neural networks (ENN) were trained using 3 prototypes for each class ω_k

(positive class) and $3(K-1)$ prototypes for the others (negative class). This revealed to be unsufficient for the Vowel dataset: then, $3(K-1)$ prototypes were used for both positive and negative classes. When using ENN, the VOTE and PROB schemes were provided with the pignistic probabilities $BetP_k$ computed from the bbas $m_k^{\Theta_k}$.

We conducted experiments on a synthetic dataset: Synth, and on three real datasets: Letter, Satimage, and Vowel (UCI Machine Learning database repository, `http://www.ics.uci.edu/~mlearn/`). The features of these datasets (dimension, number of classes, number of patterns for training and test) are summarized in Table 4. In each class, the training and test patterns were randomly chosen.

Table 4: Datasets features

dataset	dim.	#classes	#train	#test
Letter	16	26	12001	7999
Satimage	36	6	2573	3862
Synth	2	4	1700	340
Vowel	10	11	528	462

For the datasets with more than 6 classes (here, Letter and Vowel), the complexity was reduced by selecting the five classes with highest $m_k^{\Omega}(\{\omega_k\})$ and aggregating the others.

5.2 Quantitative results

Tables 5 and 6 present the results obtained when combining classification trees and evidential neural networks, respectively.

Table 5: Recognition rates (%), classification trees

Method	Letter	Satimage	Synth	Vowel
TBM	79.7	84.4	95.0	37.0
VOTE	79.8	84.5	95.0	36.1
PROB	79.7	84.5	95.0	36.8

Table 6: Recognition rates (%), evidential neural networks

Method	Satimage	Synth	Vowel
TBM	82.7	95.3	65.2
VOTE	82.8	95.3	64.7
PROB	82.8	95.3	64.7

5.3 Qualitative results

Here we analyse qualitatively the results obtained by combining the bbas m_k^{Ω}, obtained using evidential neural networks as binary classifiers, on the synthetic dataset Synth.

Figures 1, 2 and 3 show the bbms $m_3^{\Omega}(\{\omega_3\})$, $m_3^{\Omega}(\overline{\{\omega_3\}})$ and $m_3^{\Omega}(\Omega)$, respectively. Figures 4 and 5 show the bbms $m_4^{\Omega}(\{\omega_4\})$ and $m_4^{\Omega}(\overline{\{\omega_4\}})$ respectively.

Here, the bbas m_k^{Ω} are normalized. The bbms $m_k^{\Omega}(\Omega)$ are high in the regions free of any training patterns, but usually low in regions where the positive and negative classes overlap (in this case, ω_3 and $\overline{\{\omega_3\}}$): there, $m_k(\{\omega_k\})$ and $m_k(\overline{\{\omega_k\}})$ sum almost to one. Thus, imprecision is almost not quantified.

Figures 6, 7 and 8 show the bbms $\widehat{m}^{\Omega}(\{\omega_3\})$, $\widehat{m}^{\Omega}(\{\omega_4\})$ and $\widehat{m}^{\Omega}(\{\omega_3, \omega_4\})$, respectively.

It can be seen that the boundaries between the classes are almost identical to those obtained with the binary classifiers. It can be noticed that some belief (although very few) was given to the set $\{\omega_3, \omega_4\}$ in the region between ω_3 and ω_4. This suggests that combining classifiers giving a significant belief $m_k^{\Omega}(\Omega)$ in the regions where the positive and negative classes overlap, could allow to retrieve a bba $\widehat{m}^{\Omega}$ quantifying the imprecision between some classes (in this case, ω_3 and ω_4).

Figures 9 and 10 show the pignistic probabilities $BetP(\omega_3)$ and $BetP(\omega_4)$ computed from the combined bbas $\widehat{m}^{\Omega}$, respectively. Figures 11 and 12 show the probabilities $\widehat{P}(\omega_3)$ and $\widehat{P}(\omega_4)$ estimated using the probabilistic method described in [7], respectively.

It can be seen that the estimates of the posterior probabilities obtained with both meth-

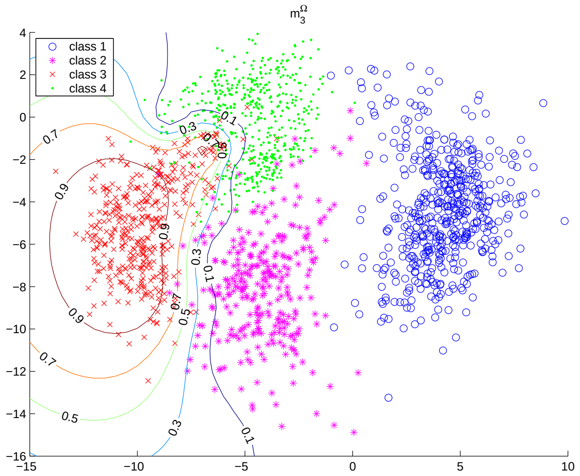

Figure 1: Masses $m_3^\Omega(\{\omega_3\})$

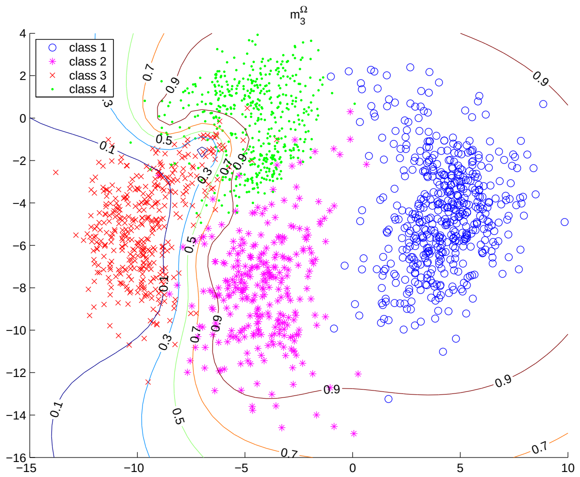

Figure 2: Masses $m_3^\Omega(\overline{\{\omega_3\}})$

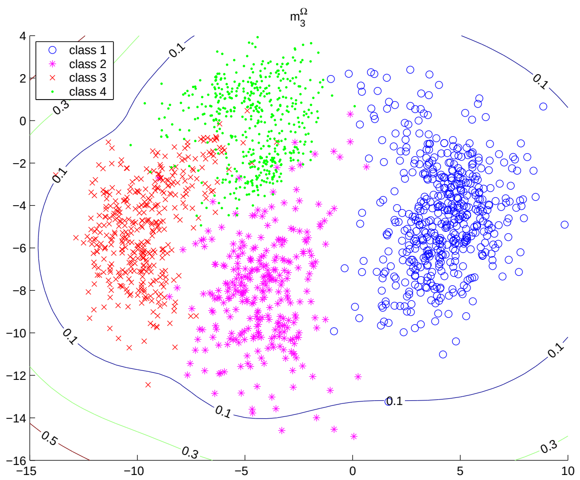

Figure 3: Masses $m_3^\Omega(\Omega)$

ods are very similar. These results, along with those presented in Section 5.2, indicate that the combination schemes studied here may be equivalent in accuracy. In the case of OVA decomposition, the information on the boundary of a class ω_k is provided by a single classifier. Hence, the binary classifiers are complementary for solving the multiclass problem, and their combination is likely to provide similar decision boundaries, whatever the scheme used. Moreover, when the bbas m_k^Ω are (close to) probabilities, their combination is likely to be itself (close to) a probability.

Combination of more complex classifiers, that quantify imprecision (between the classes they separate) and ignorance (in the regions of the space that correspond to classes they were not trained to separate), might lead to compute a bba quantifying a richer knowledge of the membership of $\mathbf{x}$. Alternatively, combination of classifiers trained to separate two sets $A \subseteq \Omega$ and $B \subseteq \Omega$ of classes, might enable to increase the robustness of the combination: the information on the boundary of a class ω_k is then brought by several classifiers. This alternative corresponds to an ECOC-combination of binary classifiers.

6 Conclusion

In this article, we presented a method for combining binary classifiers in the case of OVA decomposition. The combination scheme is formalized in the Transferable Belief Model. The results show that the accuracy of the method is similar to those of two other combination schemes. Moreover, probabilistic classifiers as well as credal classifiers may be combined using our technique.

Further work concerns the combination of credal binary classifiers quantifying imprecision or ignorance; formalizing the combination of classifiers in the general ECOC-decomposition framework should then enable to combine different types of binary classifiers, in order to increase the robustness and the accuracy of the solution.

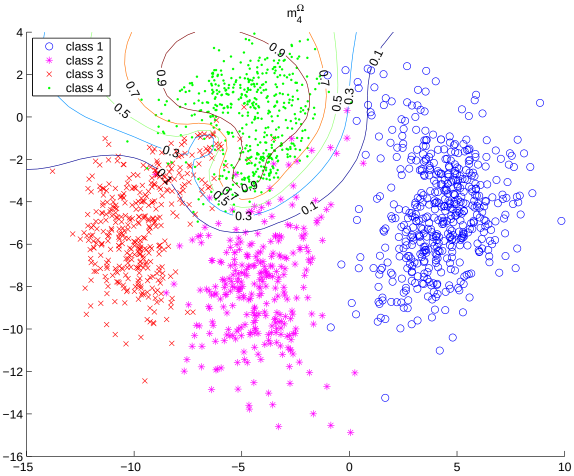

Figure 4: Masses $m_4^{\Omega}(\{\omega_4\})$

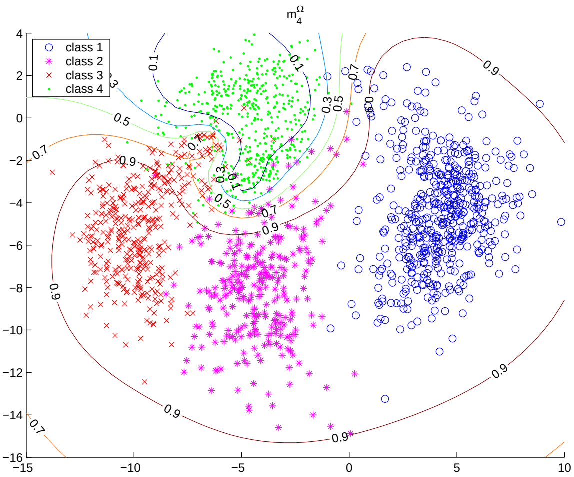

Figure 5: Masses $m_4^{\Omega}(\overline{\{\omega_4\}})$

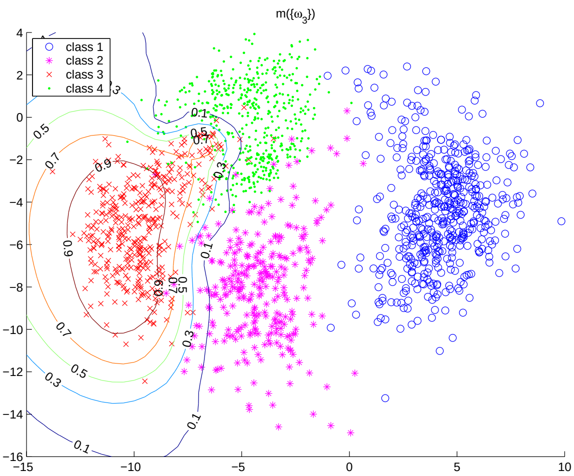

Figure 6: Combined masses $\widehat{m}^{\Omega}(\{\omega_3\})$

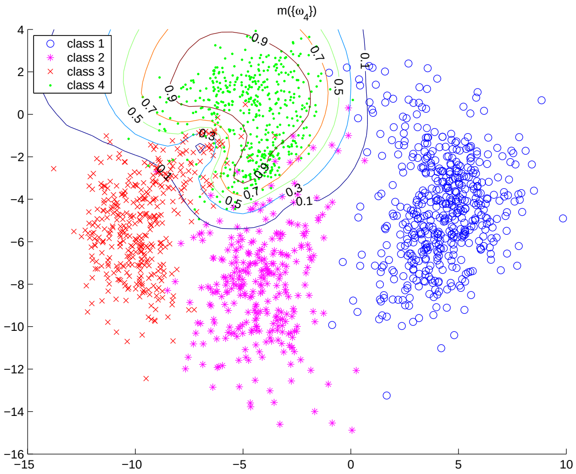

Figure 7: Combined masses $\widehat{m}^{\Omega}(\{\omega_4\})$

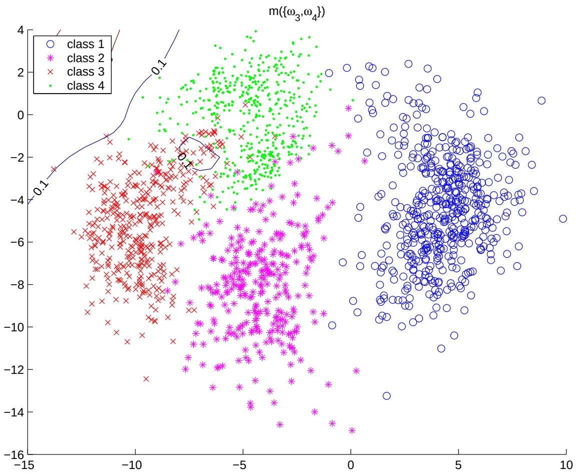

Figure 8: Combined masses $\widehat{m}^{\Omega}(\{\omega_3, \omega_4\})$

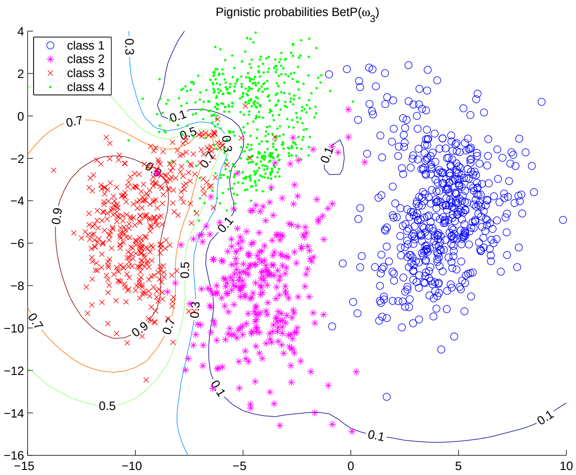

Figure 9: Pignistic probabilities $BetP(\omega_3)$

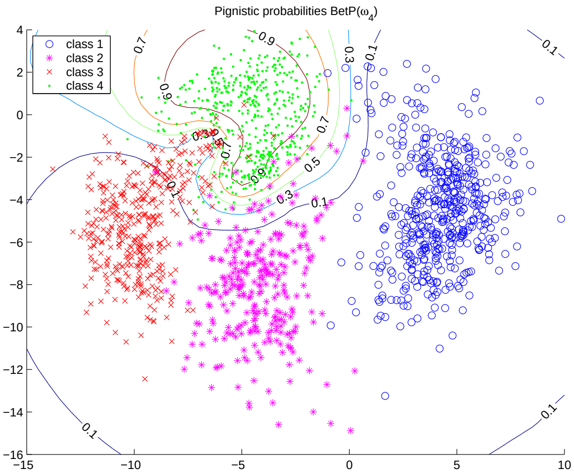

Figure 10: Pignistic probabilities $BetP(\omega_4)$

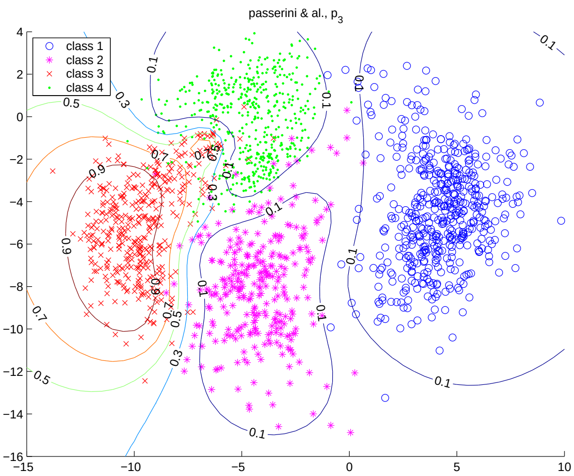

Figure 11: Probabilities $\widehat{P}(\omega_3)$

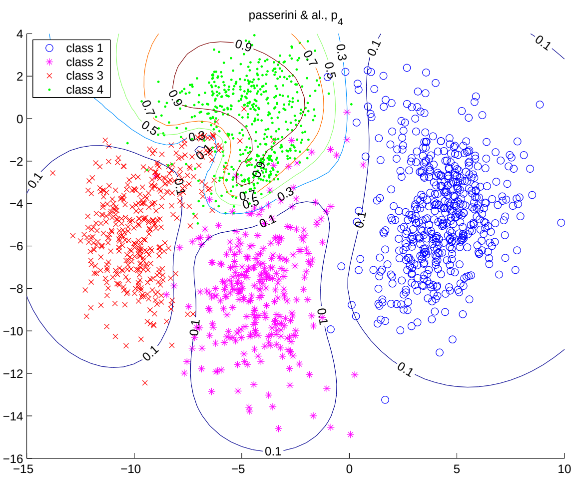

Figure 12: Probabilities $\widehat{P}(\omega_4)$

References

[1] E. Allwein, R. Schapire and Y. Singer: Reducing Multiclass to Binary: A Unifying Approach for Margin Classifiers. In *International Conference on Machine Learning*, vol. 17, pages 9-16, 2000.

[2] L. Breiman, J. Friedman, R. Olshen and C. Stone: Classification and Regression Trees. Wadsworth, Belmont (1984).

[3] T. Dietterich and G. Bakiri: Solving Multiclass Learning Problems via Error-Correcting Output Codes. In *Journal of Artificial Intelligence Research*, vol. 2, pages 263-286, 1995.

[4] T. Denœux: An evidence-theoretic neural network classifier. In *IEEE International Conference on Systems, Man and Cybernetics*, vol. 3, pages 712-717, 1995.

[5] J. Friedman: Another Approach to Polychotomous Classification. *Technical Report, Department of Statistics and Stanford Linear Accelerator Center, Stanford University*, 1996.

[6] T. Hastie and R. Tibshirani: Classification by Pairwise Coupling. In *Advances in Neural Information Processing Systems*, vol. 10, pages 507-513, Denver, USA, 1997.

[7] A. Passerini, M. Pontil and P. Frasconi: New Results on Error Correcting Output Codes of Kernel Machines. In *IEEE Transactions on Neural Networks*, vol. 15, pages 45-54, 2004.

[8] B. Quost, T. Denœux and M. Masson: Pairwise Classifier Combination in the Framework of Belief Functions. In *Eighth Conference on Information Fusion Conference*, 2005.

[9] G. Shafer: A mathematical theory of evidence. Princeton University Press (1976).

[10] P. Smets and R. Kennes: The transferable belief model. In *Artificial Intelligence*, vol. 66, pages 191-234, 1994.