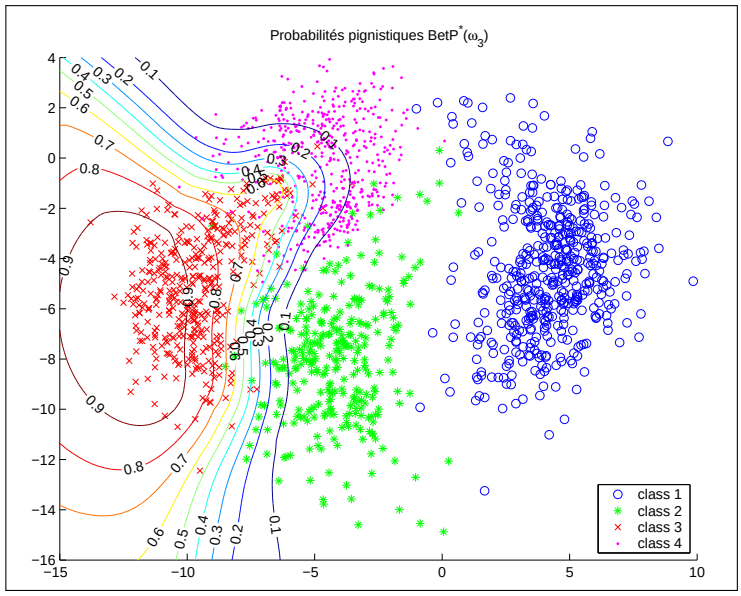
utc
Université de Technologie
Compiègne

M^r Benjamin Quost

Combinaison de classifieurs binaires dans le cadre de la théorie des fonctions de croyance

Thèse présentée pour obtenir le grade de Docteur de l'Université de Technologie de Compiègne.



Soutenue le 30 novembre 2006
Spécialité : Technologie de l'Information et des Systèmes

Combinaison de classifieurs binaires dans le cadre de la théorie des fonctions de croyance

Benjamin QUOST

Soutenue le 30 novembre 2006 devant le jury composé de :

M^{r} Alain APPRIOU	rapporteur
M^{me} Florence D'ALCHÉ-BUC	rapporteur
M^{r} Thierry DENŒUX	directeur de thèse
M^{r} Gérard GOVAERT	président du jury
M^{elle} Marie-Hélène MASSON	co-directrice de thèse
M^{r} Patrick VANNOORENBERGHE	examinateur

Remerciements

Tout d'abord, je tiens à remercier mes directeurs de thèse, Thierry Denœux et Mylène Masson, pour leur patience, leur aide, la pertinence de leurs conseils. La thèse est une expérience un peu aventureuse, à l'issue incertaine ; et c'est avant tout à eux que je dois d'avoir mené la mienne à son terme. Travailler à leurs côtés a été un réel plaisir, et j'espère en avoir de nouveau l'occasion par la suite.

Je suis également reconnaissant à Alain Appriou et Florence d'Alché-Buc pour m'avoir fait l'honneur de rapporter sur ce mémoire, ainsi qu'à Patrick Vannoorenberghe, qui a spontanément accepté de joindre ses efforts aux leurs ; je ne saurais omettre Gérard Govaert, qui a présidé le jury réuni pour ma soutenance. Je les remercie tous pour leurs remarques et leurs critiques constructives.

Nombreux sont ceux qui m'ont fait partager leur expérience d'enseignant ou de chercheur au cours de ces dernières années. Je souhaite ainsi remercier, dans le désordre alphabétique, Christophe Ambroise, Yves Grandvalet, Pierre Villon, Stéphane Mottelet, Djalil Kateb, Marc Dambrine, Franck Davoine. Je garde une pensée particulière pour Bernard Dubuisson, qui m'a fait connaître la reconnaissance de formes.

Le travail présenté dans ce mémoire est le fruit de trois ans de travail au sein du laboratoire Heudiasyc, au département de Génie Informatique de l'UTC. Je souhaite exprimer ma gratitude à tous les membres qui s'attachent à y rendent le travail plus agréable : notamment, les secrétaires et les membres de la cellule logistique, dont j'ai apprécié l'aide.

Je remercie bien sûr tous les doctorants côtoyés pendant cette thèse : mes collègues de bureau David et Stéphane, avec lesquels j'ai eu des discussions aussi passionnantes que variées ; et Erik, Romain, Marie, Astride, Pierre-Alexandre, Amel, Yoann, Morgan, Julien et David, Etienne et Alexandra, Maged, Frédéric.

Enfin, j'adresse ma profonde gratitude et ma reconnaissance à ma famille et à mes amis, pour leur soutien au cours de ces années.

Résumé

La classification supervisée a pour enjeu de construire un système, ou classifieur, capable de prédire automatiquement la classe d'un phénomène observé. Son architecture peut être modulaire : le problème abordé est décomposé en sous-problèmes plus simples, traités par des classifieurs, et la combinaison des résultats donne la solution globale. Nous nous intéressons au cas de sous-problèmes binaires, en particulier les décompositions où chaque classe est opposée à chaque autre, chaque classe est opposée à toutes les autres, et le cas général où deux groupes de classes disjoints sont opposés l'un à l'autre.

La combinaison des classifieurs est formalisée dans le cadre de la théorie des fonctions de croyance. Nous interprétons les sorties des classifieurs binaires comme des fonctions de croyance définies sur des domaines restreints, dépendant du schéma de décomposition employée. Les classifieurs sont alors combinés en déterminant la fonction de croyance la plus consistante possible avec leurs sorties.

Supervised classification aims at building a system, or classifier, able to predict the class of a phenomenon being observed. Its architecture may be modular : the problem to be tackled is decomposed into simpler sub-problems, solved by classifiers, and the combination of the results gives the global solution. We address the case of binary sub-problems in particular the decompositions where each class is opposed to each other, each class is opposed to all the others, and the general case where two disjoint groups of classes are opposed to each other.

The combination of the classifiers is formalized within the theory of evidence framework. We interpret the outputs of the binary classifiers as belief functions defined on restricted domains, according to the decomposition scheme used. The classifiers are then combined by determining the belief function which is the most consistant with their outputs.

Table des matières

Liste des tableaux

Table des figures

Introduction

Avant-propos

Le besoin de conférer une plus grande autonomie à la machine, de la substituer à l'individu pour la réalisation de tâches répétitives, a contribué au développement du traitement automatique des données, et en particulier la reconnaissance des formes. Cette discipline, qui emprunte à l'informatique, aux mathématiques appliquées, à la statistique, est à présent reconnue comme domaine de recherche à part entière. L'un des thèmes majeurs de ce domaine de recherche, la classification supervisée, a notamment pour objectif de construire un système, appelé classifieur, capable de prédire automatiquement le type d'un phénomène observé, sur la base d'exemples. L'apprentissage statistique permet ainsi de proposer aux médecins des outils d'aide au diagnostic, de doter les biologistes de systèmes d'analyse ou d'aide à la conception de molécules. Le traitement de signaux ou d'images peut être partiellement automatisé, ce qui ouvre de nouvelles perspectives aux technologies de l'information et des télécommunications ; l'identification et le suivi d'individus, la reconnaissance d'objets et de sons, constituent des avancées majeures dans les domaines de la sécurité ou de la défense.

Ces dernières années, un nombre significatif de travaux a porté sur la résolution de problèmes de classification par combinaison de classifieurs. Ainsi, plusieurs classifieurs entraînés à résoudre le même problème de classification peuvent être combinés, dans le but d'améliorer leurs performances individuelles ; on pourra citer la technique du boosting [23], qui consiste à entraîner un ensemble de classifieurs à résoudre un même problème, un nouveau classifieur s'attachant à corriger les erreurs commises par les précédents. La combinaison de classifieurs peut également être utilisée pour décomposer un problème complexe en sous-problèmes plus simples à résoudre : chaque sous-problème est résolu au moyen d'un classifieur, et les résultats ainsi obtenus sont ensuite combinés pour déterminer la solution du problème global.

Nous avons centré notre travail sur la résolution de problèmes comptant plusieurs classes au moyen de classifieurs binaires, entraînés à reconnaître deux classes : en effet, un certain nombre d'algorithmes, comme la régression logistique ou les séparateurs à vaste marge, ont une formulation plus simple dans ce cas. Nous nous sommes intéressés plus particulièrement à différents schémas de

décomposition du problème initial, en considérant des classifieurs binaires entraînés à distinguer une classe d'une autre, une classe de l'ensemble des autres, et deux groupes de classes disjoints l'un de l'autre. Soulignons que dans chacun de ces cas, les informations fournies par chaque classifieur concernent un ensemble limité de classes : elles sont donc exprimées sur un référentiel propre au classifieur, plus restreint que le domaine correspondant au problème global.

La résolution d'un problème de reconnaissance des formes nécessite de définir un cadre théorique, dans lequel les informations disponibles et les résultats obtenus sont modélisés et interprétés. Dans le cas d'une approche par décomposition du problème considéré, ce formalisme doit également permettre une modélisation claire de la combinaison des classifieurs. Le cadre le plus classique de représentation des connaissances est la théorie des probabilités, qui permet d'associer à un ensemble d'hypothèses une mesure de probabilité sur leur réalisation. Or, bien qu'il permette de résoudre de manière satisfaisante un grand nombre de problèmes de reconnaissance des formes, ce formalisme offre peu d'outils pour gérer des informations définies sur des référentiels différents.

Nous avons abordé le problème de la combinaison de classifieurs sous l'angle de la théorie des fonctions de croyance. Dans ce formalisme, parfois appelé théorie de Dempster-Shafer ou théorie de l'évidence [51, 55], les connaissances sont représentées par des fonctions de croyance définies sur des cadres de discernement. Des outils ont été définis pour gérer les correspondances entre différents cadres ; en particulier, divers opérateurs permettent de transformer une fonction de croyance sur un autre cadre que celui sur lequel elle est définie. Pour ces raisons, la théorie des fonctions de croyance semble a priori être bien adaptée à la formalisation du problème de combinaison de classifieurs entraînés sur des référentiels différents.

Ce travail a pour motivation d'étudier les possibilités offertes par le Modèle des Croyances Transférables, une interprétation subjectiviste de la théorie des fonctions de croyance, pour formaliser la combinaison de classifieurs binaires. Il se situe donc à la croisée des domaines de la classification supervisée par combinaison de classifieurs et de la gestion des connaissances mentionnés ci-dessus. Pour chacun des schémas de décomposition considérés, nous nous sommes attachés à interpréter les sorties des classifieurs dans le cadre du Modèle des Croyances Transférables, pour ensuite en déduire une règle de combinaison permettant d'élaborer la solution du problème global.

Organisation du mémoire

Le présent mémoire est articulé en cinq chapitres. Le premier est consacré au Modèle des Croyances Transférables, et plus particulièrement à la présentation des concepts auxquels les méthodes de combinaison de classifieurs développées présentées par la suite font appel. Le second chapitre est dédié à la description de méthodes de combinaison de classifieurs déjà existantes. Nous nous sommes

intéressés plus particulièrement aux méthodes formalisées dans un cadre théorique de représentation des connaissances.

Les travaux réalisés sont présentés dans les chapitres trois et quatre : les méthodes de combinaison de classifieurs binaires élaborées y sont décrites, et leurs propriétés analysées, pour les différents types de décomposition considérés. Le chapitre cinq concerne l'étude de la précision des méthodes, ainsi que leur comparaison à d'autres méthodes de combinaison de classifieurs binaires existantes. Nous présentons tout d'abord les résultats obtenus lors du traitement de jeux de données de la littérature, qui sont ensuite analysés et interprétés de manière plus approfondie.

Enfin, la conclusion présente une synthèse globale du mémoire, ainsi que les perspectives soulevées par la réalisation de ces travaux.

Chapitre 1

Le Modèle des Croyances Transférables

La nécessité de disposer d'un cadre riche et flexible de représentation de la connaissance a conduit au développement de divers formalismes : les plus notables sont la théorie des possibilités [18], la théorie des probabilités imprécises [60], et la théorie de Dempster-Shafer [51].

Cette dernière, parfois appelée théorie des fonctions de croyance ou théorie de l'évidence, permet de représenter la connaissance partielle de la valeur d'une variable y par une fonction de croyance. Le Modèle des Croyances Transférables (MCT) est une interprétation subjectiviste de cette théorie, développée par Smets [52, 55]. Divers travaux ont mis en évidence l'intérêt de ce cadre théorique pour la formalisation et la résolution de problèmes de diagnostic [53], de reconnaissance de formes [11, 12, 66], ou de fusion d'informations [4, 5, 21].

Dans ce chapitre, nous présentons le MCT, et plus particulièrement les concepts sur lesquels se basent les méthodes de combinaison de classifieurs présentées aux chapitres 3 et 4.

1.1 Représentation et manipulation des connaissances

Le MCT permet de modéliser la connaissance partielle de la valeur d'une variable y, définie sur un cadre de discernement, ou domaine, $\Omega = \{\omega_1, \ldots, \omega_K\}$. Dans le cas de la classification supervisée, la variable y correspond à la classe d'un individu décrit par un vecteur d'attributs $\mathbf{x}$.

Le MCT s'articule en deux niveaux : un niveau crédal où les connaissances concernant la valeur de y sont représentées et aggrégées, et un niveau pignistique, où elles sont utilisées pour prendre une décision.

1.1.1 Représenter les connaissances au moyen de fonctions de croyance

Définition 1.1 (fonction de masse) *Une fonction de masse de croyance est une application* $m^\Omega : 2^\Omega \to [0;1]$*, qui vérifie :*

$$\sum_{A \subseteq \Omega} m^\Omega(A) = 1. \tag{1.1}$$

La croyance relative à la valeur de y peut être quantifiée par une fonction de masse de croyance m^Ω. La quantité $m^\Omega(A)$ est interprétée comme une masse de croyance allouée spécifiquement à l'hypothèse A, sur la base d'un élément d'évidence, et qui ne peut être allouée à un sous-ensemble strict $B \subset A$ du fait d'un manque d'information. Tout sous-ensemble $A \subseteq \Omega$, tel que $m^\Omega(A) > 0$, est appelé élément focal de m^Ω. L'exposant Ω peut être omis lorsqu'il n'y a pas d'ambiguïté sur le domaine de m.

Définition 1.2 (fonction de masse catégorique) *Une fonction de masse catégorique a un unique élément focal* A *:*

$$m(A) = 1.$$

Lorsque $A = \Omega$, on obtient la fonction de croyance vide, modélisant l'ignorance totale.

Définition 1.3 (fonction de masse Bayésienne) *Les fonctions de masse Bayésiennes n'ont que des éléments focaux singletons :*

$$m(A) > 0 \Rightarrow |A| = 1.$$

Définition 1.4 (normalité) *Une fonction de masse* m *est dite normale si* $m(\emptyset) = 0$*. Dans le cas contraire,* m *est dite sous-normale ; la masse* $m(\emptyset)$ *est généralement interprétée comme le degré de conflit entre les connaissances quantifiées par* m*. Une fonction de masse sous-normale peut être transformée en fonction de masse normale, en divisant chaque masse* $m(A)$*,* $A \neq \emptyset$*, par* $1 - m(\emptyset)$ *; le résultat de cette normalisation est noté* m^* *:*

$$m^*(A) = \begin{cases} \dfrac{m(A)}{1 - m(\emptyset)} & si\ A \neq \emptyset, \\ 0 & si\ A = \emptyset. \end{cases} \tag{1.2}$$

L'hypothèse du monde clos consiste à supposer que le cadre Ω est exhaustif : la valeur réelle de y est alors nécessairement l'un des atomes $\omega_k \in \Omega$. Dans ce cas, l'hypothèse de normalité est généralement acceptée. Le cas contraire correspond à l'hypothèse du monde ouvert : on considère alors que la valeur de y peut ne pas être un atome de Ω. Dans ce cas, la masse $m(\emptyset)$ peut être interprétée comme la croyance en l'hypothèse « $y \notin \Omega$ », le degré de conflit quantifié par $m(\emptyset)$ étant dû à la non-exhaustivité de Ω.

Définition 1.5 (conditionnement) *Soit une fonction de masse m^Ω. Le conditionnement $m^\Omega[B]$ de m par rapport à un sous-ensemble $B \subseteq \Omega$ peut être calculé par :*

$$m^\Omega[B](A) = \begin{cases} \sum_{C \cap B = A} m(C) & \text{si } A \subseteq B, \\ 0 & \text{sinon.} \end{cases} \tag{1.3}$$

Toute masse de croyance initialement associée à $C \subseteq \Omega$ est ainsi transférée à $C \cap B$. La fonction de masse $m^\Omega[B]$ quantifie la connaissance de la valeur de y, sachant que $y \in B$; la masse $m^\Omega[B](\emptyset)$ représente alors la masse de croyance donnée par m aux hypothèses incompatibles avec B. On peut aussi voir le conditionnement $m^\Omega[B]$ comme une fonction de masse définie sur le cadre restreint B, toutes les masses $m^\Omega(C)$ telles que $C \not\subseteq B$ étant nulles. Dans ce cas, la masse $m^\Omega[B](\emptyset)$ est naturellement interprétée comme la croyance en l'hypothèse « $y \notin B$ ». Remarquons enfin que l'opération définie par (1.3) produit un conditionnement $m^\Omega[B]$ généralement sous-normal ; la version normalisée de cette transformation peut-être obtenue en ajoutant une étape de normalisation :

$$m^\Omega[B]^*(A) = \begin{cases} \dfrac{m^\Omega[B](A)}{1 - m^\Omega[B](\emptyset)} & \text{si } A \subseteq \Omega, A \neq \emptyset, \\ 0 & \text{si } A = \emptyset. \end{cases} \tag{1.4}$$

La figure 1.1 illustre les transferts de masses lors d'un conditionnement.

Définition 1.6 (fonctions de croyance et de plausibilité) *Les fonctions de croyance bel et de plausibilité pl sont définies, pour tout $A \subseteq \Omega$, par :*

$$bel(A) = \sum_{B \subseteq A, B \neq \emptyset} m(B), \tag{1.5}$$

$$pl(A) = \sum_{A \cap B \neq \emptyset} m(B). \tag{1.6}$$

La quantité $bel(A)$ représente le degré total de croyance spécifique et justifiée en A [55] : justifiée, car ne sont pris en compte que les masses de croyance allouées à des sous-ensembles $B \subseteq A$; et spécifique, car l'élément $\emptyset$ n'est pas considéré, étant un sous-ensemble de A et de $\overline{A}$. La quantité $pl(A)$ constitue une borne supérieure sur le degré de croyance qui pourrait être allouée à A après conditionnement : en effet, $pl(A) = bel[A](A) = \max_{B \subseteq \Omega} bel[B](A)$.

Définition 1.7 (fonctions d'implicabilité et de communalité) *Les fonctions d'implicabilité b et de communalité q sont définies, pour tout $A \subseteq \Omega$, par :*

$$b(A) = \sum_{B \subseteq A} m(B), \quad \forall A \subseteq \Omega; \tag{1.7}$$

$$q(A) = \sum_{A \subseteq B} m(B), \quad \forall A \subseteq \Omega. \tag{1.8}$$

TAB. 1.1 – Fonctions de croyance quantifiant la connaissance de l'état d'un patient.

	m_A	m^*_A	bel_A	pl_A	b_A	q_A
$\emptyset$	0.2	0	0	0	0.2	1
$\{bro\}$	0	0	0	0.5	0.2	0.5
$\{can\}$	0	0	0	0.1	0.2	0.1
$\{bro, can\}$	0	0	0	0.5	0.2	0.1
$\{tub\}$	0.3	0.375	0.3	0.8	0.5	0.8
$\{bro, tub\}$	0.4	0.5	0.7	0.8	0.9	0.5
$\{can, tub\}$	0	0	0.3	0.8	0.5	0.1
$\{bro, can, tub\}$	0.1	0.125	0.8	0.8	1	0.1

La quantité $q(A)$ peut être interprétée comme la masse de croyance pouvant être transférée spécifiquement à A [51] par conditionnement : en effet, $q(A) = m[A](A) = \max_{B \subseteq \Omega} m[B](A)$. Remarquons que l'on a $b(A) = bel(A) + m(\emptyset)$, ou encore $b(A) = 1 - pl(\overline{A})$: la quantité $b(A)$ peut donc être vue comme la quantité totale de croyance ne pouvant être transférée à $\overline{A}$ par conditionnement.

Exemple 1.1 (diagnostic médical) Nous proposons d'illustrer les notions présentées sur un exemple de diagnostic médical, très simplifié. Un médecin s'interroge sur la maladie d'un patient se plaignant de douleurs pulmonaires ; les pathologies considérées sont la bronchite ($\{bro\}$), le cancer du poumon ($\{can\}$) et la tuberculose ($\{tub\}$) (on supposera qu'on ne peut observer plus d'une maladie à la fois). Suite à l'examen du patient, le praticien modélise sa connaissance de la maladie par une fonction de masse m_A, représentée dans la table 1.1 ainsi que les fonctions qui lui sont associées. Le praticien a donc une croyance spécifique de 0.3 dans le fait que le patient souffre de tuberculose et de 0.4 dans le fait qu'il souffre de bronchite ou de tuberculose, un degré d'ignorance totale (« l'affliction est l'une des pathologies considérées ») de 0.1, et un degré de croyance de 0.2 dans le fait que l'affliction n'est pas l'une des pathologies considérées. □

Définition 1.8 (négation d'une fonction de masse [19]) *La négation $\neg m$ d'une fonction de masse m est définie par :*

$$\neg m(A) = m(\overline{A}), \quad \forall A \subseteq \Omega. \tag{1.9}$$

Propriété 1.1 *Soient $\neg b$ et $\neg q$ les fonctions d'implicabilité et de communalité associées à la négation $\neg m$ de m. On a :*

$$\neg b(A) = q(\overline{A}), \quad \forall A \subseteq \Omega, \tag{1.10}$$

$$et\ \neg q(A) = b(\overline{A}), \quad \forall A \subseteq \Omega. \tag{1.11}$$

Définition 1.9 (inclusion forte [14, 54]) *Soient deux fonctions de masse de croyance m_1 et m_2, ayant pour éléments focaux respectifs $A_1, \ldots, A_p$ et $B_1, \ldots, B_q$. La fonction de masse m_1 est dite incluse au sens fort dans m_2, s'il existe une matrice non-négative W d'éléments w_{ij} ($i \in \{1, \ldots, p\}$, $j \in \{1, \ldots, q\}$) telle que :*

$$\sum_{j=1}^{q} w_{ij} = m_1(A_i), \quad \forall i \in \{1, \ldots, p\}, \tag{1.12}$$

$$\sum_{i=1}^{p} w_{ij} = m_2(B_j), \quad \forall j \in \{1, \ldots, q\}; \tag{1.13}$$

$$w_{ij} > 0 \Rightarrow A_i \subseteq B_j. \tag{1.14}$$

On dit aussi que m_1 est une spécialisation de m_2, ou encore que m_2 est une généralisation de m_1. On note alors :

$$m_1 \subseteq m_2. \tag{1.15}$$

Intuitivement, la relation définie par les équations (1.12)-(1.14) signifie que m_1 peut être obtenue à partir de m_2 en transférant toute masse $m_2(B_j)$ à des éléments $A \subseteq B_j$; la proportion exacte de masse $m_2(B_j)$ transférée à $A_i \subseteq B_j$ est alors égale à w_{ij}.

Définition 1.10 (inclusion faible [14, 54]) *Soient deux fonctions de masse m_1 et m_2. La fonction de masse m_1 est dite faiblement incluse dans m_2 au sens des plausibilités, ce qui s'écrit $m_1 \subseteq_{pl} m_2$, si :*

$$pl_1(C) \leq pl_2(C), \quad \forall C \subseteq \Omega. \tag{1.16}$$

On dit alors parfois que m_A est plus spécifique, ou conservative, que m_B au sens des plausibilités. De même, la fonction de masse m_1 est dite incluse au sens faible dans m_2 au sens des communalités (ou encore plus spécifique, ou conservative, que m_2 au sens des communalités), ce qui s'écrit $m_1 \subseteq_q m_2$, si :

$$q_1(C) \leq q_2(C), \quad \forall C \subseteq \Omega. \tag{1.17}$$

Propriété 1.2 *Soient deux fonctions de masse m_1 et m_2 telles que $m_1 \subseteq m_2$. On a :*

$$m_1 \subseteq_{pl} m_2,$$
$$m_1 \subseteq_{q} m_2.$$

Définition 1.11 (Principe du Minimum d'Information) *Soient bel une fonction de croyance, dont la connaissance peut être incomplète, et $\mathcal{F}$ un ensemble de fonctions de croyance compatibles avec bel. Lorsqu'une seule de ces fonctions doit être choisie, le Principe du Minimum d'Information (PMI) consiste à sélectionner la moins spécifique.*

Cet axiome, qui reflète une forme de conservatisme dans l'allocation des masses, formalise l'idée qu'un sous-ensemble $A \subseteq \Omega$ ne doit pas recevoir plus de croyance qu'il n'est justifié.

1.1.2 Combinaison de fonctions de croyance

Les connaissances représentées par des fonctions de croyance peuvent être aggrégées, au niveau crédal, en utilisant un opérateur approprié. Deux règles de combinaison ont ainsi été définies pour combiner des fonctions de croyance distinctes : les sommes conjonctive et disjonctive. Par « distinctes », on entend l'absence de lien entre les sources d'information permettant de définir les fonctions. Un opérateur permettant de combiner des fonctions de croyance obtenues à partir de sources d'information non-distinctes a été récemment proposé [13].

Définition 1.12 (sommes conjonctive et disjonctive) *Soient m_1 et m_2 deux fonctions de croyance distinctes. La somme conjonctive $m_1 \textcircled{\cap}_2$ de m_1 et m_2 est définie par :*

$$m_1 \textcircled{\cap}_2(Z) = \sum_{X \cap Y = Z} m_1(X)\, m_2(Y). \tag{1.18}$$

La somme disjonctive $m_1 \textcircled{\cup}_2$ de m_1 et m_2 est définie par :

$$m_1 \textcircled{\cup}_2(Z) = \sum_{X \cup Y = Z} m_1(X)\, m_2(Y). \tag{1.19}$$

Propriété 1.3 *Soient $m_1 \textcircled{\cap}_2 = m_1 \textcircled{\cap} m_2$, et $m_1 \textcircled{\cup}_2 = m_1 \textcircled{\cup} m_2$; on a :*

$$m_1 \textcircled{\cap}_2 \subseteq m_1 \subseteq m_1 \textcircled{\cup}_2, \tag{1.20}$$

$$m_1 \textcircled{\cap}_2 \subseteq m_2 \subseteq m_1 \textcircled{\cup}_2. \tag{1.21}$$

Remarquons que la somme conjonctive généralise l'opérateur de conditionnement présenté au paragraphe 1.1.1 : toute masse $m_1(A)$ est en effet transférée aux sous-ensembles $A \cap B$ proportionnellement aux masses de croyance $m_2(B)$. Ainsi, le conditionnement par rapport à B est obtenu lorsque m_2 est une fonction de masse catégorique définie par $m_2(B) = 1$.

1.1.3 Prise de décision

Lorsqu'une décision doit être prise quant à la valeur de y, la fonction de croyance exprimant l'ensemble de la connaissance disponible est transformée en une distribution de probabilité pignistique [55].

Définition 1.13 (probabilité pignistique) *La probabilité pignistique $BetP^*$ associée à une fonction de masse m est définie par :*

$$BetP^*(\omega) = \sum_{A \subseteq \Omega : \omega \in A} \frac{m^*(A)}{|A|}, \quad \forall \omega \in \Omega. \tag{1.22}$$

La transformation pignistique ainsi définie consiste à partager équitablement chaque masse de croyance normalisée $m^*(A)$ entre les atomes composant A. Un équivalent non-normalisé $BetP$ de la probabilité pignistique $BetP^*$ associée à la fonction de masse m peut être obtenu en remplaçant m^* par m dans la relation (1.22) :

$$BetP(\omega) = \sum_{A \subseteq \Omega : \omega \in A} \frac{m(A)}{|A|}, \quad \forall \omega \in \Omega. \tag{1.23}$$

Bien qu'il ne s'agisse pas d'une distribution de probabilité – on a en effet $\sum_k BetP(\omega_k) < 1$ lorsque $m(\emptyset) > 0$, $BetP$ sera appelée par la suite probabilité pignistique non-normalisée associée à la fonction de masse m, par abus de langage.

Exemple 1.2 (diagnostic médical, suite) Un examen complémentaire permet de déterminer avec quasi-certitude que le patient ne souffre pas d'un cancer des poumons. Cette croyance peut être modélisée par la fonction de masse m_B définie par :

$$\begin{aligned} m_B(\{bro, tub\}) &= 0.9, \\ m_B(\{bro, can, tub\}) &= 0.1. \end{aligned}$$

Le praticien intègre cette nouvelle information à son diagnostic, en combinant conjonctivement m_A et m_B. Le résultat de cette opération, noté m_{AB}, est défini par :

$$\begin{aligned} m_{AB}(\emptyset) &= 0.2, \\ m_{AB}(\{tub\}) &= 0.3, \\ m_{AB}(\{bro, tub\}) &= 0.49, \\ m_{AB}(\{bro, can, tub\}) &= 0.01. \end{aligned}$$

Cette fonction de masse est une spécialisation de m_A : elle a été obtenue en transférant une masse de 0.09 de $\{bro, can, tub\}$ à $\{bro, tub\}$. On pourra constater que $pl_{AB} \leq pl_A$ et $q_{AB} \leq q_A$ pour tout $A \subseteq \Omega$.

Supposons que le praticien doive se prononcer sur la maladie dont souffre le patient. Le calcul de la probabilité pignistique normalisée $BetP^*_{AB}$ associée à m_{AB} donne :

$$\begin{aligned} BetP^*_{AB}(bro) &= 0.3104, \\ BetP^*_{AB}(can) &= 0.0042, \\ BetP^*_{AB}(tub) &= 0.6854. \end{aligned}$$

En effet, les examens menés permettent de penser que le patient présente une tuberculose plutôt qu'une bronchite ; l'hypothèse d'un cancer des poumons, quoique n'étant pas totalement écartée, paraît très peu probable. □

1.2 Opérations sur les cadres de discernement

Dans ce paragraphe, nous rappelons quelques axiomes fondamentaux du MCT, puis nous décrivons certaines transformations s'appliquant aux cadres de discernement.

Définition 1.14 (consistance) *Soient une application injective* $\Psi : 2^\Theta \rightarrow 2^\Omega$*, et une fonction de croyance* bel^Θ *définie sur* Θ*. S'il existe une fonction de croyance* bel^Ω *définie sur* Ω*, telle que :*

$$bel^\Theta(A) = bel^\Omega\left(\Psi(A)\right), \quad \forall A \subseteq \Theta, \tag{1.24}$$

les fonctions de croyance bel^Θ *et* bel^Ω *sont alors dites consistantes [51] relativement à* Ψ*.*

La propriété de consistance a été définie dans le cas des fonctions de croyance *bel* [51]. Elle peut néanmoins être étendue à tout type de fonction f : fonction de masse, de plausibilité, de communalité ou d'implicabilité.

Définition 1.15 (consistance, dans le cas général) *Soient* $\Psi : 2^\Theta \rightarrow 2^\Omega$ *une application injective, et une fonction* f^Θ *définie sur* Θ*. S'il existe une fonction* f^Ω *définie sur* Ω*, telle que :*

$$f^\Theta(A) = f^\Omega\left(\Psi(A)\right), \quad \forall A \subseteq \Theta. \tag{1.25}$$

les fonctions f^Θ *et* f^Ω *sont alors dites consistantes relativement à* Ψ *; de même, toute fonction* g *associée à* f *peut être qualifiée de* f*-consistante relativement à* Ψ*.*

En particulier, les fonctions de masse m^Θ et m^Ω, associées respectivement à deux fonctions de croyance bel^Θ et bel^Ω vérifiant (1.24), sont *bel*-consistantes. Remarquons que les différents types de consistance ne sont en général pas équivalents.

1.2.1 Déconditionnement

Soit $m^\Omega[B]$ une fonction de masse conditionnelle à $B \subseteq \Omega$. Il existe généralement plusieurs fonctions de masse dont le conditionnement est égal à $m^\Omega[B]$; le PMI permet de calculer la moins informative d'entre elles.

Définition 1.16 (déconditionnement) *Soit* $m^\Omega[B]$ *une fonction de masse conditionnelle à* $B \subseteq \Omega$*. Le déconditionnement de* $m^\Omega[B]$ *sur* Ω*, noté* $m^{B\Uparrow\Omega}$*, est défini par :*

$$\begin{cases} m^{B\Uparrow\Omega}(A \cup \overline{B}) &= m^\Omega[B](A) & \text{pour tout } A \subseteq B, \\ m^{B\Uparrow\Omega}(C) &= 0 & \text{pour tout } C \subset \overline{B}, C \neq \emptyset. \end{cases} \tag{1.26}$$

La fonction de masse $m^{B\Uparrow\Omega}$ ainsi obtenue est la moins informative de toutes les fonctions de masse dont le conditionnement est égal à $m^{\Omega}[B]$, c'est-à-dire m-consistantes avec $m^{\Omega}[B]$, relativement à l'opérateur de conditionnement.

Intuitivement, cette transformation consiste à transférer la masse allouée à tout $A \subseteq B$ au plus grand sous-ensemble de Ω qui contient les hypothèses auparavant écartées (soit $\overline{B}$) et dont l'intersection avec B est A. Remarquons que la somme disjonctive présentée au paragraphe 1.1.2 généralise l'opérateur de déconditionnement : chaque masse $m_1(A)$ est en effet transférée aux sur-ensembles $A \cup B$ proportionnellement aux masses de croyance $m_2(B)$. L'opérateur de déconditionnement d'une fonction de masse $m_1[B]$ est donc retrouvé lorsque m_2 est une fonction de masse catégorique définie par $m_2(\overline{B}) = 1$.

La figure 1.2 illustre les transferts de masses lors d'un déconditionnement.

Exemple 1.3 (diagnostic médical, suite) Pour trancher quant à la pathologie présentée par son patient, le médecin ordonne un examen radiologique des poumons. Les résultats permettent d'écarter l'hypothèse d'une tuberculose. Cette connaissance peut être modélisée par la fonction de masse catégorique m_C, définie par $m_C(\{bro, can\}) = 1$. Remarquons que m_C correspond à la négation de la fonction de masse catégorique Bayésienne m_D, définie par $m_D(\{tub\}) = 1$.

Le conditionnement $m_{AB}[\{bro, can\}]$ de m_{AB} par rapport à $\{bro, can\}$ sera noté m_{ABC} pour plus de simplicité. Cette fonction de masse est définie par :

$$\begin{aligned} m_{ABC}(\emptyset) &= 0.5, \\ m_{ABC}(\{bro\}) &= 0.49, \\ m_{ABC}(\{bro, can\}) &= 0.01. \end{aligned}$$

La fonction de masse conditionnelle m^{Ω}_{ABC} est caractérisée par une masse $m^{\Omega}_{ABC}(\emptyset)$ significative, qui représente la croyance du praticien dans le fait que l'affliction de son patient n'est ni une bronchite ni une tuberculose.

Alerté par un tel degré de conflit, le médecin remet en question la validité du cadre de discernement initial : il estime que le patient pourrait aussi souffrir d'une pleurésie ($\{ple\}$), ou encore d'une pneumonie ($\{pne\}$). Cette hypothèse mène à définir un nouveau cadre $\Omega' = \{bro, can, ple, pne, tub\}$. Les diagnostics menés jusqu'à présent ont été faits en considérant que le patient souffre soit d'une bronchite, soit d'un cancer des poumons, soit d'une tuberculose ; la fonction de masse m^{Ω}_{ABC} peut donc être interprétée comme une fonction de masse définie sur Ω' et conditionnelle à $\{bro, can, tub\}$:

$$m^{\Omega}_{ABC} = m^{\Omega'}_{ABC}[\{bro, can, tub\}].$$

Afin de prendre en compte les nouvelles pathologies, sans pour autant remettre en question les analyses déjà menées, m^{Ω}_{ABC} est déconditionnée sur Ω'. Le décon-

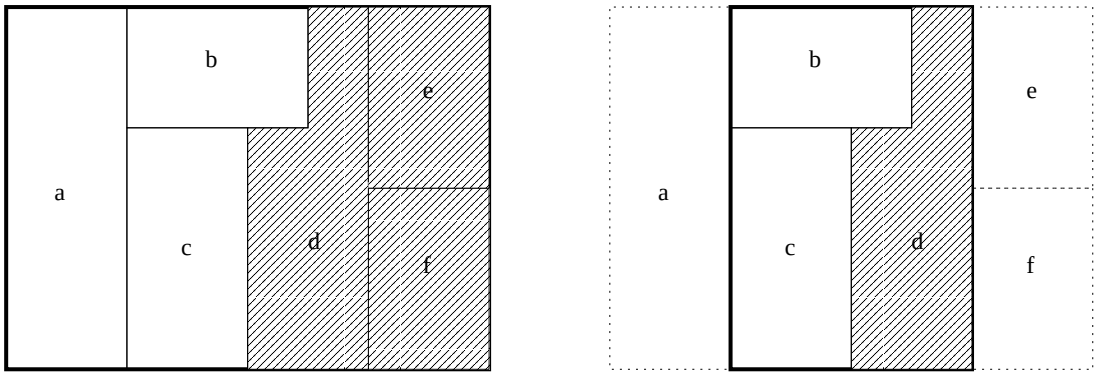


FIG. 1.1 – Conditionnement d'une fonction de masse m^Ω sur $\{b, c, d\}$: la masse $m^\Omega(\{d, e, f\})$ est transférée à $\{d\}$.

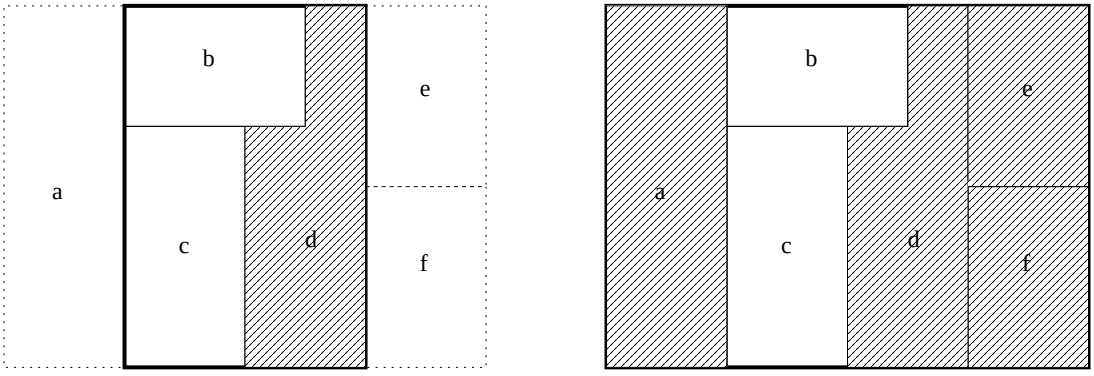


FIG. 1.2 – Déconditionnement sur $\Omega = \{a, b, c, d, e, f\}$ d'une fonction de masse $m^\Omega[\{b, c, d\}]$: la masse $m^\Omega[\{b, c, d\}](\{d\})$ est transférée à $\{a, d, e, f\}$.

ditionnement $m^{\Omega \Uparrow \Omega'}_{ABC}$, noté plus simplement $m^{\Omega'}_{ABC}$, est défini par :

$$\begin{aligned} m^{\Omega'}_{ABC}(\{ple, pne\}) &= 0.5, \\ m^{\Omega'}_{ABC}(\{bro, ple, pne\}) &= 0.49, \\ m^{\Omega'}_{ABC}(\{bro, can, ple, pne\}) &= 0.01. \end{aligned}$$

Ces deux pathologies n'ont encore fait l'objet d'aucun examen spécifique ; le praticien n'a donc pas de croyance particulière les concernant : en particulier, on pourra remarquer que $pl^{\Omega'}_{ABC}(\{ple\}) = pl^{\Omega'}_{ABC}(\{pne\}) = 1$, ce qui illustre le caractère conservatif du processus de déconditionnement. □

1.2.2 Grossissement et raffinement

Définition 1.17 (raffinement et grossissement) *Soient deux cadres Θ et Ω, et une application $\rho : 2^\Theta \to 2^\Omega$ telle que :*

1. *l'ensemble $\{\rho(\{\theta\}), \theta \in \Theta\} \subseteq 2^\Omega$ est une partition de Ω ;*
2. *pour chaque $A \subseteq \Theta$, $\rho(A) = \bigcup_{\theta \in A} \rho(\{\theta\})$.*

L'application ρ est alors appelée raffinement de Θ vers Ω ; par extension, Ω est appelé un raffinement de Θ, et Θ un grossissement de Ω.

Intuitivement, le grossissement Θ d'un cadre Ω est un cadre de discernement dans lequel certains sous-ensembles, distincts dans Ω, sont confondus les uns avec les autres.

La figure 1.3 présente un grossissement $\Theta = \{\theta_1, \theta_2\}$ d'un cadre $\Omega = \{\omega_1, \omega_2, \omega_3, \omega_4\}$, défini par $\rho(\{\theta_1\}) = \{\omega_1, \omega_2\}$, $\rho(\{\theta_2\}) = \{\omega_3, \omega_4\}$.

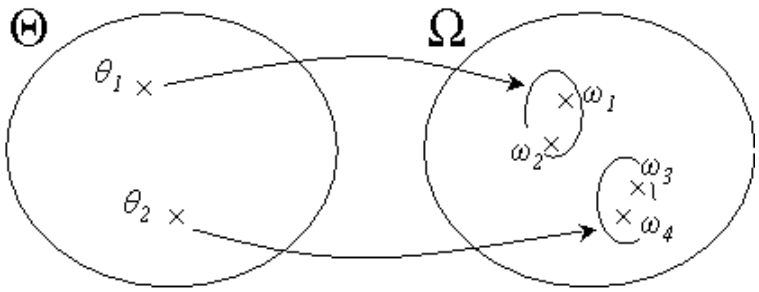


FIG. 1.3 – Grossissement $\Theta = \{\theta_1, \theta_2\}$ d'un cadre $\Omega = \{\omega_1, \omega_2, \omega_3, \omega_4\}$.

L'application ρ n'est généralement pas surjective : il peut exister des éléments $B \subseteq \Omega$ qui ne sont pas image par ρ d'un élément $A \subseteq \Theta$. Il y a donc plusieurs transformations permettant d'associer un élément $B \subseteq \Omega$ à un sous-ensemble de Θ ; deux d'entre elles présentent un intérêt particulier.

Définition 1.18 (réductions intérieure et extérieure d'un élément) *Soit* $\underline{\theta}(B) \subseteq \Theta$ *le plus grand sous-ensemble de* Θ *dont l'image par* ρ *est incluse dans* B *:*

$$\underline{\theta}(B) = \{\theta \in \Theta : \rho(\{\theta\}) \subseteq B\}. \tag{1.27}$$

Le sous-ensemble $\underline{\theta}(B)$ *est appelé* réduction intérieure *de* B *sur* Θ*. La réduction intérieure de* B *sur* Ω *est définie par* $\rho(\underline{\theta}(B))$*.*

Soit $\overline{\theta}(B) \subseteq \Theta$ *le plus petit sous-ensemble de* Θ *dont l'image par* ρ *inclut* B *:*

$$\overline{\theta}(B) = \{\theta \in \Theta : \rho(\{\theta\}) \cap B \neq \emptyset\}. \tag{1.28}$$

Le sous-ensemble $\overline{\theta}(B)$ *est appelé* réduction extérieure *de* B *sur* Θ*. La réduction extérieure de* B *sur* Ω *est définie par* $\rho(\overline{\theta}(B))$*.*

Exemple 1.4 (diagnostic médical, suite) Le médecin souhaite distinguer les infections des poumons ($\{ip\}$) comme la pneumonie ou la tuberculose, des pathologies touchant d'autres parties du système respiratoire ($\{sr\}$) comme la bronchite ou la pleurésie, et du cancer des poumons ($\{cn\}$). Cette partition de Ω' correspond au grossissement Θ défini par :

$$\begin{array}{rclcrcl} \Theta & = & \{ip, sr, cn\}, & & \rho(\{cn\}) & = & \{can\}, \\ \rho(\{ip\}) & = & \{pne, tub\}, & & \rho(\{sr\}) & = & \{bro, ple\}. \end{array}$$

□

Ces transformations peuvent être étendues aux fonctions de croyance, en associant à tout élément focal d'une fonction de masse m^Ω sa réduction intérieure ou extérieure sur Θ.

Définition 1.19 (réductions intérieure et extérieure d'une fonction de masse) *La réduction intérieure d'une fonction de masse* m^Ω *sur* Θ*, notée* $\underline{m}^\Theta$*, est définie par :*

$$\underline{m}^\Theta(A) = \sum_{B \subseteq \Omega, \underline{\theta}(B) = A} m^\Omega(B), \quad \forall A \subseteq \Theta; \tag{1.29}$$

La réduction extérieure d'une fonction de masse m^Ω *sur* Θ*, notée* $\overline{m}^\Theta$*, est définie par :*

$$\overline{m}^\Theta(A) = \sum_{B \subseteq \Omega, \overline{\theta}(B) = A} m^\Omega(B), \quad \forall A \subseteq \Theta. \tag{1.30}$$

Les figures 1.5 et 1.6 illustrent les transferts de masses respectivement lors d'une réduction intérieure et lors d'une réduction extérieure, sur un grossissement Θ de Ω présenté sur la figure 1.4.

Propriété 1.4 *Soit* $m^{\Omega *}$ *une fonction de masse normale définie sur* Ω*. La réduction extérieure* $\overline{m}^{\Theta *}$ *de* $m^{\Omega *}$ *sur* Θ *vérifie la propriété de consistance définie par l'équation (1.24) :*

$$\overline{bel}^{\Theta *}(A) = bel^{\Omega *}(\rho(A)), \quad \forall A \subseteq \Theta.$$

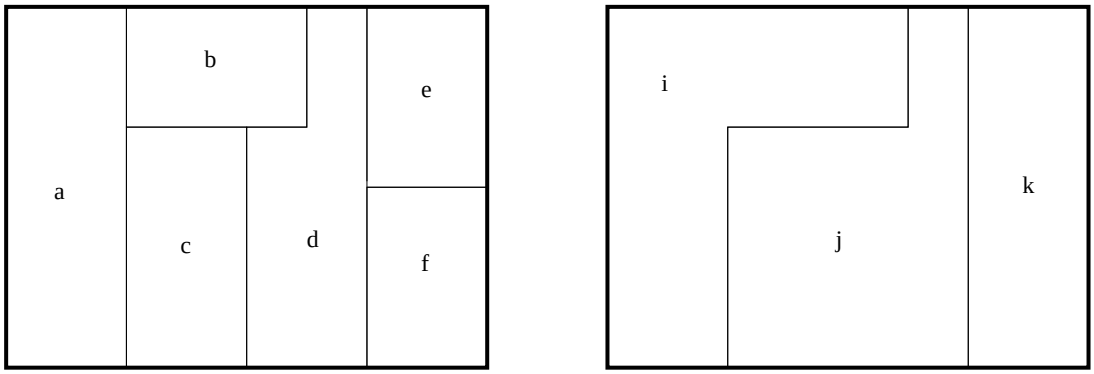


FIG. 1.4 – Cadre $\Omega = \{a, b, c, d, e, f\}$ (gauche) et grossissement $\Theta = \{i, j, k\}$ de Ω (droite), défini par $\rho(\{i\}) = \{a, b\}$, $\rho(\{j\}) = \{c, d\}$, $\rho(\{k\}) = \{e, f\}$.

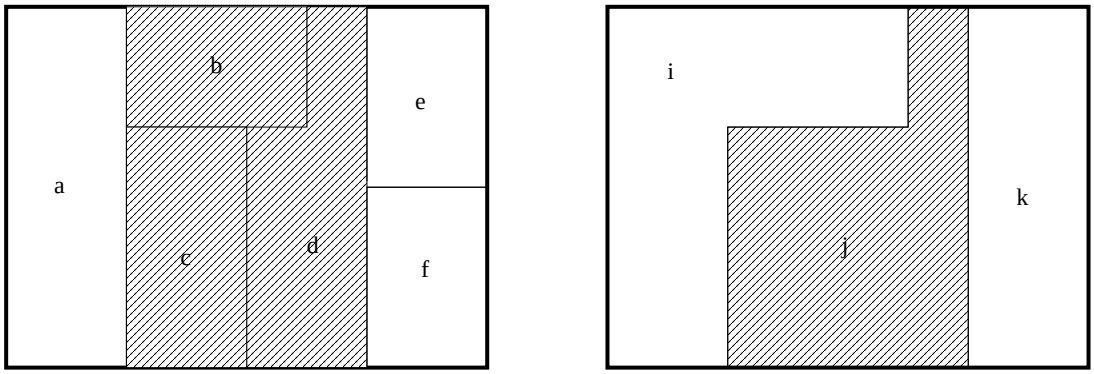


FIG. 1.5 – Réduction intérieure de m^Ω sur Θ : la masse $m^\Omega(\{b, c, d\})$ est transférée à $\{j\}$.

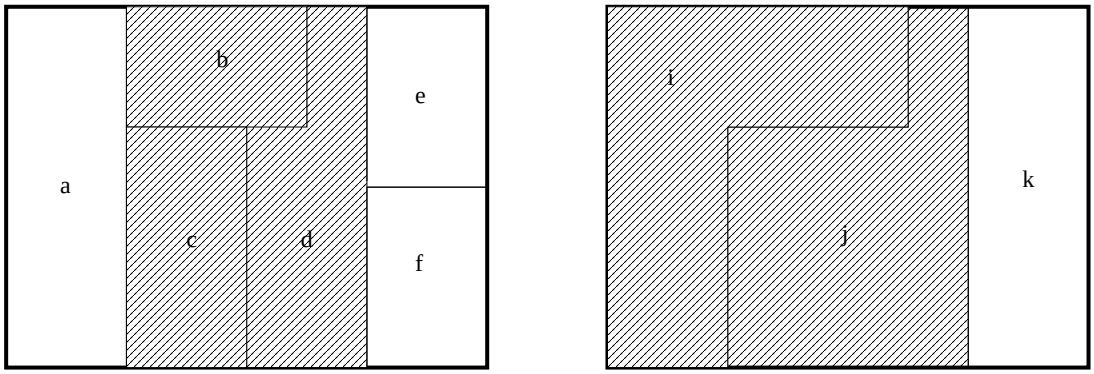


FIG. 1.6 – Réduction extérieure de m^Ω sur Θ : la masse $m^\Omega(\{b, c, d\})$ est transférée à $\{i, j\}$.

Pour cette raison, $\overline{m}^\Theta$ est parfois appelée *restriction* de m^Ω sur Θ [51, page 126]. Cette propriété peut être étendue à toute fonction de masse, non-nécessairement normale. La réduction intérieure vérifie une propriété similaire.

Propriété 1.5 *Une fonction de masse m^Ω définie sur Ω et sa réduction extérieure $\overline{m}^\Theta$ sur un grossissement Θ de Ω sont b-consistantes :*

$$\overline{b}^\Theta(A) = b^\Omega(\rho(A)), \quad \forall A \subseteq \Theta.$$

De même, une fonction de masse m^Ω et sa réduction intérieure $\underline{m}^\Theta$ sont q-consistantes :

$$\underline{q}^\Theta(A) = q^\Omega(\rho(A)), \quad \forall A \subseteq \Theta.$$

Preuve : Par définition de la réduction extérieure,

$$\begin{aligned}
\overline{b}^\Theta(A) &= \sum_{C \subseteq A} \overline{m}^\Theta(C), && (1.31)\\
&= \sum_{C \subseteq A} \sum_{\substack{B \subseteq \Omega \\ C = \overline{\theta}(B)}} m^\Omega(B), && (1.32)\\
&= \sum_{\substack{B \subseteq \Omega \\ \overline{\theta}(B) \subseteq A}} m^\Omega(B), && (1.33)\\
&= \sum_{B \subseteq \rho(A)} m^\Omega(B), && (1.34)\\
&= b^\Omega(\rho(A)). && (1.35)
\end{aligned}$$

De manière similaire, pour la réduction intérieure :

$$\begin{aligned}
\underline{q}^\Theta(A) &= \sum_{A \subseteq C} \underline{m}^\Theta(C), && (1.36)\\
&= \sum_{A \subseteq C} \sum_{\substack{B \subseteq \Omega \\ C = \underline{\theta}(B)}} m^\Omega(B), && (1.37)\\
&= \sum_{\substack{B \subseteq \Omega \\ A \subseteq \underline{\theta}(B)}} m^\Omega(B), && (1.38)\\
&= \sum_{\rho(A) \subseteq B} m^\Omega(B), && (1.39)\\
&= q^\Omega(\rho(A)). && (1.40)
\end{aligned}$$

Soit une fonction de masse m^Θ ; il existe généralement plusieurs fonctions de masse définies sur Ω qui sont m-consistantes avec m^Θ, c'est-à-dire dont la réduction (intérieure ou extérieure) sur Θ est exactement m^Θ. En accord avec le PMI, on associe alors à m^Θ la moins informative d'entre elles.

Définition 1.20 (extension vide) *Soit une fonction de masse m^Θ définie sur Θ. L'extension vide [51] de m^Θ sur Ω, notée $m^{\Theta\uparrow\Omega}$, est définie pour tout $B \subseteq \Omega$ par :*

$$m^{\Theta\uparrow\Omega}(B) = \begin{cases} m^\Theta(A) & \text{si } B = \rho(A),\ A \subseteq \Theta \\ 0 & \text{sinon.} \end{cases} \tag{1.41}$$

C'est la moins informative des fonctions de masse définies sur Ω, qui sont m-consistantes avec m^Θ.

Par commodité, les extensions vides $\underline{m}^{\Theta\uparrow\Omega}$ et $\overline{m}^{\Theta\uparrow\Omega}$ des réductions intérieure et extérieure sur Ω seront notées $\underline{m}^\Omega$ et $\overline{m}^\Omega$, respectivement.

Propriété 1.6 *Les réductions intérieure $\underline{m}^\Omega$ et extérieure $\overline{m}^\Omega$ sur Ω vérifient [14] :*

$$\underline{m}^\Omega \subseteq m^\Omega \subseteq \overline{m}^\Omega. \tag{1.42}$$

Exemple 1.5 (diagnostic médical, suite et fin) Les réductions intérieure $\underline{m}^\Theta_{ABC}$ et extérieure $\overline{m}^\Theta_{ABC}$ de la fonction de masse $m^{\Omega'}_{ABC}$ sur Θ, ainsi que les fonctions de croyance qui leur sont associées, sont représentées dans les tableaux 1.2 et 1.3, respectivement.

La fonction de masse $\underline{m}^\Theta_{ABC}$ est clairement une spécialisation de $\overline{m}^\Theta_{ABC}$: elle peut être obtenue en transférant une masse de 0.5 de $\{ip, sr\}$ à $\emptyset$, une masse de 0.49 de $\{ip, sr\}$ à $\{sr\}$, et la totalité de la masse affectée à $\{ip, sr, cn\}$ à $\{sr, cn\}$. On pourra constater en outre que $\underline{pl}^\Theta_{ABC}(A) \leq \overline{pl}^\Theta_{ABC}(A)$ et $\underline{q}^\Theta_{ABC}(A) \leq \overline{q}^\Theta_{ABC}(A)$, pour tout $A \subseteq \Theta$.

La masse affectée à $\{bro, ple, pne\}$, par exemple, est transférée à $\{sr\}$ dans le premier cas, et à $\{ip, sr\}$ dans le second cas. On peut l'interpréter comme une croyance spéficique dans le fait que la maladie appartient « au moins au groupe $\{sr\}$ » dans le premier cas, et « au moins au groupe $\{sr\}$ et peut-être au groupe $\{ip\}$ » dans le second cas. □

1.3 Synthèse

Le Modèle des Croyances Transférables est un cadre théorique, dans lequel une fonction de masse m^Ω représente la connaissance de la valeur d'une variable y à valeurs dans Ω, à différents niveaux de précision. Divers outils permettent de modéliser le caractère partiel que revêt parfois cette connaissance. L'ensemble des valeurs possibles de y peut ainsi être restreint à un sous-ensemble $B \subseteq \Omega$, en conditionnant m^Ω par rapport à B. Par ailleurs, la connaissance modélisée par m^Ω peut être exprimée de manière plus grossière, en réduisant m^Ω sur un grossissement Θ de Ω. Réciproquement, le Principe du Minimum d'Information permet de recouvrer une fonction de masse unique en déconditionnant une fonction de masse conditionnelle, ou en étendant une fonction de masse exprimée sur un cadre grossier.

TAB. 1.2 – Réduction intérieure de $m^{\Omega'}_{ABC}$ sur Θ, et fonctions de croyance associées.

	$\underline{m}_{ABC}$	$\underline{m}^*_{ABC}$	$\underline{bel}_{ABC}$	$\underline{pl}_{ABC}$	$\underline{b}_{ABC}$	$\underline{q}_{ABC}$
$\emptyset$	0.5	0	0	0	0.5	1
$\{ip\}$	0	0	0	0	0.5	0
$\{sr\}$	0.49	0.98	0.49	0.5	0.99	0.5
$\{ip, sr\}$	0	0	0.49	0.5	0.99	0
$\{cn\}$	0	0	0	0.01	0.5	0.01
$\{ip, cn\}$	0	0	0	0.01	0.5	0
$\{sr, cn\}$	0.01	0.02	0.5	0.5	1	0.01
$\{ip, sr, cn\}$	0	0	0.5	0.5	1	0

TAB. 1.3 – Réduction extérieure de $m^{\Omega'}_{ABC}$ sur Θ, et fonctions de croyance associées.

	$\overline{m}_{ABC}$	$\overline{m}^*_{ABC}$	$\overline{bel}_{ABC}$	$\overline{pl}_{ABC}$	$\overline{b}_{ABC}$	$\overline{q}_{ABC}$
$\emptyset$	0	0	0	0	0	1
$\{ip\}$	0	0	0	1	0	1
$\{sr\}$	0	0	0	1	0	1
$\{ip, sr\}$	0.99	0.99	0.99	1	0.99	1
$\{cn\}$	0	0	0	0.01	0	0.01
$\{ip, cn\}$	0	0	0	1	0	0.01
$\{sr, cn\}$	0	0	0	1	0	0.01
$\{ip, sr, cn\}$	0.01	0.01	1	1	1	0.01

Remarquons que ce formalisme de représentation de la connaissance généralise le modèle Bayésien : une distribution de probabilité peut être modélisée par une fonction de croyance Bayésienne, et le conditionnement Bayésien est un cas particulier de l'opérateur de conditionnement. Toutefois, un nombre important de notions, dont le principe de déconditionnement, qui découle de l'utilisation du Principe du Minimum d'Information, ainsi que le grossissement, le raffinement et les notions qui leur sont associées, sont propres au Modèle des Croyances Transférables.

Chapitre 2

Combinaison de classifieurs

2.1 Introduction

Dans un problème classique de reconnaissance des formes, on dispose d'un ensemble d'apprentissage composé de vecteurs (ou formes) $\mathbf{x}_p \in \mathbb{R}^d$ $(p = 1, \ldots, P)$, associés à une étiquette y_p à valeurs dans $\Omega = \{\omega_1, \ldots, \omega_K\}$. Un vecteur $\mathbf{x}_p$ peut être vu comme la réalisation d'une variable aléatoire multidimensionnelle (de loi généralement inconnue) ; l'étiquette y_p indique sa classe d'appartenance. On cherche alors à construire un classifieur, c'est-à-dire un algorithme permettant de séparer l'espace des formes en régions correspondant aux classes, dans le but de prédire la classe de tout vecteur $\mathbf{x}$ inconnu.

On distingue généralement les problèmes de classification binaires, où une classe (dite positive) doit être séparée d'une autre (dite négative), des problèmes de classification multi-classes, pour lesquels $|\Omega| > 2$. Ces derniers sont généralement plus difficiles à résoudre : la répartition des classes dans l'espace des formes peut rendre le calcul des frontières complexe, comme l'illustre la figure 2.1. La complexité d'un classifieur doit être adaptée à celle du problème abordé ; dans le cas de problèmes avec de nombreuses classes, aux frontières non-linéaires, le coût d'apprentissage des classifieurs peut devenir important.

Remarquons qu'un certain nombre de classifieurs ont une formulation simple dans le cas de problèmes binaires : c'est notamment le cas de la régression logistique [27], des arbres de décision binaires [7, 44], ou des séparateurs à vaste marge (SVM) [59, 6]. En conséquence, plutôt que de résoudre un problème multi-classes au moyen d'un unique classifieur, une méthode alternative consiste à le décomposer en sous-problèmes binaires (ou dichotomies), résolus au moyen de classifieurs binaires, puis à combiner les résultats ainsi obtenus. La fusion de capteurs hétérogènes, où l'indépendance des classifieurs combinés est généralement admise, constitue un thème de recherche différent bien que voisin. De même, nous distinguons le problème abordé dans ce mémoire de la « fusion de classifieurs »[37], qui vise à améliorer la qualité des performances de classification, en tirant parti

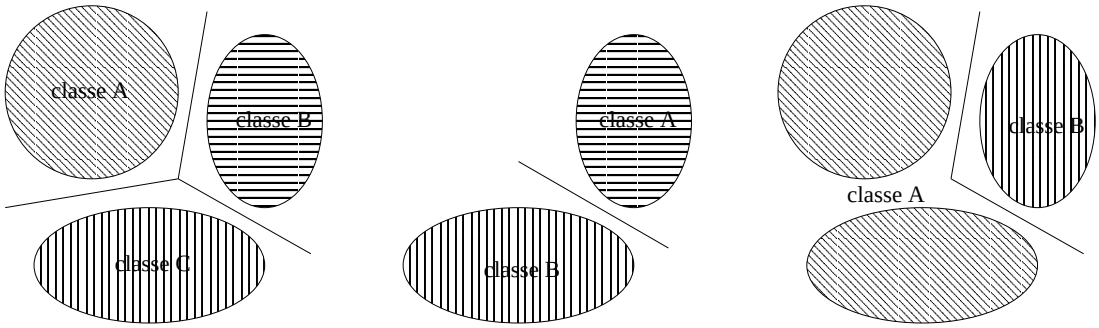


FIG. 2.1 – Un problème de classification multi-classes, à la frontière de décision non linéaire (gauche) ; un problème binaire, avec une frontière linéaire (centre) ; un problème binaire, avec une frontière non linéaire (droite).

de la complémentarité de classifieurs entraînés à résoudre un même problème ; ce domaine de recherche ne sera pas abordé.

La méthode de combinaison mise en œuvre dépend naturellement de l'algorithme d'apprentissage employé, comme du schéma de décomposition du problème initial. On distingue parfois les classifieurs suivant la nature de leurs sorties [62].

Définition 2.1 (Classifieur de type I) *Un classifieur de type I fournit une décision δ quant à la classe de $\mathbf{x}$, ou une valeur réelle f dont la magnitude indique une classe.*

Par exemple, un réseau de neurones fournit K sorties $z_1, \ldots, z_K$ et z_k est d'autant plus importante que $\mathbf{x}$ est proche de la classe ω_k ; un SVM binaire linéaire fournit une valeur $f(\mathbf{x})$ d'autant plus grande (respectivement, faible) que la distance entre $\mathbf{x}$ et la région associée à la classe négative (respectivement, positive) est importante, $\mathbf{x}$ étant affecté à la classe positive si $f(\mathbf{x}) > 0$, et à la classe négative sinon.

Définition 2.2 (Classifieur de type II) *Un classifieur de type II fournit une liste d'étiquettes $\delta_1, \ldots, \delta_k$, interclassées par ordre de préférence relativement à l'appartenance de $\mathbf{x}$.*

Définition 2.3 (Classifieur de type III) *Un classifieur de type III fournit une mesure de confiance (probabilité, possibilité, fonction de croyance) sur l'appartenance de $\mathbf{x}$ aux classes.*

Par exemple, la régression logistique permet d'estimer les probabilités d'appartenance a posteriori $\mathbb{P}(\omega_k|\mathbf{x})$.

Dans ce chapitre, nous présentons trois schémas de décomposition généralement employés ; pour chacun, nous passons en revue les méthodes de combinaison

proposées dans la litérature, en nous intéressant plus particulièrement aux classifieurs de type III.

2.2 Combinaison de classifieurs dans le cas d'une décomposition 1-1

Définition 2.4 (Décomposition 1-1) *Le schéma de décomposition un-contre-un (ou 1-1) consiste à former C_K^2 dichotomies en opposant chaque classe à chaque autre [34, 24].*

Un ensemble de C_K^2 classifieurs $\mathcal{E}_{ij}$ est donc construit ; par convention, ces classifieurs sont définis pour tout $j > i$ (pour chaque paire de classes). Remarquons que chaque dichotomie ne faisant intervenir que deux classes, le nombre de vecteurs à traiter est moins important que lorsque toutes les classes sont considérées. Par ailleurs, la frontière de décision entre les classes positive et négative peut parfois être approchée par un hyperplan. Par conséquent, suivant l'algorithme de classification employé et le nombre de classes du problème considéré, le coût d'apprentissage global peut être réduit de manière significative par rapport au problème initial. Toutefois, ce gain de complexité a un prix : un classifieur $\mathcal{E}_{ij}$, entraîné à séparer ω_i de ω_j, n'a une connaissance incomplète de Ω : les informations correspondant aux classes ω_k, $k \notin \{i, j\}$ ont été laissées de côté lors de son apprentissage.

2.2.1 Combinaison 1-1 de classifieurs de type I

Règles de vote et opérateurs d'aggrégation simples

Soient $\mathcal{E}_{ij}$ le classifieur entraîné à séparer la classe ω_i de la classe ω_j, $f_{ij}(\mathbf{x})$ la sortie de $\mathcal{E}_{ij}$ lors de l'évaluation de $\mathbf{x}$, $\delta_{ij}(\mathbf{x})$ la décision prise par $\mathcal{E}_{ij}$ pour la classe ω_i, et $\delta_{ji}(\mathbf{x}) = 1 - \delta_{ij}(\mathbf{x})$ la décision prise par $\mathcal{E}_{ij}$ pour ω_j :

- si $f_{ij}(\mathbf{x}) > 0$, $\delta_{ij}(\mathbf{x}) = 1$ et $\delta_{ji}(\mathbf{x}) = 0$: on choisit ω_i plutôt que ω_j ;
- sinon, $\delta_{ij}(\mathbf{x}) = 0$ et $\delta_{ji}(\mathbf{x}) = 1$: on choisit ω_j plutôt que ω_i.

Dans [34], la contrainte $\delta_{ji}(\mathbf{x}) = 1 - \delta_{ij}(\mathbf{x})$ est relaxée. Soit θ un seuil fixé :

- si $f_{ij}(\mathbf{x}) > \theta$, $\delta_{ij}(\mathbf{x}) = 1$ et $\delta_{ji}(\mathbf{x}) = 0$: la classe ω_i est retenue ;
- si $f_{ij}(\mathbf{x}) < -\theta$, $\delta_{ij}(\mathbf{x}) = 0$ et $\delta_{ji}(\mathbf{x}) = 1$: la classe ω_j est retenue ;
- si $f_{ij}(\mathbf{x}) \in [-\theta; \theta]$, $\delta_{ij}(\mathbf{x}) = \delta_{ji}(\mathbf{x}) = 0$: les deux classes sont écartées, le vecteur $\mathbf{x}$ est rejeté.

Les δ_{ij} sont ensuite combinées avec un opérateur logique ET : pour chaque classe ω_i, on définit $\delta_i(\mathbf{x}) = 1$ si $\delta_{ij}(\mathbf{x}) = 1$ pour tout $j \neq i$, et $\delta_i(\mathbf{x}) = 0$ sinon. S'il existe ω_i telle que $\delta_i(\mathbf{x}) = 1$, $\mathbf{x}$ lui est affecté ; sinon, il est rejeté.

Dans [24, 36], les δ_{ij} sont combinées par vote : $\mathbf{x}$ est affecté à la classe ω_k vérifiant

$$k = \arg \max_{i=1,\ldots,K} \sum_{j \neq i} \delta_{ij}(\mathbf{x}). \tag{2.1}$$

Règle de vote négatif, associée à un modèle probabiliste

Cutzu [10] propose une autre interprétation de la sortie $\delta_{ij}(\mathbf{x})$ de $\mathcal{E}_{ij}$:

$$\begin{cases} \delta_{ij}(\mathbf{x}) = 1 & \text{revient à rejeter } \omega_j, \\ \delta_{ij}(\mathbf{x}) = 0 & \text{revient à rejeter } \omega_i. \end{cases}$$

Un modèle de probabilités a posteriori est alors proposé, pour tout $j \neq i$. Soient $\mathbb{P}(\omega_k)$ la probabilité a priori de la classe ω_k, et ϵ_{ji} un facteur proportionnel à la probabilité que $\mathcal{E}_{ij}$ fournisse une sortie $\delta_{ij}(\mathbf{x}) = 0$ en évaluant un vecteur $\mathbf{x}$ dont la vraie classe est ω_i :

$$\mathbb{P}(\omega_i|\delta_{ij}(\mathbf{x}) = 0) = \epsilon_{ji}\mathbb{P}(\omega_i), \tag{2.2}$$

$$\mathbb{P}(\omega_k|\delta_{ij}(\mathbf{x}) = 0) = \frac{\mathbb{P}(\omega_k)\,(1 - \epsilon_{ji}\mathbb{P}(\omega_i))}{1 - \mathbb{P}(\omega_i)}, \text{ pour tout } k \neq i\ . \tag{2.3}$$

Le facteur ϵ_{ji} permet de déterminer la probabilité que le classifieur $\mathcal{E}_{ij}$ vote en faveur de la classe ω_i lors de l'évaluation d'un individu $\mathbf{x} \in \omega_j$; de manière similaire, ϵ_{ij} permet d'obtenir $\mathbb{P}(\omega_j|\delta_{ij}(\mathbf{x}) = 1)$.

Lorsque les $\mathcal{E}_{ij}$ fournissent une sortie continue $f_{ij}(\mathbf{x}) \in [0; 1]$, il est proposé d'interpoler entre les cas limites $\mathbb{P}(\omega_k|\delta_{ij}(\mathbf{x}) = 0)$ et $\mathbb{P}(\omega_k|\delta_{ij}(\mathbf{x}) = 1)$ pour obtenir $\mathbb{P}(\omega_k|\delta_{ij}(\mathbf{x}) \in [0; 1])$:

$$\begin{aligned}
\mathbb{P}(\omega_i|\delta_{ij}(\mathbf{x})) &= (1 - f_{ij}(\mathbf{x}))\epsilon_{ji}\mathbb{P}(\omega_i) + f_{ij}(\mathbf{x})\frac{\mathbb{P}(\omega_i)(1 - \epsilon_{ji}\mathbb{P}(\omega_j))}{1 - \mathbb{P}(\omega_j)}, \\
\mathbb{P}(\omega_j|f_{ij}(\mathbf{x})) &= f_{ij}(\mathbf{x})\epsilon_{ji}\mathbb{P}(\omega_j) + (1 - f_{ij}(\mathbf{x}))\frac{\mathbb{P}(\omega_j)(1 - \epsilon_{ji}\mathbb{P}(\omega_i))}{1 - \mathbb{P}(\omega_i)}, \\
\mathbb{P}(\omega_k|f_{ij}(\mathbf{x})) &= (1 - f_{ij}(\mathbf{x}))\frac{\mathbb{P}(\omega_k)(1 - \epsilon_{ji}\mathbb{P}(\omega_i))}{1 - \mathbb{P}(\omega_i)} + \\
&\quad f_{ij}(\mathbf{x})\frac{\mathbb{P}(\omega_k)(1 - \epsilon_{ji}\mathbb{P}(\omega_j))}{1 - \mathbb{P}(\omega_j)}, \quad \forall k \notin \{i, j\}.
\end{aligned}$$

Dans le cas de sorties discrètes, les probabilités a posteriori sont obtenues en remplaçant les probabilités $\mathbb{P}(\omega_k|f_{ij}(\mathbf{x}))$ par les expressions définies par les relations 2.2 et 2.3. On obtient alors :

$$\begin{aligned}
\log \mathbb{P}(\omega_i|\mathbf{x}) = c + \log \mathbb{P}(\omega_i) + \sum_{j \neq i} \log \left(\epsilon_{ji}(1 - \delta_{ij}(\mathbf{x})) + \frac{1 - \epsilon_{ji}\mathbb{P}(\omega_j)}{1 - \mathbb{P}(\omega_j)}\delta_{ij}(\mathbf{x}) \right) \\
+ \sum_{k,j \neq i} \log \left(\frac{1 - \epsilon_{kj}\mathbb{P}(\omega_j)}{1 - \mathbb{P}(\omega_j)}\delta_{kj}(\mathbf{x}) + \frac{1 - \epsilon_{jk}\mathbb{P}(\omega_k)}{1 - \mathbb{P}(\omega_k)}(1 - \delta_{kj}(\mathbf{x})) \right).
\end{aligned} \tag{2.4}$$

Dans le cas de sorties continues, les probabilités a posteriori sont obtenues en remplaçant δ_{ij} par f_{ij}, pour tout $i \neq j$. Si toutes les classes ont même probabilité a priori $\mathbb{P}(\omega_k) = 1/K$ et si tous les ϵ_{ji} sont égaux, la décision prise est équivalente à celle obtenue par vote [24].

Transformation des décisions en fonctions d'appartenance

Abe et Inoue [1] proposent une méthode de combinaison 1-1 de machines à vecteurs supports (SVM), où la sortie $f_{ij}(\mathbf{x})$ d'un SVM $\mathcal{E}_{ij}$ évaluant le vecteur $\mathbf{x}$ est transformée en une fonction d'appartenance $\eta_{ij}(\mathbf{x})$ de $\mathbf{x}$ à ω_i :

$$\eta_{ij}(\mathbf{x}) = \begin{cases} 1 & \text{pour } f_{ij}(\mathbf{x}) \geq 1, \\ f_{ij}(\mathbf{x}) & \text{sinon .} \end{cases} \tag{2.5}$$

La fonction d'appartenance $\eta_i(\mathbf{x})$ du vecteur $\mathbf{x}$ à chaque classe ω_i est ensuite calculée au moyen de l'opérateur min :

$$\eta_i(\mathbf{x}) = \min_{j=1,\ldots,K} \eta_{ij}(\mathbf{x}), \tag{2.6}$$

et le vecteur $\mathbf{x}$ est affecté à la classe ω_k pour laquelle $\eta_k(\mathbf{x})$ est maximale.

Construction d'un graphe orienté acyclique

La méthode proposée dans [41] consiste à construire un graphe orienté acyclique, dont chaque nœud interne est associé à un classifieur $\mathcal{E}_{ij}$, et dont chacune des feuilles correspond à une classe. Soient, à l'itération n, $\mathcal{L}^n$ la liste des classes possibles pour $\mathbf{x}$ (à l'initialisation, $\mathcal{L}^0 = \Omega$), et d_n et f_n les indices des classes en début et fin de liste, respectivement. À chaque itération, $\mathbf{x}$ est évalué par le classifieur entraîné à séparer ω_{d_n} de ω_{f_n} :

- si $\delta_{d_n\, f_n} > 0$, ω_{f_n} est éliminée de la liste ; la nouvelle liste est $\mathcal{L}^{n+1} = \{\omega_{d_n}, \ldots, \omega_{f_n - 1}\}$: à l'étape suivante, $\mathbf{x}$ sera évalué par le classifieur $\mathcal{E}_{d_n\, f_n - 1}$;
- sinon, on a $\mathcal{L}^{n+1} = \{\omega_{d_n + 1}, \ldots, \omega_{f_n}\}$: $\mathbf{x}$ sera alors évalué par le classifieur $\mathcal{E}_{d_n + 1\, f_n}$.

Après $K - 1$ étapes d'évaluation et de rejet, une feuille du graphe est atteinte, et $\mathbf{x}$ est affecté à la classe correspondante. Cette méthode a fait l'objet de travaux ultérieurs [33, 56].

2.2.2 Combinaison 1-1 de classifieurs de type III

Soient $\mathcal{E}_{ij}$ le classifieur entraîné à séparer la classe ω_i de la classe ω_j, $r_{ij}(\mathbf{x})$ la sortie fournie par $\mathcal{E}_{ij}$ lors de l'évaluation d'un vecteur $\mathbf{x}$, et $r_{ji}(\mathbf{x}) = 1 - r_{ij}(\mathbf{x})$. Soient $\mathbb{P}(\cdot|\mathbf{x})$ la distribution de probabilité a posteriori quantifiant la connaissance de la classe de $\mathbf{x}$, $p_i = \mathbb{P}(\omega_i|\mathbf{x})$ et $\mu_{ij} = \mathbb{P}(\omega_i|\omega_i \text{ ou } \omega_j, \mathbf{x})$:

$$\mu_{ij} = \frac{p_i}{p_i + p_j}.$$

La sortie $r_{ij}(\mathbf{x})$ peut être interprétée comme une estimation de μ_{ij} :

$$r_{ij} \simeq \mu_{ij}, \quad \forall j > i. \tag{2.7}$$

Le système défini par l'égalité stricte (2.7) compte K variables et C_K^2 contraintes d'égalité. Par conséquent, la recherche de la distribution de probabilité $\mathbb{P}$ satisfaisant :

$$p_i \geq 0, \quad \forall i = 1, \ldots, K, \tag{2.8}$$

$$\sum_i p_i = 1, \tag{2.9}$$

et vérifiant (2.7), est un problème surdéterminé, qui n'a généralement pas de solution.

Combinaison des probabilités par sélection des sorties des classifieurs

Il est donc proposé dans [47] de sélectionner $K-1$ équations parmi les C_K^2 définies par (2.7) ; en tirant parti de la relation $\mu_{ij}/\mu_{ji} = p_i/p_j$, et en adjoignant les contraintes (2.8)-(2.9), on peut définir un système linéaire dont la résolution permet d'estimer les p_i. Comme cela est souligné dans [42], les résultats obtenus par cette méthode de combinaison sont très dépendants des $K-1$ équations sélectionnées.

Price et al. [42] proposent donc une méthode d'estimation de $\mathbb{P}$ utilisant la totalité des sorties r_{ij} : en considérant que

$$\sum_{j=1, j\neq i}^{K} \mathbb{P}(\{\omega_i, \omega_j\}|\mathbf{x}) - (K-2)\mathbb{P}(\omega_i) = \sum_{j=1}^{K} \mathbb{P}(\omega_j|\mathbf{x}) = 1,$$

et en utilisant l'approximation (2.7), on obtient :

$$p_i = \frac{1}{\sum_{j\neq i} \frac{1}{r_{ij}} - (K-2)}. \tag{2.10}$$

Remarquons qu'il est nécessaire de normaliser les p_i ainsi obtenues pour obtenir une distribution de probabilité.

Combinaison des probabilités par une procédure itérative

La méthode exposée dans [26] consiste à calculer des estimations $\widehat{p_i}$ des p_i, en minimisant la distance de Kullback-Leibler négative pondérée $\mathcal{L}$ entre les μ_{ij} et les r_{ij} fournies par les classifieurs, par une procédure itérative de descente de gradient. Soit $\widehat{\mathbf{p}}$ le vecteur des $\widehat{p_i}$:

$$\widehat{\mathbf{p}} = \arg\min \mathcal{L}(\mathbf{p}), \tag{2.11}$$

sous les contraintes :

$$\sum_i p_i = 1, \tag{2.12}$$

$$p_i \geq 0, \quad \text{pour tout } i \in \{1, \ldots, K\}; \tag{2.13}$$

où le critère $\mathcal{L}$ est défini par :

$$\mathcal{L}(\mathbf{p}) = \sum_{i<j} n_{ij} \left(r_{ij} \log \frac{r_{ij}}{\mu_{ij}} + (1 - r_{ij}) \log \frac{1 - r_{ij}}{1 - \mu_{ij}} \right). \tag{2.14}$$

Cette méthode sera appelée par la suite méthode PCpl. Hastie et Tibshirani remarquent que les p_i peuvent être écrits sous la forme :

$$p_i = \sum_{j \neq i} \left(\frac{p_i + p_j}{K - 1} \right) \left(\frac{p_i}{p_i + p_j} \right). \tag{2.15}$$

En remplaçant $p_i + p_j$ par $2/K$ dans le premier terme et chacun des seconds termes par les r_{ij} correspondants, on obtient des estimations simples des p_i, notées $p_i^\star$, à partir des r_{ij} :

$$p_i^\star = \frac{2}{K(K-1)} \sum_{j \neq i} r_{ij}. \tag{2.16}$$

Il est montré dans [26] que les $p_i^\star$ sont dans le même ordre que les $\widehat{p_i}$: pour tout (i, j), $p_i^\star \geq p_j^\star$ si et seulement si $\widehat{p_i} \geq \widehat{p_j}$. Ils peuvent donc servir de valeurs de départ pour la procédure de minimisation itérative, ou lorsque seules des décisions sont requises.

Combinaison des probabilités par résolution de systèmes linéaire

Dans [61], Wu, Lin et Weng proposent deux méthodes non-itératives permettant d'estimer les p_i, en se basant sur le travail présenté dans [26]. La première (qui sera appelée par la suite méthode PEst1) consiste à ne remplacer que μ_{ij} par r_{ij} dans l'équation (2.15) :

$$p_i = \sum_{j \neq i} \left(\frac{p_i + p_j}{K - 1} \right) r_{ij}, \quad \forall i. \tag{2.17}$$

La théorie des chaînes de Markov permet de montrer que la solution du système défini par (2.17), sous les contraintes (2.8)-(2.9), est l'unique solution d'un système linéaire :

$$Q\mathbf{p} = \mathbf{p}, \tag{2.18}$$

$$\sum_{i=1}^{K} p_i = 1, \tag{2.19}$$

où la matrice Q est définie par :

$$Q(i,j) = \begin{cases} \dfrac{r_{ij}}{K-1} & \text{si } j \neq i, \\ \dfrac{\sum_{k:k\neq i} r_{ik}}{K-1} & \text{si } j = i. \end{cases} \tag{2.20}$$

Wu et al. montrent que la solution $\widehat{\mathbf{p}}$ de ce problème est le minimum global unique d'un problème d'optimisation convexe :

$$\widehat{\mathbf{p}} = \arg\min_{\mathbf{p}} \sum_{i=1}^{K} \left(\sum_{j:j\neq i} r_{ji} p_i - \sum_{j:j\neq i} r_{ij} p_j \right)^2, \tag{2.21}$$

sous les contraintes (2.8)-(2.9). Motivés par cette formulation, les auteurs proposent une autre méthode (appelée par la suite méthode PEst2), pour combiner les r_{ij} :

$$\widehat{\mathbf{p}} = \arg\min_{\mathbf{p}} \frac{1}{2} \sum_{i=1}^{K} \sum_{j:j\neq i} \left(r_{ji} p_i - r_{ij} p_j \right)^2, \tag{2.22}$$

sous les contraintes (2.8)-(2.9). Les auteurs démontrent la redondance des contraintes de positivité (2.8) ; le problème est donc un problème quadratique convexe avec une contrainte d'égalité linéaire. D'après les conditions d'optimalité de Kuhn et Tucker, le minimum global $\widehat{\mathbf{p}}$ de (2.22)-(2.9) peut donc être déterminé en résolvant un système linéaire :

$$\begin{bmatrix} Q & \mathbf{e} \\ \mathbf{e}^\top & 0 \end{bmatrix} \cdot \begin{bmatrix} \mathbf{p} \\ b \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ 1, \end{bmatrix} \tag{2.23}$$

où $\mathbf{e}$ et $\mathbf{0}$ sont des vecteurs de taille $k \times 1$ dont les éléments sont respectivement des 1 et des 0, b est le multiplicateur de Lagrange associé à la contrainte d'égalité (2.9), et Q est la matrice définie par :

$$Q(i,j) = \begin{cases} -r_{ji} r_{ij} & \text{si } i \neq j, \\ \displaystyle\sum_{s:s\neq i} r_{si}^2 & \text{si } i = j. \end{cases} \tag{2.24}$$

Combinaison des probabilités corrigées par l'opérateur moyenne

Comme cela est souligné dans [26], un classifieur 1-1 peut fournir des estimations r_{ij} erronées, s'il n'a pas été entraîné à reconnaître la classe réelle du vecteur $\mathbf{x}$ évalué. Dans [39], il est donc proposé d'entraîner des classifieurs supplémentaires, dits correcteurs, à séparer les classes $\{\omega_i, \omega_j\}$ de l'ensemble des autres, et donc à calculer des estimations q_{ij} des probabilités $\mathbb{P}(\{\omega_i, \omega_j\}|\mathbf{x})$. En remplaçant dans (2.15) le numérateur $p_i + p_j$ par q_{ij}, on obtient des estimations $p_i^\dagger$ des p_i :

$$p_i^\dagger = \frac{1}{K-1} \sum_{j>i} r_{ij} q_{ij}. \tag{2.25}$$

Bien que susceptible d'améliorer la précision du processus de classification, cette méthode (appelée par la suite méthode PEstCorr) nécessite d'entraîner C_K^2 classifieurs supplémentaires sur tout l'ensemble d'apprentissage, ce qui peut être coûteux lorsque K est grand.

2.3 Combinaison de classifieurs dans le cas d'une décomposition 1-T

Définition 2.5 (Décomposition 1-T) *Le schéma de décomposition un-contre-tous (ou 1-T) consiste à former K dichotomies en opposant chaque classe à toutes les autres [8, 3].*

La décomposition un-contre-tous ou 1-T consiste à construire un ensemble de K classifieurs (un pour chaque classe). Le classifieur $\mathcal{E}_k$ est entraîné à séparer ω_k de l'ensemble des autres classes : il est donc construit sur la base de tous les exemples d'apprentissage. Cependant, il a une connaissance incomplète de Ω en ce sens qu'il est incapable de discerner les classes ω_l, $l \neq k$. Les classifieurs sont moins nombreux dans le cas 1-T, mais peuvent être plus coûteux à entraîner. En outre, les frontières de décision sont généralement plus complexes que dans le cas d'une décomposition 1-1. Néanmoins, le traitement de sous-problèmes binaires permet généralement de réduire le coût d'apprentissage par rapport au problème initial.

Une comparaison du coût d'apprentissage global de ces deux méthodes a été proposée par Fürnkranz [25].

2.3.1 Combinaison 1-T de classifieurs de type I

Règles de vote

La règle du vote est la méthode la plus couramment utilisée pour combiner des classifieurs 1-T. Soit $\mathcal{E}_k$ le classifieur entraîné à séparer la classe ω_k de $\overline{\{\omega_k\}}$, et $\delta_k(\mathbf{x})$ la décision fournie lors de l'évaluation d'un vecteur $\mathbf{x}$:

$$\delta_k(\mathbf{x}) = \begin{cases} 1 & \text{revient à affecter } \mathbf{x} \text{ à } \{\omega_k\} \ , \\ 0 & \text{revient à affecter } \mathbf{x} \text{ à } \overline{\{\omega_k\}} \ ; \end{cases} \tag{2.26}$$

le vecteur $\mathbf{x}$ est ainsi affecté à la classe ω_k telle que $\delta_k(\mathbf{x}) = 1$. En cas de conflits (plusieurs classes remportent le nombre maximal de votes), il est proposé dans [8] de recourir à une méthode probabiliste et d'affecter $\mathbf{x}$ à la classe ayant la plus grande probabilité a priori.

Alternativement, la magnitude des sorties des classifieurs binaires peut être utilisée pour résoudre les conflits. Anand et al. [3] proposent de remplacer un réseau de neurones multi-classes par un ensemble de réseaux 1-T ; de manière

similaire, il est proposé dans [6] de résoudre les problèmes multi-classes au moyen de SVM binaires 1-T. Dans les deux cas, $\mathbf{x}$ est affecté à la classe ω_k maximisant $f_k(\mathbf{x})$.

Transformation des décisions en fonctions d'appartenance

Inoue et Abe [30] considèrent la combinaison de SVM, et proposent de résoudre les conflits en transformant les sorties $f_j(\mathbf{x})$ en fonctions d'appartenance. Soit $\eta_{ij}(\mathbf{x})$ la fonction d'appartenance à ω_i, obtenue à partir de la sortie $f_j(\mathbf{x})$ du classifieur $\mathcal{E}_j$ $(j = 1, \ldots, K)$:

$$\begin{aligned} \eta_{ii}(\mathbf{x}) &= \begin{cases} 1 & \text{si } f_i(\mathbf{x}) > 1, \\ f_i(\mathbf{x}) & \text{sinon;} \end{cases} \\ \eta_{ij}(\mathbf{x}) &= \begin{cases} 1 & \text{si } f_j(\mathbf{x}) < -1, \\ -f_j(\mathbf{x}) & \text{sinon,} \end{cases} \qquad \text{pour tout } j \neq i. \end{aligned}$$

L'appartenance $\eta_i(\mathbf{x})$ de $\mathbf{x}$ à chaque classe ω_i est ensuite obtenue en combinant les η_{ij} au moyen de l'opérateur min :

$$\eta_i(\mathbf{x}) = \min_{j=1,\ldots,K} \eta_{ij}(\mathbf{x}), \tag{2.27}$$

et $\mathbf{x}$ est affecté à la classe ω_k pour laquelle $\eta_k(\mathbf{x})$ est maximale.

2.3.2 Combinaison 1-T de classifieurs de type III

Vannoorenberghe et Denœux [58] proposent de combiner les sorties d'arbres de décision binaires 1-T au moyen de l'opérateur moyenne. Chaque arbre de décision fournit une fonction de masse $m_k^{\Theta_k}$, définie sur un grossissement $\Theta_k = \left\{\theta_k^+, \theta_k^-\right\}$ de Ω :

$$\begin{aligned} \rho_k\left(\{\theta_k^+\}\right) &= \{\omega_k\}, \\ \rho_k\left(\{\theta_k^-\}\right) &= \overline{\{\omega_k\}}, \end{aligned}$$

où ρ_k est le raffinement transformant Θ_k en Ω. Les fonctions de masse $m_k^{\Theta_k}$ sont ensuite exprimées sur Ω au moyen de l'extension vide (présentée au paragraphe 1.2.2) :

$$\begin{aligned} m_k^{\Omega}(\{\omega_k\}) &= m_k^{\Theta_k}(\{\theta_k^+\}), \\ m_k^{\Omega}(\Omega \setminus \{\omega_k\}) &= m_k^{\Theta_k}(\{\theta_k^-\}), \\ m_k^{\Omega}(\Omega) &= m_k^{\Theta_k}(\Theta_k); \end{aligned}$$

ces fonctions de masse sont ensuite combinées par l'opérateur moyenne :

$$m^{\Omega} = \frac{1}{K} \sum_{k=1}^{K} m_k^{\Omega}. \tag{2.28}$$

2.4 Combinaison de classifieurs dans le cas d'une décomposition par codes correcteurs d'erreurs

Définition 2.6 (Décomposition par codes correcteurs d'erreurs) *La décomposition par codes correcteurs d'erreurs (CCE) [16, 2] consiste à former N dichotomies : la i^e dichotomie est formée d'un ensemble $A_i^+ \subseteq \Omega$ de classes positives et d'un ensemble $A_i^- \subseteq \Omega$ de classes négatives.*

Il est évident que les deux ensembles sont nécessairement disjoints et non vides :

$$A_i^+, A_i^- \neq \emptyset, \tag{2.29}$$
$$A_i^+ \cap A_i^- = \emptyset; \tag{2.30}$$

Dans la version proposée par Dietterich et Bakiri [16], chaque classifieur est entraîné à partir de l'ensemble des exemples d'apprentissage : $A_i^+ \cup A_i^- = \Omega$. Allwein, Schapire et Singer [2] proposent une approche plus générale, en relaxant cette contrainte :

$$A_i^+ \cup A_i^- \subseteq \Omega. \tag{2.31}$$

Définition 2.7 (Matrice de codes) *Une décomposition CCE est caractérisée par une matrice de codes $\mathcal{M} = (e_{ki})$ ($k = 1, \ldots, K$, $i = 1, \ldots, N$), dont la i^e colonne indique le rôle de chaque classe ω_k dans l'ensemble d'apprentissage du classifieur $\mathcal{E}_i$.*

Ainsi, dans l'approche la plus générale définie dans [2], la matrice $\mathcal{M}$ est définie par :

- $e_{ki} = +1$ si $\omega_k \in A_i^+$,
- $e_{ki} = -1$ si $\omega_k \in A_i^-$,
- $e_{ki} = 0$ si $\omega_k \notin A_i^+ \cup A_i^-$.

Une ligne $\mathbf{e}_i$ de $\mathcal{M}$ ($i = 1, \ldots, K$) définit le code associé à la classe ω_i, qui peut être vu comme la signature de cette classe. Lors de l'évaluation d'un vecteur $\mathbf{x}$, un code $\mathbf{e}_\mathbf{x}$ est généré à partir des sorties des classifieurs $\mathcal{E}_1, \ldots, \mathcal{E}_N$, puis comparé aux $\mathbf{e}_i$; le vecteur est affecté à la classe ω_k telle que $d(\mathbf{e}_k, \mathbf{e}_\mathbf{x})$ est minimale, où d est une distance définissant une *fonction de décodage*.

Les propriétés de la matrice de codes déterminent celles du schéma de décomposition associé. En particulier, une distance de Hamming élevée entre deux lignes de la matrice garantit qu'un nombre important de classifieurs sont entraînés à séparer les classes correspondantes, l'une des classes étant classe positive, et l'autre classe négative, lors de l'apprentissage de ces classifieurs. De même, une distance de Hamming élevée entre la colonne de la matrice associée au classifieur $\mathcal{E}_i$ et celle associée au classifieur $\mathcal{E}_j$, de même qu'entre la colonne associée à $\mathcal{E}_i$ et le *complément à un* de celle associée à $\mathcal{E}_j$, est liée à l'indépendance des deux classifieurs : elle marque en effet la différence de composition entre les ensembles

TAB. 2.1 – Matrice de codes 1-1 pour un problème à quatre classes.

	$\mathcal{E}_{12}$	$\mathcal{E}_{13}$	$\mathcal{E}_{14}$	$\mathcal{E}_{23}$	$\mathcal{E}_{24}$	$\mathcal{E}_{34}$
ω_1	+1	+1	+1	0	0	0
ω_2	-1	0	0	+1	+1	0
ω_3	0	-1	0	-1	0	+1
ω_4	0	0	-1	0	-1	-1

TAB. 2.2 – Matrice de codes 1-T pour un problème à quatre classes.

	$\mathcal{E}_1$	$\mathcal{E}_2$	$\mathcal{E}_3$	$\mathcal{E}_4$
ω_1	+1	-1	-1	-1
ω_2	-1	+1	-1	-1
ω_3	-1	-1	+1	-1
ω_4	-1	-1	-1	+1

d'apprentissage des deux classifieurs. Les méthodes les plus récentes de construction d'une matrice de codes [9, 43] ne seront pas présentées dans ce mémoire.

Nous considérons dans ce mémoire le cas le plus général défini par Allwein et al. [2], et nous adoptons par conséquent les notations associées. Dans ce cas, les schémas de décomposition 1-1 et 1-T peuvent être vus comme des cas particuliers du schéma de décomposition CCE. En particulier, on retrouve le premier en construisant toutes les dichotomies telles que $|A_i^+| = |A_i^-| = 1$, et le second en considérant toutes les dichotomies telles que $|A_i^+| = 1$ et $|A_i^-| = K - 1$. Les tableaux 2.1 et 2.2 présentent les matrices de codes respectivement associées aux schémas de décomposition 1-1 et 1-T. On pourra ainsi constater que la matrice associée au schéma 1-1 est caractérisée par une meilleure séparation des lignes et des colonnes que le schéma 1-T. Cela suggère que les classes sont mieux séparées, et les classifieurs moins dépendants les uns des autres, dans le premier cas.

2.4.1 Combinaison CCE de classifieurs de type I

Dietterich et Bakiri proposent dans [16] de générer le code associé à $\mathbf{x}$ en concaténant les décisions $\delta_i \in \{+1, -1\}$ prises par les classifieurs $\mathcal{E}_i$:

$$\mathbf{e}_\mathbf{x} = (\delta_1, \ldots, \delta_N)\,; \tag{2.32}$$

la distance de Hamming $d_H(\mathbf{e}_\mathbf{x}, \mathbf{e}_k)$ entre $\mathbf{e}_\mathbf{x}$ et chacun des $\mathbf{e}_k$ est alors calculée :

$$d_H(\mathbf{e}_\mathbf{x}, \mathbf{e}_k) = \sum_{i=1}^{N} \frac{1 - e_{\mathbf{x}i} e_{ki}}{2}, \quad \forall k = 1, \ldots, K, \tag{2.33}$$

et $\mathbf{x}$ est affecté à la classe ω_k pour laquelle $d_H(\mathbf{e_x}, \mathbf{e}_k)$ est minimale. Dans le cas général où $e_{ki} \in \{+1, 0, -1\}$, on remarquera que tout élément $e_{ki} = 0$ de $\mathcal{M}$ contribue à faire augmenter la distance $d_H(\mathbf{e_x}, \mathbf{e}_k)$ définie par (2.33), quelle que soit la valeur de δ_i.

Allwein et al. [2] remarquent que d_H ne tient pas compte de la magnitude des sorties $f_i(\mathbf{x})$ des $\mathcal{E}_i$. Ils définissent par conséquent une fonction de décodage d_L tenant compte de cette valeur, par :

$$d_L(\mathbf{e_x}, \mathbf{e}_k) = \sum_{i=1}^{N} \frac{1 - f_i(\mathbf{x}) e_{ki}}{2}, \quad \forall k = 1, \ldots, K. \tag{2.34}$$

Cette fonction revient à utiliser (2.33) en générant le code $\mathbf{e_x}$ par concaténation des $f_i(\mathbf{x})$.

2.4.2 Combinaison CCE de classifieurs de type III

Combinaison de classifieurs probabilistes

Passerini, Pontil et Frasconi [40] proposent d'utiliser une fonction de décodage basée sur les probabilités conditionnelles fournies par les classifieurs CCE. Soit $\mathbf{f}$ le vecteur de sorties formé par concanétation des $f_i(\mathbf{x})$.

Soit $\mathcal{O}$ l'ensemble de tous les codes $\mathbf{e}$ possibles ; les probabilités a posteriori des classes peuvent s'écrire :

$$\mathbb{P}(\omega_k|\mathbf{f}) = \sum_{\mathbf{e} \in \mathcal{O}} \mathbb{P}(\omega_k|\mathbf{e_x} = \mathbf{e}, \mathbf{f}) \mathbb{P}(\mathbf{e_x} = \mathbf{e}|\mathbf{f}). \tag{2.35}$$

En supposant que les classes sont indépendantes de $\mathbf{f}$ étant donné un code $\mathbf{e}$, les auteurs obtiennent :

$$\mathbb{P}(\omega_k|\mathbf{f}) = \sum_{\mathbf{e} \in \mathcal{O}} \mathbb{P}(\omega_k|\mathbf{e_x} = \mathbf{e}) \mathbb{P}(\mathbf{e_x} = \mathbf{e}|\mathbf{f}). \tag{2.36}$$

En considérant que les valeurs $e_{ki} = 0$ peuvent correspondre indifféremment à une valeur $e_{\mathbf{x}i} = +1$ ou $e_{\mathbf{x}i} = -1$, l'ensemble $\mathcal{C}_k$ des codes valides pour chaque classe ω_k est alors défini par l'ensemble des codes obtenus par substitutions des valeurs $e_{ki} = 0$ par $+1$ ou -1. Soit $\overline{\mathcal{C}}$ l'ensemble des codes invalides, c'est-à-dire ne correspondant à aucune classe. Les auteurs proposent alors le modèle suivant :

$$\mathbb{P}(\omega_k|\mathbf{e_x} = \mathbf{e}) = \begin{cases} 1 & \text{si } \mathbf{e} \in \mathcal{C}_k, \\ 0 & \text{si } \mathbf{e} \in \mathcal{C}_l, \text{ pour } l \neq k, \\ \frac{1}{K} & \text{si } \mathbf{e} \in \overline{\mathcal{C}}. \end{cases} \tag{2.37}$$

Soit $\mathbb{P}(\overline{\mathcal{C}}) = \sum_{\mathbf{e}\in\overline{\mathcal{C}}} \mathbb{P}(\mathbf{e_x} = \mathbf{e}|\mathbf{f})$ la masse de probabilité correspondant aux codes invalides, et $\alpha = \mathbb{P}(\overline{\mathcal{C}})/K$: cet élément collecte la masse de probabilité allouée aux codes invalides, et constitue donc une mesure du conflit entre les différents classifieurs. Cela induit :

$$\mathbb{P}(\omega_k|\mathbf{f}) = \sum_{\mathbf{e}\in\mathcal{C}_k} \mathbb{P}(\mathbf{e_x} = \mathbf{e}|\mathbf{f}) + \alpha. \tag{2.38}$$

En supposant que chaque élément e_i de $\mathbf{e}$ est conditionnellement indépendant des autres sachant $\mathbf{f}$, et ne dépend que de l'élément f_i de $\mathbf{f}$ correspondant, les probabilités a posteriori sont obtenues par :

$$\mathbb{P}(\omega_k|\mathbf{f}) = \sum_{\mathbf{e}\in\mathcal{C}_k} \prod_{i=1}^{N} \mathbb{P}(e_{\mathbf{x}i} = e_i|f_i) + \alpha. \tag{2.39}$$

La probabilité d'un élément e_i correspondant à une valeur $e_{ki} = 0$ est indépendante de la sortie $f_k(\mathbf{x})$ du classifieur $\mathcal{E}_k$: la classe ω_k ne figure pas dans l'ensemble d'aprentissage de $\mathcal{E}_k$. En supposant que cette probabilité est uniformément distribuée entre les réalisation possibles $+1$ ou -1, les codes valides $\mathbf{e} \in \mathcal{C}_k$ sont tous équiprobables, ce qui amène :

$$\mathbb{P}(\omega_k|\mathbf{f}) = \prod_{i:e_{ki}\neq 0} \mathbb{P}(e_{\mathbf{x}i} = e_{ki}|f_i) + \alpha. \tag{2.40}$$

La fonction de décodage proposée est alors :

$$d_P(\mathbf{e}_k, \mathbf{e_x}) = -\log \mathbb{P}(\omega_k|\mathbf{f}). \tag{2.41}$$

Affecter $\mathbf{x}$ à la classe minimisant $d_P(\mathbf{e}_k, \mathbf{e_x})$ revient à l'affecter à la classe maximisant les estimations des probabilités a posteriori $\mathbb{P}(\omega_k|\mathbf{f})$, obtenues par l'équation (2.40) à partir des probabilités conditionnelles $\mathbb{P}(e_{\mathbf{x}i} = e_{ki}|f_i)$. Cette méthode sera appelée par la suite méthode PEstP.

Zadrozny [65] propose de combiner les sorties de classifieurs CCE probabilistes par une méthode similaire à celle proposée par Hastie et Tibshirani dans le cas 1-1 [26]. Soient $p_k = \mathbb{P}(\omega_k|\mathbf{x})$, $\mathbf{p} = (p_1, \ldots, p_K)^\top$, et q_i la probabilité conditionnelle définie par :

$$\begin{aligned} q_i &= \mathbb{P}\left(A_i^+ | A_i^+ \cup A_i^-, \mathbf{x}\right) && (2.42)\\ &= \frac{\sum_{\omega_k\in A_i^+} p_k}{\sum_{\omega_k\in A_i^+\cup A_i^-} p_k}. && (2.43) \end{aligned}$$

L'auteur propose alors un algorithme itératif, inspiré de celui proposé dans [26], pour calculer une estimation $\widehat{\mathbf{p}}$ de $\mathbf{p}$ minimisant la distance de Kullback-Leibler entre les q_i et les estimations r_i des q_i fournies par les classifieurs :

$$\mathcal{L}(\mathbf{p}) = \sum_{i=1}^{N} n_i \left(r_i \log \frac{r_i}{q_i} + (1 - r_i) \log \frac{1 - r_i}{1 - q_i} \right), \tag{2.44}$$

où n_i représente le nombre de vecteurs $\mathbf{x}$ appartenant aux classes $\omega_k \in A_i^+ \cup A_i^-$.

Huang, Weng et Lin [29] remarquent que l'algorithme proposé dans [65] ne permet pas de toujours déterminer les $\widehat{p_k}$ minimisant (2.44). Ils proposent alors un autre algorithme, permettant de calculer la solution de ce problème par minimisation itérative de la distance de Kullback-Leibler entre les r_i et les q_i. Dans le cas 1-T, le résultat peut être obtenu par résolution d'une équation non-linéaire à une inconnue. Dans le cas 1-1, la procédure se réduit à celle proposée dans [26]. Cette méthode sera appelée par la suite méthode PCplBT.

Combinaison de fonctions de croyance par déconditionnement et somme conjonctive

Dans [31, 32], Janez et Appriou proposent plusieurs méthodes pour combiner des sources d'information, fournissant des fonctions de croyance $m_1^{\Omega_1}, \ldots, m_N^{\Omega_N}$ définies sur des cadres de discernement différents. Les fonctions de masse peuvent être déconditionnées sur Ω :

$$m_i^{\Omega}\left(A \cup (\Omega \setminus \Omega_i)\right) = m_i^{\Omega_i}(A), \quad \forall A \subseteq \Omega_i, \quad \forall \Omega_i. \tag{2.45}$$

Leur combinaison par la somme conjonctive peut alors être effectuée.

Un processus moins conservatif que le déconditionnement peut être utilisé pour exprimer les fonctions de masse sur Ω, en utilisant une connaissance supplémentaire de compatibilité entre les atomes des différents cadres Ω_i et ceux du cadre Ω. Soit $m_i^{\Omega_i}$ une fonction de masse, que l'on cherche à déconditionner sur le cadre commun Ω pour la combiner. Pour tout $\omega \in \Omega_i$, soit $c_i(\omega) \subseteq \Omega \setminus \Omega_i$ l'ensemble des hypothèses manquantes (non présentes dans Ω_i) fortement compatibles avec ω, et $c_i(A) = \bigcup_{\omega \in A} c_i(\omega)$. La fonction de masse $m_i^{\Omega_i}$ est alors définie sur Ω par :

$$m_i^{\Omega}\left(A \cup c_i(A)\right) = m_i^{\Omega_i}(A), \quad \forall A \subseteq \Omega_i, \quad \forall \Omega_i. \tag{2.46}$$

Dans les deux cas, les fonctions de masse exprimées sur Ω sont alors combinées par la somme conjonctive :

$$m^{\Omega} = m_1^{\Omega} \textcircled{\cap} \ldots \textcircled{\cap} m_N^{\Omega}. \tag{2.47}$$

Combinaison de fonctions de croyance par correction des plausibilités

Une méthode dite de correction des plausibilités, permettant de combiner les classifieurs sans effectuer de déconditionnement préalable, est introduite dans [31]. Soient $pl_i^{\Omega_i}$ et $pl_j^{\Omega_j}$ les plausibilités obtenues à partir des sorties des classifieurs $\mathcal{E}_i$ et $\mathcal{E}_j$, respectivement ; soit $\Omega_{i \cup j,c} = \Omega_i \cap \Omega_j$ l'ensemble des hypothèses communes à Ω_i et Ω_j. Cette méthode se base sur une propriété de la règle de combinaison conjonctive normalisée pour les plausibilités :

$$pl_{ij}^{\Omega}(\{\omega_k\}) = K^{\Omega} pl_i^{\Omega}(\{\omega_k\}) pl_j^{\Omega}(\{\omega_k\}), \quad \forall \omega_k \in \Omega,$$

où $K^{\Omega} = 1/(1 - m_{ij}^{\Omega}(\emptyset))$; et sur l'expression de la règle de de conditionnement pour les plausibilités :

$$pl[A](B) = \frac{pl^{\Omega}(B)}{pl^{\Omega}(A)}, \quad \forall B \subseteq A \subseteq \Omega.$$

Ici, $\mathcal{E}_i$ est considéré comme classifieur de référence, et $\mathcal{E}_j$ comme apportant une information supplémentaire correctrice. La fonction de plausibilité pl_{ij}^{Ω} est réexprimée en ce sens. On obtient, pour tout $\omega_k \in \Omega_i \setminus \Omega_{i \cup j,c}$ (hypothèse considérée seulement par $\mathcal{E}_i$) :

$$pl_{ij}^{\Omega}(\{\omega_k\}) = K^{\Omega_i} pl_{ij}^{\Omega}(\Omega_i) pl_j^{\Omega_i}(\{\omega_k\}) pl_i^{\Omega_i}(\{\omega_k\}); \tag{2.48}$$

pour tout $\omega_k \in \Omega_{i \cup j,c}$ (hypothèse commune à $\mathcal{E}_i$ et $\mathcal{E}_j$) :

$$pl_{ij}^{\Omega}(\{\omega_k\}) = K^{\Omega_i} pl_{ij}^{\Omega}(\Omega_i) pl_j^{\Omega_i}(\Omega_{i \cup j,c}) pl_i^{\Omega_i}(\{\omega_k\}) pl_j^{\Omega_j}(\{\omega_k\}); \tag{2.49}$$

pour tout $\omega_k \in \Omega_j \setminus \Omega_{i \cup j,c}$ (hypothèse considérée seulement par $\mathcal{E}_j$) :

$$pl_{ij}^{\Omega}(\{\omega_k\}) = K^{\Omega_i} pl_{ij}^{\Omega}(\Omega_i) \frac{pl_j^{\Omega_i}(\Omega_{i \cup j,c})}{pl_i^{\Omega_j}(\Omega_{i \cup j,c})} pl_i^{\Omega_j}(\{\omega_k\}) \frac{pl_i^{\Omega_i}(\Omega_{i \cup j,c})}{pl_j^{\Omega_j}(\Omega_{i \cup j,c})} pl_j^{\Omega_j}(\{\omega_k\}). \tag{2.50}$$

Les termes $pl_j^{\Omega_i}(\{\omega_k\})$, $pl_j^{\Omega_i}(\Omega_{i \cup j,c})$ et $\frac{pl_j^{\Omega_i}(\Omega_{i \cup j,c})}{pl_i^{\Omega_j}(\Omega_{i \cup j,c})} pl_i^{\Omega_j}(\{\omega_k\})$, inconnus, sont assimilés à la fonction de croyance vide $pl^{\Omega}(\Omega) = 1$. En outre, considérant que les termes $K^{\Omega_i} pl_{ij}^{\Omega}(\Omega_i)$ sont communs à toutes les expressions et n'ont de ce fait pas d'influence sur la décision, les auteurs proposent de ne pas les estimer. Des plausibilités relatives, incomplètes, notées ci-dessous $\tilde{pl}_{ij}^{\Omega}$, sont alors calculées :

$$\tilde{pl}_{ij}^{\Omega}(\{\omega_k\}) = pl_i^{\Omega_i}(\{\omega_k\}), \quad \forall \omega_k \in \Omega_i \setminus \Omega_{i \cup j,c}; \tag{2.51}$$

$$\tilde{pl}_{ij}^{\Omega}(\{\omega_k\}) = pl_i^{\Omega_i}(\{\omega_k\})\ pl_j^{\Omega_{i \cup j,c}}(\{\omega_k\}), \quad \forall \omega_k \in \Omega_{i \cup j,c}; \tag{2.52}$$

$$\tilde{pl}_{ij}^{\Omega}(\{\omega_k\}) = \frac{pl_i^{\Omega_i}(\Omega_{i \cup j,c})}{pl_j^{\Omega_j}(\Omega_{i \cup j,c})}\ pl_j^{\Omega_j}(\{\omega_k\}), \quad \forall \omega_k \in \Omega_j \setminus \Omega_{i \cup j,c}. \tag{2.53}$$

L'aggrégation de plus de deux classifieurs se fait de manière séquentielle ; les auteurs remarquent que les performances dépendent de l'ordre dans lequel elle est effectuée, l'opérateur étant non associatif.

Dans [32], Janez et Appriou proposent d'étendre cette méthode au cas où une connaissance supplémentaire de compatibilité entre les atomes des différents cadres est disponible. Les relations de compatibilité définies par $c_i(\omega_k)$, pour tout (i, k), permettent d'éviter certaines des approximations faites dans [31]. Ainsi, la plausibilité de chaque hypothèse reconnue par le classifieur de référence seul

est réajustée, en fonction de la plausibilité des hypothèses qui lui sont fortement compatibles : pour tout $\omega_k \in \Omega_i \setminus \Omega_{i\cup j,c}$ (hypothèse considérée seulement par $\mathcal{E}_i$),

$$\tilde{pl}_{ij}^{\Omega}(\{\omega_k\}) = pl_i^{\Omega_i}(\{\omega_k\})\ pl_j^{\Omega_j}\left(\bigcup_{c_j(\omega_l)\cap\{\omega_k\}\neq\emptyset} \{\omega_l\}\right). \tag{2.54}$$

Les plausibilités de chaque hypothèse commune aux deux classifieurs sont combinées : pour tout $\omega_k \in \Omega_{i\cup j,c}$ (hypothèse commune à $\mathcal{E}_i$ et $\mathcal{E}_j$),

$$\tilde{pl}_{ij}^{\Omega}(\{\omega_k\}) = pl_i^{\Omega_i}(\{\omega_k\})\ pl_j^{\Omega_{i\cup j,c}}(\{\omega_k\}). \tag{2.55}$$

Enfin, la plausibilité de chaque hypothèse reconnue par le classifieur additionnel seul est réajustée, en fonction du rapport des plausibilités données à $\Omega_{i\cup j,c}$ par chacun des classifieurs et de la plausibilité des hypothèses qui lui sont fortement compatibles : pour tout $\omega_k \in \Omega_j \setminus \Omega_{i\cup j,c}$ (hypothèse considérée seulement par $\mathcal{E}_j$),

$$\tilde{pl}_{ij}^{\Omega}(\{\omega_k\}) = \frac{pl_i^{\Omega_i}(\Omega_{i\cup j,c})}{pl_j^{\Omega_j}(\Omega_{i\cup j,c})}\ pl_i^{\Omega_i}\left(\bigcup_{c_i(\omega_l)\cap\{\omega_k\}\neq\emptyset} \{\omega_l\}\right)\ pl_j^{\Omega_j}(\{\omega_k\}). \tag{2.56}$$

Insistons que le fait que les valeurs ainsi obtenues ne constituent pas une distribution de plausibilité. Seules, des valeurs $\tilde{pl}_{ij}^{\Omega}(\{\omega_k\})$, proportionnelles aux plausibilités des singletons $\omega_k \in \Omega$ sont déterminées. Le vecteur $\mathbf{x}$ est ensuite affecté à la classe de plausibilité relative maximale. Remarquons enfin que les méthodes développées dans [31, 32] ont été proposées pour résoudre un problème de fusion de données multicapteurs. L'hypothèse d'indépendance des capteurs est généralement faite, ce qui permet d'utiliser la règle de combinaison conjonctive pour combiner les informations disponibles. Dans le cas de classifieurs entraînés à partir de données communes, l'hypothèse d'indépendance ne peut être a priori acceptée.

Combinaison de fonctions de croyance par l'opérateur d'extension

Appriou propose dans [5] une méthode générique pour résoudre un problème de fusion de données multicapteurs, permettant de combiner un nombre quelconque de classifieurs distincts, définis sur des cadres de discernement différents. Un opérateur appelé *extension* est construit, à partir des mécanismes de conditionnement et déconditionnement, et de projection et extension vide. Nous présentons ici le cas de la combinaison de deux classifieurs $\mathcal{E}_i$ et $\mathcal{E}_j$.

On suppose disposer de deux fonctions de masse $m_i^{\Omega_i}$ et $m_j^{\Omega_j}$ fournies respectivement par les classifieurs $\mathcal{E}_i$ et $\mathcal{E}_j$. Soient $B_i \subseteq \Omega_i$ et $B_j \subseteq \Omega_j$, deux groupes d'attributs, et $B = B_i \times B_j$. si les classifieurs sont indépendants, on peut définir $m_{ij}^{\Omega_i\times\Omega_j}(B) = m_i^{\Omega_i}(B_i)m_j^{\Omega_j}(B_j)$; sinon, la connaissance des dépendances entre Ω_i et Ω_j permet de déterminer $m_{ij}^{\Omega_i\times\Omega_j}$ par $m_{ij}^{\Omega_i\times\Omega_j}(B) = m_i^{\Omega_i}[B_j](B_i)m_j^{\Omega_j}(B_j)$. On notera pl_{ij} la fonction de masse associée à m_{ij}.

Soit κ_{ij} l'ensemble des couples (ω_i, ω_j) de $\Omega_i \times \Omega_j$ qui sont considérés comme admissibles, suivant le problème traité. Soit $pl_s^{\Omega}[B]$ une mesure de plausibilité, qui formalise la connaissance des relations entre l'ensemble $B = B_i \times B_j$ observé par les classifieurs $\mathcal{E}_i$ et $\mathcal{E}_j$, et la classe prédite suite à leur combinaison.

L'opérateur d'extension est défini par :

$$pl^{\Omega \times \Omega_i \times \Omega_j}(A \times B) = \frac{pl_s^{\Omega}[B](A)\ pl^{\Omega_i \times \Omega_j}(B)}{pl^{\Omega_i \times \Omega_j}(\kappa_{ij})}. \tag{2.57}$$

La fonction de masse $m^{\Omega \times \Omega_i \times \Omega_j}$ associée à $pl^{\Omega \times \Omega_i \times \Omega_j}$ est ensuite déterminée ; dans le cas où cette dernière est incomplète (et ne permet donc pas de déterminer une fonction de masse unique), on calcule la fonction de masse la moins spécifique qui satisfait aux contraintes définies par (2.57). Enfin, la projection de $m^{\Omega \times \Omega_i \times \Omega_j}$ sur Ω peut être calculée, de manière à obtenir une fonction de masse quantifiant la connaissance de la classe du vecteur $\mathbf{x}$ évalué :

$$m^{\Omega}(A) = \sum_{B \subseteq \Omega_i \times \Omega_j} m^{\Omega \times \Omega_i \times \Omega_j}(A \times B). \tag{2.58}$$

Remarquons que le choix des cadres de discernement Ω_i, Ω_j et Ω, ainsi que la connaissance des relations entre ces cadres (modélisée par $pl_s^{\Omega}[B]$) et des couples admissibles (déterminés par κ_{ij}) détermine la nature de la combinaison effectuée. Appriou montre en particulier que les sommes conjonctive et disjonctive, ainsi que d'autres opérateurs, comme celui proposé par Dubois et Prade [20], ou celui proposé par Yager [63], peuvent être modélisés. Soulignons que la question du choix d'une règle particulière pour la résolution d'un problème donné n'est pas élucidée.

L'opérateur d'extension ainsi défini a été proposé dans le cadre de la fusion de données multicapteurs ; il constitue donc une méthode générale de combinaison de sources distinctes. Cependant, cette hypothèse de distinction ne peut être faite lorsque les classifieurs sont entraînés à partir de données en partie communes. Les résultats de la combinaison sont alors vraisemblablement tributaires de la règle de combinaison modélisée, et en particulier de son comportement vis-à-vis de l'interdépendance des classifieurs, qui n'est généralement pas connu a priori pour un problème donné.

2.5 Synthèse

Dans ce chapitre, nous avons détaillé les différentes méthodes de combinaison de classifieurs binaires proposées dans la littérature, dans le cas de décompositions 1-1, 1-T et CCE. Nous nous sommes intéressés plus particulièrement aux méthodes permettant de combiner des mesures de confiance (probabilités, possibilités ou fonctions de croyance), et tenant compte des dépendances entre les

différents classifieurs. Plusieurs méthodes proposées dans la littérature pour combiner des classifieurs probabilistes ont retenu notre attention.

- Une méthode itérative pour combiner des classifieurs CCE probabilistes, appelée par la suite méthode PCplBT, consiste à minimiser la distance de Kullback-Leibler entre les sorties des classifieurs et les probabilités conditionnelles correspondantes [29] (dans le cas 1-1, elle est identique à celle proposée dans [26]).
- Une méthode non-itérative pour combiner des classifieurs CCE probabilistes, appelée par la suite méthode PEstP, est basée sur un certain nombre d'hypothèses concernant l'indépendance des classifieurs et les distributions considérées [40].
- Deux méthodes non-itératives pour combiner des classifieurs 1-1 probabilistes, appelées par la suite méthodes PEst1 et PEst2, minimisent une distance entre les sorties des classifieurs et les probabilités conditionnelles correspondantes [61].
- Une méthode non-itérative permettant de combiner des classifieurs CCE probabilistes, appelée par la suite méthode PEstCorr, inclut une étape de correction des classifieurs, consistant à estimer la probabilité que le vecteur évalué $\mathbf{x}$ appartienne à leur ensemble d'apprentissage [39].

Les analyses comparatives menées dans la plupart des articles [25, 57, 28] montrent que le schéma 1-1 donne généralement de meilleurs résultats que le schéma 1-T ; de même, Dietterich et Bakiri [16] mettent en évidence la supériorité du schéma CCE sur le schéma 1-T. Cependant, Rifkin et Klautau soutiennent dans une publication récente [48] que les résultats d'une combinaison 1-T sont équivalents à ceux obtenus par les autres schémas de décomposition, lorsque les classifieurs sont correctement régularisés. Cette polémique met en évidence le comportement inégal de la combinaison de classifieurs selon le schéma de décomposition et les caractéristiques des classifieurs binaires employés, pour un problème donné. Nous aborderons ce point dans le chapitre 5.

Chapitre 3

Combinaison de classifieurs binaires dans le cadre du Modèle des Croyances Transférables : cas d'une décomposition un-contre-un

Dans ce chapitre, nous présentons une approche pour la combinaison de classifieurs binaires dans un cadre évidentiel, lorsque le schéma de décomposition un-contre-un (ou 1-1) est employé : l'ensemble Ω des classes est décomposé en paires de classes $\{\omega_i, \omega_j\}$; par la suite, nous adopterons la notation $\Omega_{ij} = \{\omega_i, \omega_j\}$. Pour tout $j > i$, un classifieur $\mathcal{E}_{ij}$ est entraîné à séparer ω_i de ω_j, à partir des vecteurs d'apprentissage appartenant à Ω_{ij}.

Rappelons que la problématique abordée doit être distinguée de la fusion d'informations issues de sources distinctes. Dans le cas présent, nous cherchons à identifier la classe d'un vecteur $\mathbf{x}$ évalué parmi un ensemble Ω, en combinant les informations fournies par les classifieurs binaires. Chaque classifieur $\mathcal{E}_{ij}$ apporte donc des informations incomplètes, étant entraîné à reconnaître deux classes ω_i et ω_j seulement ; et non distinctes, dans la mesure où les informations utilisées pour le construire apparaissent également dans l'ensemble d'apprentissage d'autres classifieurs.

Le MCT, qui permet de représenter divers types de connaissance partielle, semble donc être un cadre théorique bien adapté pour modéliser le caractère incomplet des informations disponibles, puis les combiner en tenant compte de leur incomplétude et de leur caractère indistinct. Nous proposons ainsi plusieurs méthodes de combinaison, dont les propriétés sont illustrées sur un jeu de données synthétiques Synth représenté sur la figure 3.1. Deux types de classifieurs binaires ont été utilisés : la régression logistique et les réseaux de neurones évidentiels, présentés au paragraphe 5.1.2 du chapitre 5.

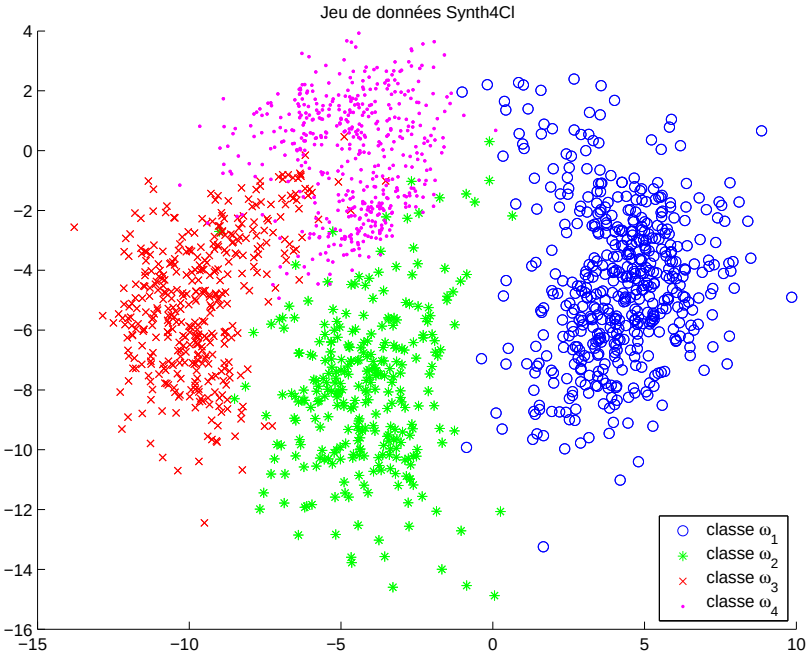


FIG. 3.1 – Jeu de données Synth, utilisé pour analyser les méthodes de combinaison.

3.1 Les sorties des classifieurs vues comme des fonctions de masse conditionnelles

En assimilant le classifieur $\mathcal{E}_{ij}$ à un expert, on pourrait qualifier sa connaissance sur Ω d'incomplète, en ce sens qu'il ignore tout des classes de Ω autres que ω_i et ω_j. Les informations fournies par $\mathcal{E}_{ij}$ lors de l'évaluation d'un vecteur $\mathbf{x}$ peuvent donc être modélisées par une fonction de masse m_{ij}, valide à condition que $\mathbf{x}$ appartienne à ω_i ou ω_j. Cette fonction de masse est donc par définition conditionnelle à Ω_{ij}. Supposons que la connaissance de la classe réelle de $\mathbf{x}$ soit modélisée par une fonction de masse m^Ω ; m_{ij}, ne représentant qu'une fraction de cette connaissance, peut être interprétée comme le conditionnement de m^Ω sur Ω_{ij} :

$$m_{ij} = m^\Omega[\Omega_{ij}], \quad \forall\ j > i. \tag{3.1}$$

Le caractère partiel de la connaissance qu'a $\mathcal{E}_{ij}$ de Ω se traduit donc par le fait que ses sorties sont modélisées sur un cadre Ω_{ij}, obtenu en restreignant l'ensemble des classes considérées à $\{\omega_i, \omega_j\}$.

Remarquons en outre que la plupart des méthodes de classification n'intègrent pas de processus de détection de nouveauté : un classifieur $\mathcal{E}_{ij}$ est généralement incapable de déterminer si le vecteur $\mathbf{x}$ évalué est d'un type nouveau qu'il n'a pas été entraîné à reconnaître, c'est-à-dire si la classe réelle de $\mathbf{x}$ n'est ni ω_i ni ω_j. Une fonction de masse fournie par un tel classifieur est donc définie sur un

cadre Ω_{ij} exhaustif : l'hypothèse de la non-appartenance de $\mathbf{x}$ à Ω_{ij} n'est pas envisagée. En d'autres termes, cette fonction de masse (que l'on notera m^*_{ij}) peut être interprétée comme le conditionnement normalisé d'une fonction de masse m^Ω sur Ω_{ij} :

$$\begin{aligned} m^*_{ij}(A) &= \frac{m^\Omega[\Omega_{ij}](A)}{1 - m^\Omega[\Omega_{ij}](\emptyset)}, \quad \forall A \subseteq \Omega_{ij}, A \neq \emptyset, \quad \forall\, j > i; & (3.2) \\ m^*_{ij}(\emptyset) &= 0. & (3.3) \end{aligned}$$

Rappelons que chaque classe ω_k apparaît dans l'ensemble d'apprentissage de $K-1$ classifieurs exactement : les fonctions de masse m_{ik} ($\forall k \neq i$) ne sont donc pas indépendantes, étant basées sur des informations provenant de sources non distinctes. Elles ne peuvent donc être simplement dénormalisées puis combinées par la règle de combinaison conjonctive (1.18). Remarquons que l'équation (3.2) peut s'écrire :

$$m^\Omega[\Omega_{ij}](A) = m^*_{ij}(A)(1 - m^\Omega[\Omega_{ij}](\emptyset)), \quad \forall A \subseteq \Omega_{ij}, A \neq \emptyset, \quad \forall i > j. \qquad (3.4)$$

Chaque cadre Ω_{ij} compte quatre sous-ensembles : $\emptyset$, $\{\omega_i\}$, $\{\omega_j\}$ et Ω_{ij}. La relation (3.4) définit donc un système linéaire à $3 \times C_K^2$ équations et $2^K - 1$ inconnues, les 2^K masses $m^\Omega(C)$ étant liées par la contrainte $\sum_{C \subseteq \Omega} m^\Omega(C) = 1$. La fonction de masse définie par $m^\Omega(\emptyset) = 1$ vérifie :

$$\left\{ \begin{array}{lcll} m^\Omega[\Omega_{ij}](\emptyset) & = & 1, & \forall\, \Omega_{ij}; \\ m^\Omega[\Omega_{ij}](A) & = & 0, & \forall A \subseteq \Omega_{ij}, A \neq \emptyset, \quad \forall\, \Omega_{ij}. \end{array} \right.$$

Elle est donc une solution triviale du système défini par (3.4) ; pour l'écarter, une contrainte de normalité $m^\Omega(\emptyset) = 0$ peut être imposée. Les fonctions de masse m^*_{ij} ne sont généralement pas consistantes : il n'existe en général aucune fonction m^Ω dont les conditionnements normalisés vérifient $m[\Omega_{ij}]^* = m^*_{ij}$ pour tout $j > i$. Dans ce cas, le système défini par l'équation (3.4) n'a pas de solution. Il n'est donc également pas possible de combiner les masses m_{ij} par résolution directe de ce système.

Tout comme dans la méthode présentée au paragraphe 2.2.2, nous proposons de déterminer une solution approchée du système (3.4), en recherchant la fonction de masse $\widehat{m^*}^\Omega$ la plus consistante possible avec les informations fournies par les classifieurs. Formellement, $\widehat{m^*}^\Omega$ est définie comme la solution du problème d'optimisation quadratique à $3 \times C_K^2$ équations et $2^K - 2$ inconnues consistant à minimiser les écarts entre les conditionnements normalisés $m[\Omega_{ij}]^*$ et les estimations m^*_{ij} :

$$\widehat{m^*}^\Omega = \arg\min_{m^\Omega} \sum_{j>i} \sum_{\emptyset \neq A \subseteq \Omega_{ij}} \left(m^\Omega[\Omega_{ij}](A) - m^*_{ij}(A) \left(1 - m^\Omega[\Omega_{ij}](\emptyset)\right)\right)^2 ; \qquad (3.5)$$

sous les contraintes :

$$\begin{aligned} m^{\Omega}(A) &\geq 0, \quad \forall A \subseteq \Omega, && (3.6)\\ \sum_{A\subseteq\Omega} m^{\Omega}(A) &= 1; && (3.7)\\ m^{\Omega}(\emptyset) &= 0. && (3.8) \end{aligned}$$

Cette méthode sera appelée par la suite méthode MCT1-1.

Exemple 3.1 Soit un problème à trois classes $\{\omega_1, \omega_2, \omega_3\} = \Omega$, et un ensemble de classifieurs $\mathcal{E}_{ij}$, ayant fourni les masses de croyance conditionnelles normalisées suivantes :

$$\begin{array}{lllllllll} m^*_{12}(\{\omega_1\}) &=& 0.174, & m^*_{13}(\{\omega_1\}) &=& 0.002, & m^*_{23}(\{\omega_2\}) &=& 0.073,\\ m^*_{12}(\{\omega_2\}) &=& 0.781, & m^*_{13}(\{\omega_3\}) &=& 0.997, & m^*_{23}(\{\omega_3\}) &=& 0.915,\\ m^*_{12}(\Omega_{12}) &=& 0.045; & m^*_{13}(\Omega_{13}) &=& 0.001; & m^*_{23}(\Omega_{23}) &=& 0.012; \end{array}$$

La combinaison de ces masses, par la méthode définie par (3.5)-(3.8), donne la fonction de masse suivante :

$$\begin{array}{lllllllll} \widehat{m^*}^{\Omega}(\emptyset) &=& 0.000, & \widehat{m^*}^{\Omega}(\{\omega_3\}) &=& 0.908,\\ \widehat{m^*}^{\Omega}(\{\omega_1\}) &=& 0.006, & \widehat{m^*}^{\Omega}(\Omega_{13}) &=& 0.004,\\ \widehat{m^*}^{\Omega}(\{\omega_2\}) &=& 0.071, & \widehat{m^*}^{\Omega}(\Omega_{23}) &=& 0.010,\\ \widehat{m^*}^{\Omega}(\Omega_{12}) &=& 0.000, & \widehat{m^*}^{\Omega}(\Omega) &=& 0.001; \end{array}$$

cette masse est la plus proche des estimations m^*_{ij}, au sens du critère (3.5). □

La figure 3.2 montre les probabilités conditionnelles r^*_{23}, r^*_{24} et r^*_{34} obtenues par régression logistique lors du traitement des données Synth. Les fonctions de masse conditionnelles Bayésiennes m^*_{23}, m^*_{24} et m^*_{34} en sont directement déduites : on a $m^*_{ij}(\emptyset) = m^*_{ij}(\{\omega_i, \omega_j\}) = 0$, et $m^*_{ij}(\{\omega_i\}) = r^*_{ij}$, $m^*_{ij}(\{\omega_j\}) = 1 - r^*_{ij}$, pour tout $j > i$.

Les masses $\widehat{m}^{\Omega *}(\{\omega_3\})$ et $\widehat{m}^{\Omega *}(\{\omega_4\})$, obtenues en combinant les fonctions de masse m^*_{ij} par la méthode MCT1-1, sont représentées sur la figure 3.3. Les masses attribuées aux éléments focaux non singletons sont négligeables et n'ont pas été représentées. On constate que $m^{\Omega}(\{\omega_3\})$ est très proche de $m_{23}(\{\omega_3\})$, à la frontière entre ω_2 et ω_3 ; et de $m_{34}(\{\omega_3\})$, à la frontière entre ω_3 et ω_4 : dans ces régions, les courbes de niveau correspondent à celles de r_{12} et r_{23}. La résolution du problème d'optimisation permet de déterminer un compromis entre les informations fournies par les classifieurs binaires. À la frontière entre ω_3 et ω_4, la masse $\widehat{m}^*(\{\omega_3\})$ obtenue par combinaison correspond donc aux informations fournies par le classifieur $\mathcal{E}_{34}$. À la frontière entre les classes ω_2, ω_3 et ω_4, un compromis est déterminé entre les sorties des classifieurs $\mathcal{E}_{23}$, $\mathcal{E}_{24}$ et $\mathcal{E}_{34}$, ce qui se traduit par des courbes de niveau non linéaires.

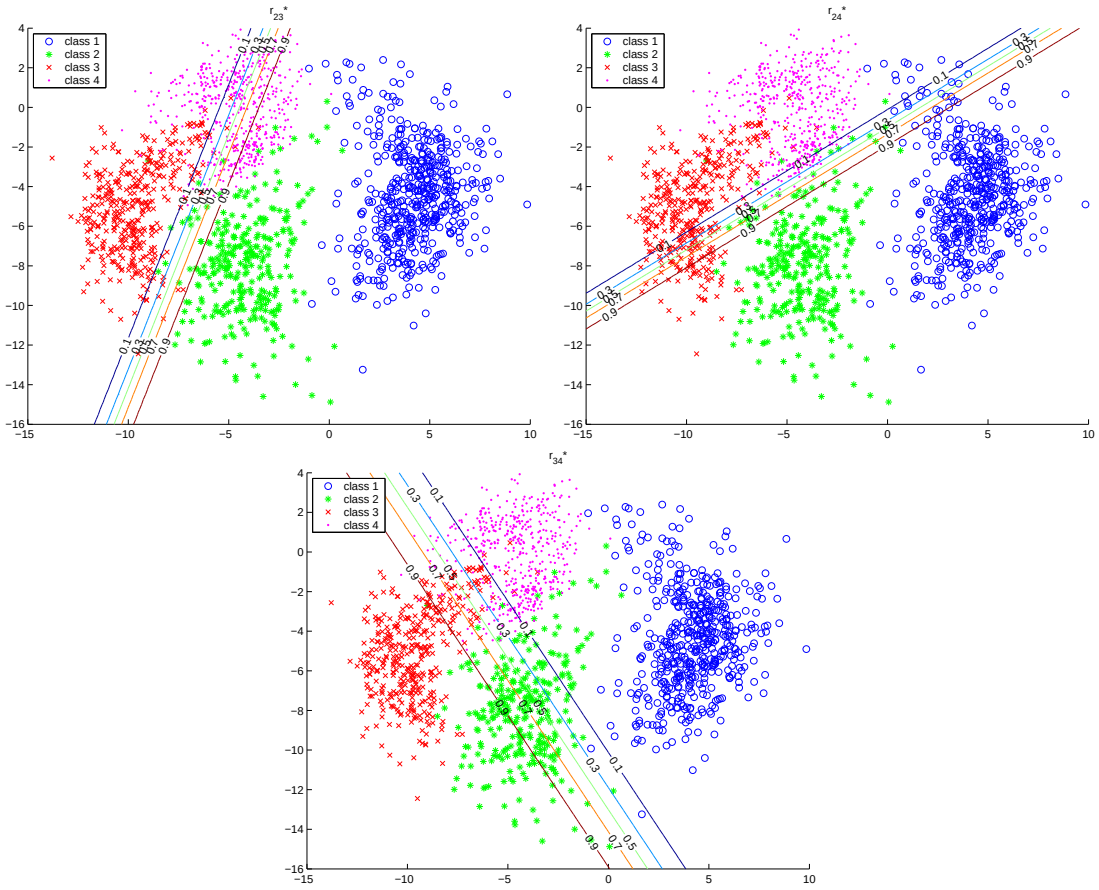


FIG. 3.2 – Courbes de niveau des probabilités conditionnelles r^*_{23} (haut-gauche), r^*_{24} (haut-droite), et r^*_{34} (bas), obtenues par régression logistique.

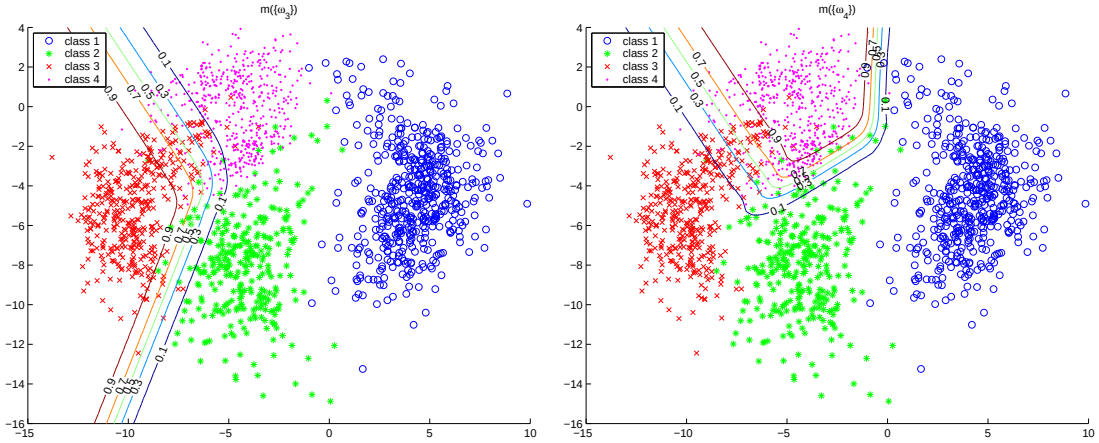


FIG. 3.3 – Courbes de niveau des masses combinées $\widehat{m}^{\Omega *}(\{\omega_3\})$ (gauche) et $\widehat{m}^{\Omega *}(\{\omega_4\})$ (droite), obtenues par la méthode MCT1-1.

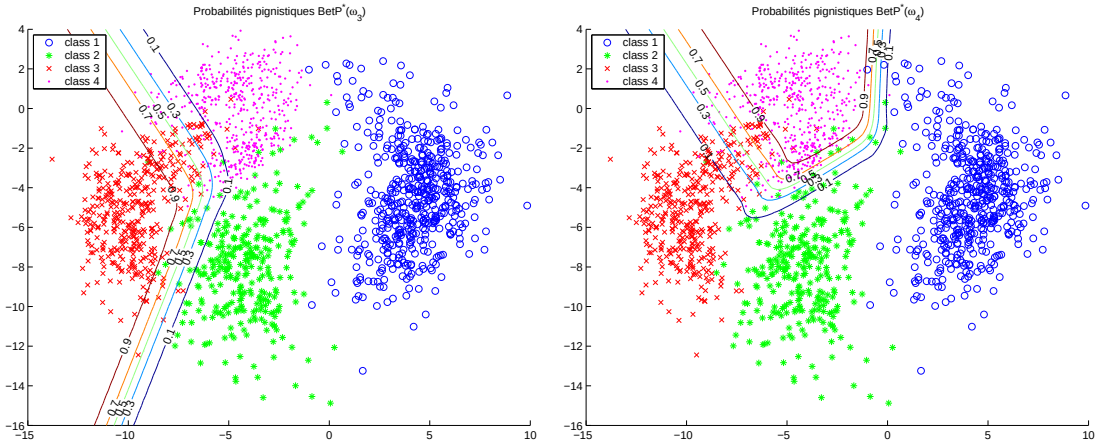


FIG. 3.4 – Courbes de niveau des probabilités pignistiques $BetP^*(\omega_3)$ (gauche) et $BetP^*(\omega_4)$ (droite), obtenues par la méthode MCT1-1.

Les probabilités pignistiques correspondant $BetP^*(\omega_3)$ et $BetP^*(\omega_4)$ sont représentées sur la figure 3.4. La combinaison fournit une fonction de masse presque Bayésienne, une masse de croyance négligeable étant parfois allouée aux éléments focaux non singletons. Par conséquent, les probabilités pignistiques calculées sont très proches de cette fonction de masse.

Remarquons qu'ici, les classifieurs pertinents pour reconnaître une classe ω_k sont aussi nombreux que les autres. On peut tout de même relever des cas où un classifieur, a priori non pertinent, s'avère influencer le classement de $\mathbf{x}$.

Exemple 3.2 La figure 3.5 montre un individu $\mathbf{x}$ pour lequel

$$\begin{array}{lllllllll} r^*_{12} & = & 0, & r^*_{13} & = & 0, & r^*_{14} & = & 0; \\ r^*_{23} & = & 0.535, & r^*_{24} & = & 0.587, & r^*_{34} & = & 0.808. \end{array}$$

La classe ω_2 est préférée à chaque autre par le classifieur correspondant. Cependant, la combinaison des masses Bayésiennes obtenues à partir des r^*_{ij} donne :

$$\begin{array}{lllllllll} \widehat{m}^{\Omega *}(\{\omega_1\}) & = & 0.003, & \widehat{m}^{\Omega *}(\{\omega_2\}) & = & 0.408, & \widehat{m}^{\Omega *}(\{\omega_3\}) & = & 0.412; \\ \widehat{m}^{\Omega *}(\{\omega_4\}) & = & 0.165, & \widehat{m}^{\Omega *}(\{\omega_3,\omega_4\}) & = & 0.012. & & & \end{array}$$

La probabilité pignistique associée est définie par $BetP^*(\omega_1) = 0.003$, $BetP^*(\omega_2) = 0.408$, $BetP^*(\omega_3) = 0.418$ et $BetP^*(\omega_4) = 0.171$: $\mathbf{x}$ est donc affecté à ω_3. Ce résultat est dû au fait que les probabilités r^*_{23} et r^*_{24} sont proches de 0.5, tandis que r^*_{34}, plus élevée, joue un rôle déstabilisateur dans le calcul de la solution. Bien qu'il soit ici difficile d'apprécier visuellement l'appartenance de $\mathbf{x}$, cet exemple suggère que l'occurence de tels cas de déséquilibre pourrait augmenter avec le nombre de classifieurs non pertinents, qui croît plus vite que le nombre de classifieurs pertinents en fonction du nombre de classes. □

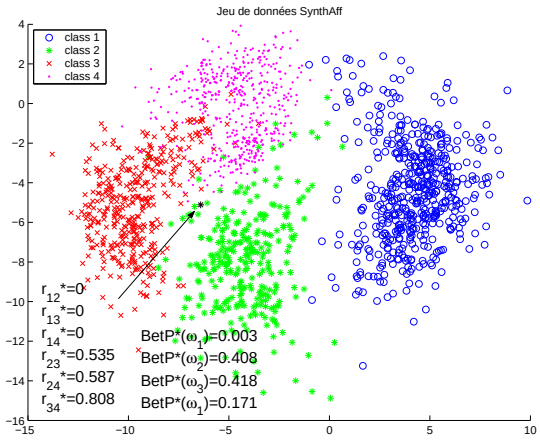


FIG. 3.5 – Exemple d'influence d'un classifieur sur la décision après combinaison.

3.2 Estimation de la pertinence des classifieurs binaires

La résolution du problème défini par (3.5)-(3.8) donne une solution $\widehat{m^*}^{\Omega}$ compatible avec les fonctions de masse normales m^*_{ij} fournies par les classifieurs $\mathcal{E}_{ij}$. Or, comme le soulignent Hastie et Tibshirani [26], un classifieur $\mathcal{E}_{ij}$ évaluant un vecteur $\mathbf{x}$ peut fournir des informations erronées, s'il n'a pas été entraîné à reconnaître la classe réelle de $\mathbf{x}$. Assimilons un tel classifieur à un expert : il peut être considéré comme ignorant, en ce sens qu'il ne dispose pas de la connaissance nécessaire à une évaluation pertinente de $\mathbf{x}$. L'ignorance de $\mathcal{E}_{ij}$ dépend donc clairement de l'appartenance de $\mathbf{x}$ à l'ensemble d'apprentissage Ω_{ij} de $\mathcal{E}_{ij}$. Il semble donc naturel d'évaluer la pertinence de $\mathcal{E}_{ij}$ en estimant la *plausibilité* que $\mathbf{x}$ appartienne à Ω_{ij} ; la relation suivante met en évidence le lien entre ces deux concepts :

$$
\begin{aligned}
m^{\Omega}[\Omega_{ij}](\emptyset) &= \sum_{A \cap \Omega_{ij} = \emptyset} m^{\Omega}(A) && (3.9)\\
&= 1 - \sum_{A \cap \Omega_{ij} \neq \emptyset} m^{\Omega}(A) && (3.10)\\
&= 1 - pl^{\Omega}(\Omega_{ij}). && (3.11)
\end{aligned}
$$

Cette étape d'estimation de la pertinence $pl(\Omega_{ij})$ de $\mathcal{E}_{ij}$, pour tout $j > i$, peut évoquer le calcul des estimations $q_{ij} = \widehat{\mathbb{P}}(\Omega_{ij})$ [39] présenté au paragraphe 2.2.2. De manière similaire, cette information peut être intégrée aux fonctions de masse m^*_{ij} en les *dénormalisant* : en reprenant (3.4), on obtient alors une estimation sous-normale m_{ij} des conditionnements $m[\Omega_{ij}]$, pour tout $j > i$, par :

$$
\begin{aligned}
m_{ij}(A) &= pl(\Omega_{ij}) m^*_{ij}(A), \quad \forall A \subseteq \Omega_{ij}, A \neq \emptyset; && (3.12)\\
m_{ij}(\emptyset) &= 1 - pl(\Omega_{ij}). && (3.13)
\end{aligned}
$$

L'opérateur de conditionnement défini par l'équation (1.3) étant une application linéaire, chaque terme $m[\Omega_{ij}](A)$ peut être écrit comme une combinaison linéaire de masses $m^\Omega(C)$, avec $C \subseteq \Omega$. Un cadre Ω_{ij} comptant quatre sous-ensembles, les relations (3.12)-(3.13) définissent un système de $4 \times C_K^2$ équations linéaires et $2^K - 1$ inconnues, les 2^K masses $m^\Omega(C)$ étant liées par la contrainte $\sum_{C \subseteq \Omega} m^\Omega(C) = 1$. Comme dans le cas précédent, les fonctions de masse m_{ij} ne sont généralement pas consistantes : il n'existe en général aucune fonction m^Ω dont les conditionnements vérifient $m^\Omega[\Omega_{ij}] = m_{ij}$ pour tout $j > i$. Dans ce cas, le système défini par l'équation (3.1) n'a pas de solution exacte.

On propose donc de déterminer une solution approchée en résolvant le problème d'optimisation défini par (3.14)-(3.16), consistant à minimiser les écarts entre les conditionnements $m^\Omega[\Omega_{ij}]$ et les estimations m_{ij} [45] :

$$\widehat{m}^\Omega = \arg\min_{m^\Omega} \sum_{j>i} \sum_{A \subseteq \Omega_{ij}} \left(m^\Omega[\Omega_{ij}](A) - m_{ij}(A) \right)^2, \tag{3.14}$$

sous les contraintes :

$$m^\Omega(A) \geq 0, \quad \forall A \subseteq \Omega, \tag{3.15}$$

$$\sum_{A \subseteq \Omega} m^\Omega(A) = 1. \tag{3.16}$$

Cette méthode sera appelée par la suite méthode MCTCorr1-1. Soulignons qu'elle peut être utilisée pour combiner des classifieurs fournissant directement des fonctions de masse sous-normales.

Exemple 3.3 Reprenons l'exemple 3.1. On suppose disposer des plausibilités $pl_{12} = 0.754$, $pl_{13} = 1$ et $pl_{23} = 1$. On obtient les estimations sous-normales m_{ij} suivantes :

$m_{12}(\emptyset)$	=	0.246,	$m_{13}(\emptyset)$	=	0.000,	$m_{23}(\emptyset)$	=	0.000,
$m_{12}(\{\omega_1\})$	=	0.131,	$m_{13}(\{\omega_1\})$	=	0.002,	$m_{23}(\{\omega_2\})$	=	0.073,
$m_{12}(\{\omega_2\})$	=	0.589,	$m_{13}(\{\omega_3\})$	=	0.997,	$m_{23}(\{\omega_3\})$	=	0.915.
$m_{12}(\Omega_{12})$	=	0.034;	$m_{13}(\Omega_{13})$	=	0.001;	$m_{23}(\Omega_{23})$	=	0.012.

La combinaison de ces estimations, par résolution de (3.14)-(3.16), donne :

$\widehat{m}(\{\omega_2\})$	=	0.074,	$\widehat{m}(\Omega_{13})$	=	0.116,
$\widehat{m}(\{\omega_3\})$	=	0.534,	$\widehat{m}(\Omega_{23})$	=	0.276,

les autres masses étant nulles. Les conditionnements de $\widehat{m}$ sur les Ω_{ij} donnent :

$\widehat{m}[\Omega_{12}](\emptyset)$	=	0.534,	$\widehat{m}[\Omega_{13}](\emptyset)$	=	0.074,	$\widehat{m}[\Omega_{23}](\emptyset)$	=	0.000,
$\widehat{m}[\Omega_{12}](\{\omega_1\})$	=	0.116,	$\widehat{m}[\Omega_{13}](\{\omega_1\})$	=	0.000,	$\widehat{m}[\Omega_{23}](\{\omega_2\})$	=	0.074,
$\widehat{m}[\Omega_{12}](\{\omega_2\})$	=	0.350,	$\widehat{m}[\Omega_{13}](\{\omega_3\})$	=	0.810,	$\widehat{m}[\Omega_{23}](\{\omega_3\})$	=	0.650,
$\widehat{m}[\Omega_{12}](\Omega_{12})$	=	0.000;	$\widehat{m}[\Omega_{13}](\Omega_{13})$	=	0.116;	$\widehat{m}[\Omega_{23}](\Omega_{23})$	=	0.276,

qui sont les meilleures approximations des m_{ij} au sens du critère (3.14). □

L'estimation d'une distribution de plausibilité à partir de données est un problème qui fait actuellement l'objet de recherches. Plusieurs solutions ont été proposées dans le cas de données monodimensionnelles [17, 38] ; l'extension de ces méthodes au cas multidimensionnel reste à ce jour un problème non résolu.

Plutôt que d'estimer directement chaque plausibilité $pl(\Omega_{ij})$ que $\mathbf{x}$ appartienne au domaine Ω_{ij}, nous proposons d'utiliser des classifieurs à une classe, pour associer à chaque classe $\omega_k \in \Omega$ une plausibilité d'appartenance $pl(\{\omega_k\})$. Un classifieur à une classe n'est pas un outil de classification supervisée : il permet de déterminer une région de l'espace correspondant à un groupe d'individus, représentés par des vecteurs d'attributs non étiquetés. Il s'agit donc davantage d'un outil de description des données que de discrimination.

Ces plausibilités $pl(\{\omega_k\})$ peuvent alors être combinées par paires pour obtenir les $pl(\Omega_{ij})$. Remarquons que toute plausibilité $pl(\Omega_{ij})$ satisfait la relation suivante :

$$\max\left[pl(\{\omega_i\}), pl(\{\omega_j\})\right] \leq pl(\Omega_{ij}) \leq \min\left[1, pl(\{\omega_i\}) + pl(\{\omega_j\})\right]. \tag{3.17}$$

Les $pl(\{\omega_k\})$ peuvent donc être combinées au moyen d'une conorme triangulaire, ou t-conorme. Un tel opérateur est défini comme une application $\odot : [0,1] \times [0,1] \rightarrow [0,1]$, commutative, associative, monotone et admettant 0 pour élément neutre :

$$\begin{aligned} x \odot y &= y \odot x, \\ x \odot (y \odot z) &= (x \odot y) \odot z; \\ x \odot y \leq z \odot t &\text{ si } x \leq z \text{ et } y \leq t, \\ x \odot 0 &= x. \end{aligned}$$

On peut montrer que l'opérateur vérifie en outre :

$$\begin{aligned} x \odot y &\geq \max(x, y), \\ x \odot 1 &= 1. \end{aligned}$$

Les plausibilités $pl(\{\omega_k\})$ peuvent donc être combinées en utilisant une t-conorme vérifiant $x \odot y \leq \min[1, x + y]$. L'utilisation de la t-conorme probabiliste donne le résultat suivant :

$$pl(\Omega_{ij}) = pl(\{\omega_i\}) + pl(\{\omega_j\}) - pl(\{\omega_i\})pl(\{\omega_j\}). \tag{3.18}$$

La figure 3.6 montre les plausibilités $pl^{\Omega}(\{\omega_3\})$ et $pl^{\Omega}(\{\omega_4\})$, obtenues au moyen de classifieurs à une classe ; et la plausibilité $pl^{\Omega}(\Omega_{34})$, obtenue en combinant $pl^{\Omega}(\{\omega_2\})$ et $pl^{\Omega}(\{\omega_3\})$ au moyen de la t-conorme probabiliste. Ces données permettent de dénormaliser les masses m^*_{ij} fournies par les classifieurs binaires. Les masses sous-normales $m_{34}(\{\omega_3\})$ et $m_{34}(\{\omega_4\})$ sont représentées sur la figure 3.7 (rappelons que $m_{ij}(\emptyset) = 1 - pl(\Omega_{ij})$, pour tout $j > i$). On constate que les

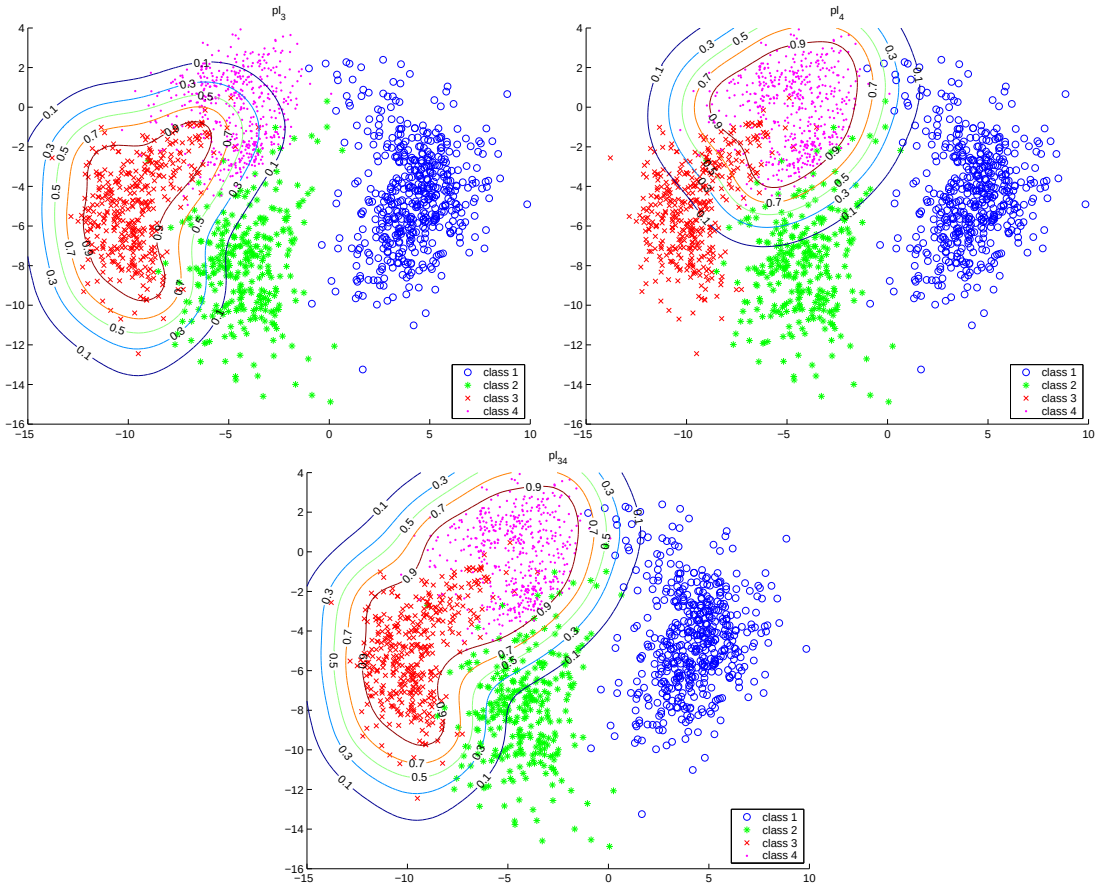


FIG. 3.6 – Courbes de niveau des plausibilités $pl^{\Omega}(\{\omega_3\})$ (haut-gauche) et $pl^{\Omega}(\{\omega_4\})$ (haut-droite), obtenues au moyen de 1-SVM ; et $pl^{\Omega}(\Omega_{34})$ (bas), obtenue par combinaison des deux précédentes par la t-conorme probabiliste.

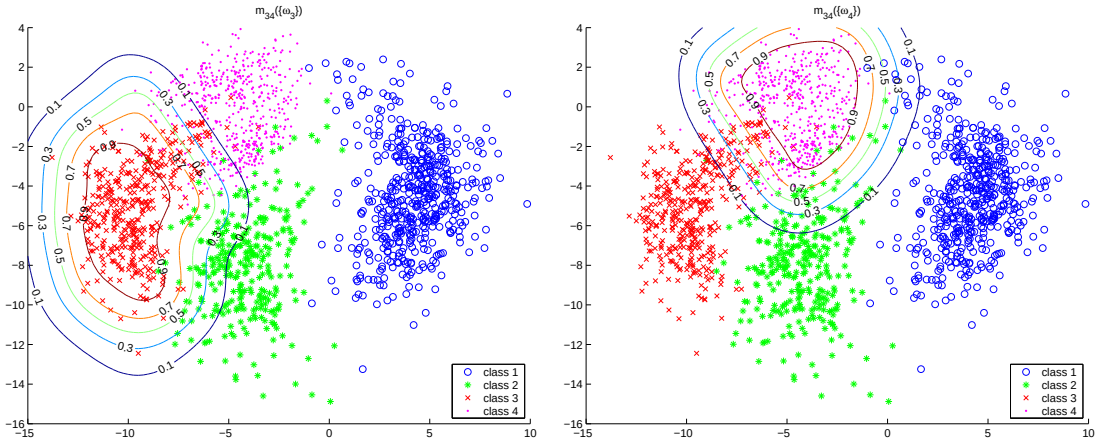


FIG. 3.7 – Courbes de niveau des masses $m_{34}(\{\omega_3\})$ (gauche) et $m_{34}(\{\omega_4\})$ (droite), obtenues en dénormalisant la fonction de masse m^*_{34}.

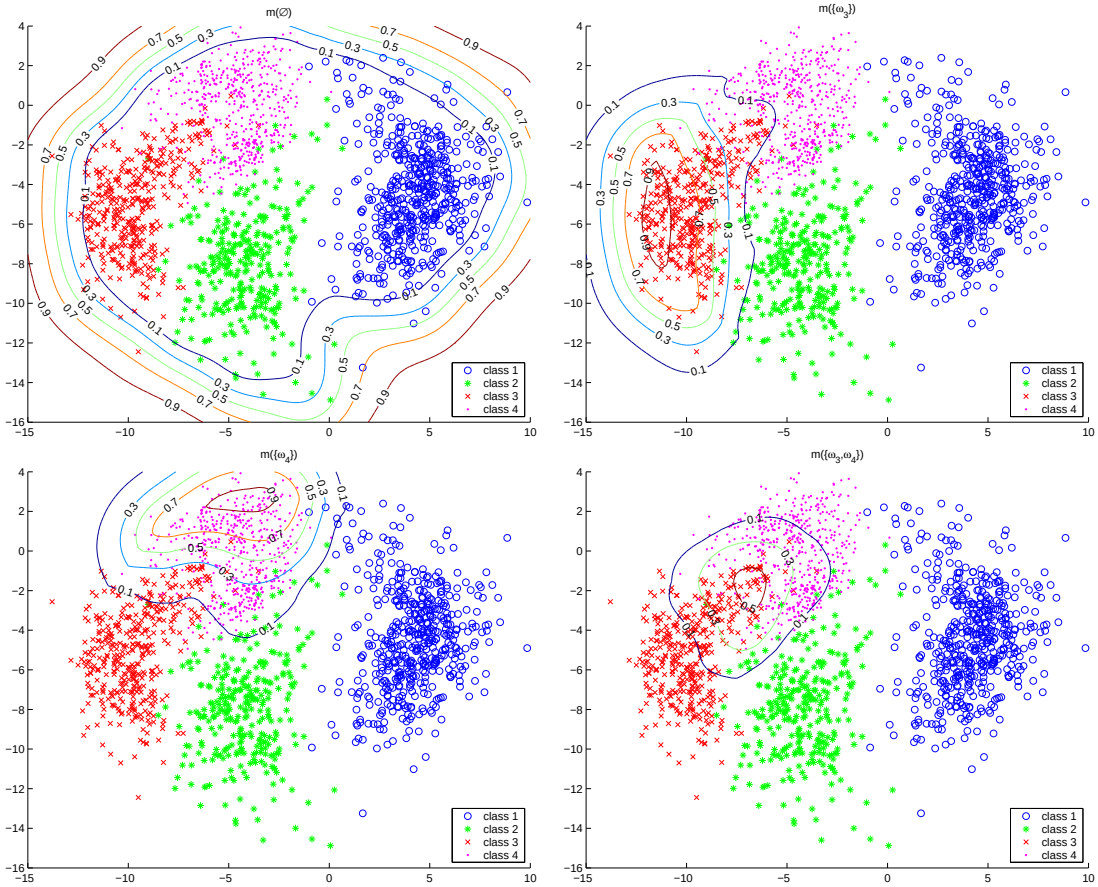


FIG. 3.8 – Courbes de niveau des masses combinées $\widehat{m}^{\Omega}(\emptyset)$ (haut-gauche), $\widehat{m}^{\Omega}(\{\omega_3\})$ (haut-droite), $\widehat{m}^{\Omega}(\{\omega_4\})$ (bas-gauche) et $\widehat{m}^{\Omega}(\{\omega_3,\omega_4\})$ (bas-droite), obtenues par la méthode MCTCorr1-1.

masses $m_{34}(\{\omega_3\})$ et $m_{34}(\{\omega_4\})$ sont élevées dans les régions de forte densité des classes ω_3 et ω_4, respectivement. On peut en outre constater qu'à la frontière entre ω_3 et ω_4, les courbes de niveau de ces masses correspondent à celles de $m_{34}(\{\omega_3\})$; dans les autres régions frontalières de ces deux classes, elles correspondent à celles de $pl^{\Omega}(\Omega_{34})$. Les informations fournies par $\mathcal{E}_{34}$ sont donc utilisées pour séparer ω_3 de ω_4, et celles fournies par les classifieurs à une classe, pour séparer Ω_{34} du reste de l'espace. Contrairement à la méthode MCT1-1, la méthode MCTCorr1-1 permet de déterminer le degré de participation de chaque classifieur dans l'élaboration de la solution.

Les masses $\widehat{m}^{\Omega}(\emptyset)$, $\widehat{m}^{\Omega}(\{\omega_3\})$, $\widehat{m}^{\Omega}(\{\omega_4\})$ et $\widehat{m}^{\Omega}(\{\omega_3,\omega_4\})$, obtenues en combinant les fonctions de masse sous-normales m_{ij} par la méthode MCTCorr1-1, sont représentées sur la figure 3.8. Il est intéressant de noter que la masse $\widehat{m}(\{\omega_3,\omega_4\})$ est significative à la frontière des classes ω_3 et ω_4. L'attribution d'une masse de croyance aux éléments non singletons permet d'améliorer l'adéquation entre les

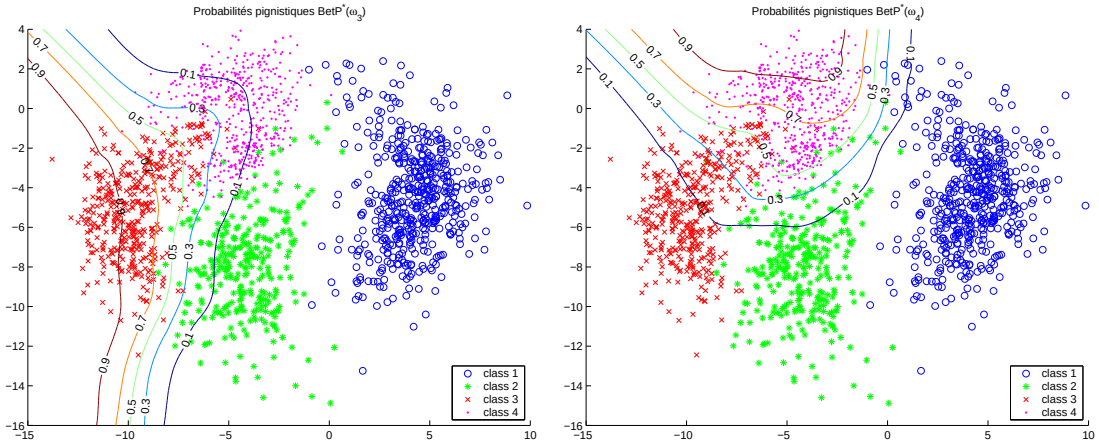


FIG. 3.9 – Courbes de niveau des probabilités pignistiques $BetP^*(\omega_3)$ (gauche) et $BetP^*(\omega_4)$ (droite), obtenues par la méthode MCTCorr1-1.

fonctions de masse m_{ij} et les conditionnements $\widehat{m}[\Omega_{ij}]$ de la fonction de masse recherchée : lors d'un conditionnement, cette masse est plus libre que celle allouée aux singletons. On pourra constater que les probabilités pignistiques $BetP^*(\omega_3)$ et $BetP^*(\omega_4)$, représentées sur la figure 3.9, sont beaucoup plus proches des frontières des classes que dans le cas de la méthode MCT1-1.

Exemple 3.4 Reprenons l'exemple 3.2. Les plausibilités d'appartenance aux paires de classes sont $pl(\Omega_{12}) = pl(\Omega_{23}) = pl(\Omega_{24}) = 1$, $pl(\Omega_{13}) = 0.653$, $pl(\Omega_{14}) = 0.619$, et $pl(\Omega_{34}) = 0.867$. Les fonctions de masse dénormalisées sont définies par $m_{12} = m^*_{12}$, $m_{13} = m^*_{13}$, $m_{23} = m^*_{23}$, et :

$$\begin{array}{lllllllll} m_{13}(\emptyset) & = & 0.347, & m_{14}(\emptyset) & = & 0.381, & m_{34}(\emptyset) & = & 0.132, \\ m_{13}(\{\omega_3\}) & = & 0.653, & m_{14}(\{\omega_3\}) & = & 0.619, & m_{34}(\{\omega_3\}) & = & 0.701, \\ & & & & & & m_{34}(\{\omega_4\}) & = & 0.167. \end{array}$$

Leur combinaison donne la fonction de masse suivante :

$$\begin{array}{lllllllll} \widehat{m}^{\Omega}(\{\omega_2\}) & = & 0.169, & \widehat{m}^{\Omega}(\{\omega_2,\omega_3\}) & = & 0.342, & \widehat{m}^{\Omega}(\{\omega_3,\omega_4\}) & = & 0.209. \\ \widehat{m}^{\Omega}(\{\omega_1,\omega_3\}) & = & 0.057, & \widehat{m}^{\Omega}(\{\omega_2,\omega_4\}) & = & 0.223, & & & \end{array}$$

La probabilité pignistique associée est définie par $BetP^*(\omega_1) = 0.028$, $BetP^*(\omega_2) = 0.452$, $BetP^*(\omega_3) = 0.304$, $BetP^*(\omega_4) = 0.216$. Remarquons que l'attribution d'une masse de croyance non nulle à $\{\omega_1,\omega_3\}$, $\{\omega_2,\omega_3\}$, $\{\omega_2,\omega_4\}$ et $\{\omega_3,\omega_4\}$ reflète la difficulté de distinguer la classe d'appartenance de $\mathbf{x}$. À présent, $\mathbf{x}$ est affecté à ω_2, l'influence du classifieur $\mathcal{E}_{34}$ sur la décision finale ayant été amoindrie par l'estimation des plausibilités d'appartenance. □

3.3 Cas de classifieurs binaires probabilistes

Un classifieur binaire $\mathcal{E}_{ij}$ probabiliste fournit une distribution de probabilité définie sur Ω_{ij}. Soit r^*_{ij} la probabilité de la classe ω_i fournie par $\mathcal{E}_{ij}$, et $r^*_{ji} = 1 - r^*_{ij}$ (les notations du paragraphe 2.2.1 ont été adaptées pour insister sur le caractère normalisé des sorties de $\mathcal{E}_{ij}$). On peut l'interpréter comme une fonction de masse m^*_{ij} Bayésienne, définie par $m^*_{ij}(\{\omega_i\}) = r^*_{ij}$, $m^*_{ij}(\{\omega_j\}) = r^*_{ji}$. Ces m^*_{ij} peuvent alors être combinées en utilisant la méthode définie par (3.14)-(3.16), après estimation de la pertinence $pl(\Omega_{ij})$ des $\mathcal{E}_{ij}$.

Par ailleurs, les sorties des $\mathcal{E}_{ij}$ peuvent aussi être interprétées comme des estimations des probabilités pignistiques $BetP^*_{ij}$ associées aux conditionnements $m^\Omega[\Omega_{ij}]$:

$$r^*_{ij} = BetP^*_{ij}(\omega_i), \tag{3.19}$$
$$r^*_{ji} = BetP^*_{ij}(\omega_j). \tag{3.20}$$

On rappelle que $BetP^*_{ij}$ est définie par :

$$BetP^*_{ij}(\omega_i) = \frac{BetP_{ij}(\omega_i)}{1 - m^\Omega[\Omega_{ij}](\emptyset)}, \tag{3.21}$$
$$BetP^*_{ij}(\omega_j) = \frac{BetP_{ij}(\omega_j)}{1 - m^\Omega[\Omega_{ij}](\emptyset)}; \tag{3.22}$$

ici, $BetP_{ij}$ est la distribution de probabilité pignistique « non normalisée », calculée à partir de $m^\Omega[\Omega_{ij}]$, définie par :

$$BetP_{ij}(\omega_i) = m^\Omega[\Omega_{ij}](\{\omega_i\}) + \frac{m^\Omega[\Omega_{ij}](\Omega_{ij})}{2} \tag{3.23}$$
$$BetP_{ij}(\omega_j) = m^\Omega[\Omega_{ij}](\{\omega_j\}) + \frac{m^\Omega[\Omega_{ij}](\Omega_{ij})}{2}. \tag{3.24}$$

en faisant apparaître explicitement l'ignorance des $\mathcal{E}_{ij}$ dans les équations (3.19)-(3.20), on obtient :

$$r^*_{ij}(1 - m[\Omega_{ij}](\emptyset)) = BetP_{ij}(\omega_i) \tag{3.25}$$
$$r^*_{ji}(1 - m[\Omega_{ij}](\emptyset)) = BetP_{ij}(\omega_j). \tag{3.26}$$

L'estimation de la pertinence $pl(\Omega_{ij})$ des $\mathcal{E}_{ij}$ permet de calculer les estimations non normalisées $r_{ij} = pl(\Omega_{ij}) r^*_{ij}$. De même que précédemment, on peut alors rechercher la fonction de masse $\widehat{m}^\Omega$ la plus consistante possible avec les r_{ij}, en imposant en outre que chaque masse $m^\Omega[\Omega_{ij}](\emptyset)$ soit proche de l'estimation $m_{ij}(\emptyset) = 1 - pl(\Omega_{ij})$ correspondante :

$$\widehat{m}^\Omega = \arg\min_{m^\Omega} \sum_{\Omega_{ij} \subseteq \Omega} (BetP_{ij}(\omega_i) - r_{ij})^2 + (BetP_{ij}(\omega_j) - r_{ji})^2 + \left(m^\Omega[\Omega_{ij}](\emptyset) - 1 + pl(\Omega_{ij})\right)^2, \tag{3.27}$$

sous les contraintes (3.15) et (3.16). Cette méthode sera appelée par la suite méthode MCTProb.

Exemple 3.5 Considérons un problème à trois classes $\{\omega_1, \omega_2, \omega_3\} = \Omega$. Supposons que des classifieurs probabilistes ont donné les probabilités conditionnelles : $r^*_{12} = 0.843$, $r^*_{13} = 0.810$ et $r^*_{23} = 0.665$, et les classifieurs à une classe, les estimations suivantes des plausibilités d'appartenance : $pl^{\Omega}(\{\omega_1\}) = 1$, $pl^{\Omega}(\{\omega_2\}) = 0.444$, $pl^{\Omega}(\{\omega_3\}) = 0.624$.

En utilisant la t-conorme probabiliste définie par (3.18), on obtient :

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

desquelles nous déduisons $r_{12} = r^*_{12}$, $r_{21} = 1 - r^*_{12}$; $r_{13} = r^*_{13}$, $r_{31} = 1 - r^*_{13}$; et

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

La minimisation de (3.27) donne la fonction de masse suivante :

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

toutes les autres masses étant nulles. En conditionnant $\widehat{m}^{\Omega}$ sur chaque Ω_{ij}, et en calculant les probabilités pignistiques correspondantes, on obtient $BetP^*_{12}(\omega_1) = 0.768$, $BetP^*_{13}(\omega_1) = 0.843$, et $BetP^*_{23}(\omega_2) = 0.597$, qui approchent les sorties des $\mathcal{E}_{ij}$. □

La figure 3.10 montre les masses $\widehat{m}^{\Omega}(\{\omega_3\})$, $\widehat{m}^{\Omega}(\{\omega_4\})$, $\widehat{m}^{\Omega}(\{\omega_3, \omega_4\})$ et $\widehat{m}^{\Omega}(\{\omega_2, \omega_3, \omega_4\})$, obtenues en combinant les r^*_{ij} et les $pl(\Omega_{ij})$ par la méthode MCTProb1-1. On constate que les éléments focaux non singletons reçoivent plus de masse par la méthode MCTProb1-1 que par la méthode MCTCorr1-1 : la masse donnée à $\{\omega_3, \omega_4\}$ est plus élevée, et l'élément $\{\omega_2, \omega_3, \omega_4\}$ reçoit une masse significative, tandis qu'elle est nulle dans le cas de la méthode MCTCorr1-1.

Les probabilités pignistiques correspondantes $BetP^*(\omega_3)$ et $BetP^*(\omega_4)$ sont représentées sur la figure 3.11. Soulignons que bien que les masses de croyance calculées par les deux méthodes soient différentes, les probabilités pignistiques correspondantes présentent une certaine ressemblance.

3.4 Réduction de la complexité

Le nombre de sous-ensembles de Ω augmente exponentiellement avec $K = |\Omega|$; cela constitue un obstacle à la résolution de problèmes de classification comptant un grand nombre de classes. Ainsi, pour un problème de reconnaissance de caractères comptant $K = 26$ classes, le nombre d'éléments focaux de la fonction de masse m^{Ω} modélisant la connaissance de la classe réelle d'un vecteur évalué, peut atteindre $2^{26} = 67108864$. Nous proposons de réduire cette complexité en limitant le nombre d'éléments focaux de la fonction de masse $\widehat{m}^{\Omega}$ recherchée.

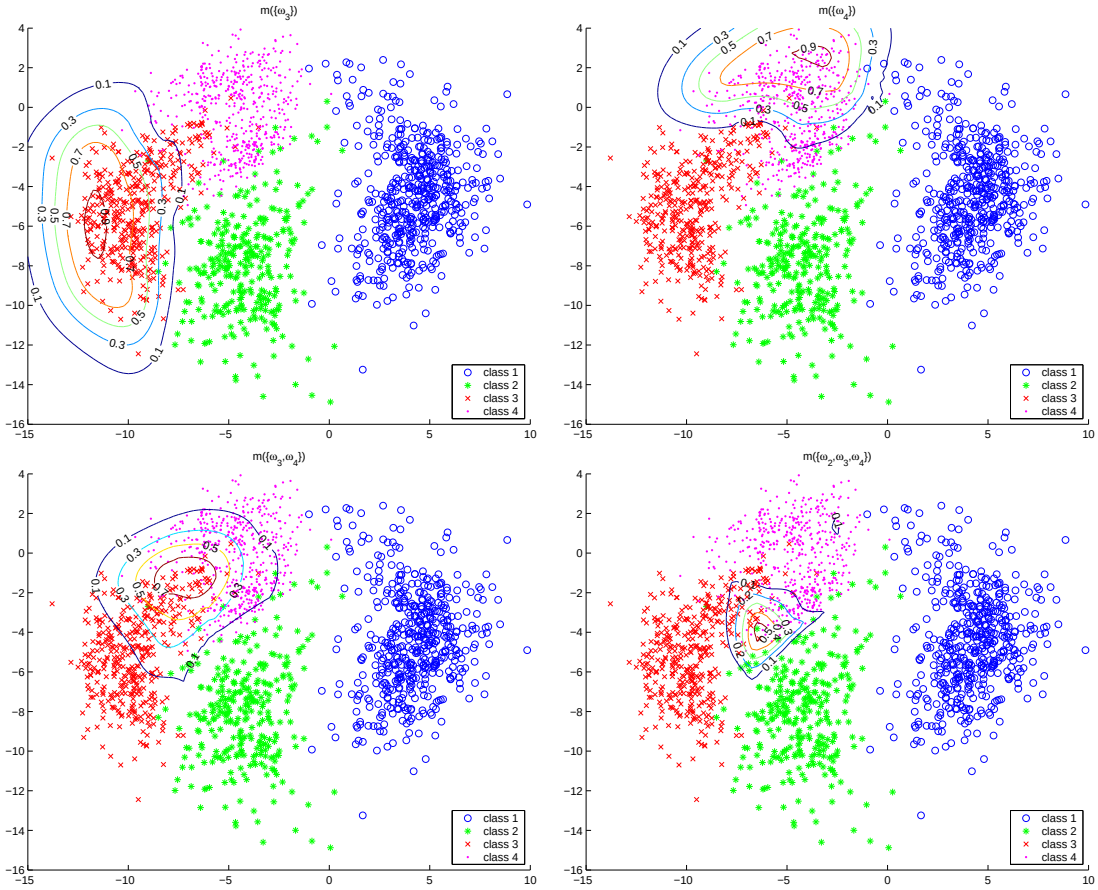


FIG. 3.10 – Courbes de niveau des masses $\widehat{m}^{\Omega}(\{\omega_3\})$ (haut-gauche), $\widehat{m}^{\Omega}(\{\omega_4\})$ (haut-droite), $\widehat{m}^{\Omega}(\{\omega_3,\omega_4\})$ (bas-gauche), et $\widehat{m}^{\Omega}(\{\omega_2,\omega_3,\omega_4\})$ (bas-droite), obtenues par la méthode MCTProb1-1.

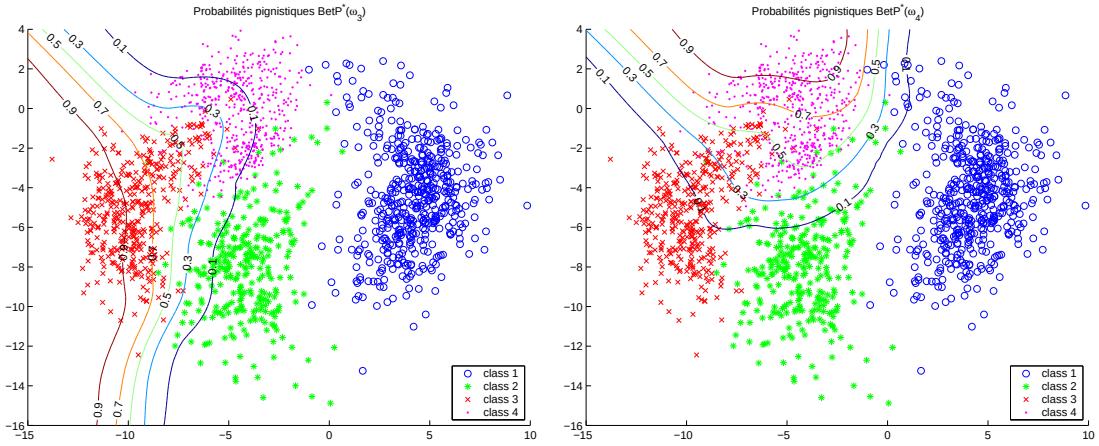


FIG. 3.11 – Courbes de niveau des probabilités pignistiques $BetP^*(\omega_3)$ (gauche) et $BetP^*(\omega_4)$ (droite), obtenues par la méthode MCTProb1-1.

Cela peut être fait en identifiant les $L \leq K$ classes ω_i de plus grande plausibilité $pl^\Omega(\{\omega_i\})$, et en traitant les $K - L$ autres comme une classe unique. Soient $\omega_{(1)}, \ldots, \omega_{(K)}$ les classes ordonnées par plausibilités décroissantes, c'est-à-dire vérifiant :

$$pl^\Omega(\{\omega_{(1)}\}) \geq \cdots \geq pl^\Omega(\{\omega_{(K)}\}). \tag{3.28}$$

Soient $\theta_i = \{\omega_{(i)}\}$, pour $i \in \{1, \ldots, L\}$, et $\theta_{L+1} = \{\omega_{(L+1)}, \ldots, \omega_{(K)}\}$. L'ensemble $\Theta = \{\theta_1, \ldots, \theta_{L+1}\}$ constitue une partition de Ω. Les classes de θ_{L+1} n'ont pas besoin d'être discernées les unes des autres : il est peu plausible que l'une d'elles soit la classe de $\mathbf{x}$. Nous définissons ainsi les sous-ensembles de Θ comme étant les éléments focaux potentiels de m^Ω ; le nombre de variables du problème d'optimisation est donc réduit de 2^K à 2^{L+1}.

Exemple 3.6 Considérons un problème $\Omega = \{\omega_1, \omega_2, \omega_3, \omega_4, \omega_5\}$. Soient $pl^\Omega(\{\omega_1\}) = 0$, $pl^\Omega(\{\omega_2\}) = 0.1$, $pl^\Omega(\{\omega_3\}) = 0.7$, $pl^\Omega(\{\omega_4\}) = 0.3$ et $pl^\Omega(\{\omega_5\}) = 0.05$ les plausibilités calculées en évaluant un vecteur $\mathbf{x}$. Supposons que l'on souhaite conserver les deux classes pour lesquelles la plausibilité d'appartenance est la plus élevée.

On définit donc $\theta_1 = \{\omega_3\}$, $\theta_2 = \{\omega_4\}$ et $\theta_3 = \{\omega_1, \omega_2, \omega_5\}$, et on autorise $\widehat{m}^\Omega$ à attribuer de la masse aux huit éléments suivants : $\emptyset$, θ_1, θ_2, θ_3, $\theta_1 \cup \theta_2 = \{\omega_3, \omega_4\}$, $\theta_1 \cup \theta_3 = \{\omega_1, \omega_2, \omega_3, \omega_5\}$, $\theta_2 \cup \theta_3 = \{\omega_1, \omega_2, \omega_4, \omega_5\}$, et Ω. □

Lorsque les plausibilités $pl^\Omega(\{\omega_k\})$ ne sont pas estimées pour les besoins de la méthode, on propose de les calculer à partir des fonctions de masse m_{ij}. Les fonctions de masse m_{ij} sont tout d'abord déconditionnées sur le cadre Ω :

$$\begin{aligned} m_{ij}^\Omega(\overline{\{\omega_j\}}) &= m_{ij}^*(\{\omega_i\}), \\ m_{ij}^\Omega(\overline{\{\omega_i\}}) &= m_{ij}^*(\{\omega_j\}), \\ m_{ij}^\Omega(\Omega) &= m_{ij}^*(\Omega_{ij}). \end{aligned}$$

La plausibilité pl_{ij}^{Ω} associée à une fonction de masse m_{ij}^{Ω} dénormalisée est obtenue par :

$$\begin{aligned} pl_{ij}^{\Omega}(\{\omega_i\}) &= m_{ij}^{\Omega}(\overline{\{\omega_j\}}) + m_{ij}^{\Omega}(\Omega), \\ pl_{ij}^{\Omega}(\{\omega_j\}) &= m_{ij}^{\Omega}(\overline{\{\omega_i\}}) + m_{ij}^{\Omega}(\Omega), \\ pl_{ij}^{\Omega}(\{\omega_k\}) &= 1, \quad \forall k \notin \{i,j\}. \end{aligned}$$

On propose de combiner les différentes fonctions de plausibilité obtenues par l'opérateur moyenne : pour tout $k \in \{1,\ldots,K\}$,

$$pl^{\Omega}(\{\omega_k\}) = \frac{1}{C_K^2} \sum_{j>i} pl_{ij}^{\Omega}(\{\omega_k\}).$$

3.5 Synthèse

Dans ce chapitre, nous avons formalisé la combinaison de classifieurs binaires dans le cadre du MCT, lorsque le schéma de décomposition un-contre-un est employé. Les sorties de chaque classifieur sont interprétées comme des informations définies sur des conditionnements du cadre initial Ω : la connaissance modélisée par de telles fonctions sont donc incomplètes, certaines hypothèses sur la classe de l'individu $\mathbf{x}$ évalué étant ignorées.

Les classifieurs sont combinées en calculant la fonction de masse $\widehat{m}^{\Omega}$ la plus consistante avec les informations disponibles, par résolution d'un problème d'optimisation quadratique. Nous proposons plusieurs méthodes suivant le nombre et la nature de ces informations. Les fonctions de masse fournies par des classifieurs crédaux peuvent être directement combinées, ou préalablement dénormalisées au moyen de plausibilités d'appartenance, de manière à modéliser la pertinence des classifieurs. Une variante permet de combiner des classifieurs probabilistes, dont les sorties sont vues comme des estimations de probabilités pignistiques.

La complexité peut être réduite en limitant le nombre d'éléments focaux de $\widehat{m}^{\Omega}$, en identifiant les classes les plus plausibles et en aggrégeant les autres en une classe unique.

Chapitre 4

Combinaison de classifieurs binaires dans le cadre du MCT : décompositions un-contre-tous et par codes correcteurs d'erreurs

Dans ce chapitre, nous formalisons la combinaison de classifieurs binaires dans le cadre du MCT, lorsque le problème est décomposé suivant un schéma un-contre-tous (ou 1-T), puis suivant un schéma par codes correcteurs d'erreurs (ou CCE).

Dans le premier cas, chaque dichotomie de Ω est obtenue en opposant une classe ω_k à l'ensemble des autres. Un classifieur $\mathcal{E}_k$ est alors entraîné à séparer $\{\omega_k\}$ de $\overline{\{\omega_k\}}$, à partir de tous les vecteurs d'apprentissage. Les informations qu'il fournit sont donc incomplètes, dans la mesure où $\mathcal{E}_k$ est incapable de discerner les unes des autres les classes du sous-ensemble $\overline{\{\omega_k\}}$. Ces informations sont également indistinctes, tous les classifieurs ayant été entraînés à partir de la totalité des individus d'apprentissage.

Dans le cas d'une décomposition CCE, chaque classifieur $\mathcal{E}_i$ est entraîné à reconnaître deux groupes de classes A_i^+ et A_i^- l'un de l'autre. Le classifieur est donc incapable de discerner les classes au sein de chaque groupe ; il peut de plus ignorer certaines classes de Ω, qui n'appartiennent ni à A_i^+ ni à A_i^-. Dans ce schéma de décomposition, on retrouve donc le caractère incomplet des informations manipulées dans les cas de décompositions 1-1 et 1-T.

Comme dans le chapitre 3, les méthodes présentées sont accompagnées d'exemples numériques et visuels tirés du jeu de données synthétiques Synth. Les classifieurs binaires utilisés ici sont les arbres de décision et les réseaux de neurones évidentiels, décrits au paragraphe 5.1.2 du chapitre 5. Afin de faciliter l'appréciation des propriétés des méthodes, les graphiques présentés dans ce chapitre ont été obtenus à partir d'arbres de décision simples, élagués de manière à avoir une profondeur inférieure ou égale à 2.

4.1 Combinaison de classifieurs dans le cas d'une décomposition 1-T

Nous considérons dans ce paragraphe le cas d'une décomposition 1-T : les dichotomies de Ω sont obtenues en opposant chaque classe ω_k à l'ensemble $\overline{\omega_k}$ des autres. Un classifieur $\mathcal{E}_k$ est alors entraîné à séparer ω_k de $\overline{\omega_k}$, à partir de tous les vecteurs d'apprentissage.

4.1.1 Les sorties des classifieurs vues comme des réductions extérieures sur des cadres grossiers

Lors de l'apprentissage du classifieur $\mathcal{E}_k$, aucune distinction n'est faite entre les classes appartenant au sous-ensemble $\overline{\{\omega_k\}}$. Les sorties fournies par $\mathcal{E}_k$ lors de l'évaluation d'un vecteur $\mathbf{x}$ peuvent donc être interprétées comme une fonction de masse $m_k^{\Theta_k}$, définie sur un grossissement Θ_k du cadre initial Ω [58, 46] :

$$\Theta_k = \left\{\theta_k^+, \theta_k^-\right\}; \tag{4.1}$$

$$\rho_k\left(\{\theta_k^+\}\right) = \{\omega_k\}, \tag{4.2}$$

$$\rho_k\left(\{\theta_k^-\}\right) = \overline{\{\omega_k\}}, \tag{4.3}$$

où l'application ρ_k est le raffinement permettant de transformer Θ_k en Ω. La figure 4.1 représente le grossissement Θ_k associé au classifieur $\mathcal{E}_k$. Ici, le caractère

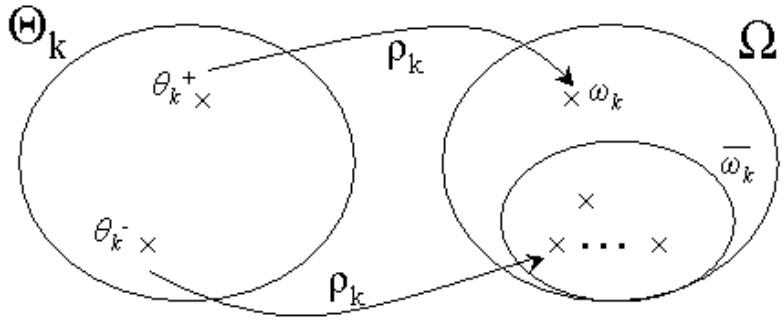


FIG. 4.1 – Grossissement Θ_k associé au classifieur $\mathcal{E}_k$.

partiel de la connaissance quantifiée par $m_k^{\Theta_k}$ se traduit donc par le fait que le degré de granularité de Θ_k ne permet pas de discerner les classes $\omega_l \in \overline{\{\omega_k\}}$.

Le tableau 4.1 illustre les correspondances entre les éléments focaux $A \subseteq \Omega$ d'une fonction de masse m^Ω et ceux de sa réduction intérieure $\underline{m}^{\Theta_k}$ sur Θ_k. Il montre en particulier qu'une partie de la masse $\underline{m}^{\Theta_k}(\{\theta_k^+\})$ peut provenir d'un élément $\{\omega_k\} \cup C$, $C \subset \overline{\{\omega_k\}}$: on ne peut donc assimiler $\underline{m}^{\Theta_k}(\{\theta_k^+\})$ à la croyance que $\mathbf{x}$ appartient à ω_k. De plus, une partie de $\underline{m}^{\Theta_k}(\emptyset)$ pouvant avoir été transférée d'un élément $C \subset \overline{\{\omega_k\}}$, $\underline{m}^{\Theta_k}(\emptyset)$ ne peut être assimileée à la croyance que $\mathbf{x}$

TAB. 4.1 – Correspondance entre les éléments focaux de m^Ω et $\underline{m}^{\Theta_k}$

éléments focaux B de $\underline{m}^{\Theta_k}$	éléments focaux de m^Ω dont provient $\underline{m}^{\Theta_k}(B)$
$\emptyset$	$\{A \subseteq \Omega : \omega_k \notin A, \overline{\omega_k} \nsubseteq A\}$
θ_k^+	$\{A \subseteq \Omega : \omega_k \in A, \overline{\omega_k} \nsubseteq A\}$
θ_k^-	$\{A \subseteq \Omega : \omega_k \notin A, \overline{\omega_k} \subseteq A\} = \overline{\{\omega_k\}}$
Θ_k	$\{A \subseteq \Omega : \omega_k \in A, \overline{\omega_k} \subseteq A\} = \Omega$

TAB. 4.2 – Correspondance entre les éléments focaux de m^Ω et $\overline{m}^{\Theta_k}$

éléments focaux B de $\overline{m}^{\Theta_k}$	éléments focaux de m^Ω dont provient $\overline{m}^{\Theta_k}(B)$
$\emptyset$	$\{A \subseteq \Omega : \omega_k \notin A, \overline{\omega_k} \cap A = \emptyset\} = \emptyset$
θ_k^+	$\{A \subseteq \Omega : \omega_k \in A, \overline{\omega_k} \cap A = \emptyset\} = \{\omega_k\}$
θ_k^-	$\{A \subseteq \Omega : \omega_k \notin A, \overline{\omega_k} \cap A \neq \emptyset\}$
Θ_k	$\{A \subseteq \Omega : \omega_k \in A, \overline{\omega_k} \cap A \neq \emptyset\}$

n'appartient pas à Ω. Ces propriétés sont dues à la q-consistance de la fonction de masse m^Ω avec sa réduction intérieure $\underline{m}^{\Theta_k}$.

Exemple 4.1 Soit un cadre $\Omega = \{\omega_1, \omega_2, \omega_3, \omega_4\}$ et $\Theta_3 = \{\theta_3^+, \theta_3^-\}$ le grossissement de Ω défini par $\rho_3(\{\theta_3^+\}) = \{\omega_3\}$, $\rho_3(\{\theta_3^-\}) = \overline{\{\omega_3\}}$. Soit une masse m^Ω définie par :

$$\begin{array}{lcllcl} m^\Omega(\{\omega_1\}) & = & 0.10, & m^\Omega(\{\omega_3\}) & = & 0.25, \\ m^\Omega(\{\omega_1,\omega_2\}) & = & 0.15, & m^\Omega(\{\omega_1,\omega_3\}) & = & 0.20, \\ m^\Omega(\{\omega_1,\omega_2,\omega_4\}) & = & 0.10, & m^\Omega(\Omega) & = & 0.20; \end{array}$$

la réduction intérieure $\underline{m}_3^{\Theta_3}$ de m^Ω sur Θ_3 est définie par :

$$\begin{array}{lcl} \underline{m}_3^{\Theta_3}(\emptyset) & = & m^\Omega(\{\omega_1\}) + m^\Omega(\{\omega_1,\omega_2\}) = 0.25, \\ \underline{m}_3^{\Theta_3}(\{\theta_3^+\}) & = & m^\Omega(\{\omega_3\}) + m^\Omega(\{\omega_1,\omega_3\}) = 0.45, \\ \underline{m}_3^{\Theta_3}(\{\theta_3^-\}) & = & m^\Omega(\{\omega_1,\omega_2,\omega_4\}) = 0.10, \\ \underline{m}_3^{\Theta_3}(\Theta_3) & = & m^\Omega(\Omega) = 0.20. \end{array}$$

□

Le tableau 4.2 illustre les correspondances entre les éléments focaux $A \subseteq \Omega$ de la fonction de masse m^Ω et ceux de sa réduction extérieure $\overline{m}^{\Theta_k}$ sur Θ_k. On constate à présent que la masse $\overline{m}^{\Theta_k}(\emptyset)$ ne peut provenir que de l'élément $\emptyset$, et peut donc être assimilée à la croyance que $\mathbf{x}$ n'appartient pas à Ω ; de même, la masse $\overline{m}^{\Theta_k}(\{\theta_k^+\})$ étant nécessairement transférée de l'élément $\{\omega_k\}$, elle peut être vue

comme la croyance que $\mathbf{x}$ appartient à ω_k. En outre, les ensembles $\{\theta_k^-\}$ et Θ étant respectivement associés aux éléments $C \subseteq \overline{\{\omega_k\}}$ et $\{\omega_k\} \cup C$ (avec $C \subseteq \overline{\{\omega_k\}}$), les masses $\overline{m}^{\Theta_k}(\{\theta_k^-\})$ et $\overline{m}^{\Theta_k}(\Theta_k)$ peuvent être mises en correspondance avec la croyance que $\mathbf{x}$ appartient à l'une des classes $\omega_l \in \overline{\{\omega_k\}}$, et l'ignorance totale quant à l'appartenance de $\mathbf{x}$. Ces correspondances entre les éléments focaux de m^Ω et $\overline{m}^{\Theta_k}$ reflètent la propriété de b-consistance vérifée par ces deux fonctions de masse.

Exemple 4.2 Reprenons l'exemple 4.1. La réduction extérieure $\overline{m}_3^{\Theta_3}$ de m^Ω sur Θ_3 est définie par :

$$\begin{array}{lcl}
\overline{m}_3^{\Theta_3}(\emptyset) & = & m^\Omega(\emptyset) = 0, \\
\overline{m}_3^{\Theta_3}(\{\theta_3^+\}) & = & m^\Omega(\{\omega_3\}) = 0.25, \\
\overline{m}_3^{\Theta_3}(\{\theta_3^-\}) & = & m^\Omega(\{\omega_1\}) + m^\Omega(\{\omega_1,\omega_2\}) + m^\Omega(\{\omega_1,\omega_2,\omega_4\}) = 0.35, \\
\overline{m}_3^{\Theta_3}(\Theta_3) & = & m^\Omega(\{\omega_1,\omega_3\}) + m^\Omega(\Omega) = 0.40.
\end{array}$$

□

En conclusion, la réduction intérieure $\underline{m}^{\Theta_k}$ est trop peu conservative pour correspondre à l'attribution des masses faites par le classifieur $\mathcal{E}_k$. En revanche, plusieurs arguments intuitifs et théoriques justifient l'interprétation de $m_k^{\Theta_k}$ comme l'estimation de la réduction extérieure $\overline{m}^{\Theta_k}$ d'une fonction de masse m^Ω, qui représente la connaissance de l'appartenance de l'individu évalué $\mathbf{x}$ aux classes.

4.1.2 Combinaison des masses fournies par les différents classifieurs

Soit $\overline{\theta}_k$ l'opérateur de réduction extérieure d'une fonction de masse sur Θ_k. Formellement, on a :

$$m_k^{\Theta_k} = \overline{\theta}_k\left(m^\Omega\right), \quad \forall k \in \{1,\ldots,K\}, \tag{4.4}$$

c'est-à-dire, pour tout $B \subseteq \Theta_k$:

$$m_k^{\Theta_k}(B) \;=\; \sum_{A\subseteq\Omega:\rho_k(B)\cap A\neq\emptyset} m^\Omega(A). \tag{4.5}$$

L'opérateur de réduction extérieure est donc linéaire. Chaque cadre Θ_k comptant quatre sous-ensembles ($\emptyset$, $\{\theta_k^+\}$, $\{\theta_k^-\}$, et Θ_k), la relation (4.4) définit donc un système linéaire de $4 \times K$ équations à $2^K - 1$ inconnues. Comme dans le cas d'une décomposition 1-1 considéré au chapitre 3, les fonctions de masse m_k ne sont pas indépendantes : il n'est donc pas correct de les combiner au moyen de la somme conjonctive. De même, elles ne sont généralement pas consistantes : il n'existe alors pas de fonction de masse m^Ω dont les réductions extérieures $\overline{m}^{\Theta_k} = \overline{\theta}_k(m^\Omega)$

correspondent exactement aux $m_k^{\Theta_k}$, pour tout k. On peut donc calculer une solution approchée du système (4.4), en résolvant un problème d'optimisation quadratique :

$$\widehat{m}^{\Omega} = \arg\min_{m^{\Omega}} \sum_{k=1}^{K} \sum_{B \subseteq \Theta_k} \left(\overline{\theta}_k \left(m^{\Omega} \right) (B) - m_k^{\Theta_k}(B) \right)^2, \tag{4.6}$$

sous les contraintes :

$$\begin{aligned} m^{\Omega}(A) &\geq 0, \quad \forall\, A \subseteq \Omega & (4.7) \\ \sum_{A \subseteq \Omega} m^{\Omega}(A) &= 1. & (4.8) \end{aligned}$$

Ce problème peut être résolu au moyen d'un algorithme usuel d'optimisation quadratique. La méthode ainsi définie sera appelée par la suite méthode MCT1-T.

Exemple 4.3 Soit un problème $\Omega = \{\omega_1, \omega_2, \omega_3\}$. On dispose de trois classifieurs binaires $\mathcal{E}_1$, $\mathcal{E}_2$ et $\mathcal{E}_3$, séparant respectivement $\{\omega_1\}$ de $\{\omega_2, \omega_3\}$, $\{\omega_2\}$ de $\{\omega_1, \omega_3\}$ et $\{\omega_3\}$ de $\{\omega_1, \omega_2\}$. Supposons que, lors de l'évaluation d'un vecteur $\mathbf{x}$, ces classifieurs ont fourni les fonctions de masse (normalisées) suivantes :

$$\begin{array}{lllllllll} m_1^{\Theta_1}(\{\theta_1^+\}) &=& 0.008, & m_2^{\Theta_2}(\{\theta_2^+\}) &=& 0.652, & m_3^{\Theta_3}(\{\theta_3^+\}) &=& 0.436, \\ m_1^{\Theta_1}(\{\theta_1^-\}) &=& 0.991, & m_2^{\Theta_2}(\{\theta_2^-\}) &=& 0.338, & m_3^{\Theta_3}(\{\theta_3^-\}) &=& 0.556, \\ m_1^{\Theta_1}(\Theta_1) &=& 0.001; & m_2^{\Theta_2}(\Theta_2) &=& 0.010; & m_3^{\Theta_3}(\Theta_3) &=& 0.008. \end{array}$$

En utilisant la méthode de combinaison présentée ci-dessus, on obtient la fonction de masse $\widehat{m}^{\Omega}$ suivante :

$$\begin{array}{llll} \widehat{m}^{\Omega}(\{\omega_2\}) &=& 0.604, & \widehat{m}^{\Omega}(\{\omega_3\}) &=& 0.387, \\ \widehat{m}^{\Omega}(\{\omega_2, \omega_3\}) &=& 0.004, & \widehat{m}^{\Omega}(\Omega) &=& 0.005. \end{array}$$

Les réductions extérieures de cette fonction de masse sur les différents Θ_k donnent :

$$\begin{array}{lllllllll} \widehat{m}_1^{\Theta_1}(\{\theta_1^+\}) &=& 0.000, & \widehat{m}_2^{\Theta_2}(\{\theta_2^+\}) &=& 0.604, & \widehat{m}_3^{\Theta_3}(\{\theta_3^+\}) &=& 0.387, \\ \widehat{m}_1^{\Theta_1}(\{\theta_1^-\}) &=& 0.995, & \widehat{m}_2^{\Theta_2}(\{\theta_2^-\}) &=& 0.387, & \widehat{m}_3^{\Theta_3}(\{\theta_3^-\}) &=& 0.604, \\ \widehat{m}_1^{\Theta_1}(\Theta_1) &=& 0.005; & \widehat{m}_2^{\Theta_2}(\Theta_2) &=& 0.009; & \widehat{m}_3^{\Theta_3}(\Theta_3) &=& 0.009. \end{array}$$

qui sont les meilleures approximations de $m_1^{\Theta_1}$, $m_2^{\Theta_2}$ et $m_3^{\Theta_3}$ au sens du critère (4.6). □

La figure 4.2 montre les masses Bayésiennes $m_1^{\Theta_1}(\{\theta_1^+\})$, $m_2^{\Theta_2}(\{\theta_2^+\})$, $m_3^{\Theta_3}(\{\theta_3^+\})$ et $m_4^{\Theta_4}(\{\theta_4^+\})$, calculées au moyen d'arbres de décision binaires. Rappelons que la fonction de masse $m_k^{\Theta_k}$ est exprimée sur un cadre grossier défini par $\Theta_k =$

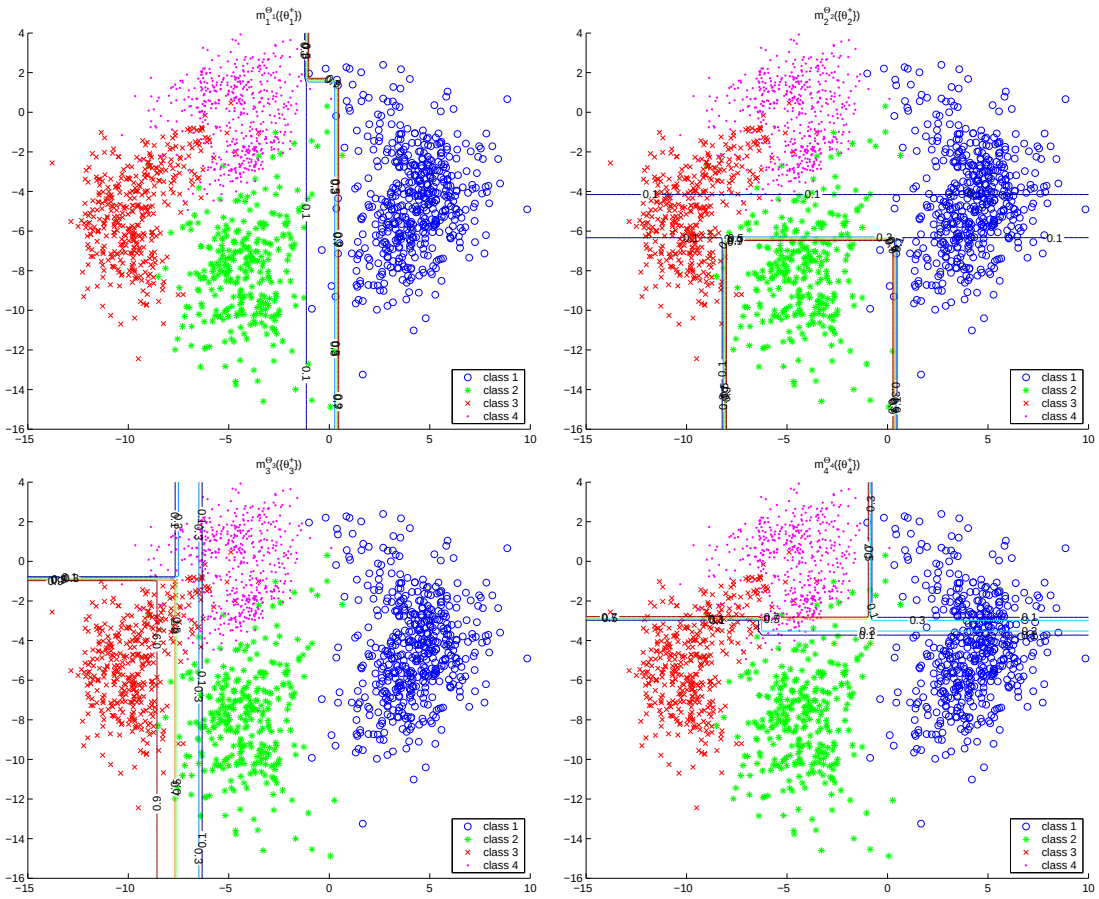


FIG. 4.2 – Courbes de niveau des masses $m_1^{\Theta_1}(\{\theta_1^+\})$, $m_2^{\Theta_2}(\{\theta_2^+\})$, $m_3^{\Theta_3}(\{\theta_3^+\})$ et $m_4^{\Theta_4}(\{\theta_4^+\})$, calculées au moyen d'arbres de décision.

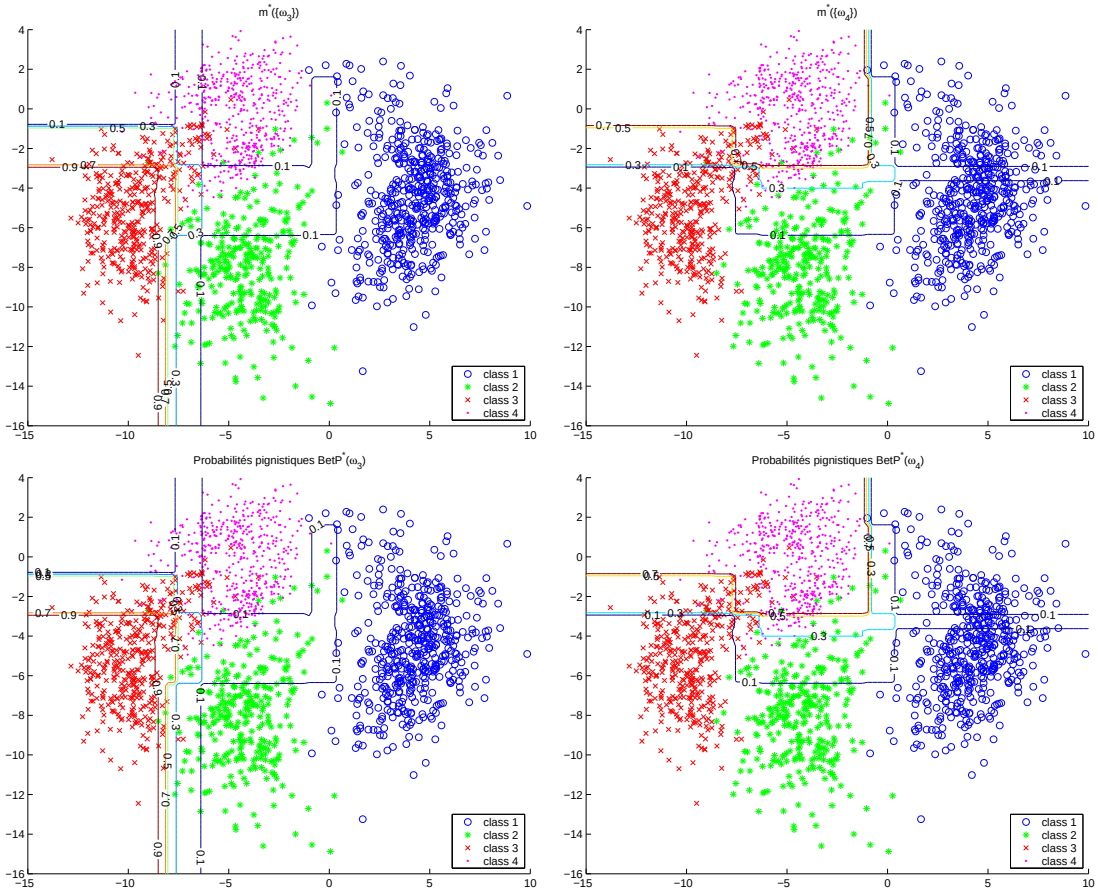


FIG. 4.3 – Courbes de niveau des masses $\widehat{m}^{\Omega}(\{\omega_3\})$ et $\widehat{m}^{\Omega}(\{\omega_4\})$, obtenues par la méthode MCT1-T, et des probabilités pignistiques $BetP^*(\omega_3)$ et $BetP^*(\omega_4)$ correspondantes.

$\{\theta_k^+, \theta_k^-\}$, $\rho_k(\{\theta_k^+\}) = \{\omega_k\}$, $\rho_k(\{\theta_k^-\}) = \overline{\{\omega_k\}}$, ρ_k étant le raffinement permettant de transformer Θ_k en Ω. Rappelons que la réduction extérieure d'un élément $A \subseteq \Omega$ sur Θ_k est notée $\overline{\theta}_k(A)$.

Les masses $\widehat{m}^{\Omega}(\{\omega_3\})$ et $\widehat{m}^{\Omega}(\{\omega_4\})$, ainsi que les probabilités pignistiques $BetP^*(\omega_3)$ et $BetP^*(\omega_4)$ correspondantes, sont représentées sur la figure 4.3. Les masses fournies par les classifieurs étant Bayésiennes, la fonction de masse obtenue par optimisation n'affecte qu'une croyance négligeable aux éléments focaux non singletons. Les masses obtenues par combinaison étant quasiment Bayésiennes, les probabilités pignistiques correspondantes en sont très proches.

4.2 Combinaison de classifieurs dans le cas général d'une décomposition par codes correcteurs d'erreurs

Dans ce paragraphe, nous considérons le schéma de décomposition par codes correcteurs d'erreurs (CCE) : un ensemble de N dichotomies de Ω est formé, la i$^{\text{e}}$ dichotomie étant obtenue en opposant deux sous-ensembles de classes $A_i^+ \subseteq \Omega$ et $A_i^- \subseteq \Omega$, tels que $A_i^+ \cap A_i^- = \emptyset$ et $A_i^+ \cup A_i^- \subseteq \Omega$ (par la suite, nous adopterons la notation $\Omega_i = A_i^+ \cup A_i^-$). Pour chaque dichotomie, un classifieur binaire $\mathcal{E}_i$ est entraîné à séparer A_i^+ de A_i^-, sur la base des vecteurs d'apprentissage appartenant à $A_i^+ \cup A_i^-$.

Les schémas 1-1 et 1-T peuvent donc être vus comme des cas particuliers du schéma CCE : ils sont formés en considérant toutes les dichotomies telles que $|A_i^+| = |A_i^-| = 1$ dans le premier cas, ou $|A_i^+| = 1$ et $|A_i^-| = K-1$ dans le second.

4.2.1 Interprétation des sorties des classifieurs binaires CCE

Les sorties fournies par le classifieur $\mathcal{E}_i$ lors de l'évaluation d'un vecteur $\mathbf{x}$ peuvent être modélisées par une fonction de masse $m_i^{\Theta_i}$, définie sur un cadre Θ_i. Ce cadre est, par définition, le grossissement d'un conditionnement Ω_i de Ω. La figure 4.4 illustre cette correspondance entre Θ_i, Ω_i et Ω.

Exemple 4.4 Soient un problème $\Omega = \{\omega_1, \omega_2, \omega_3, \omega_4\}$, et la décomposition non-triviale définie par la matrice de codes présentée dans le tableau 4.3. Les sorties du classifieur $\mathcal{E}_4$ peuvent être modélisées par une fonction de masse $m_4^{\Theta_4}$, où Θ_4 est défini par :

$$\begin{aligned} \Theta_4 &= \{\theta_4^+, \theta_4^-\}; \\ \rho_4(\theta_4^+) = A_4^+ &= \{\omega_1, \omega_3\}, \\ \rho_4(\theta_4^-) = A_4^- &= \{\omega_2\}, \\ \Omega_4 = \{A_4^+, A_4^-\} &= \{\omega_1, \omega_2, \omega_3\}. \end{aligned}$$

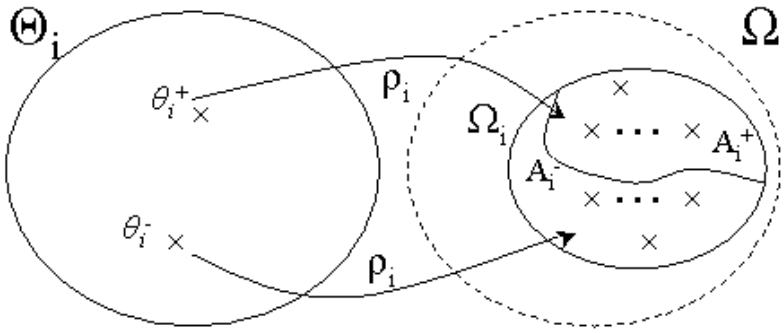


FIG. 4.4 – Grossissement Θ_i d'un conditionnement Ω_i de Ω, associé au classifieur $\mathcal{E}_i$, dans le cas d'une décomposition par codes correcteurs d'erreurs.

TAB. 4.3 – Matrice de codes CCE pour un problème à quatre classes.

	$\mathcal{E}_1$	$\mathcal{E}_2$	$\mathcal{E}_3$	$\mathcal{E}_4$	$\mathcal{E}_5$	$\mathcal{E}_6$	$\mathcal{E}_7$
ω_1	-1	+1	0	+1	0	+1	-1
ω_2	0	+1	+1	-1	0	-1	0
ω_3	0	-1	0	+1	-1	0	+1
ω_4	+1	0	-1	0	+1	0	0

□

Le classifieur $\mathcal{E}_i$ a donc une connaissance partielle de Ω, en ce sens qu'il ne sait reconnaître que les vecteurs dont la classe réelle se trouve dans Ω_i, et ne sait discerner parmi celles-ci que les classes A_i^+ des classes A_i^-.

Soit la fonction de masse m^Ω, quantifiant la connaissance de la classe réelle de $\mathbf{x}$. On peut interpréter $m_i^{\Theta_i}$ comme la réduction d'un conditionnement de m^Ω. Les tableaux 4.4 et 4.5 illustrent les correspondances entre les éléments focaux $A \subseteq \Omega$ de m^Ω et ceux des réductions intérieure $\underline{m}_i^\Theta$ et extérieure $\overline{m}_i^\Theta$, respectivement. Des arguments similaires à ceux avancés dans le cas d'une décomposition 1-T justifient donc l'interprétation de $m_i^{\Theta_i}$ comme estimation de la réduction extérieure

TAB. 4.4 – Correspondance entre les éléments focaux de m^Ω et $\underline{m}^{\Theta_i}$

éléments focaux B de $\underline{m}^{\Theta_i}$	éléments focaux de m^Ω dont provient $\underline{m}^{\Theta_i}(B)$
$\emptyset$	$\left\{A \subset \Omega : A_i^+ \not\subseteq A, A_i^- \not\subseteq A\right\}$
θ_i^+	$\left\{A \subset \Omega : A_i^+ \subseteq A, A_i^- \not\subseteq A\right\}$
θ_i^-	$\left\{A \subset \Omega : A_i^+ \not\subseteq A, A_i^- \subseteq A\right\}$
Θ_i	$\left\{A \subset \Omega : A_i^+ \subseteq A, A_i^- \subseteq A\right\} = \Omega_i$

TAB. 4.5 – Correspondance entre les éléments focaux de m^{Ω} et $\overline{m}^{\Theta_i}$

éléments focaux B de $\overline{m}^{\Theta_i}$	éléments focaux de m^{Ω} dont provient $\overline{m}^{\Theta_i}(B)$
$\emptyset$	$\left\{A \subset \Omega : A_i^+ \cap A = \emptyset, A_i^- \cap A = \emptyset\right\} = \emptyset$
θ_i^+	$\left\{A \subset \Omega : A_i^+ \cap A \neq \emptyset, A_i^- \cap A = \emptyset\right\}$
θ_i^-	$\left\{A \subset \Omega : A_i^+ \cap A = \emptyset, A_i^- \cap A \neq \emptyset\right\}$
Θ_i	$\left\{A \subset \Omega : A_i^+ \cap A \neq \emptyset, A_i^- \cap A \neq \emptyset\right\}$

de m^{Ω}, la réduction intérieure semblant trop peu conservative pour refléter la connaissance, exprimée sur Ω, de l'appartenance de $\mathbf{x}$.

4.2.2 Combinaison des masses fournies par les différents classifieurs

Dans le cas le plus courant où l'algorithme de classification employé n'intègre pas de processus de détection de nouveauté, et lorsqu'un cadre Ω_i est un sous-ensemble strict de Ω, le classifieur $\mathcal{E}_i$ correspondant n'est pas entraîné à reconnaître les vecteurs $\mathbf{x} \notin \Omega_i$. Les sorties de $\mathcal{E}_i$ peuvent alors être interprétées comme des fonctions de masse $m_i^{\Theta_i *}$ normalisées, qui sont des estimations de réductions extérieures de conditionnements normalisés de $\widehat{m}^{\Omega}$:

$$m_i^{\Theta_i} = \overline{\theta}_i \left(m^{\Omega}[\Omega_i]^*\right), \quad \forall i \in \{1, \ldots, N\}. \tag{4.9}$$

L'individu $\mathbf{x}$ est supposé appartenir nécessairement à une classe $\omega \in \Omega_i$, ou de manière équivalente à une classe $\theta \in \Theta_i$. Remarquons que la normalisation de m^{Ω} peut être effectuée indifféremment avant ou après réduction sur Θ_i : comme le montre le tableau 4.5,

$$\overline{m}^{\Theta_i}(\emptyset) = m[\Omega_i](\emptyset). \tag{4.10}$$

Il est clair que la même information est perdue si l'on choisit d'ignorer les classes $\omega \notin \Omega_i$ ou si l'on choisit d'ignorer les classes $\theta \notin \Theta_i$. On a donc, pour tout

$B \subseteq \Theta_i$:

$$\begin{aligned}
\overline{\theta}_i\left(m[\Omega_i]^*\right)(B) &= \sum_{C\subseteq\Omega_i:\rho_i(B)\cap C\neq\emptyset} m^{\Omega}[\Omega_i]^*(C), && (4.11)\\
&= \sum_{C\subseteq\Omega_i:\rho_i(B)\cap C\neq\emptyset} \frac{m^{\Omega}[\Omega_i](C)}{1-m^{\Omega}[\Omega_i](\emptyset)}, && (4.12)\\
&= \frac{1}{1-m^{\Omega}[\Omega_i](\emptyset)} \sum_{C\subseteq\Omega_i:\rho_i(B)\cap C\neq\emptyset} m^{\Omega}[\Omega_i](C), && (4.13)\\
&= \frac{1}{1-\overline{m}^{\Theta_i}(\emptyset)} \sum_{C\subseteq\Omega_i:\rho_i(B)\cap C\neq\emptyset} m^{\Omega}[\Omega_i](C), && (4.14)\\
&= m_i^{\Theta_i *}(B). && (4.15)
\end{aligned}$$

On peut alors combiner les $m_i^{\Theta_i *}$ en résolvant le problème d'optimisation quadratique :

$$\widehat{m^*}^{\Omega} = \arg\min_{m^{\Omega}} \sum_{i=1}^{N} \sum_{B\subseteq\Theta_i} \left(\overline{\theta}_i\left(m[\Omega_i]\right)(B) - m_i^{\Theta_i}(B)\left(1-\overline{\theta}_i\left(m[\Omega_i]\right)(\emptyset)\right)\right)^2; \quad (4.16)$$

sous les contraintes :

$$\begin{aligned}
m^{\Omega}(A) &\geq 0, \quad \forall A\subseteq\Omega, && (4.17)\\
\sum_{A\subseteq\Omega} m^{\Omega}(A) &= 1; && (4.18)\\
m^{\Omega}(\emptyset) &= 0. && (4.19)
\end{aligned}$$

La méthode de combinaison définie par les systèmes (4.16)-(4.19) sera appelée par la suite méthode MCTCCE.

Exemple 4.5 Soient un problème $\Omega = \{\omega_1, \omega_2, \omega_3\}$, et la décomposition définie par la matrice de codes suivante :

	$\mathcal{E}_1$	$\mathcal{E}_2$	$\mathcal{E}_3$	$\mathcal{E}_4$
ω_1	+1	+1	0	+1
ω_2	-1	0	+1	-1
ω_3	0	-1	-1	+1

Supposons que les classifieurs $\mathcal{E}_i$ ont fourni les masses de croyance normalisées suivantes :

$$\begin{array}{llll}
m_1^{\Theta_1 *}(\{\theta_1^+\}) = 0.029, & & m_2^{\Theta_2 *}(\{\theta_2^+\}) = 0.054,\\
m_1^{\Theta_1 *}(\{\theta_1^-\}) = 0.954, & & m_2^{\Theta_2 *}(\{\theta_2^-\}) = 0.898,\\
m_1^{\Theta_1 *}(\Theta_1) = 0.017; & & m_2^{\Theta_2 *}(\Theta_2) = 0.048;
\end{array}$$

$$
\begin{array}{lcllcl}
m_3^{\Theta_3 *}(\{\theta_3^+\}) & = & 0.511, & m_4^{\Theta_4 *}(\{\theta_4^+\}) & = & 0.447, \\
m_3^{\Theta_3 *}(\{\theta_3^-\}) & = & 0.454, & m_4^{\Theta_4 *}(\{\theta_4^-\}) & = & 0.519, \\
m_3^{\Theta_3 *}(\Theta_3) & = & 0.035; & m_4^{\Theta_4 *}(\Theta_4) & = & 0.034.
\end{array}
$$

En utilisant la méthode de combinaison présentée ci-dessus, on obtient la fonction de masse $\widehat{m^*}^{\Omega}$ suivante :

$$
\begin{array}{lcllcllcl}
\widehat{m^*}^{\Omega}(\{\omega_1\}) & = & 0.015, & \widehat{m^*}^{\Omega}(\{\omega_2\}) & = & 0.512, & \widehat{m^*}^{\Omega}(\{\omega_3\}) & = & 0.432, \\
\widehat{m^*}^{\Omega}(\Omega_{13}) & = & 0.007, & \widehat{m^*}^{\Omega}(\Omega_{23}) & = & 0.020, & \widehat{m^*}^{\Omega}(\Omega) & = & 0.014.
\end{array}
$$

□

4.2.3 Estimation de l'ignorance des classifieurs binaires

Comme dans le cas d'une décomposition un-contre-un, nous proposons d'évaluer la plausibilité $pl^{\Omega}(\Omega_i)$ qu'un vecteur $\mathbf{x}$ évalué appartienne aux cadres Ω_i, par une méthode similaire à celle décrite dans le paragraphe 3.1. Nous adopterons par la suite la notation $pl_{\Omega_i} = pl^{\Omega}(\Omega_i)$. Soient $pl^{\Omega}(\{\omega_k\})$ les plausibilités d'appartenance aux classes, fournies par des classifieurs à une classe, et $\odot$ un opérateur de combinaison, correspondant à une conorme triangulaire. Une t-conorme étant commutative et associative, on peut calculer pl_{Ω_i} par :

$$
pl_{\Omega_i} = \bigodot_{k:\omega_k \in \Omega_i} pl(\{\omega_k\}). \tag{4.20}
$$

Ces estimations peuvent ensuite être utilisées pour dénormaliser les fonctions de masse m_i :

$$
m_i^{\Theta_i} = pl_{\Omega_i} m_i^{\Theta_i *}, \quad \forall i \in \{1, \ldots, N\}. \tag{4.21}
$$

Remarquons que lorsqu'un classifieur $\mathcal{E}_i$ est entraîné à partir de la totalité des vecteurs d'apprentissage, le cadre Θ_i correspondant est défini comme un grossissement de Ω. La dénormalisation de la fonction de masse $m_i^{\Theta_i}$ permet alors de déterminer si le vecteur $\mathbf{x}$ évalué ne fait pas partie de l'ensemble Ω, lorsque l'hypothèse du monde ouvert est acceptée.

Remarquons que les opérateurs de conditionnement et de réduction extérieure étant linéaires, leur composition l'est aussi. Soit $\overline{\theta}_i$ l'opérateur de réduction extérieure sur Θ_i d'une fonction de masse conditionnelle $m^{\Omega}[\Omega_i]$:

$$
m_i^{\Theta_i} = \overline{\theta}_i \left(m^{\Omega}[\Omega_i] \right), \quad \forall i \in \{1, \ldots, N\}, \tag{4.22}
$$

c'est-à-dire, pour tout $B \subseteq \Theta_i$:

$$
\begin{aligned}
m_i^{\Theta_i}(B) &= \sum_{C \subseteq \Omega_i : \rho_i(B) \cap C \neq \emptyset} m^{\Omega}[\Omega_i](C), && (4.23)\\
&= \sum_{C \subseteq \Omega_i : \rho_i(B) \cap C \neq \emptyset} \left(\sum_{A \subseteq \Omega : A \cap \Omega_i = C} m^{\Omega}(A) \right), && (4.24)\\
&= \sum_{A \subseteq \Omega : \rho_i(B) \cap A \neq \emptyset} m^{\Omega}(A). && (4.25)
\end{aligned}
$$

Chaque cadre Θ_k comptant quatre sous-ensembles ($\emptyset$, $\{\theta_k^+\}$, $\{\theta_k^-\}$, et Θ_k), la relation (4.22) définit donc un système linéaire de $4 \times K$ équations à $2^K - 1$ inconnues. Toutefois, les fonctions de masse $m_i^{\Theta_i}$ n'étant généralement pas consistantes, ce système n'a en général pas de solution exacte. Une solution approchée peut donc être calculée en résolvant un problème d'optimisation quadratique :

$$
\widehat{m}^{\Omega} = \arg\min_{m^{\Omega}} \sum_{i=1}^{N} \sum_{B \subseteq \Theta_i} \left(\overline{\theta}_i \left(m^{\Omega}[\Omega_i] \right)(B) - m_i^{\Theta_i}(B) \right)^2, \qquad (4.26)
$$

sous les contraintes :

$$
\begin{aligned}
m^{\Omega}(A) &\geq 0, \quad \forall\, A \subseteq \Omega && (4.27)\\
\sum_{A \subseteq \Omega} m^{\Omega}(A) &= 1. && (4.28)
\end{aligned}
$$

Cette méthode, permettant de prendre en compte la pertinence des classifieurs binaires, sera appelée par la suite méthode MCTCorrCCE. Remarquons que cette méthode généralise la méthode MCTCorr1-1 présentée au paragraphe 3.2 du chapitre 3. De même, elle permet également de combiner des classifieurs capables de détecter la nouveauté. Enfin, lorsque la décomposition est telle que chaque classifieur est entraîné à partir de tous les exemples d'apprentissage, on a $\Omega_i = \Omega$, pour tout $i \in \{1, \ldots, N\}$. Les masses normales fournies par des classifieurs binaires n'intégrant pas de détection de nouveauté peuvent alors être directement combinées, sans estimation préalable de la plausibilité d'appartenance aux classes.

Exemple 4.6 Reprenons l'exemple 4.5. Supposons que des classifieurs $\mathcal{E}_i$ ont fourni les masses de croyance normalisées suivantes :

$$
\begin{array}{llll}
m_1^{\Theta_1 *}(\{\theta_1^+\}) = 0.033, & & m_2^{\Theta_2 *}(\{\theta_2^+\}) = 0.059, & \\
m_1^{\Theta_1 *}(\{\theta_1^-\}) = 0.946, & & m_2^{\Theta_2 *}(\{\theta_2^-\}) = 0.883, & \\
m_1^{\Theta_1 *}(\Theta_1) = 0.021; & & m_2^{\Theta_2 *}(\Theta_2) = 0.058; & \\
 & & & \\
m_3^{\Theta_3 *}(\{\theta_3^+\}) = 0.479, & & m_4^{\Theta_4 *}(\{\theta_4^+\}) = 0.458, & \\
m_3^{\Theta_3 *}(\{\theta_3^-\}) = 0.480, & & m_4^{\Theta_4 *}(\{\theta_4^-\}) = 0.502, & \\
m_3^{\Theta_3 *}(\Theta_3) = 0.041; & & m_4^{\Theta_4 *}(\Theta_4) = 0.040. &
\end{array}
$$

Supposons en outre que des classifieurs à une classe ont fourni les plausibilités suivantes : $pl_1 = 0.075$, $pl_2 = 0.754$, $pl_3 = 0.764$. La combinaison de ces plausibilités au moyen de la t-conorme probabiliste définie par (4.20) donne :

$$\begin{array}{rclcrcl} pl_{\Omega_1} & = & 0.772, & & pl_{\Omega_2} & = & 0.782, \\ pl_{\Omega_3} & = & 0.942, & & pl_{\Omega_4} & = & 0.946. \end{array}$$

On en déduit les fonctions de masses dénormalisées :

$$\begin{array}{rclcrcl} m_1^{\Theta_1}(\emptyset) & = & 0.228, & & m_2^{\Theta_2}(\emptyset) & = & 0.218, \\ m_1^{\Theta_1}(\{\theta_1^+\}) & = & 0.025, & & m_2^{\Theta_2}(\{\theta_2^+\}) & = & 0.046, \\ m_1^{\Theta_1}(\{\theta_1^-\}) & = & 0.730, & & m_2^{\Theta_2}(\{\theta_2^-\}) & = & 0.690, \\ m_1^{\Theta_1}(\Theta_1) & = & 0.017; & & m_2^{\Theta_2}(\Theta_2) & = & 0.046; \end{array}$$

$$\begin{array}{rclcrcl} m_3^{\Theta_3}(\emptyset) & = & 0.058, & & m_4^{\Theta_4}(\emptyset) & = & 0.054, \\ m_3^{\Theta_3}(\{\theta_3^+\}) & = & 0.451, & & m_4^{\Theta_4}(\{\theta_4^+\}) & = & 0.433, \\ m_3^{\Theta_3}(\{\theta_3^-\}) & = & 0.452, & & m_4^{\Theta_4}(\{\theta_4^-\}) & = & 0.475, \\ m_3^{\Theta_3}(\Theta_3) & = & 0.039; & & m_4^{\Theta_4}(\Theta_4) & = & 0.038. \end{array}$$

La combinaison de ces masses dénormalisées donne la fonction de masse $\widehat{m}^{\Omega}$ suivante :

$$\begin{array}{rclcrcl} \widehat{m}^{\Omega}(\{\omega_1\}) & = & 0.019, & \widehat{m}^{\Omega}(\{\omega_2\}) & = & 0.414, & \widehat{m}^{\Omega}(\{\omega_3\}) = 0.378, \\ \widehat{m}^{\Omega}(\Omega_{13}) & = & 0.027, & \widehat{m}^{\Omega}(\Omega_{23}) & = & 0.162. & \end{array}$$

En conditionnant cette fonction de masse sur les Ω_i puis en réduisant le résultat sur les différents Θ_i, on obtient :

$$\begin{array}{rclcrcl} \widehat{m}_1^{\Theta_1}(\emptyset) & = & 0.378, & & \widehat{m}_2^{\Theta_2}(\emptyset) & = & 0.414, \\ \widehat{m}_1^{\Theta_1}(\{\theta_1^+\}) & = & 0.045, & & \widehat{m}_2^{\Theta_2}(\{\theta_2^+\}) & = & 0.019, \\ \widehat{m}_1^{\Theta_1}(\{\theta_1^-\}) & = & 0.577, & & \widehat{m}_2^{\Theta_2}(\{\theta_2^-\}) & = & 0.541, \\ \widehat{m}_1^{\Theta_1}(\Theta_1) & = & 0.000; & & \widehat{m}_2^{\Theta_2}(\Theta_2) & = & 0.026; \end{array}$$

$$\begin{array}{rclcrcl} \widehat{m}_3^{\Theta_3}(\emptyset) & = & 0.019, & & \widehat{m}_4^{\Theta_4}(\emptyset) & = & 0.000, \\ \widehat{m}_3^{\Theta_3}(\{\theta_3^+\}) & = & 0.414, & & \widehat{m}_4^{\Theta_4}(\{\theta_4^+\}) & = & 0.424, \\ \widehat{m}_3^{\Theta_3}(\{\theta_3^-\}) & = & 0.405, & & \widehat{m}_4^{\Theta_4}(\{\theta_4^-\}) & = & 0.414, \\ \widehat{m}_3^{\Theta_3}(\Theta_3) & = & 0.162. & & \widehat{m}_4^{\Theta_4}(\Theta_4) & = & 0.162, \end{array}$$

qui sont les meilleures approximations des $m_i^{\Theta_i}$ au sens du critère (4.26). □

La figure 4.5 permet de comparer les masses $\widehat{m}^{\Omega}(\{\omega_3\})$ et $\widehat{m}^{\Omega}(\{\omega_3, \omega_4\})$, obtenues en combinant les fonctions de masse sous-normales fournies par des réseaux de neurones évidentiels en utilisant les méthodes MCTCorr1-1 et MCTCorrCCE. On constate que les masses allouées à $\{\omega_3\}$ sont plus importantes dans le cas de la décomposition CCE ; au contraire, les masses données $\{\omega_3, \omega_4\}$ sont plus importantes dans le cas de la décomposition 1-1. Le nombre de classifieurs binaires,

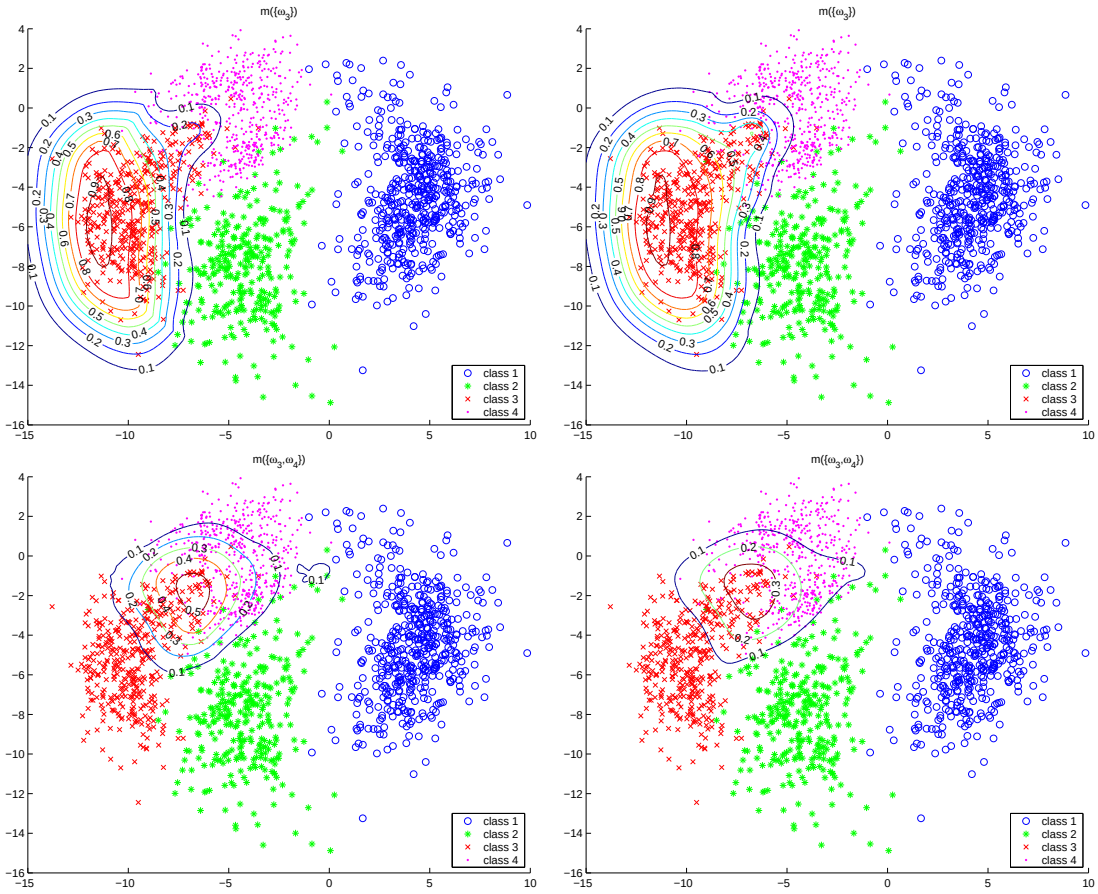


FIG. 4.5 – Courbes de niveau des masses $\widehat{m}^{\Omega}(\{\omega_3\})$ (haut-gauche) et $\widehat{m}^{\Omega}(\{\omega_3, \omega_4\})$ (bas-gauche), obtenues par la méthode MCTCorr1-1, et par la méthode MCTCorrCCE avec matrice de codes creuse (droite).

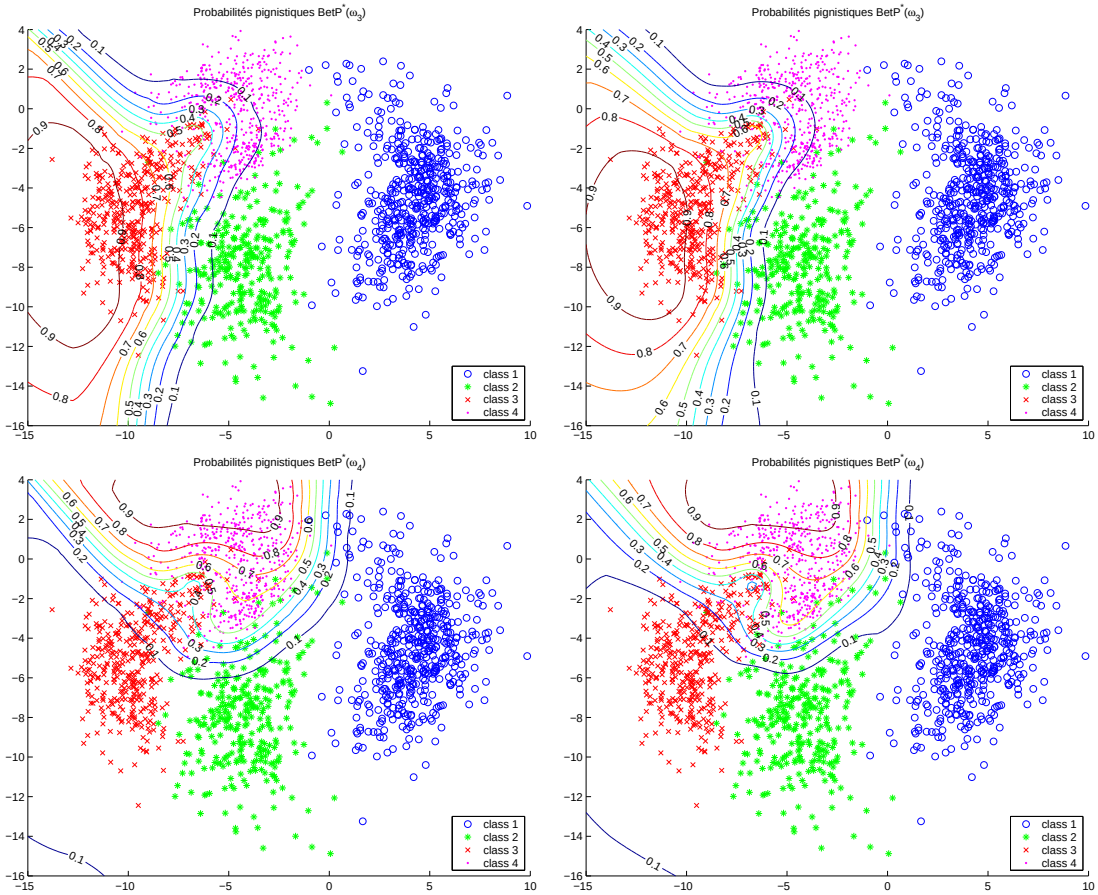


FIG. 4.6 – Courbes de niveau des probabilités pignistiques $BetP^*(\omega_3)$ (haut-gauche) et $BetP^*(\omega_4)$ (bas-gauche), obtenues dans le cas d'une décomposition 1-1, et dans le cas d'une décomposition CCE avec matrice de codes creuse (droite).

et donc d'informations dirigeant la masse de croyance vers les éléments focaux singletons lors de la procédure d'optimisation, est plus important dans le premier cas.

Les probabilités pignistiques correspondantes, $BetP^*(\omega_3)$ et $BetP^*(\omega_4)$, sont représentées sur la figure 4.6. Les frontières de décision obtenues dans le cas d'une décomposition CCE semblent mieux épouser les contours des régions associées aux classes. Cela est vraisemblablement dû au fait que la masse de croyance est dirigée davantage vers les éléments focaux singletons lors de la procédure d'optimisation.

4.2.4 Cas de classifieurs binaires probabilistes

On peut choisir d'interpréter les sorties d'un classifieur binaire $\mathcal{E}_i$ probabiliste comme des estimations des probabilités pignistiques normalisées $BetP_i^*$ calculées à partir des fonctions de masse $m_i^{\Theta_i}$:

$$r_i^{+*} = BetP_i^*(\theta_i^+), \tag{4.29}$$
$$r_i^{-*} = BetP_i^*(\theta_i^-), \tag{4.30}$$

où $BetP_i^*$ est définie par :

$$BetP_i^*(\theta) = \frac{BetP_i(\theta)}{1 - m^{\Theta_i}(\emptyset)}, \quad \forall \theta \in \Theta_i, \forall \Theta_i;$$

et où $BetP_i$ est définie par :

$$BetP_i(\theta_i^+) = \overline{\theta}_i \left(m^\Omega[\Omega_i]\right)(\{\theta_i^+\}) + \frac{\overline{\theta}_i \left(m^\Omega[\Omega_i]\right)(\Theta_i)}{2}, \tag{4.31}$$
$$BetP_i(\theta_i^-) = \overline{\theta}_i \left(m^\Omega[\Omega_i]\right)(\{\theta_i^-\}) + \frac{\overline{\theta}_i \left(m^\Omega[\Omega_i]\right)(\Theta_i)}{2}. \tag{4.32}$$

De même que dans le cas d'une décomposition 1-1, on peut alors rechercher la fonction de masse la plus consistante possible avec les sorties dénormalisées r_i^+ et r_i^- des classifieurs binaires, en résolvant :

$$\widehat{m}^\Omega = \arg\min_{m^\Omega} \sum_{i=1}^{N} \left(BetP_i(\theta_i^+) - r_i^+\right)^2 + \left(BetP_i(\theta_i^-) - r_i^-\right)^2 + \left(m^\Omega[\Omega_i](\emptyset) - 1 + pl_{\Omega_i}\right)^2, \tag{4.33}$$

sous les contraintes (4.27) et (4.28).

Exemple 4.7 Reprenons l'exemple 4.5. Supposons que des classifieurs $\mathcal{E}_i$ ont fourni les estimations suivantes des probabilités pignistiques :

$$r_1^{+*} = 0.000, \quad r_2^{+*} = 0.044, \quad r_3^{+*} = 0.750, \quad r_4^{+*} = 0.189,$$
$$r_1^{-*} = 1.000; \quad r_2^{-*} = 0.956; \quad r_3^{-*} = 0.250; \quad r_4^{-*} = 0.811.$$

Supposons en outre que des classifieurs à une classe ont fourni les plausibilités suivantes : $pl_1 = 0.437$, $pl_2 = 1.000$, $pl_3 = 0.990$. La combinaison de ces plausibilités au moyen de la t-conorme probabiliste (définie par (4.20)) donne :

$$pl_{\Omega_1} = 1.000, \quad pl_{\Omega_2} = 0.994,$$
$$pl_{\Omega_3} = 1.000, \quad pl_{\Omega_4} = 1.000.$$

On en déduit les probabilités pignistiques dénormalisées : $r_i^+ = r_i^{+*}$ et $r_i^- = r_i^{-*}$, pour $i \neq 2$; et $r_2^+ = 0.043$, $r_2^- = 0.951$.

La combinaison de ces informations, par minimisation de (4.33), donne la fonction de masse $\widehat{m}^{\Omega}$ suivante :

$$\begin{array}{rclcrcl} \widehat{m}^{\Omega}(\{\omega_2\}) & = & 0.197, & & \widehat{m}^{\Omega}(\Omega_{12}) & = & 0.015, \\ \widehat{m}^{\Omega}(\Omega_{23}) & = & 0.787. & & & & \end{array}$$

Le calcul des probabilités pignistiques $BetP_i$, définies sur les différents Θ_i, donne :

$$\begin{array}{rclcrcl} BetP_1(\theta_1^+) & = & 0.008, & & BetP_2(\theta_2^+) & = & 0.015, \\ BetP_3(\theta_3^+) & = & 0.606, & & BetP_4(\theta_4^+) & = & 0.401; \end{array}$$

ces estimations sont les meilleures approximations des r_k^+, au sens du critère (4.33). □

La figure 4.7 montre les masses $\widehat{m}^{\Omega}(\{\omega_3\})$, $\widehat{m}^{\Omega}(\{\omega_3,\omega_4\})$, et $\widehat{m}^{\Omega}(\{\omega_2,\omega_3,\omega_4\})$, obtenues en combinant les probabilités pignistiques conditionnelles obtenues à partir de réseaux de neurones évidentiels, par les méthodes MCTProb1-1 et MCT-ProbCCE. Les probabilités pignistiques correspondantes, $BetP^*(\omega_3)$ et $BetP^*(\omega_4)$, sont représentées sur la figure 4.8. De même que pour la méthode MCTCorr, on constate que les masses allouées aux éléments $\{\omega_3,\omega_4\}$ et $\{\omega_2,\omega_3,\omega_4\}$ sont moins importantes dans le cas CCE que dans le cas 1-1. On peut également constater que les courbes de niveau de la masse affectée à $\{\omega_3\}$ épousent plus les formes de la classe ω_3, à la frontière entre ω_3 et ω_4, que celles obtenues dans le cas d'une d"écomposition 1-1. Cette caractéristique peut également être observée sur les courbes de niveau des probabilités pignistiques, notamment $BetP^*(\omega_4)$, à la frontière entre ω_3 et ω_4.

4.3 Réduction de la complexité

De même que dans le cas d'une décomposition 1-1 (voir paragraphe 3.4), nous proposons de réduire la complexité en limitant le nombre d'éléments focaux de m^{Ω}.

Nous détaillons ci-dessous le calcul des plausibilités $pl^{\Omega}(\{\omega_k\})$, lorsqu'elles ne sont pas déterminées au moyen de classifieurs à une classe. Dans le cas d'une décomposition 1-T, on a :

$$m_k^{\Omega}(\rho_k(A)) = m_k^{\Theta_k}(A), \quad \forall A \subseteq \Theta_k, \forall \Theta_k.$$

En utilisant l'équation (1.6), on en déduit :

$$\begin{array}{rcl} pl_k^{\Omega}(\{\omega_k\}) & = & m_k^{\Omega}(\{\omega_k\}) + m_k^{\Omega}(\Omega), \\ & = & m_k^{\Theta_k}(\{\theta_k^+\}) + m_k^{\Theta_k}(\Theta_k); \\ pl_l^{\Omega}(\{\omega_k\}) & = & m_l^{\Omega}(\overline{\{\omega_l\}}) + m_l^{\Omega}(\Omega), \quad \forall l \neq k, \\ & = & m_l^{\Theta_l}(\{\theta_l^-\}) + m_l^{\Theta_l}(\Theta_l), \quad \forall l \neq k. \end{array}$$

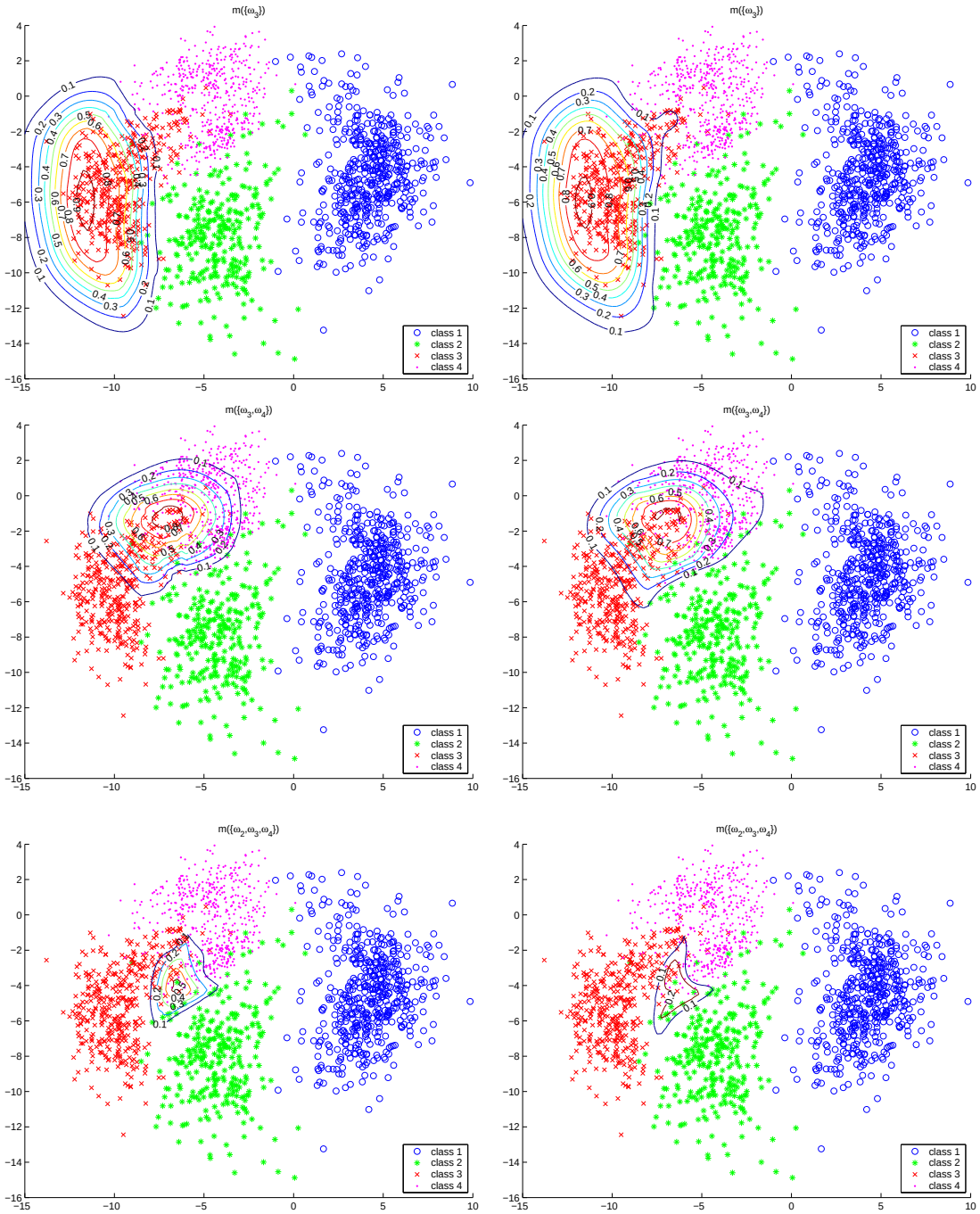


FIG. 4.7 – Courbes de niveau des masses $\widehat{m}^{\Omega}(\{\omega_3\})$ (haut-gauche), $\widehat{m}^{\Omega}(\{\omega_3,\omega_4\})$ (centre-gauche), et $\widehat{m}^{\Omega}(\{\omega_2,\omega_3,\omega_4\})$ (bas-gauche), obtenues par la méthode MCTProb1-1, et par la méthode MCTProbCCE avec matrice de codes creuse (droite).

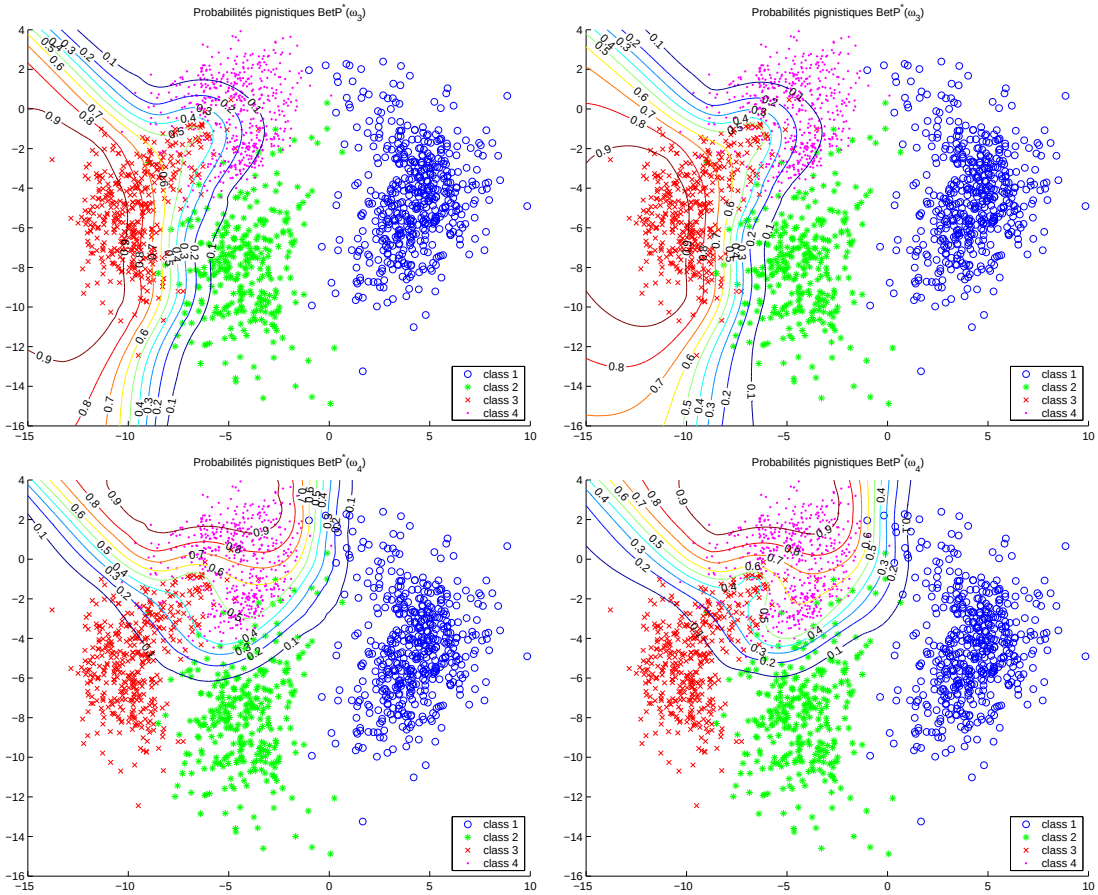


FIG. 4.8 – Courbes de niveau des probabilités pignistiques $BetP^*(\omega_3)$ (haut-gauche) et $BetP^*(\omega_4)$ (bas-gauche), obtenues dans le cas d'une décomposition 1-1, et dans le cas d'une décomposition CCE avec matrice de codes creuse (droite).

Dans le cas d'une décomposition CCE, les fonctions de masse $m_i^{\Theta_i}$ sont exprimées sur Ω, par extension vide et le cas échéant par déconditionnement :

$$\begin{aligned} m_i^{\Omega}(\overline{\Omega_i}) &= m_i^{\Theta_i}(\emptyset), \\ m_i^{\Omega}(A_i^+ \cup \overline{\Omega_i}) &= m_i^{\Theta_i}(\{\theta_i^+\}), \\ m_i^{\Omega}(A_i^- \cup \overline{\Omega_i}) &= m_i^{\Theta_i}(\{\theta_i^-\}), \\ m_i^{\Omega}(\Omega) &= m_i^{\Theta_i}(\Theta_i). \end{aligned}$$

La plausibilité pl_i^{Ω} associée à une fonction de masse m_i^{Ω} est obtenue par :

$$\begin{aligned} pl_i^{\Omega}(\{\omega_k\}) &= m_i^{\Omega}(A_i^+ \cup \overline{\Omega_i}) + m_i^{\Omega}(\Omega), \quad \forall \omega_k \in A_i^+; \\ pl_i^{\Omega}(\{\omega_k\}) &= m_i^{\Omega}(A_i^- \cup \overline{\Omega_i}) + m_i^{\Omega}(\Omega), \quad \forall \omega_k \in A_i^-; \\ pl_i^{\Omega}(\{\omega_k\}) &= 1, \quad \forall \omega_k \notin \Omega_i. \end{aligned}$$

Les différentes fonctions de plausibilité obtenues sont ensuite combinées par l'opérateur moyenne :

$$pl^{\Omega}(\{\omega_k\}) = \frac{1}{N} \sum_{i=1}^{N} pl_i^{\Omega}(\{\omega_k\}),$$

où N représente le nombre de dichotomies associées au schéma de décomposition : en particulier, $N = K$ dans le cas d'une décomposition 1-T.

4.4 Synthèse

Dans ce chapitre, nous avons formalisé la combinaison de classifieurs binaires dans le cadre du MCT, lorsque des décompositions un-contre-tous et par codes correcteurs d'erreurs sont employées. Dans le cas d'une décomposition un-contre-tous, les sorties de chaque classifieur sont interprétées comme des réductions extérieures sur des grossissements de Ω d'une fonction de masse m^{Ω} modélisant la connaissance de la classe de $\mathbf{x}$. Le schéma de décomposition par codes correcteurs d'erreurs peut être vu comme une généralisation des schémas un-contre-un et un-contre-tous. Les sorties des classifieurs sont alors interprétées comme des réductions extérieures de conditionnements de m^{Ω}.

Comme dans le cas d'une décomposition un-contre-un, les classifieurs sont combinés en déterminant la fonction de masse $\widehat{m}^{\Omega}$ la plus consistante avec leurs sorties, par résolution d'un problème d'optimisation quadratique. Lorsque certains classifieurs ont été entraînés à ne reconnaître qu'un nombre réduit de classes, l'estimation de la plausibilité d'appartenance à ces classes permet d'évaluer la pertinence des informations qu'ils fournissent. Comme précédemment, les sorties des classifieurs probabilistes peuvent être vues comme des estimations de probabilités pignistiques et combinées par une méthode spécifique.

L'identification des classes les plus plausibles permet de réduire la complexité, de manière à traiter des problèmes comptant un nombre significatif de classes.

Chapitre 5

Analyses

Dans ce chapitre, nous analysons les méthodes de combinaison présentées aux chapitres 3 et 4. Dans un premier temps, nous décrivons la mise en œuvre des tests réalisés, en détaillant notamment le mode d'apprentissage des classifieurs. Nous nous livrons ensuite à une étude des résultats obtenus lors du traitement de plusieurs jeux de données réelles. L'objectif est de comparer les méthodes de combinaison proposées dans ce mémoire à plusieurs méthodes déjà existantes, et d'en dégager des caractéristiques en termes de précision et de robustesse.

5.1 Protocole expérimental

5.1.1 Méthodes de combinaison comparées

Méthodes de combinaison évidentielles

Les méthodes de combinaison proposées dans ce mémoire peuvent être utilisées pour résoudre un problème multiclasses décomposé de diverses manières.

La méthode MCT, qui permet de combiner les sorties des classifieurs binaires interprétées comme des fonctions de masse, s'applique aux cas de décompositions un-contre-un (méthode MCT1-1, présentée au paragraphe 3.1), un-contre-tous (méthode MCT1-T, présentée au paragraphe 4.1.2), et par codes correcteurs d'erreurs (méthode MCTCCE, présentée au paragraphe 4.2.2).

La méthode MCTCorr permet de combiner des classifieurs binaires, en prenant en compte la plausibilité d'appartenance d'un individu $\mathbf{x}$ évalué à l'ensemble d'apprentissage des classifieurs. La méthode MCTProb permet de combiner des classifieurs probabilistes, en interprétant leurs sorties comme des probabilités pignistiques conditionnelles, et en les dénormalisant au moyen des plausibilités d'apartenance. Ces deux méthodes s'appliquent aux cas de décompositions un-contre-un (méthode MCTCorr1-1, paragraphe 3.2) et par codes correcteurs d'erreurs (méthode MCTCorrCCE, paragraphe 4.2.3).

Nous avons également combiné les masses issues des sorties des classifieurs

par la règle de combinaison conjonctive. L'emploi de cette méthode nécessite d'exprimer les masses sur un cadre de discernement commun. Lorsque les masses sont définies sur des grossissements de Ω (dans les cas d'une décomposition 1-T ou CCE), leur extension vide sur Ω est donc calculée. Lorsque les masses définies sur Ω sont conditionnelles (dans le cas d'une décomposition 1-1, ou d'une décomposition CCE avec matrice de codes creuse après calcul de l'extension vide), elles sont déconditionnées sur le domaine Ω. La règle de combinaison conjonctive a ainsi été appliquée aux masses fournies par les classifieurs (méthode Conj), et aux masses dénormalisées au moyen des plausibilités d'appartenance (méthode ConjCorr).

Méthodes de combinaison probabilistes

Plusieurs méthodes de combinaison de classifieurs probabilistes dans le cas CCE ont également été évaluées : la méthode itérative proposée par Huang, Weng et Lin (méthode PCplBT), présentée au paragraphe 2.4.2, et qui correspond à la méthode proposée par Hastie et Tibshirani dans le cas 1-1 ; et la méthode non-itérative proposée par Passerini, Pontil et Frasconi (méthode PEstP), présentée au paragraphe 2.4.2.

De même, plusieurs méthodes de combinaison de classifieurs probabilistes spécifiques à la décomposition 1-1 ont été testées : les méthodes non-itératives proposées par Wu, Lin et Weng (méthodes PEst1 et PEst2), présentées au paragraphe 2.2.2, et la méthode non-itérative de combinaison de classifieurs probabilistes incluant une étape de correction, proposée par Moreira et Mayoraz (méthode PEstCorr), présentée au paragraphe 2.2.2.

Le tableau 5.1 récapitule le type d'entrées et de sorties des différentes méthodes de combinaison évaluées.

5.1.2 Algorithmes employés pour la construction des classifieurs binaires

Régression logistique

La régression logistique permet d'estimer des probabilités d'appartenance aux classes positive et négative. La frontière de décision déterminée est linéaire. Pour cette raison, ce classifieur binaire n'a pas été employé dans les cas 1-T et CCE : il semble en effet peu raisonnable d'approcher la frontière de décision par un hyperplan.

Arbres de décision binaires

Les arbres de décision binaires permettent d'estimer des probabilités d'appartenance aux classes positives et négatives. La frontière de décision déterminée est linéaire par morceaux. Les arbres de décision binaires ont donc été utilisés dans le

TAB. 5.1 – Récapitulatif des entrées et des sorties des méthodes de combinaison évaluées.

	entrées	sorties
MCT, Conj	fonctions de masse normales fournies par les classifieurs binaires	fonction de masse normale, probabilités pignistiques associées
MCTCorr, ConjCorr	fonctions de masse sous-normales ou dénormalisées	fonction de masse sous-normale, probabilités pignistiques associées
MCTProb	probabilités pignistiques conditionnelles, plausibilités d'appartenance aux cadres restreints	fonction de masse sous-normale, probabilités pignistiques associées
PCplBT, PEstP ; PEst1, PEst2 (schéma 1-1)	probabilités conditionnelles fournies par les classifieurs binaires	probabilités a posteriori
PEstCorr (schéma 1-1)	probabilités conditionnelles corrigées par les probabilités correctrices	probabilités a posteriori

cas de décompositions 1-1, 1-T et CCE. Les procédures MatLab ont été utilisées pour construire les arbres : l'apprentissage a été fait selon l'algorithme CART [7], et l'arbre de coût minimal a été déterminé par validation croisée à dix coupes.

Réseaux de neurones évidentiels

Les réseaux de neurones évidentiels [12] permettent de déterminer des fonctions de croyance quantifiant l'appartenance d'un individu $\mathbf{x}$ aux classes positive et négative. La frontière de décision est non-linéaire ; ce classifieur a donc été utilisé dans le cas de décompositions 1-1, 1-T et CCE. Le code Matlab pour l'implémentation des réseaux de neurones évidentiels est disponible à l'URL `http ://www.hds.utc.fr/~tdenoeux/software.htm`.

Un réseau de neurones évidentiel caractérise les classes qu'il sépare par des prototypes. Soient A_i^+ et A_i^- les groupes de classes positives et négatives constituant l'ensemble d'apprentissage du classifieur $\mathcal{E}_i$. D'une manière générale, $3|A_i^+|$ et $3|A_i^-|$ prototypes ont été respectivement utilisés pour caractériser A_i^+ et A_i^-. Ainsi, dans le cas 1-1, 3 prototypes ont été utilisés pour décrire ω_i comme ω_j, pour tout $j > i$; dans le cas 1-T, 3 prototypes ont été utilisés pour décrire θ_k^+ et $3(K-1)$ pour θ_k^-, pour tout k, sauf dans le cas du jeu de données Vowel où $3(K-1)$ prototypes ont été utilisés pour décrire θ_k^+ comme θ_k^-. Nous avons utilisé un étiquetage classique : chaque individu d'apprentissage a été associé à sa classe réelle d'appartenance.

La transformation pignistique (équation (1.22)) a permis de déterminer des probabilités à partir des sorties de chaque réseau de neurones évidentiel, pour évaluer les méthodes probabilistes (PCplBT et PEstP dans le cas général, ainsi que PCpl, PEst1, PEst2, et PEstCorr dans le cas 1-1).

Le tableau 5.2 résume les traitements appliqués aux sorties des différents classifieurs binaires pour les évaluer.

5.1.3 Algorithmes employés pour estimer la pertinence des classifieurs

Dans la méthode MCT, les plausibilités $pl^{\Omega}(\{\omega_k\})$ ont été calculées au moyen de séparateurs à vaste marge à une classe (1-SVM) [50]. Un 1-SVM estime le support d'une classe ω_k, et le décrit au moyen d'un sous-ensemble choisi d'individus de ω_k, appelés vecteurs de support. La distance signée $f_k(\mathbf{x})$ d'un individu $\mathbf{x}$ au support de ω_k peut alors être calculée. Un paramètre ν choisi arbitrairement permet de fixer une borne inférieure sur la fraction de vecteurs de support et une borne supérieure sur la fraction de points aberrants ; ici, ν a été fixé à 0.2. La plausibilité $pl^{\Omega}(\{\omega_k\})$ a été calculée par :

$$pl^{\Omega}(\{\omega_k\}) = \frac{f_k(\mathbf{x}) + \rho}{\rho}, \tag{5.1}$$

TAB. 5.2 – Traitement appliqué aux sorties des classifieurs, pour les différentes méthodes évaluées.

	traitement appliqué aux sorties pour l'évaluation des méthodes crédales	traitement appliqué aux sorties pour l'évaluation des méthodes probabilistes
régression logistique, arbres de décision (sorties : probabilités)	transformation en fonctions de masse Bayésiennes ; Conj-Corr, MCTCorr : dénormalisation par les plausibilités d'appartenance	combinaison des probabilités ; MCTProb : dénormalisation par les plausibilités d'appartenance, PEstCorr : correction par les probabilités correctrices
réseaux de neurones évidentiels (sorties : fonctions de masse normales)	combinaison des fonctions de masse ; MCTCorr, Conj-Corr : dénormalisation par les plausibilités d'appartenance	transformation en probabilités pignistiques conditionnelles ; MCTProb : dénormalisation par les plausibilités d'appartenance, PEstCorr : correction par les probabilités correctrices

où ρ est un paramètre obtenu lors de l'apprentissage du 1-SVM (détails dans [50]). Ces plausibilités ont ensuite été combinées au moyen de la conorme triangulaire (t-conorme) probabiliste pour obtenir les plausibilités $pl_{ij} = pl^{\Omega}(\Omega_{ij})$ dans le cas 1-1, et $pl_i = pl^{\Omega}(\Omega_i)$ dans le cas CCE. Rappelons que cet opérateur de combinaison est défini par :

$$x \odot y = x + y - xy.$$

Dans la méthode PEstCorr, les probabilités correctrices q_{ij} ont été calculées à partir d'estimations g_k des densités de probabilité des classes ω_k, par une méthode de noyaux :

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

n_i et n_j étant le nombre d'individus dans les classes ω_i et ω_j, respectivement.

Dans les deux cas (1-SVM et méthode des noyaux), une estimation $\widehat{\sigma}_k$ de la largeur de bande des noyaux a été obtenue pour chaque classe ω_k par la méthode proposée dans [35]. Pour chaque classe, la largeur de bande a ensuite été déterminée par :

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

Cette méthode heuristique semble donner les résultats les plus robustes pour l'ensemble des jeux de données sur lesquelles les méthodes de combinaison ont été évaluées.

5.1.4 Détermination des matrices de codes

Les propriétés de la matrice de codes associée à un schéma de décomposition traduisent la capacité à bien classer un individu évalué. Une bonne séparation (au sens de la distance de Hamming) des lignes signifie que les classes sont bien séparées les unes des autres par l'ensemble des classifieurs, et la séparation des colonnes traduit l'indépendance des classifieurs, et donc la robustesse de l'ensemble.

Dans le cas de décompositions 1-1 et 1-T, la matrice de codes est naturellement imposée. Dans le cas d'une décomposition CCE quelconque, nous avons utilisé deux types de matrices de codes décrits dans [2].

- Les *matrices denses* sont constituées d'éléments $e_{ki} \in \{-1; 1\}$. Chaque élément est déterminé aléatoirement selon la loi uniforme : $\mathbb{P}(\{-1\}) = 0.5$, $\mathbb{P}(\{1\}) = 0.5$.
- Les *matrices creuses* sont constituées d'éléments $e_{ki} \in \{-1; 0; 1\}$. Chaque élément est déterminé aléatoirement selon $\mathbb{P}(\{-1\}) = 0.25$, $\mathbb{P}(\{0\}) = 0.5$, $\mathbb{P}(\{1\}) = 0.25$.

Dans les deux cas, T matrices $\mathcal{M}_t$ $(t = 1, \ldots, T)$ sont ainsi déterminées aléatoirement ; puis les colonnes dont tous les éléments sont positifs ou négatifs, ainsi que les colonnes identiques ou complémentaires, sont supprimées. La matrice de codes optimale $\mathcal{M}^*$ est alors sélectionnée selon la procédure suivante. Soient $d_L(k)$ et $d_C(i)$ les distances de Hamming minimales entre la k^{e} ligne (respectivement, la i^{e} colonne) d'une matrice et l'ensemble des autres :

$$\begin{aligned} d_L(k) &= \min_{\substack{l=1,\ldots,K \\ l \neq k}} d_H(\mathbf{e}_{k\cdot}, \mathbf{e}_{l\cdot}), \\ d_C(i) &= \min_{\substack{j=1,\ldots,N \\ j \neq i}} d_H(\mathbf{e}_{\cdot i}, \mathbf{e}_{\cdot j}), \end{aligned}$$

où $d_H(\mathbf{e}_{k\cdot}, \mathbf{e}_{l\cdot})$ (respectivement, $d_H(\mathbf{e}_{\cdot i}, \mathbf{e}_{\cdot j})$) représente la distance de Hamming entre les lignes de la matrice associées aux classes ω_k et ω_l (respectivement, les colonnes de la matrice associées aux classifieurs $\mathcal{E}_i$ et $\mathcal{E}_j$). On sélectionne tout d'abord l'ensemble des matrices maximisant la somme des distances entre lignes :

$$\{\mathcal{M}\}_1 = \left\{ \mathcal{M} = \arg \max_{t=1,\ldots,T} \sum_{k=1}^{K} d_L(k) \right\}.$$

Si cet ensemble correspond à une seule matrice : $|\{\mathcal{M}\}_1| = 1$, on a $\mathcal{M}^* = \{\mathcal{M}\}_1$. Sinon, on sélectionne dans $\{\mathcal{M}\}_1$ la matrice qui maximise la somme des distances

entre colonnes :

$$\{\mathcal{M}\}_2 = \left\{ \mathcal{M} = \arg\max_{\{\mathcal{M}\}_1} \sum_{i=1}^{N} d_C(i) \right\}.$$

De nouveau, si $\{\mathcal{M}\}_2$ correspond à une seule matrice, on a $\mathcal{M}^* = \{\mathcal{M}\}_2$; sinon, $\mathcal{M}^*$ est sélectionnée aléatoirement dans $\{\mathcal{M}\}_2$.

Nous avons généré des ensembles de $T = 5000$ matrices, de taille initiale (avant suppression des colonnes non pertinentes ou redondantes) $K \times 10\log_2(K)$ pour les matrices denses, et $K \times 15\log_2(K)$ pour les matrices creuses.

5.1.5 Réduction de la complexité

La complexité a été réduite en sélectionnant les cinq classes auxquelles la plausibilité d'appartenance de $\mathbf{x}$ est la plus élevée, et en aggrégeant les autres en une classe unique.

5.1.6 Caractéristiques des jeux de données

Le tableau 5.3 présente les caractéristiques des jeux de données utilisés lors des tests. Tous proviennent de la base de données du département d'Apprentissage Statistique de l'UCI [1], excepté le jeu de données Synth, généré par mélange de Gaussiennes.

TAB. 5.3 – Caractéristiques des jeux de données

données	dimension	nb. classes	nb. individus / appr.	nb. individus / test
Ecoli	7	8	201	135
Glass	9	6	139	75
Letter	16	26	7800	10400
Satimage	36	6	2573	3862
Segment	19	7	1400	910
Synth3Cl	2	3		
Synth	2	4	1700	340
Synth5Cl	2	5		
Vowel	10	11	528	462
Waveform	21	3	1500	3500

[1] Disponible à l'URL `http ://www.ics.uci.edu/~mlearn`.

5.2 Présentation des résultats

Nous analysons à présent les résultats obtenus en appliquant les méthodes de combinaison présentées dans ce mémoire à plusieurs jeux de données réelles, pour différents types de classifieurs binaires. Les méthodes comparées fournissent toutes une distribution de probabilité sur l'ensemble des classes, suite à l'évaluation d'un individu $\mathbf{x}$ – dans le cas des méthodes de combinaison crédales, cette probabilité est obtenue par transformation pignistique des masses issues de la combinaison. À l'issue de la combinaison, $\mathbf{x}$ a été affecté à la classe de probabilité maximale. En cas d'équiprobabilité, une classe a été déterminée au hasard parmi les classes en compétition. Les fonctions de masse obtenues par combinaison telles que $\widehat{m}^{\Omega}(\emptyset) = 1$ ont été transformées en probabilités pignistiques uniformes $BetP^*(\omega_k) = 1/K$, pour tout $k \in \{1, \ldots, K\}$. Les taux de bonne classification ont ainsi été mesurés, pour chaque schéma de décomposition, pour les différents types de classifieurs binaires employés.

La significativité des différences entre les taux a été évaluée au moyen du *test de Mc Nemar* [15] au niveau 5%. Cette variante du test du Chi2 permet de comparer les proportions associées à deux échantillons appariés, sur la base des paires de valeurs présentant une différence. Pour chaque évaluation, le meilleur résultat obtenu a été identifié, et apparaît souligné dans les tableaux ; le taux de bonne classification donné par chaque autre méthode lui a été comparé, et est représenté en gras si la différence constatée n'est pas jugée significative.

5.2.1 Décomposition un-contre-un

Les tableaux 5.4, 5.5 et 5.6 présentent les taux de bonne classification obtenus dans le cas d'une décomposition 1-1, après combinaison des informations fournies par la régression logistique, les arbres de décision, et les réseaux de neurones évidentiels, respectivement. La méthode MCTProb1-1 a été testée uniquement avec les classifieurs binaires probabilistes (régression logistique et arbres de décision).

Il apparaît clairement que les méthodes MCTCorr et MCTProb donnent globalement les meilleurs résultats. L'une des deux au moins obtient les meilleurs résultats pour les cinq jeu de données lorsque la régression logistique est utilisée, et pour quatre jeux de données avec les arbres de décision et les réseaux de neurones évidentiels. Lorsque les résultats ne sont pas les meilleurs, ils ne sont pas significativement moins bons que les meilleurs résultats obtenus. Remarquons en outre que les méthodes faisant intervenir une étape de correction des classifieurs (MCTCorr et MCTProb, ConjCorr, et PEstCorr) donnent globalement de meilleurs résultats que les autres. Les méthodes MCTCorr et MCTProb donnent toutefois globalement les meilleurs résultats, et la méthode PEstCorr semble moins performante que toutes les autres méthodes sur le jeu de données Segment. Enfin, la différence entre les résultats fournis par les méthodes MCTCorr et MCTProb n'est pas jugée significative, hormis lors du traitement du jeu de données Waveform au

TAB. 5.4 – Taux de bonne classification (%), décomposition 1-1, régression logistique

Méthode	Satimage	Segment	Synth	Vowel	Waveform
MCT1-1	86.9	**95.6**	**94.4**	52.8	**85.1**
MCTCorr1-1	**93.1**	**96.0**	**95.9**	**66.5**	**85.2**
MCTProb1-1	**93.1**	**96.0**	**95.6**	**65.8**	**85.3**
Conj	86.9	**95.8**	**94.4**	50.6	**85.1**
ConjCorr	91.6	**95.9**	**95.3**	55.6	**85.2**
PCplBT	87.1	**96.0**	**94.4**	51.3	**85.1**
PEstP	86.9	**95.8**	**94.4**	49.8	**85.1**
PEst1	86.9	**95.6**	**94.4**	50.9	**85.1**
PEst2	86.9	**95.6**	**94.4**	52.6	**85.1**
PEstCorr	90.8	90.3	**95.6**	60.6	**85.2**

moyen d'arbres de décision ; même dans ce cas, la différence entre les résultats obtenus avec ces deux méthodes est bien moins importante que celle constatée pour les autres.

5.2.2 Décomposition un-contre-tous

Les tableaux 5.7 et 5.8 permettent de comparer les méthodes de combinaison proposées dans le cas d'une décomposition 1-T. Les arbres de décision et la régression logistique ont été utilisés comme classifieurs binaires.

On peut constater que les résultats obtenus pour les différentes méthodes de combinaison, dans le cas d'une décomposition 1-T, sont équivalents. Ces résultats sont strictement identiques pour trois des cinq jeux de données lorsque les classifieurs binaires sont des arbres de décision, pour deux jeux de données lorsque ce sont des réseaux de neurones évidentiels. Dans les autres cas, les différences ne sont pas jugées significatives.

5.2.3 Décomposition par codes correcteurs d'erreurs, avec matrice de codes dense

Les résultats obtenus dans le cas d'une décomposition CCE avec matrice de codes dense ont été consignés dans les tableaux 5.9 et 5.10, qui présentent respectivement les résultats de la combinaison d'arbres de décision et de réseaux de

TAB. 5.5 – Taux de bonne classification (%), décomposition 1-1, arbres de décision

Méthode	Satimage	Segment	Synth	Vowel	Waveform
MCT1-1	86.7	93.6	**95.9**	46.3	74.9
MCTCorr1-1	**93.0**	**95.5**	**95.9**	**66.9**	80.0
MCTProb1-1	**93.0**	**95.3**	**95.9**	**65.6**	**80.7**
Conj	86.9	93.0	**95.9**	45.2	74.9
ConjCorr	91.0	93.6	**95.9**	51.1	76.7
PCplBT	86.9	93.0	**95.9**	48.3	74.9
PEstP	86.9	93.1	**95.9**	41.3	74.9
PEst1	86.8	93.2	**95.9**	46.3	74.9
PEst2	86.7	93.6	**95.9**	46.5	74.9
PEstCorr	90.4	89.0	**96.5**	60.6	76.3

neurones évidentiels.

Lorsque les arbres de décision sont utilisés comme classifieurs binaires, les résultats sont globalement similaires pour trois jeux de données. La méthode PEstP donne des résultats significativement moins bons sur le jeu de données Segment, et les méthodes Conj et PEstP sur le jeu de données Vowel. Lorsque les classifieurs binaires sont des réseaux de neurones évidentiels, ausune différence significative ne peut être constatée lors du traitement de trois jeux de données. La méthode PEstP donne des résultats significativement moins bons que les autres méthodes sur les données Satimage, et les méthodes Conj et PEstP sur les données Segment. D'une manière générale, les résultats donnés par les méthodes MCT et PCplBT sont proches. Une différence, qui n'est pas jugée significative, peut être constatée lors du traitement des données Vowel au moyen de réseaux de neurones évidentiels.

5.2.4 Décomposition par codes correcteurs d'erreurs, avec matrice de codes creuse

Les tableaux 5.11 et 5.12 présentent respectivement les résultats de la combinaison d'arbres de décision et de réseaux de neurones évidentiels, dans le cas d'une décomposition CCE avec matrice de codes creuse. Tout comme dans le cas d'une décomposition 1-1, la méthode MCTProbCCE a été testée uniquement avec les arbres de décision comme classifieurs binaires.

On constate ici encore la supériorité des méthodes MCTCorr et MCTProb :

TAB. 5.6 – Taux de bonne classification (%), décomposition 1-1, réseaux de neurones évidentiels

Méthode	Satimage	Segment	Synth	Vowel	Waveform
MCT1-1	87.5	85.8	**95.6**	**64.9**	**86.3**
MCTCORR1-1	**93.1**	**89.8**	**96.2**	**66.7**	**85.9**
CONJ	86.9	84.0	**95.9**	**63.0**	**86.3**
CONJCORR	92.4	**88.9**	**95.6**	**65.8**	**86.1**
PCPLBT	86.5	84.9	**95.6**	**63.0**	**86.4**
PESTP	86.9	85.2	**95.6**	**63.6**	**86.3**
PEST1	86.9	85.2	**95.6**	**63.4**	**86.3**
PEST2	87.3	85.3	**95.9**	**65.6**	**86.2**
PESTCORR	90.0	82.3	**95.9**	61.3	85.7

l'une d'entre elles au moins donne les meilleurs résultats pour les cinq jeux de données lorsque les arbres de décision sont utilisés, ces résultats étant jugés significativement meilleurs que tous les autres dans trois cas. La méthode MCTCorr donne les meilleurs résultats pour trois jeux de données avec les réseaux de neurones évidentiels ; dans les deux autres cas, les différences constatées ne sont pas jugées significatives.

L'évaluation de la pertinence des classifieurs semble avoir ici moins d'impact que dans le cas 1-1 : les différences constatées entre les méthodes MCT et MCTCorr sont globalement moindres. L'étape de correction permet de déterminer les classifieurs pertinents, mais également de disposer de davantage d'informations pour l'élaboration de la solution. L'augmentation du nombre de classifieurs

TAB. 5.7 – Taux de bonne classification (%), décomposition 1-T, arbres de décision

Méthode	Satimage	Segment	Synth	Vowel	Waveform
MCT1-T	**84.8**	**95.1**	**95.3**	**35.5**	**74.8**
CONJ	**84.8**	**95.1**	**95.3**	**34.8**	**74.8**
PCPLBT	**84.8**	**95.1**	**95.3**	**34.4**	**74.8**
PESTP	**84.8**	**95.1**	**95.3**	**34.0**	**74.8**

TAB. 5.8 – Taux de bonne classification (%), décomposition 1-T, réseaux de neurones évidentiels

Méthode	Satimage	Segment	Synth	Vowel	Waveform
MCT1-T	**84.9**	**75.8**	**96.2**	**66.5**	**86.3**
CONJ	**85.1**	**75.7**	**95.9**	**66.5**	**86.3**
PCPLBT	**85.0**	**76.2**	**96.2**	**66.5**	**86.3**
PESTP	**85.0**	**76.2**	**96.2**	**66.5**	**86.3**

TAB. 5.9 – Taux de bonne classification (%), décomposition CCE avec matrice de codes dense, arbres de décision

Méthode	Satimage	Segment	Synth	Vowel	Waveform
MCTCCE	**89.0**	**95.8**	**95.3**	**49.4**	**76.3**
CONJ	**88.9**	**95.4**	**95.0**	31.4	**76.2**
PCPLBT	**89.0**	**95.8**	**95.3**	**49.4**	**76.2**
PESTP	**88.9**	94.2	**95.0**	23.2	**76.2**

binaires par rapport au cas 1-1 amoindrit vraisemblablement le gain dû à l'utilisation de classifieurs supplémentaires.

On peut constater que les méthodes MCT et PCplBT donnent des résultats proches ; néanmoins, la similitude de ces résultats est moins importante que dans le cas de décompositions 1-T ou CCE avec matrice de codes dense. Ces deux méthodes consistent à combiner les informations disponibles en recherchant une solution consistante avec ces informations. La différence fondamentale réside dans la prise en compte du caractère incomplet des informations combinées, lorsque les classifieurs n'ont pas été entraînés à partir de tout l'ensemble d'apprentissage. Les différences constatées entre les deux méthodes proviennent vraisemblablement de ce traitement.

5.2.5 Comparaison des différents schémas de combinaison

On constate que les résultats obtenus pour les méthodes de combinaison ne faisant pas intervenir de correction des classifieurs sont meilleurs lorsque les schémas de décomposition par codes correcteurs sont utilisés. Les informations disponibles sont alors plus nombreuses que dans le cas de décompositions 1-1 ou 1-T. Cela

TAB. 5.10 – Taux de bonne classification (%), décomposition CCE avec matrice de codes dense, réseaux de neurones évidentiels

Méthode	Satimage	Segment	Synth	Vowel	Waveform
MCTCCE	**86.1**	**84.4**	**95.6**	**61.9**	**86.0**
CONJ	**86.3**	83.5	**95.6**	**63.4**	**86.1**
PCPLBT	**86.1**	**84.6**	**95.6**	**63.4**	**86.0**
PESTP	85.9	83.1	**95.6**	**63.4**	**86.0**

TAB. 5.11 – Taux de bonne classification (%), décomposition CCE avec matrice de codes creuse, arbres de décision

Méthode	Satimage	Segment	Synth	Vowel	Waveform
MCTCCE	89.0	**95.4**	**96.2**	47.8	77.9
MCTCORRCCE	**92.3**	**95.5**	**96.5**	**63.2**	**81.1**
MCTPROBCCE	**92.3**	**95.9**	**96.5**	**62.8**	**81.1**
CONJ	89.2	**95.7**	**95.6**	38.1	78.3
CONJCORR	85.8	94.5	**95.6**	34.2	79.3
PCPLBT	88.9	**95.4**	**96.2**	47.0	77.7
PESTP	88.9	**95.9**	**95.6**	31.8	78.3

permet vraisemblablement de mieux compenser la connaissance incomplète de Ω qui caractérise un classifieur, en la complétant par les informations apportées par les autres.

Les méthodes faisant intervenir une correction semblent donner de meilleurs résultats avec la décomposition 1-1 qu'avec la décomposition CCE avec matrice de codes creuse. Cela est vraisemblablement dû au fait que les classifieurs entraînés dans le cas 1-1 sont moins dépendants les uns des autres que dans le cas CCE : en effet, les ensembles d'apprentissage de deux classifieurs différents peuvent avoir au plus une classe en commun. Dans le cas des méthodes sans correction, la perte de précision causée par la méconnaissance de certaines classes est généralement plus importante que le gain dû à l'indépendance des classifieurs.

Enfin, le schéma de décomposition 1-T donne généralement de médiocres résultats, excepté lors de la combinaison de réseaux de neurones évidentiels. Cela suggère que les performances obtenues lors de la combinaison de classifieurs per-

TAB. 5.12 – Taux de bonne classification (%), décomposition CCE avec matrice de codes creuse, réseaux de neurones évidentiels

Méthode	Satimage	Segment	Synth	Vowel	Waveform
MCTCCE	86.5	83.3	**95.6**	**66.9**	**86.1**
MCTCorrCCE	**92.6**	**85.8**	**95.6**	**66.9**	**86.1**
Conj	86.6	83.3	**95.6**	**66.2**	**86.2**
ConjCorr	91.0	**84.1**	**95.3**	**66.7**	**86.1**
PCplBT	86.5	**84.4**	**95.3**	**67.5**	**86.2**
PEstP	86.4	83.5	**95.6**	**66.0**	**86.2**

formants par différents schémas de décomposition sont globalement homogènes, comme l'ont constaté Rifkin et Klautau [48].

5.3 Analyse détaillée et interprétation des résultats

5.3.1 Impact de la correction des classifieurs binaires

Les méthodes MCTCorr1-1, MCTProb1-1, ConjCorr et PEstCorr, faisant intervenir une étape de correction des classifieurs, donnent des résultats globalement meilleurs que ceux fournis par les autres méthodes. On constate toutefois des différences dans la qualité de ces résultats, parfois significatives. Elles peuvent être expliquées par la manière dont la pertinence des classifieurs est quantifiée et intégrée aux informations fournies par les classifieurs binaires.

Analyse du fonctionnement des méthodes MCTCorr, MCTProb et PEstCorr

Dans la méthode PEstCorr, la pertinence d'un classifieur binaire $\mathcal{E}_{ij}$ est évaluée par un classifieur dit correcteur, entraîné à séparer la paire de classes $\Omega_{ij} = \{\omega_i, \omega_j\}$ des autres : elle est ainsi quantifiée par la probabilité $q_{ij} = \mathbb{P}(\Omega_{ij}|\mathbf{x})$ qu'un individu $\mathbf{x}$ évalué appartienne à Ω_{ij}. Soit $q_{ji} = \mathbb{P}(\overline{\Omega_{ij}}|\mathbf{x})$ la probabilité que $\mathbf{x}$ n'appartienne pas à Ω_{ij} ; la mesure de probabilité est additive, ce qui impose : $q_{ij} + q_{ji} = 1$. Par conséquent, la proximité d'une classe ω_k ($k \notin \{i, j\}$) à ω_i ou ω_j peut influer sur la valeur de q_{ij}. La figure 5.1 illustre ce phénomène de variation des q_{ij} sur un jeu de données synthétiques à trois classes, dont les densités sont connues. Dans un cas, les classes ω_2 et ω_3 sont nettement séparées ; dans l'autre,

la classe ω_3 a subi une translation et ω_2 et ω_3 se recouvrent partiellement. La probabilité correctrice q_{12} est calculée par la règle de Bayes. Pour un même in-

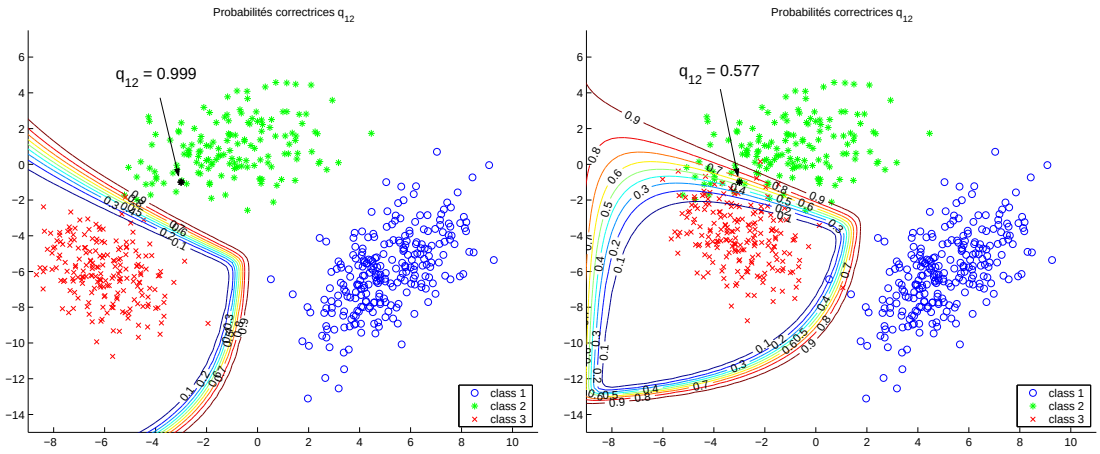


FIG. 5.1 – Courbes de niveau des probabilités correctrices q_{12}, calculées sur un jeu de trois classes (gauche) ; probabilités correctrices q_{12}, sur le même jeu de données, après translation de la classe ω_3 pour provoquer un recouvrement avec la classe ω_2 (droite).

dividu $\mathbf{x}$ de classe réelle ω_2, elle est bien plus élevée lorsque ω_2 et ω_3 sont bien séparées ($q_{12} = 0.999$) que lorsqu'elles se recouvrent partiellement ($q_{12} = 0.577$). Ainsi, lorsque les données sont mal séparées, l'étape de correction des classifieurs binaires peut avoir une influence sur le classement de $\mathbf{x}$, aboutissant parfois à choisir une autre classe que celle déterminée par les classifieurs binaires pertinents.

Dans les méthodes MCTCorr1-1 et MCTProb1-1, la pertinence d'un classifieur binaire $\mathcal{E}_{ij}$ est quantifiée par la plausibilité pl_{ij} que $\mathbf{x}$ appartienne à Ω_{ij}. La plausibilité est une mesure sur-additive : la seule contrainte imposée est $0 \leq pl_{ij}(\mathbf{x}) \leq 1$, pour tout $j > i$. Par conséquent, les méthodes MCTCorr1-1 et MCTProb1-1 sont insensibles à ce phénomène de recouvrement : dans le cas traité ci-dessus, on aura vraisemblablement $pl(\{\omega_2\}) \approx pl(\{\omega_3\}) \approx 1$; l'emploi d'une t-conorme donnera donc $pl_{ij} \approx 1$, pour tout $j > i$. Dans ce cas, l'individu $\mathbf{x}$ ne sera classé que sur la base des informations fournies par les classifieurs binaires entraînés spécifiquement pour cette tâche. Remarquons que dans le cas où aucune information ne permet de déterminer les plausibilités d'appartenance pl_{ij}, le Principe du Minimum d'Information amène à fixer $pl_{ij} = 1$ pour tout $j > i$: un individu $\mathbf{x}$ est alors classé sur la seule base des informations fournies par les classifieurs binaires, et la méthode est l'équivalent crédal de la méthode PCplBT. Au contraire, lorsque $\mathbf{x}$ n'appartient à aucune classe de Ω, les plausibilités d'appartenance sont toutes telles que $pl_{ij} \approx 0$; la fonction de masse déterminée par combinaison vérifie donc $m^{\Omega}(\emptyset) \approx 1$. Cette fonction de masse quantifie bien la croyance que $\mathbf{x}$ n'appartient pas à l'une des classes considérées.

Soulignons en outre que suivant le type de classifieurs correcteurs employés, la méthode PEstCorr peut nécessiter d'apprendre C_K^2 classifieurs correcteurs supplémentaires, chacun étant entraîné sur la base des P individus d'apprentissage. La détermination des plausibilités d'appartenance permet de ne prendre en compte qu'une seule fois l'information apportée par chaque individu d'apprentissage.

Enfin, soulignons que les méthodes MCTCorr et MCTProb comme la méthode PEstCorr de celle des q_{ij} sont d'une manière générale tributaires de la précision des pl_{ij}. Pour évaluer le nombre de mauvais classements dûs à une correction erronée des classifieurs, nous avons déterminé le nombre d'individus mal classés à l'issue de la combinaison, pour lesquels les classifieurs binaires pertinents donnent des résultats corrects et les classifieurs correcteurs des résultats erronés. Nous avons considéré la correction comme erronée lorsque la plausibilité $pl(\{\omega_k\})$ maximale ou la densité g_k maximale ne correspond pas à la classe réelle de $\mathbf{x}$. Les résultats de cette expérience sont consignés dans les tableaux 5.13, 5.14 et 5.15 ; le pourcentage correspondant au décompte effectué apparaît entre parenthèses dans chaque case. On constate que d'une manière générale, les méthodes MCTCorr1-1

TAB. 5.13 – Nombre et pourcentage d'individus pour lesquels les classifieurs binaires pertinents sont justes, les classifieurs correcteurs donnent des résultats erronés, et le classement final est faux ; la régression logistique est utilisée comme classifieur binaire 1-1.

Méthode	Satimage	Segment	Synth	Vowel	Waveform
MCTCorr	58 (1.50%)	12 (1.32%)	0	32 (6.93%)	41 (1.17%)
MCTProb	64 (1.66%)	13 (1.43%)	1 (0.29%)	39 (8.44%)	43 (1.23%)
PEstCorr	59 (1.53%)	67 (7.36%)	3 (0.88%)	27 (5.84%)	34 (0.97%)

TAB. 5.14 – Nombre et pourcentage d'individus pour lesquels les classifieurs binaires pertinents sont justes, les classifieurs correcteurs donnent des résultats erronés, et le classement final est faux ; les arbres de décision sont utilisés comme classifieurs binaires 1-1.

Méthode	Satimage	Segment	Synth	Vowel	Waveform
MCTCorr	29 (0.75%)	10 (1.10%)	1 (0.29%)	22 (4.76%)	13 (0.37%)
MCTProb	36 (0.93%)	11 (1.21%)	2 (0.59%)	25 (5.41%)	18 (0.51%)
PEstCorr	28 (0.73%)	65 (7.14%)	0	13 (2.81%)	2 (0.06%)

TAB. 5.15 – Nombre et pourcentage d'individus pour lesquels les classifieurs binaires pertinents sont justes, les classifieurs correcteurs donnent des résultats erronés, et le classement final est faux ; les classifieurs binaires 1-1 sont des réseaux de neurones évidentiels.

Méthode	Satimage	Segment	Synth	Vowel	Waveform
MCTCorr	38 (0.98%)	8 (0.88%)	1 (0.29%)	43 (9.31%)	54 (1.54%)
PEstCorr	28 (0.73%)	33 (3.63%)	3 (0.88%)	48 (10.39%)	64 (1.83%)

et MCTProb1-1 sont plus sujettes que la méthode PEstCorr à une dégradation du classement par l'étape de correction, excepté lorsque les réseaux de neurones évidentiels sont utilisés comme classifieurs binaires. Cependant, les résultats de classification obtenus avec ces méthodes étant généralement meilleurs, cela suggère que la méthode de combinaison a une importance prépondérante dans le classement d'un individu. Le jeu de données Segment constitue toutefois une exception : on constate en effet un nombre important de cas où l'étape de correction conduit à un mauvais classement par la méthode PEstCorr. Cela explique la dégradation globale des résultats par cette méthode, observée lors du traitement de ce jeu de données.

Analyse de la somme conjonctive des fonctions de masse dénormalisées

Lors d'une décomposition 1-1 (tableaux 5.4, 5.5 et 5.6), la méthode ConjCorr semble donner de meilleurs résultats que ceux obtenus par les méthodes ne faisant pas intervenir de correction. Dans le cas d'une décomposition CCE avec matrice de codes creuse, et lorsque les arbres de décision sont utilisés (tableau 5.11), les résultats semblent au contraire moins bons que ceux obtenus sans correction.

Ces classifieurs associent des probabilités d'appartenance à des régions de l'espace d'entrée, entre lesquelles la variation de probabilité est très brusque. Les zones où les informations fournies par les différents classifieurs sont conflictuelles sont donc plus nombreuses qu'avec les réseaux de neurones évidentiels. Cette tendance est d'autant plus marquée que le nombre de classifieurs est beaucoup plus important avec ce schéma de décomposition qu'avec les autres ; en outre, la correction des fonctions de masse par les plausibilités d'appartenance tend à accentuer le caractère conflictuel des fonctions de masse combinées.

Par conséquent, un nombre important de fonctions de masse combinées obtenues par la méthode ConjCorr vérifient $\widehat{m}(\emptyset) \simeq 1$. Lors de la prise de décision, de telles fonctions de masse sont associées à une distribution de probabilité pignistique uniforme sur l'ensemble des classes, et l'individu évalué est donc affecté à une classe au hasard.

5.3.2 Comportement de l'erreur par rapport au rejet en ambiguïté

Nous avons mis en place une procédure de rejet en ambiguïté lors de la prise de décision, qui consiste à affecter un individu $\mathbf{x}$ à ω_k si la probabilité $\widehat{p}_k$ maximale, ou la probabilité pigistique $BetP^*(\omega_k)$ maximale, est supérieure à un seuil λ_0 fixé. En faisant varier λ_0, nous avons pu mesurer les taux d'erreur et de rejet et tracer les courbes correspondantes.

Les courbes obtenues constituent donc une information plus riche que les simples taux de bonne classification, lors d'une étude de la précision des méthodes de combinaison. Elles permettent en outre de comparer la diversité des distributions de probabilité, ou de probabilité pignistique, obtenues par combinaison. En effet, l'appartenance d'un individu aux classes est variable selon sa localisation dans l'espace d'entrée. Pour certains, l'une des probabilités d'appartenance a une valeur significativement plus élevée que les autres ; d'autres au contraire partagent les caractéristiques de plusieurs classes à différents degrés, et les probabilités correspondantes sont associées à des valeurs significatives. Des taux de rejet répartis entre 0 et 1 caractérisent donc des données pour lesquelles les distributions de probabilité obtenues sont diverses : les différences entre les deux probabilités d'appartenance les plus élevées sont variables, selon les individus considérés. À l'inverse, l'observation d'un mode dans la répartition des taux de rejet suggère que la distribution de probabilité obtenue est semblable pour un grand nombre d'individus.

La figure 5.2 présente l'évolution des taux associés aux méthodes sans correction, obtenus lors du traitement des données Vowel par combinaison 1-1 de régression logistique ; la figure 5.3, les taux associés aux méthodes avec correction, obtenus en traitant les données Satimage au moyen de régression logistique, dans le cas d'une décomposition 1-1 également. On constate que les méthodes de combinaison MCT, PCplBT, PEst1 et PEst2, qui recherchent une solution consistante avec les données initiales, sont d'une précision semblable ; les couples de valeurs calculées correspondent à des taux de rejet répartis globalement entre 0 et 0.5. La variété des distributions de probabilité obtenues par ces méthodes reflète leur capacité à restituer la diversité des informations fournies par les classifieurs binaires.

Les méthodes MCTCorr et MCTProb sont les plus précises sur les données Satimage : à taux de rejet fixé, elles présentent les taux d'erreurs les moins importants, seule la méthode PEstCorr donnant des résultats légèrement meilleurs pour des taux de rejet compris entre 0.4 et 0.6. Remarquons en outre que les distributions de probabilité pignistique obtenues par ces méthodes présentent une diversité plus importante encore que pour les méthodes ne faisant pas intervenir de correction : les couples de valeur erreur-rejet correspondent à des taux de rejet répartis globalement entre 0 et 0.8. Les informations non pertinentes, écartées lors de la combinaison, n'étant pas reflétées par la solution, la distribution de proba-

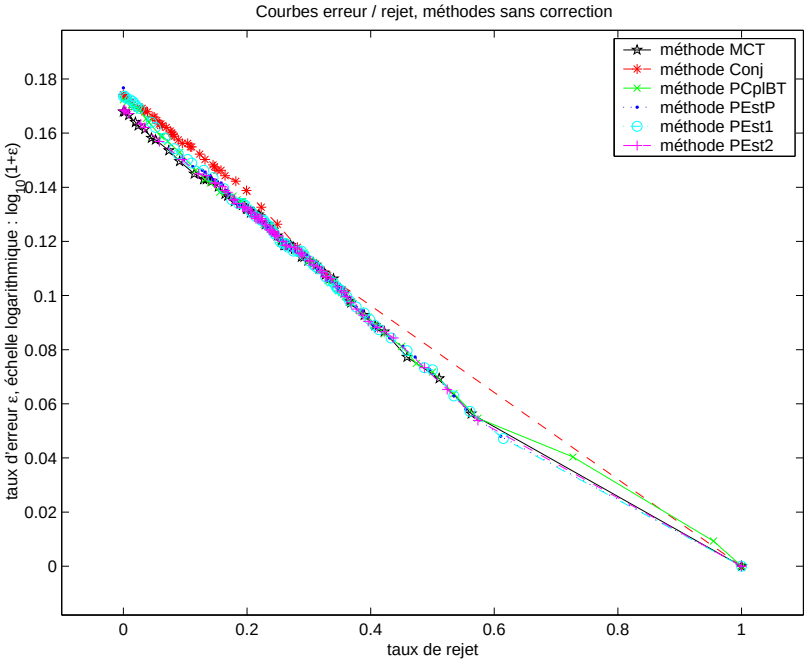


FIG. 5.2 – Courbes d'erreur-rejet, obtenues lors du traitement des données Vowel par combinaison 1-1 de régression logistique, par les méthodes MCT, Conj, PCplBT, PEstP, PEst1, et PEst2.

bilité pignistique déterminée est davantage représentative de l'appartenance de l'individu évalué aux différentes classes.

On peut remarquer que les couples de valeurs erreur-rejet obtenus pour la méthode Conj correspondent à des taux de rejet faibles, ou à un taux de rejet maximal ; un phénomène similaire peut être constaté pour la méthode ConjCorr. Cela suggère que les différences constatées entre les deux probabilités maximales sont soit faibles, soit très élevées. Rappelons qu'un nombre significatif de masses obtenues par somme conjonctive sont associées à une distribution de probabilité pignistique uniforme, les individus étant alors classés au hasard. Alternativement, lorsque les informations fournies par les classifieurs binaires ne sont pas totalement conflictuelles, la combinaison a tendance à déterminer des probabilités pignistiques très tranchées.

Enfin, on constate que les couples de valeurs caractérisant la méthode PEstCorr correspondent à des taux de rejet majoritairement inférieurs à 0.4. Rappelons que cette méthode consiste à calculer la moyenne des informations corrigées. Cela suggère que le compromis déterminé en faisant la moyenne des probabilités fournies par les classifieurs donne des probabilités d'appartenance moins tranchées que le consensus atteint par les méthodes MCTCorr et MCTProb.

La figure 5.4 présente les taux obtenus lors du traitement des données Vowel par combinaison de réseaux de neurones évidentiels, dans le cas d'une décompo-

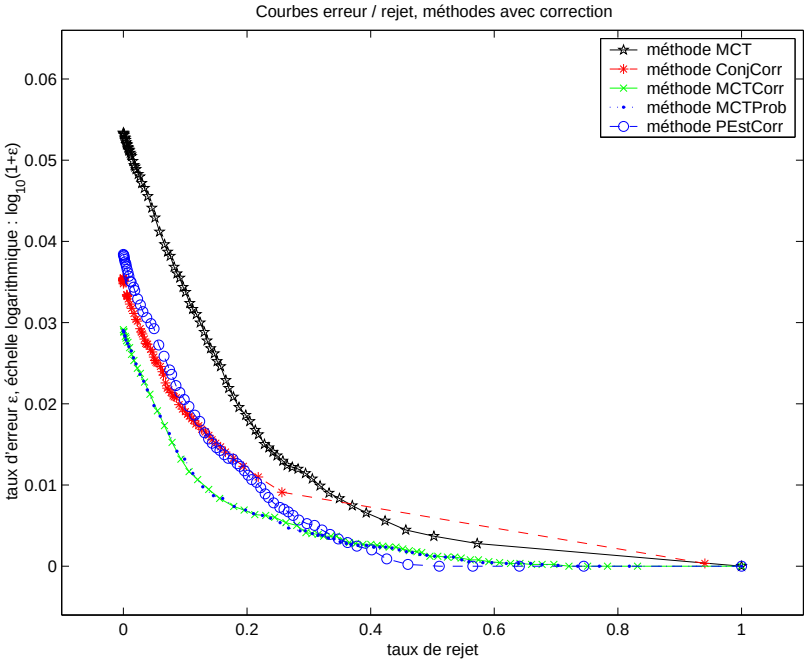


FIG. 5.3 – Courbes d'erreur-rejet, obtenues lors du traitement des données Satimage par régression logistique dans le cas d'une décomposition 1-1, par les méthodes MCT, ConjCorr, MCTCorr, MCTProb, et PEstCorr.

sition CCE avec matrice de codes dense. Les méthodes MCT, Conj et PCplBT donnent des résultats de précision similaire. Remarquons que les couples d'erreur-rejet obtenus pour la méthode Conj rendent compte d'une plus grande diversité dans les distributions de probabilité obtenues par combinaison que lorsque les arbres de décision sont utilisés : comme nous l'avons évoqué précédemment, le caractère conflictuel des informations fournies par ces derniers est plus marqué qu'avec les réseaux de neurones évidentiels. On constate par ailleurs que les distributions de probabilité obtenues par la méthode PEstP sont très proches de la distribution uniforme sur l'ensemble des classes, à de rares exceptions près où elles sont alors très tranchées. Cette méthode de combinaison suppose l'indépendance des classifieurs binaires, hypothèse qui n'est pas vérifiée dans le cas d'une décomposition CCE avec matrice de codes dense.

5.3.3 Adéquation de la solution aux données initiales

L'adéquation de la solution aux données initiales peut être évaluée en comparant les résultats obtenus à l'issue de la combinaison aux informations fournies par les classifieurs. Lorsque les sorties d'un classifieur sont interprétées comme des fonctions de masse conditionnelles ou grossières, la fonction de masse obtenue par combinaison peut être conditionnée ou réduite sur le domaine associé.

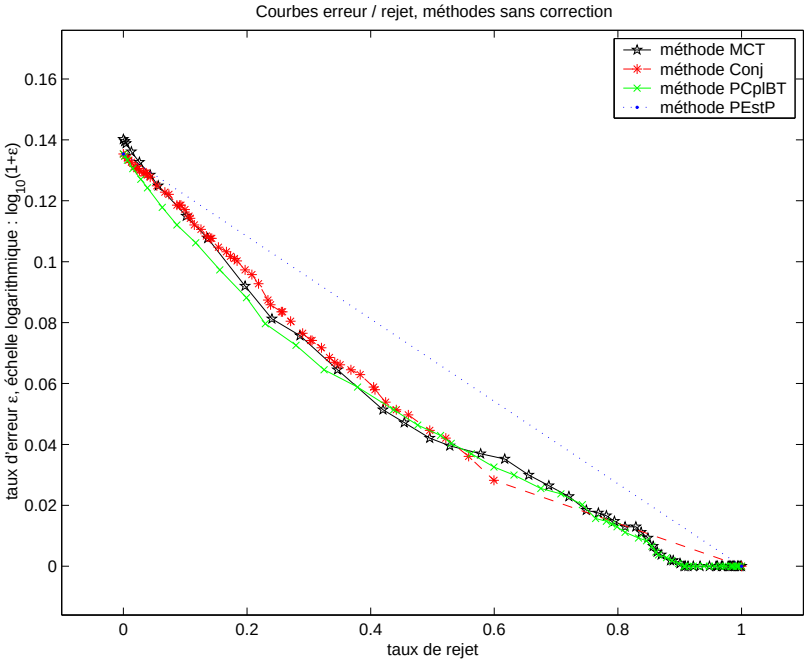


FIG. 5.4 – Courbes d'erreur-rejet, obtenues lors du traitement des données Vowel par combinaison de réseaux de neurones évidentiels (décomposition CCE avec matrice de codes dense), par les méthodes MCT, Conj, PCplBT, et PEstP.

Lorsqu'elles sont vues comme des probabilités conditionnelles, la distribution de probabilité obtenue peut être reconditionnée par rapport aux groupes de classes. Chaque classifieur peut ainsi être associé à un diagramme de Shepard : cette méthode simple de positionnement multidimensionnel permet de représenter les similitudes entre les informations obtenues et les données initiales sous forme d'un nuage de points.

Nous avons étudié la qualité de la reconstruction des informations fournies par les classifieurs, sur le jeu de données Synth, représenté sur la figure 3.1 du chapitre 3. Les couples de valeurs correspondant aux masses initiales et reconstruites ont été différenciés selon la classe réelle d'appartenance des individus correspondants.

Méthodes MCT, MCTCorr et MCTProb

Les figures 5.5 et 5.6 montrent la reconstruction des masses par la méthode MCT, dans le cas de décompositions 1-T et CCE avec matrice de codes dense, respectivement. Les réseaux de neurones évidentiels ont été utilisés comme classifieurs binaires. Dans le cas de la décomposition CCE dense, sept classifieurs ont été entraînés.

La reconstruction des masses obtenues par la méthode MCT est globalement satisfaisante. Remarquons que la qualité de la reconstruction dépend du schéma

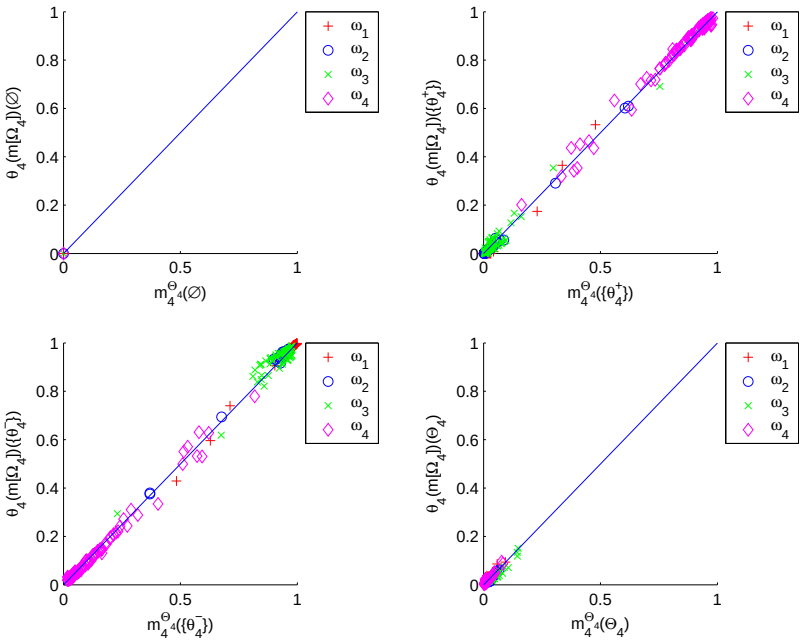


FIG. 5.5 – Méthode MCT, décomposition 1-T : masses $\widehat{m}^{\Theta_4}$ en fonction de $m_4^{\Theta_4}$, obtenues en traitant les données Synth au moyen de réseaux de neurones évidentiels.

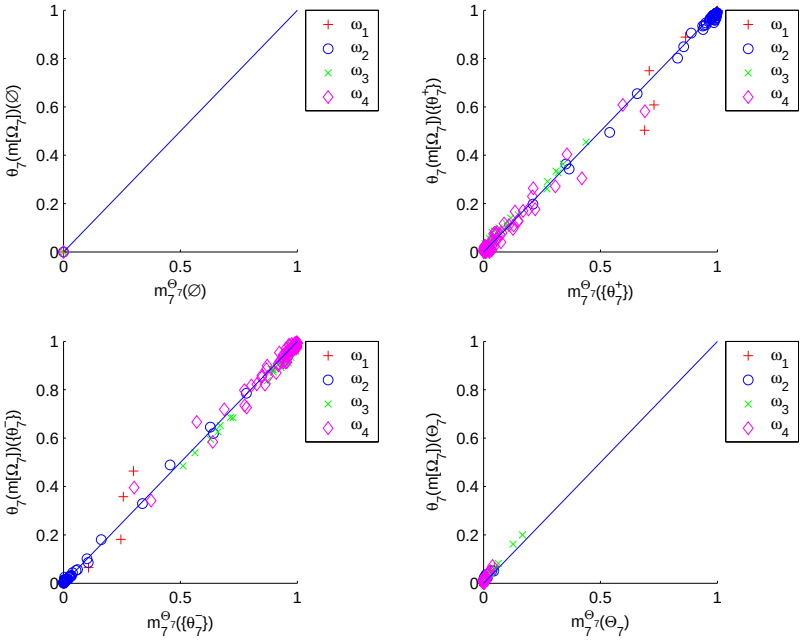


FIG. 5.6 – Méthode MCT, décomposition CCE avec matrice de codes dense : masses $\widehat{m}^{\Theta_7}$ en fonction de $m_7^{\Theta_7}$, obtenues en traitant les données Synth au moyen de réseaux de neurones évidentiels.

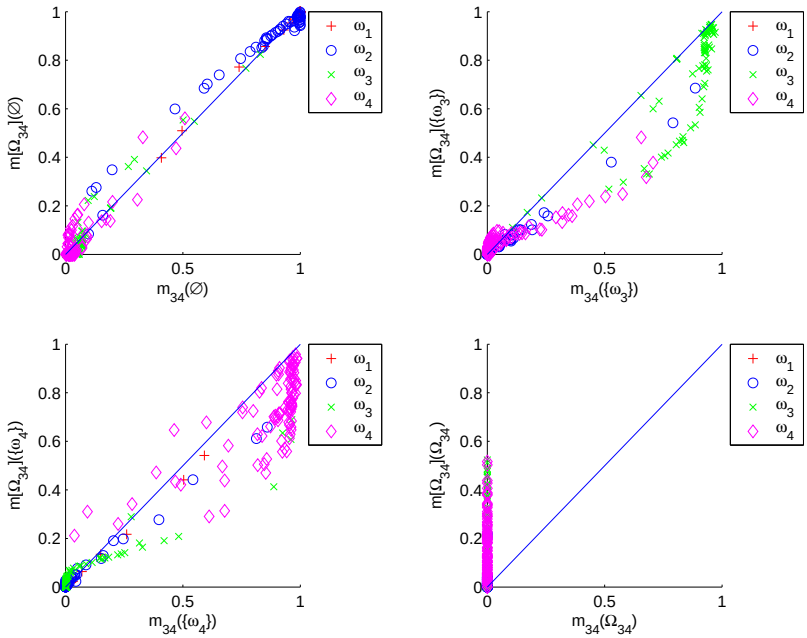


FIG. 5.7 – Méthode MCTCorr, décomposition 1-1 : masses $\widehat{m}[\Omega_{34}]$ en fonction de m_{34}, obtenues en traitant les données Synth par régression logistique.

de décomposition : l'augmentation du nombre d'informations combinées rend généralement la détermination d'un compromis plus difficile. Ainsi, la qualité de la reconstruction semble légèrement moins bonne dans le cas de la décomposition CCE avec matrice de codes dense, comme le montre la figure 5.6.

Les figures 5.7 et 5.8 montrent les masses $\widehat{m}[\Omega_{34}]$ en fonction des masses m_{34} obtenues par régression logistique, dans le cas d'une décomposition 1-1, par les méthodes MCTCorr et MCTProb, respectivement. La figure 5.9 montre les masses $\widehat{m}^{\Theta_{16}}$ en fonction des masses $m_{16}^{\Theta_{16}}$ obtenues par un arbre de décision. La décomposition CCE avec matrice de codes creuse compte dix-sept dichotomies : le nombre de classifieurs à combiner est donc beaucoup plus important que dans le cas d'une décomposition 1-1.

Dans le cas des méthodes MCTCorr et MCTProb, la reconstruction des masses est globalement satisfaisante pour les individus bien séparés par les classifieurs binaires. Les individus $\mathbf{x} \in \omega_3$ et $\mathbf{x} \in \omega_4$, situés à la frontière entre ω_3 et ω_4, sont caractérisés par des masses $m_{34}(\{\omega_3\})$ et $m_{34}(\{\omega_4\})$ comprises globalement entre 0.3 et 0.7. On constate pour ces individus une mauvaise reconstruction des données initiales, due au caractère conflictuel des informations combinées. En effet, l'affectation d'une masse de croyance significative à l'élément focal non singleton Ω_{34} dans la zone de recouvrement entre ω_3 et ω_4 induit des différences significatives entre les masses $m_{34}(\{\omega_3\})$ et $m_{34}(\{\omega_4\})$, comme le montrent les figures 5.7 et 5.8. Remarquons que ce phénomène est plus prononcé dans le cas de la méthode MCTProb, qui a tendance à diriger davantage de masse vers les élé-

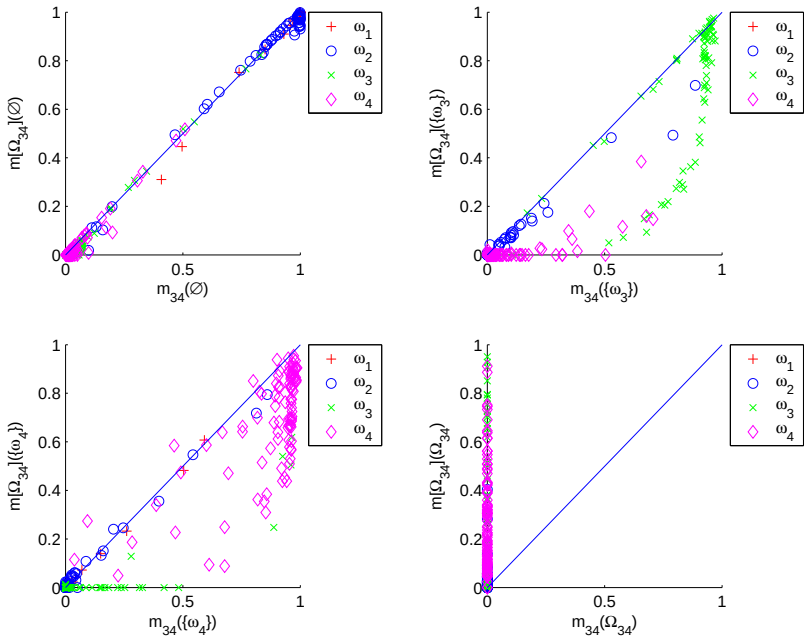


FIG. 5.8 – Méthode MCTProb, décomposition 1-1 : masses $\widehat{m}[\Omega_{34}](\emptyset)$ en fonction de $m_{34}(\emptyset)$, probabilités pignistiques non normalisées $BetP_{34}(\omega_3)$ et $BetP_{34}(\omega_4)$ en fonction de r_{34} et r_{43}, et masses $\widehat{m}[\Omega_{34}](\Omega_{34})$, obtenues en traitant les données Synth par régression logistique.

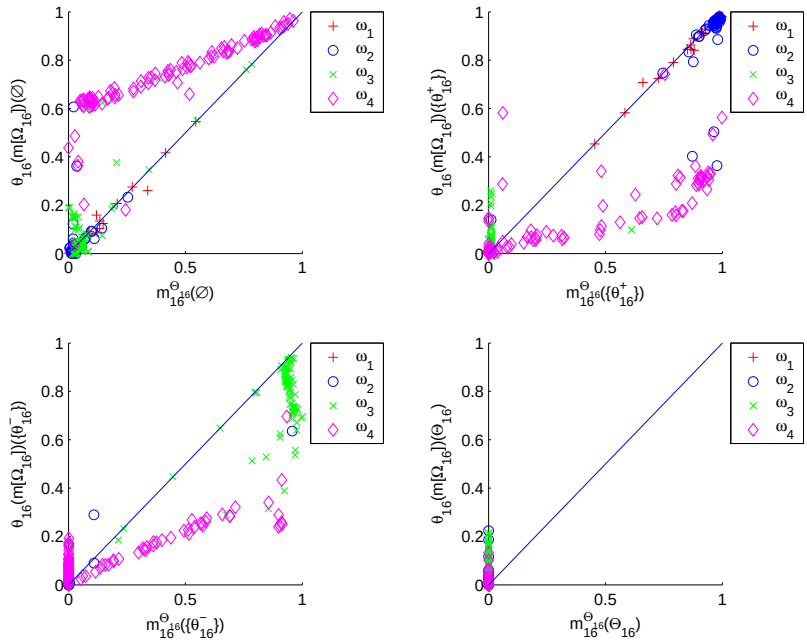


FIG. 5.9 – Méthode MCTCorr, décomposition CCE avec matrice de codes creuse : masses $\widehat{m}^{\Theta_{16}}$ en fonction de $m_{16}^{\Theta_{16}}$, obtenues en traitant les données Synth avec des arbres de décision.

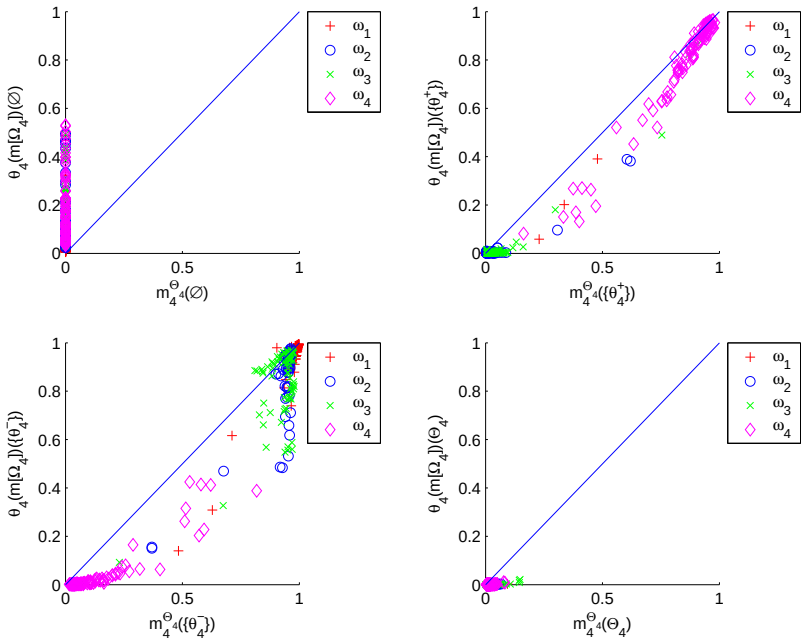


FIG. 5.10 – Méthode Conj, décomposition 1-T : masses $\widehat{m}^{\Theta_4 *}$ en fonction de $m_4^{\Theta_4}$, obtenues en traitant les données Synth par réseaux de neurones évidentiels.

ments focaux non singletons. Ici encore, l'augmentation du nombre de classifieurs combinés rend plus difficile la détermination d'un compromis entre les informations qu'ils fournissent, ce qui explique la baisse de la qualité de la reconstruction des masses obtenues par la méthode MCTCorr (figures 5.7 et 5.9).

Méthodes Conj et ConjCorr

Les figures 5.10 et 5.11 présentent la reconstruction des masses obtenues sur le jeu de données Synth, dans le cas de décompositions 1-T et CCE avec matrice de codes dense. Chaque classifieur binaire – ici, un réseau de neurones évidentiel – est donc entraîné à reconnaître l'ensemble des classes. On constate que la qualité de la reconstruction des masses diminue fortement lorsque le nombre de classifieurs combinés devient plus important – rappelons que sept classifieurs binaires ont été combinés dans le cas de la décomposition CCE avec matrice de codes dense, contre quatre dans le cas d'une décomposition 1-T. Conjointement, la masse allouée à l'ensemble vide augmente alors de manière significative, comme le montre la figure 5.11. Cela illustre le comportement de la somme conjonctive lors de la combinaison d'informations non distinctes. Le même phénomène peut être constaté lors de l'utilisation de la méthode ConjCorr. Remarquons que la correction des informations combinées semble donner lieu à une meilleure reconstruction des masses : les informations combinées sont en effet moins conflictuelles que lorsque seuls les classifieurs binaires sont considérés.

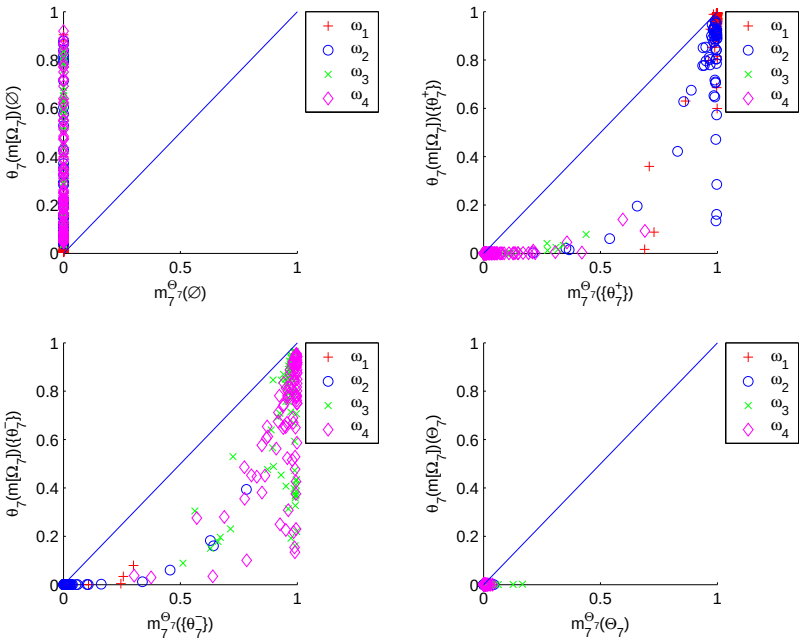


FIG. 5.11 – Méthode Conj, décomposition CCE avec matrice de codes dense : masses $\widehat{m}^{\Theta_7}$ en fonction de $m_7^{\Theta_7}$, obtenues en traitant les données Synth par réseaux de neurones évidentiels.

Méthodes PCplBT, PEst1, PEst2 et PEstCorr

La figure 5.12 montre les probabilités obtenues par les méthodes PCplBT, PEst1, PEst2 et PEstCorr reconditionnées, en fonction des probabilités fournies par les classifieurs, pour le schéma de décomposition 1-1. Les classifieurs binaires utilisés sont la régression logistique.

Lorsque le classifieur binaire est entraîné à reconnaître la classe d'un individu évalué, les probabilités reconditionnées sont généralement très proches des probabilités conditionnelles initiales ; les individus $\mathbf{x} \in \omega_2$, pour lesquels r_{34} est comprise entre 0.3 et 0.7, sont caractérisés par une adéquation légèrement moins bonne. La qualité de la reconstruction est médiocre pour les individus dont la classe réelle n'appartient pas à l'ensemble d'apprentissage du classifieur. L'adéquation entre les probabilités combinées par la méthode PCplBT puis reconditionnées et les sorties des classifieurs semble toutefois légèrement meilleure que pour les autres méthodes.

Méthode PEstP

La figure 5.13 montre les probabilités obtenues par la méthode PEstP puis reconditionnées, en fonction des probabilités fournies par les classifieurs, pour les différents schémas de décomposition considérés. Les classifieurs binaires utilisés

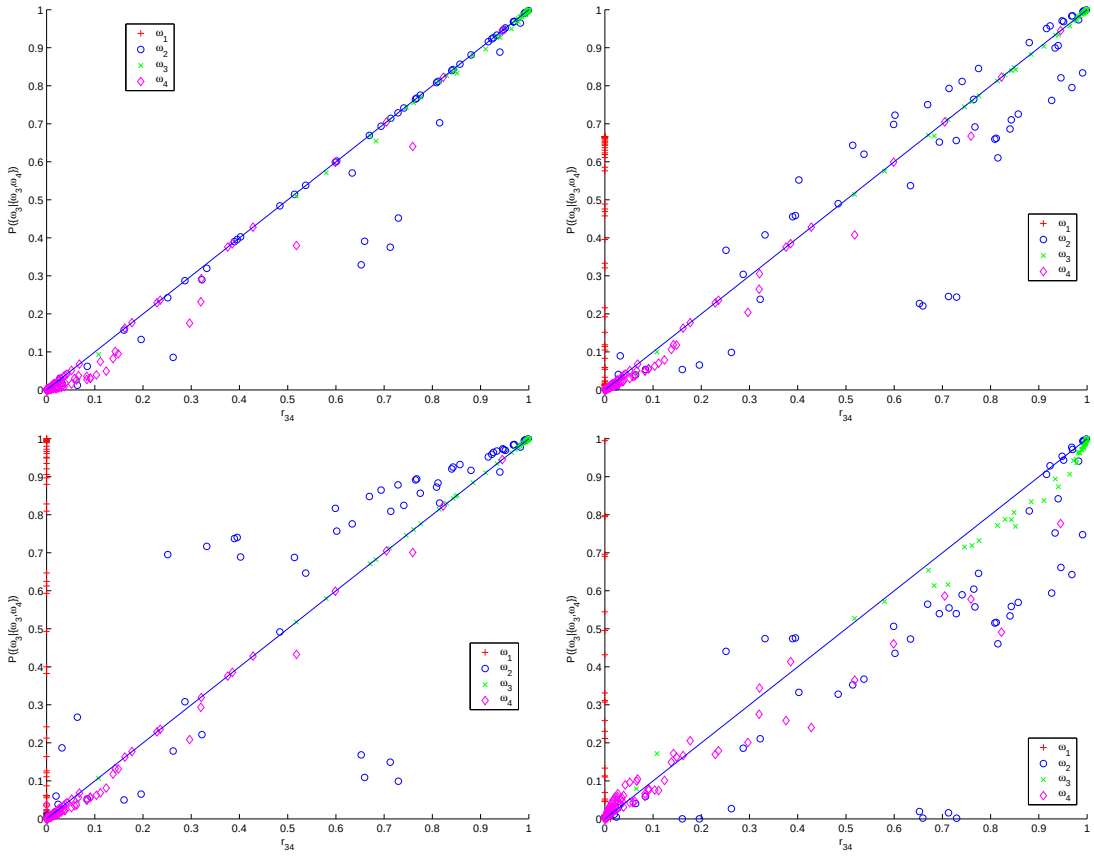


FIG. 5.12 – Probabilités combinées reconditionnées $\widehat{\mathbb{P}}(\{\omega_3\}|\{\omega_3,\omega_4\})$ exprimées en fonction des probabilités r_{34} fournies par le classifieur $\mathcal{E}_{34}$ (décomposition 1-1), pour les méthodes PCplBT (haut-gauche), PEst1 (haut-droite), PEst2 (bas-gauche), et PEstCorr (bas-droite).

sont la régression logistique pour la décomposition 1-1, et les réseaux de neurones évidentiels dans les autres cas.

On constate que la qualité de la reconstruction diminue avec l'augmentation du nombre de classifieurs : l'adéquation entre les masses initiales et les masses reconstruites est ainsi beaucoup moins bonne dans le cas d'une décomposition CCE avec matrice de codes dense, pour laquelle sept classifieurs sont combinés, que dans le cas d'une décomposition 1-T, où seuls quatre classifieurs sont considérés. Rappelons que dans ces deux cas, le classifieur binaire a été entraîné à reconnaître toutes les classes. Ce phénomène met donc en évidence l'instabilité de la méthode, qui suppose l'indépendance des classifieurs, lors de la combinaison d'informations non distinctes.

On remarque en outre que les probabilités reconditionnées sont constantes et indépendantes des probabilités fournies par le classifieur binaire considéré, pour un nombre significatif d'individus que ce dernier n'a pas été entraîné à reconnaître. Ce phénomène peut être constaté pour les individus $\mathbf{x} \in \omega_2$ dans le cas d'une décomposition 1-1, le classifieur $\mathcal{E}_{34}$ ayant été entraîné à reconnaître $\{\omega_3\}$ et $\{\omega_4\}$; et particulièrement pour les individus $\mathbf{x} \in \omega_4$ dans le cas d'une décomposition CCE avec matrice de codes creuse, le classifieur $\mathcal{E}_{16}$ sachant séparer $\{\omega_1, \omega_2\}$ de $\{\omega_3\}$. Rappelons que la méthode de combinaison PEstP définie par l'équation (2.40), présentée au paragraphe 2.4.2 du chapitre 2, consiste à exprimer la probabilité $\widehat{p_k}$ d'appartenance à ω_k comme la somme d'un terme $\prod_{i:e_{ki}\neq 0} \mathbb{P}(e_{\mathbf{x}i} = e_{ki})$ correspondant à la combinaison des probabilités fournies par les classifieurs binaires, et d'un terme α mesurant le conflit entre les différents classifieurs. Prenons l'exemple d'un individu $\mathbf{x} \in \omega_2$, dans le cas d'une décomposition 1-1. Lorsque les probabilités conditionnelles r_{12}, r_{23} et r_{24} modélisent l'appartenance de $\mathbf{x}$ à ω_2, les probabilités $\widehat{p_1}$, $\widehat{p_3}$ et $\widehat{p_4}$ obtenues par combinaison tendent vers α. En conséquence, les probabilités reconditionnées $(\widehat{p_1} + \widehat{p_2})/(\widehat{p_1} + \widehat{p_2} + \widehat{p_3})$ tendent vers $2/3$. Ce phénomène est plus marqué dans le cas d'une décomposition CCE avec matrice de codes creuse. Le type de classifieur utilisé est alors plus précis que la régression logistique : la proportion d'individus $\mathbf{x} \in \omega_4$ pour lesquels $\widehat{p_1}$, $\widehat{p_2}$ et $\widehat{p_3}$ tendent vers α est alors plus élevée. De plus, le nombre de classifieurs étant plus important que dans le cas d'une décomposition 1-1, davantage d'informations modélisent l'appartenance d'un individu $\mathbf{x} \in \omega_4$ à la classe ω_4.

Ce phénomène suggère que la méthode PEstP fournit des probabilités peu représentatives de la distribution des classes. En outre, on peut imaginer que cette méthode est peu robuste aux erreurs des classifieurs : lors de l'évaluation d'un individu $\mathbf{x} \in \omega_k$, chaque classifieur entraîné à reconnaître ω_k fournissant une faible probabilité d'appartenance à cette classe contribue à faire diminuer la probabilité $\widehat{p_k}$ obtenue par combinaison.

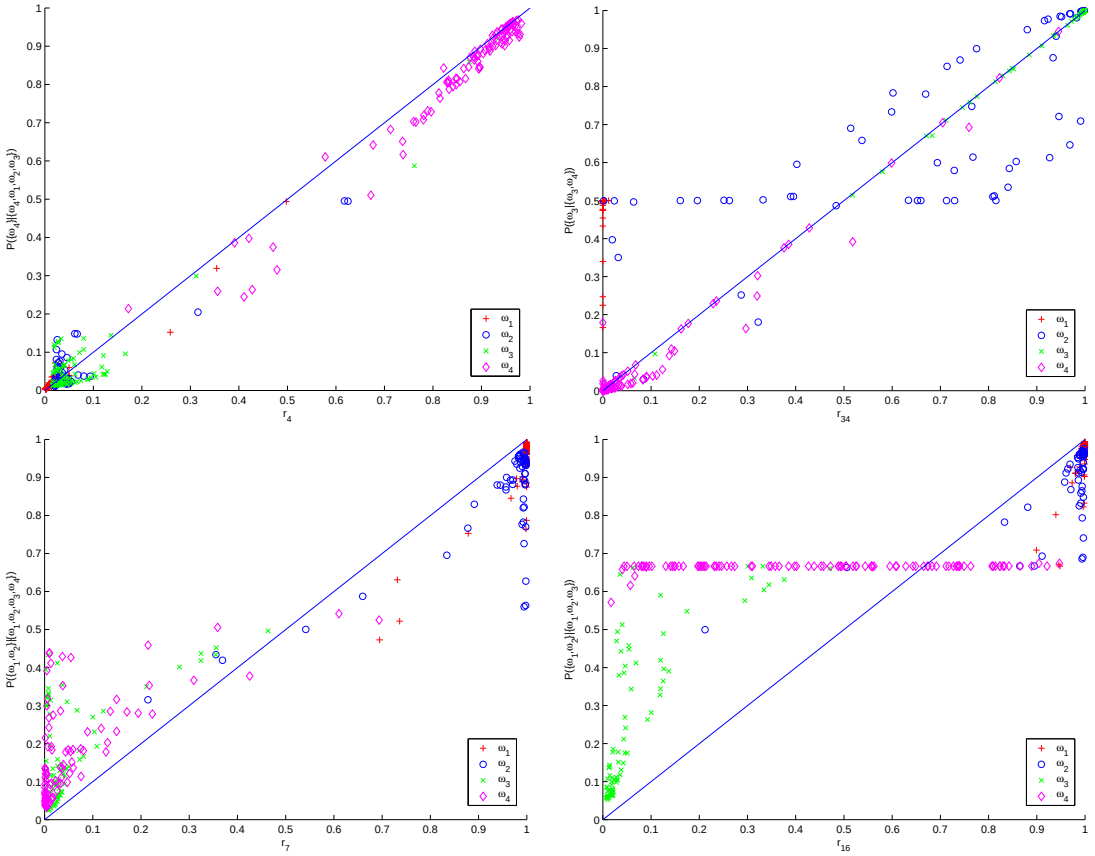


FIG. 5.13 – Probabilités combinées exprimées en fonction des probabilités fournies par les classifieurs, pour la méthode PEstP. Décomposition 1-T, réseaux de neurones évidentiels (haut-gauche) : $\widehat{\mathbb{P}}(\{\omega_4\})$ en fonction de r_4 ; décomposition 1-1, régression logistique (haut-droite) : $\widehat{\mathbb{P}}(\{\omega_3\}|\{\omega_3,\omega_4\})$ en fonction de r_{34} ; décomposition CCE avec matrice de codes dense, réseaux de neurones évidentiels (bas-gauche) : $\widehat{\mathbb{P}}(\{\omega_1,\omega_2\})$ en fonction de r_7 ; décomposition CCE avec matrice de codes creuse, réseaux de neurones évidentiels (bas-droite) : $\widehat{\mathbb{P}}(\{\omega_1,\omega_2\}|\{\omega_1,\omega_2,\omega_3\})$ en fonction de r_{16}.

5.4 Synthèse

Dans ce chapitre, les méthodes de combinaison de classifieurs binaires présentées dans ce mémoire ont été évaluées, et leurs résultats ont été comparés à ceux de méthodes déjà existantes.

Les tests effectués mettent en évidence l'intérêt de prendre en compte la pertinence des classifieurs, dans les cas où certaines classes sont laissées de côté lors de la phase d'apprentissage. L'évaluation de la précision de ces méthodes, et l'analyse de l'adéquation du résultat de la combinaison aux données initiales, mettent en évidence la supériorité des méthodes qui recherchent une solution consistante avec les informations fournies par les classifieurs binaires.

Les méthodes Conj et PEstP fournissent des résultats dont qualité dépend de l'indépendance des classifieurs combinés. L'une comme l'autre supposant l'indépendance de ces classifieurs, la solution obtenue ne modélise généralement pas l'appartenance de l'individu évalué aux différentes classes de manière satisfaisante, contrairement aux méthodes MCT, PCplBT, PEst1 et PEst2. Les résultats obtenus par ces méthodes sont meilleurs lorsqu'un schéma de décomposition par codes correcteurs d'erreurs est utilisé : le caractère incomplet des informations fournies par les classifieurs est compensé par leur nombre.

Les méthodes MCTCorr, MCTProb, ConjCorr, et PEstCorr fournissent généralement de meilleurs résultats que les autres, grâce à l'étape supplémentaire permettant d'évaluer la pertinence des classifieurs binaires. La qualité des résultats obtenus par la méthode ConjCorr est toutefois tributaire de l'indépendance des classifieurs. Ces méthodes donnent de meilleurs résultats lorsque le schéma de décomposition un-contre-un est utilisé. Cela est vraisemblablement dû au fait que les classifieurs entraînés sont alors moins dépendants les uns des autres que dans le cas d'une décomposition par codes correcteurs d'erreurs.

Le schéma de décomposition 1-1 étant plus simple à mettre en œuvre – la procédure de détermination d'une matrice de codes peut être coûteuse, l'implémentation des méthodes de combinaison crédales faisant intervenir une étape de correction semble donc être une approche intéressante pour la résolution d'un problème de classification.

Conclusion

Synthèse générale

Le travail présenté dans ce mémoire a permis d'étudier la combinaison de classifieurs binaires dans le cadre du Modèle des Croyances Transférables.

Ce cadre théorique permet de modéliser de manière simple et intuitive le caractère incomplet des informations données par les classifieurs binaires. Si un classifieur n'est pas entraîné à reconnaître la totalité des classes correspondant au problème initial, les informations qu'il fournit peuvent alors être interprétées comme des fonctions de croyance conditionnelles, définies sur un cadre de discernement restreint. Si son ensemble d'apprentissage est formé de groupes de plusieurs classes, le classifieur est incapable de faire la différence entre les classes appartenant à un même groupe, et ses sorties peuvent alors être vues comme des fonctions de croyance définies sur un cadre plus grossier que le cadre de discernement initial.

Les classifieurs sont combinés en recherchant la fonction de masse la plus consistante possible avec les informations disponibles, par résolution d'un problème d'optimisation. Selon l'algorithme de classification employé, les classifieurs binaires n'étant pas entraînés à reconnaître la classe réelle de l'individu évalué peuvent donner des informations erronées ; la combinaison directe de leurs sorties peut mener au calcul d'une solution biaisée. La pertinence des informations fournies par un classifieur peut alors être évaluée : elle est assimilée à la plausibilité que l'individu évalué appartienne à l'une des classes composant l'ensemble d'apprentissage du classifieur. Du point de vue du Modèle des Croyances Transférables, cela revient à dénormaliser la fonction de masse modélisant ces informations avant de la combiner avec les autres. Les plausibilités d'appartenance permettent également de combiner des classifieurs probabilistes, en interprétant leurs sorties comme des probabilités pignistiques définies sur les cadres correspondants.

Les résultats obtenus sur plusieurs jeux de données de la littérature mettent en évidence la robustesse des méthodes développées. La combinaison des fonctions de masse normales donne globalement des résultats similaires à d'autres méthodes de combinaison fonctionnant sur le même principe, par recherche d'une solution consistante avec les données disponibles. La dénormalisation des fonctions de masse au moyen des plausibilités d'appartenance permet généralement d'amélio-

rer les résultats de classification de manière significative. Une étude plus détaillée des résultats obtenus montre en outre que la fonction de masse déterminée par combinaison reflète les informations fournies par les classifieurs de manière satisfaisante.

Remarquons enfin que les méthodes de combinaison proposées ne se limitent pas à la simple combinaison de classifieurs binaires : elles permettent en effet de déterminer la solution d'un problème lorsque les informations disponibles sont exprimées sur des domaines différents, ne se limitant pas nécessairement à deux éléments. Les méthodes peuvent ainsi être aisément adaptées lorsque les cadres de définition des sorties des classifieurs sont de taille quelconque, en utilisant les opérateurs de conditionnement et de réduction appropriés. De même qu'une distribution de probabilité peut être vue comme une fonction de croyance Bayésienne, des distributions de possibilité ou de nécessité peuvent être interprétées dans le cadre du Modèle des Croyances Transférables. Les méthodes de combinaison proposées peuvent donc également être appliquées à la combinaison de classifieurs fournissant des données hétérogènes, procédant de formalismes différents.

En conclusion, le travail réalisé dans ce mémoire a permis de proposer plusieurs méthodes de combinaison permettant de fusionner des sources d'informations non indépendantes, de natures diverses, pouvant modéliser une connaissance imparfaite.

Perspectives

Un certain nombre d'aspects des méthodes proposées peuvent faire l'objet de travaux ultérieurs.

L'estimation des plausibilités d'appartenance est une étape cruciale permettant d'améliorer les résultats de manière significative. Or, la méthode utilisée dans ce mémoire est purement empirique : il existe à ce jour très peu de travaux portant sur l'estimation d'une distribution de plausibilité — ou, de manière équivalente, d'une distribution de possibilité — à partir de données. Remarquons que l'algorithme de SVM à une classe, utilisé pour estimer la plausibilité d'appartenance à une classe, fait l'objet d'un nombre croissant de travaux, notamment pour la détection de nouveauté. Cet algorithme a été initialement proposé pour estimer le support d'une distribution multidimensionnelle ; son extension au calcul de la plausibilité d'appartenance au support d'une classe semble donc être une piste prometteuse.

Soulignons que les méthodes de combinaison proposées nécessitent actuellement de résoudre un problème d'optimisation quadratique ; bien que des algorithmes performants permettent de s'en acquitter, l'utilisation d'une règle de combinaison non itérative constituerait naturellement un atout précieux pour le traitement de tâches nécessitant de la rapidité. Une règle de combinaison conjonc-

tive prudente a récemment été proposée par Denœux [13]. Cet opérateur consiste à n'utiliser qu'une seule fois un même élément d'information apporté par les différentes sources : il est donc bien mieux adapté que la règle conjonctive pour combiner des classifieurs non indépendants. Il semble donc intéressant de tester l'utilisation de cet opérateur pour combiner les sorties des classifieurs binaires.

L'utilisation des méthodes proposées dans ce mémoire pour résoudre d'autres problèmes de combinaison pourrait également être envisagée. L'élicitation d'avis d'experts est une approche consistant à représenter l'avis d'un individu par une mesure de confiance ; de récents travaux [64] ont montré l'intérêt de la théorie des fonctions de croyance pour modéliser ces opinions. Il est souvent plus judicieux de ne soumettre à un expert qu'un ensemble limité d'hypothèses, de manière à le confronter à un problème simple, correspondant à son domaine d'expertise ; les méthodes de combinaison proposées dans ce mémoire semblent donc être l'outil adapté pour obtenir une fonction de croyance représentant l'ensemble des connaissances des différents experts, à partir des fonctions de croyance élicitées sur des problèmes plus simples.

D'une manière générale, la théorie des fonctions de croyance s'est montrée intéressante pour formaliser la combinaison de classifieurs entraînés à reconnaître des ensembles de classes différents, de même que pour d'autres techniques comme le bagging [22]. Il semble donc prometteur d'aborder d'autres problèmes d'apprentissage statistique, permettant de combiner des classifieurs entraînés à résoudre le même problème dans le but d'améliorer leurs performances individuelles, dans une perspective évidentielle. Nous pensons en particulier à la technique du boosting, qui peut être utilisée conjointement à la méthode des codes correcteurs d'erreurs pour résoudre des problèmes de classification multiclasses [49].

Bibliographie

[1] S. Abe and T. Inoue. Fuzzy support vector machines for multiclass problems. In M. Verleysen, editor, *ESANN*, pages 113–118, 2002.

[2] E. L. Allwein, R. E. Schapire, and Y. Singer. Reducing multiclass to binary : a unifying approach for margin classifiers. In *Proc. 17th International Conf. on Machine Learning*, pages 9–16. Morgan Kaufmann, San Francisco, CA, 2000.

[3] R. Anand, K. Mehrotra, C. K. Mohan, and S. Ranka. Efficient classification for multiclass problems using modular neural networks. *IEEE Transactions on Neural Networks*, 6(1) :117–124, Jan. 1995.

[4] A. Appriou. Probabilités et incertitude en fusion de données multi-senseurs. *Revue Scientifique et Technique de la Défense*, 11 :27–40, 1991.

[5] A. Appriou. Approche générique de la gestion de l'incertain dans les processus de fusion multisenseur. *Revue Traitement du Signal*, 22(4) :307–319, 2005.

[6] L. Bottou, C. Cortes, J. S. Denker, H. Drucker, I. Guyon, L. Jackel, Y. LeCun, U. A. Muller, E. Sackinger, P. Simard, and V. Vapnik. Comparison of classifier methods : a case study in handwritten digit recognition. In *Proceedings of the 12th IAPR International Conference on Pattern Recognition, Conference B : Computer Vision & Image Processing.*, volume 2, pages 77–82, Jerusalem, October 1994. IEEE.

[7] L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone. *Classification and Regression Trees*. Wadsworth International Group, Belmont, CA, 1984.

[8] P. Clark and R. Boswell. Rule induction with CN2 : Some recent improvements. In Y. Kodratoff, editor, *EWSL*, volume 482, pages 151–163. Springer, 1991.

[9] K. Crammer and Y. Singer. On the learnability and design of output codes for multiclass problems. *Machine Learning*, 47(2-3) :201–233, 2002.

[10] F. Cutzu. Polychotomous classification with pairwise classifiers : a new voting principle. In *Multiple Classifier Systems*, pages 115–124, 2003.

[11] T. Denœux. Application du modèle des croyances transférables en reconnaissance de formes. *Traitement du Signal*, 14(5) :443–451, 1998.

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

[13] T. Denœux. A cautious rule of combination and some extensions. In *International Conference on Information Fusion (Fusion'06)*, Florence, July 2006.

[14] T. Denœux and A. B. Yaghlane. Approximating the combination of belief functions using the fast möbius transform in a coarsened frame. *International Journal of Approximate Reasoning*, 31(1), 2002.

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

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

[17] D. Dubois, L. Foulloy, G. Mauris, and H. Prade. Probability-possibility transformations, triangular fuzzy sets, and probabilistic inequalities. *Reliable Computing*, 10 :273–297, 2004.

[18] D. Dubois and H. Prade. *Théorie des Possibilités. Applications à la Représentation des Connaissances en Informatique*. Collection Méthode + Programmes, Masson, Paris, 2 edition, 1985. Seconde édition revue et augmentée, Masson, Paris, 1987.

[19] D. Dubois and H. Prade. A set-theoretic view of belief functions : logical operations and approximations by fuzzy sets. *International Journal of General Systems*, 12 :193–226, 1986.

[20] D. Dubois and H. Prade. Representation and combination of uncertainty with belief functions and possibility measures. *Computational Intelligence*, 4 :244–264, 1988.

[21] S. Fabre, A. Appriou, and X. Briottet. Presentation and description of two classification methods using data fusion based on sensor management. *Information Fusion*, 2(1) :49–71, 2001.

[22] J. François, Y. Grandvalet, T. Denœux, and J.-M. Roger. Bagging belief structures in Dempster-Shafer K-NN rule. In *Proceedings of IPMU'2000*, volume 1, pages 111–118, Madrid, July 2000.

[23] Y. Freund and R. E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. In *European Conference on Computational Learning Theory*, pages 23–37, 1995.

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

[25] J. 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.

[26] 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.

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

[28] C.-W. Hsu and C.-J. Lin. A comparison of methods for multi-class support vector machines. *IEEE Transactions on Neural Networks*, 13(2) :415–425, 2002.

[29] T.-K. Huang, R. C. Weng, and C.-J. Lin. Generalized Bradley-Terry models and multi-class probability estimates. *Journal of Machine Learning Research*, 7 :85–115, 2006.

[30] T. Inoue and S. Abe. Fuzzy support vector machines for pattern classification. In *IJCNN*, pages 1449–1454, 2001.

[31] F. Janez and A. Appriou. Théorie de l'évidence et cadres de discernement non exhaustifs. *Revue Traitement du Signal*, 13(3) :237–250, 1996.

[32] F. Janez and A. Appriou. Theory of evidence and non-exhaustive frames of discernment : Plausibilities correction methods. *International Journal of Approximate Reasoning*, 18(1-2) :1–19, 1998.

[33] B. Kijsirikul, N. Ussivakul, and S. Meknavin. Adaptive directed acyclic graphs for multiclass classification. In M. Ishizuka and A. Sattar, editors, *PRICAI*, volume 2417 of *Lecture Notes in Computer Science*, pages 158–168. Springer, 2002.

[34] S. Knerr, L. Personnaz, and G. Dreyfus. Single-layer learning revisited : a stepwise procedure for building and training a neural network. In F. Fogelman-Soulie and J. Herault, editors, *Neurocomputing : Algorithms, Architectures and Applications – NATO ASI Series*, volume F68. Springer Verlag, Berlin, Germany, 1990.

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

[36] U. Kreßel. Pairwise classification and support vector machines. In C. J. C. B. B. Scholkopf and A. J. Smola, editors, *Advances in Kernel Methods : Support Vector Learning*. The MIT Press, Cambridge, MA, 1999.

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

[38] M. Masson and T. Denœux. Inferring a possibility distribution from empirical data. *Fuzzy Sets and Systems*, 157(3) :319–340, 2006.

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

[40] A. Passerini, M. Pontil, and P. Frasconi. New results on error correcting output codes of kernel machines. *IEEE Transactions on Neural Networks*, 15(1) :45–54, 2004.

[41] 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.

[42] D. Price, S. Knerr, L. Personnaz, and G. Dreyfus. Pairwise neural network classifiers with probabilistic outputs. In G. Tesauro, D. Touretzky, and T. Leen, editors, *Advances in Neural Information Processing Systems*, volume 7, pages 1109–1116. The MIT Press, 1995.

[43] O. Pujol, P. I. Radeva, and J. Vitria. Discriminant ECOC : A heuristic method for application dependent design of error correcting output codes. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 28(6) :1007–1012, 2006.

[44] J. R. Quinlan. *C4.5 : Programs for Machine Learning*. Morgan Kaufmann, 1992.

[45] B. Quost, T. Denœux, and M. Masson. Pairwise classifier combination in the framework of belief functions. In *Proceedings of FUSION'2005*, Philadelphia, PA, July 2005.

[46] B. Quost, T. Denœux, and M. Masson. One-against-all classifier combination in the framework of belief functions. In *Proceedings of IPMU'2006*, volume 1, pages 356–363, Paris, July 2006.

[47] P. Réfrégier and F. Vallet. Probabilistic approach for multiclass classification with neural networks. In *Proceedings of International Conference on Artificial Networks*, pages 1003–1007, 1991.

[48] R. Rifkin and A. Klautau. In defense of one-vs-all classification. *Journal of Machine Learning Research*, 5 :101–141, 2004.

[49] R. E. Schapire. Using output codes to boost multiclass learning problems. In *Proceedings of the 14th International Conference on Machine Learning*, pages 313–321. Morgan Kaufmann, 1997.

[50] 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.

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

[52] P. Smets. Belief functions : the disjunctive rule of combination and the generalized bayesian theorem. *International Journal of Approximate Reasoning*, 9 :1–35, 1993.

[53] P. Smets. The application of the transferable belief model to diagnostic problems. *International Journal of Intelligent Systems*, 13 :127–157, 1998.

[54] P. Smets. The transferable belief model for quantified belief representation. In D. M. Gabbay and P. Smets, editors, *Handbook of Defeasible Reasoning and Uncertainty Management Systems, Vol. 1*, pages 267–301. Kluwer, Doordrecht, The Netherlands, 1998.

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

[56] F. Takahashi and S. Abe. Optimizing directed acyclic graph support vector machines. In *Proceedings of Conference on Artificial Neural Networks in Pattern Recognition*, pages 166–170, 2003.

[57] D. Tsujinishi, Y. Koshiba, and S. Abe. Why pairwise is better than one-against-all or all-at-once. In *Proceedings of IEEE International Joint Conference on Neural Networks*, volume 1, pages 693–698, 2004.

[58] P. Vannoorenberghe and T. Denœux. Handling uncertain labels in multiclass problems using belief decision trees. In *IPMU'2002*, volume 3, pages 1919–1926, 2002.

[59] V. Vapnik. *The Nature of Statistical Learning Theory*. Springer, N.Y., 1995.

[60] P. Walley. *Statistical Reasoning with Imprecise Probabilities*. Chapman and Hall, London, 1991.

[61] 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.

[62] L. Xu, A. Krzyzak, and C. Y. Suen. Methods of combining multiple classifiers and their applications to handwriting recognition. *IEEE Transactions on Systems, Man and Cybernetics*, 22 :418–435, 1992.

[63] R. R. Yager. On the Dempster-Shafer framework and new combination rules. *Information Sciences*, 41(2) :93–137, 1987.

[64] A. B. Yaghlane, T. Denœux, and K. Mellouli. Elicitation of expert opinions for constructing belief functions. In *Information Processing with Managing Uncertainty*, volume 1, pages 403–411, Paris, France, July 2006.

[65] B. Zadrozny. Reducing multiclass to binary by coupling probability estimates. In T. G. Dietterich, S. Becker, and Z. Ghahramani, editors, *Advances in Neural Information Processing Systems*, volume 14, Cambridge, MA, 2002. MIT Press.

[66] L. M. Zouhal and T. Denœux. An evidence-theoretic k-NN rule with parameter optimization. *IEEE Trans. on Systems, Man and Cybernetics C*, 28(2) :263–271, 1998.