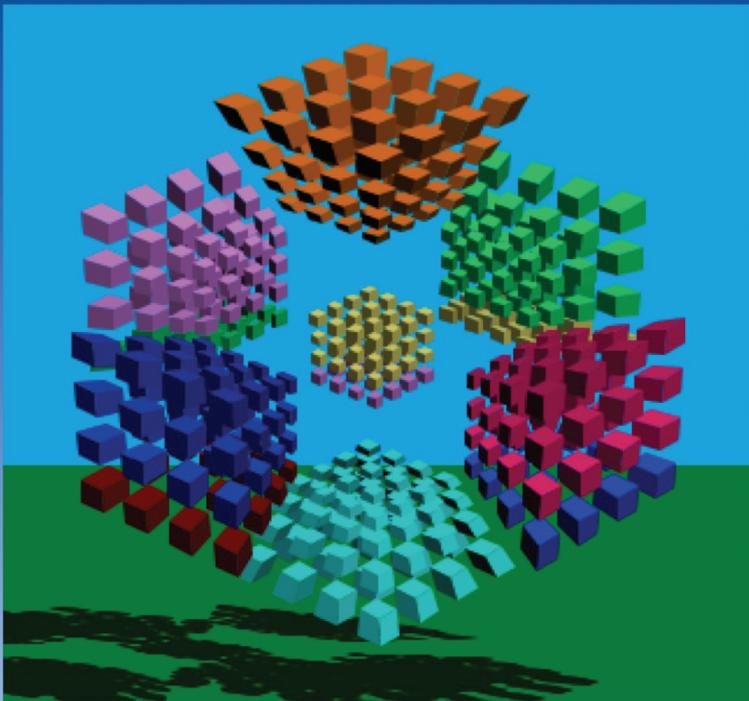




**EGC'2013**

**13<sup>e</sup> Conférence Francophone sur  
l'Extraction et la Gestion de Connaissances**  
Université Paul Sabatier – IRIT – Toulouse

**29 janvier 2013 – Journée Ateliers/Tutoriels**



**Fouille de Données  
Complexes, complexité  
liée aux données  
multiples et massives  
(FDC)**







**Extraction et Gestion des Connaissances**

**Toulouse, 29 janvier 2013**

## **10ème Atelier sur la Fouille de Données Complexes**

### **Complexité liée aux données multiples et massives**

#### **Responsables**

Guillaume Cleuziou (LIFO, Université d'Orléans)

Cyril de Runz (CreSTIC, Université de Reims)

Germain Forestier (MIPS, Université de Haute Alsace)

Mustapha Lebbah (LIPN, Université Paris 13)





# 10ème Atelier sur la Fouille de Données Complexes

## Complexité liée aux données multiples et massives

Guillaume Cleuziou\*, Cyril de Runz\*\*, Germain Forestier\*\*\*, Mustapha Lebbah\*\*\*\*

\*LIFO, Université d'Orléans  
guillaume.cleuziou@univ-orleans.fr

\*\*CReSTIC, Université de Reims  
cyril.de-runz@univ-reims.fr

\*\*\*MIPS, Université de Haute Alsace  
germain.forestier@uha.fr

\*\*\*\*LIPN, Université Paris 13  
mustapha.lebbah@univ-paris13.fr

**Résumé.** L'atelier sur la fouille de données complexes est proposé à l'instigation du groupe de travail EGC "Fouille de données complexes". Chaque année les organisateurs proposent une thématique de recherche qui suscite l'intérêt des chercheurs et des industriels.

Les précédentes éditions de l'atelier ayant permis de fédérer avec succès des recherches portant sur la problématique de complexité liée à la multiplicité des données, nous avons choisi de renouveler l'affichage de cette priorité thématique pour cette dixième édition.

## 1 Le groupe de travail "Fouille de Données Complexes"

La dixième édition de l'atelier sur la fouille de données complexes est organisée par le groupe de travail EGC "Fouille de Données Complexes". Ce groupe de travail rassemble une communauté de chercheurs et d'industriels désireux de partager leurs expériences et problématiques dans le domaine de la fouille de données complexes telles que le sont les données non-structurées (ou faiblement), les données obtenues à partir de plusieurs sources d'information ou plus généralement les données spécifiques à certains domaines d'application et nécessitant un processus d'extraction de connaissance sortant des itinéraires usuels de traitement.

Les activités du groupe de travail s'articulent autour de trois champs d'action progressifs :

- l'organisation de **journées scientifiques** une fois par an (vers le mois de juin) où sont présentés des travaux en cours ou plus simplement des problématiques ouvertes et pendant lesquelles une large place est faite aux doctorants,
- l'organisation de l'**atelier "Fouille de Données Complexes"** associé à la conférence EGC qui offre une tribune d'expression pour des travaux plus avancés et sélectionnés sur la base d'articles scientifiques par un comité de relecture constitué pour l'occasion,
- la préparation de **numéros spéciaux** de revue nationale, dans lesquels pourront être publiés les études abouties présentées dans un format long et évaluées plus en profondeur par un comité scientifique. Le second numéro spécial a été publié en 2011 et le prochain sera lancé en 2013.

## 2 Contenu scientifique de l'atelier

Nous avons reçu cette année 9 propositions, chacune d'elle a été relue par au moins deux rapporteurs. Pour la grande majorité des propositions nous avons été en mesure de proposer trois rapports d'experts afin d'offrir un processus scientifique constructif aux auteurs. Nous avons retenu 7 propositions en fonction de leur degré d'avancement ainsi que de leur cohérence avec les thématiques de l'atelier.

Les articles qui vous sont proposés cette année explorent une variété édifiante de complexités, aussi bien dans les données que dans les processus de fouille envisagés :

La **complexité liée à la multiplicité des sources d'information** - ce qui correspond au thème prioritaire de l'atelier - et bien sûr abordée avec d'une part deux études considérant l'analyse et l'intégration de la dépendance entre sources d'informations et d'autre part une contribution méthodologique pour la recherche de classifications pertinentes en présence de données décrites par plusieurs sources.

Cette tâche de classification, qui est au cœur de nombreux processus de fouille de données, est encore plus largement présente dans le contenu scientifique de l'atelier. Cette problématique peut en effet induire une **complexité liée à la structure des classes** recherchée (e.g. la recherche de classes recouvrantes) ou bien nécessiter des traitements spécifiques de part la **complexité liée à la nature des données** elles-mêmes (e.g. données biologiques, séries temporelles, images à haute résolution). Tous ces sujets sont traités par les auteurs dans les actes qui suivent.

### **3 Comité de programme**

- Hanane Azzag (LIPN, Université Paris 13)
- Omar Boussaïd (ERIC, Université de Lyon 2)
- Martine Cadot (LORIA, Nancy)
- Guillaume Cleuziou (LIFO, Université d'Orléans)
- Cyril De Runz (CReSTIC, Université de Reims Champagne-Ardenne)
- Sylvie Desprès (LIM&BIO, Université Paris 13)
- Mounir Dhibi (ISSAT, Gafsa)
- Matthieu Durut (Telecom ParisTech)
- Zied Elouedi (ISG, Université de Tunis)
- Sami Faiz (INSAT, Université du 7 Novembre de Carthage)
- Cécile Favre (ERIC, Université Lyon 2)
- Germain Forestier (MIPS, Université de Haute Alsace)
- Nicolas Labroche (LIP6, Université Pierre et Marie Curie, CLEAR)
- Anne Laurent (LIRMM, Université de Montpellier 2)
- Mustapha Lebbah (LIPN, Université Paris 13)
- Eric Lefèvre (LGI2A, Université d'Artois)
- Arnaud Martin (IRISA, Université Rennes 1)
- Florent Masségli (AxIS-Inria Sophia Antipolis)
- Christophe Osswald (ENSIETA, Brest)
- Sébastien Régis (LAMIA, Université des Antilles et de la Guyane)
- Brigitte Trousse (AxIS, INRIA Sophia Antipolis)
- Cédric Wemmert (ICube, Université de Strasbourg)

### **4 Remerciements**

Nous tenons à remercier les auteurs pour la qualité de leurs contributions, les membres du comité de programme et plus généralement tous les relecteurs de cet atelier pour le travail accompli et pour la qualité de leurs prestations.

Nous remercions également les responsables des ateliers pour EGC 2013.

Enfin nous remercions vivement les présidents : Christel Vrain, présidente du comité de programme, Florence Sèdes et André Péninou co-présidents du comité d'organisation d'EGC 2013.

## 5 Programme

- 14h15 Ouverture de l'atelier
- 14h30 Généralisation d'un opérateur d'agrégation totalement renforcé pour des séries temporelles  
*Richard Emilion, Sébastien Regis et Andrei Doncescu*
- 14h55 Mesure de dépendance positive et négative de sources crédibilistes  
*Mouna Chebbah, Arnaud Martin et Boutheina Ben Yaghlane*
- 15h20 Intégration d'une mesure d'indépendance pour la fusion d'informations  
*Mouloud Kharoune et Arnaud Martin*
- 15h45 Étude comparative d'algorithmes filtres pour la sélection de motifs biologiques  
*Hanen Mhamdi et Faouzi Mhamdi*
- 16h15-16h45 Pause
- 16h45 Une plateforme collaborative pour la recherche de consensus ou d'alternatives en classification non-supervisée  
*Jacques-Henri Sublemontier*
- 17h10 Apprentissage de règles floues pour caractériser des objets d'intérêt dans une image de télédétection  
*Bruno Belarte, Cédric Wemmert et Germain Forestier*
- 17h35 Un modèle générique pour les k-moyennes à recouvrements ajustables  
*Chiheb-Eddine Ben N'Cir, Guillaume Cleuziou et Nadia Essoussi*
- 18h00 Discussion et conclusion de l'atelier

## Summary

The workshop on mining complex data is done at the incitement of the work group EGC "Complex data mining". Each year the organizers propose a topic that interests the researchers and companies.

Previous years the workshop focused with success on the complexity associated with multiple data. Indeed we decided to keep this research field as one of the priority for the present edition of the workshop.





# Généralisation d'un opérateur d'agrégation totalement renforcé pour des séries temporelles.

Richard Emilion\*, Sébastien Régis\*\*, Andrei Doncescu\*\*\*

\*MAPMO

Université d'Orléans,  
Bâtiment de mathématiques  
Rue de Chartres B.P. 6759 - 45067 Orléans cedex 2  
richard.emilion@univ-orleans.fr  
<http://www.univ-orleans.fr/mapmo/>

\*\*LAMIA

Université des Antilles et de la Guyane,  
Fouillole BP 250 - 97157 Pointe-à-Pitre  
sregis@univ-ag.fr  
<http://lamia.servhome.org/>

\*\*\*LAAS-CNRS

7, avenue du Colonel Roche 31077 Toulouse Cedex 4  
adoncesc@laas.fr  
<http://laas.fr>

**Résumé.** Dans cet article, nous présentons une généralisation de l'opérateur d'agrégation  $\mathcal{G}3\mathcal{I}$ . Cette généralisation est appelée  $\mathcal{G}3\mathcal{I}$ , et peut être utilisée pour résumer des éléments d'une série temporelle sur un intervalle. Quelques-unes des inégalités de cette généralisation sont présentées et démontrées.

## 1 Introduction

Les avancées technologiques au niveau des capteurs et du stockage ont permis une véritable explosion des données numériques, augmentant sans cesse la quantité, la qualité et la diversité de ces informations. Ainsi, au début du siècle la question était "Peut-on récupérer et stocker ces données ?" et aujourd'hui la problématique est plutôt la suivante : "comment exploiter toutes ces données ?". En effet l'adage qui stipule que trop d'information tue l'information n'a jamais été aussi vrai. Les masses de données, leur volume et leur complexité sont telles aujourd'hui qu'elles nécessitent des outils informatiques de plus en plus sophistiqués. Diverses approches issues de l'informatique et des mathématiques appliquées ont été proposées pour gérer ces informations. Par exemple, les opérateurs d'agrégation issus de la théorie des sous-ensembles flous permettent une synthèse de ces informations diverses et variées. Il existe de nombreux opérateurs d'agrégation et le choix d'un de ces opérateurs dépend de nombreux paramètres. Mais ce choix est avant tout lié à la *fusion* elle-même. Avant d'aller plus loin il nous semble important de rappeler la définition de la fusion. Nous reprenons la définition

## Généralisation d'un opérateur d'agrégation pour des séries temporelles

proposée par (Bloch et Hunter, 2001) qui fut donnée lors des travaux du Groupe Européen de Travail sur la Fusion (FUSION) :

*La fusion consiste à réunir ou agréger des informations provenant de différentes sources, et à exploiter cette information réunie ou agrégée, dans diverses applications comme la réponse à une question, la prise de décision, une estimation numérique, etc.*

Cette définition met l'accent sur deux éléments principaux. D'abord, elle met l'emphase sur la combinaison de l'information. Puis l'accent est mis sur l'objectif de la fusion.

Un opérateur d'agrégation peut être définie comme une fonction  $f : [0, 1]^n \rightarrow [0, 1]$ ,  $n \in \mathbb{N}$ , qui vérifient au moins la condition de monotonie, c'est-à-dire que si les valeurs marginales à fusionner augmentent, l'opérateur de fusion doit augmenter aussi, ou du moins, ne pas diminuer (Dubois et Prade, 2004). Les opérateurs d'agrégation  $f$  (qu'on peut aussi noter  $f(x_1, \dots, x_n)$ ) peuvent être divisés en 4 groupes à partir d'une comparaison avec les opérateurs min (minimum) et max (maximum) (Pradera et Trillas, 2006; Beliakov et al., 2007) (voir aussi Martin (2005)) :

- les opérateurs *conjonctifs* qui vérifient  $f \leq \min$ . Les normes triangulaires et les copulas (Nelsen, 1999) appartiennent à cette classe.
- les opérateurs *disjonctifs* qui vérifient  $f \geq \max$ . Les conormes triangulaires appartiennent à ce groupe.
- les opérateurs *moyenne* qui vérifient  $\min \leq f \leq \max$ . Par exemple les OWA (Yager, 1988, 2004) appartiennent à cette classe.
- Les opérateurs *hybrides*. Ce groupe représente les autres opérateurs qui ne sont pas forcément comparables aux opérateurs min et max. Par exemple les connecteurs mixtes (Zimmerman et Zynso, 1980; Piera-Carreté et al., 1988), les uninormes (Yager, 1994; Yager et Rybalov, 1996, 1998), les nullnorms (Calvo et al., 2001) et les sommes symétriques (Silvert, 1979), appartiennent à ce groupe.

Ces opérateurs sont parfaitement adaptés pour agréger des informations ponctuelles issues de plusieurs variables. Cependant les données sont telles aujourd'hui qu'elles sont souvent organisées en séries temporelles, voire en flux de données. On cherche donc à traiter maintenant des plages de données sur des intervalles de temps plutôt qu'une information ponctuelle. Ces opérateurs d'agrégation sont peu usités pour ces données arrivant en continu. On notera cependant qu'ils sont utilisés sur des séries temporelles afin de :

- lisser des données sur ces séries temporelles (Yager, 2008)
- faire de la prévision sur ces celles-ci (Komornik et al., 2006; Jilani et Burney, 2008)

En d'autres termes, on peut dire que les opérateurs d'agrégation sont surtout utilisés d'un point de vue local et non pour résumer l'information sur un intervalle temporel. La question qui en découle est la suivante : Peut-on trouver ou définir des opérateurs d'agrégation qui synthétisent des plages de données issues de séries temporelles ?

En effet, agréger des informations d'une série temporelle sur un intervalle donné, devrait permettre d'avoir une vision plus globale plutôt que d'avoir une vision locale du problème. Par ailleurs, nous nous sommes intéressés à un opérateur totalement renforcé appelé 3II (ou triple II) et à un opérateur de type moyenne issu de ce 3II et nommé M3II. La question sous-jacente est donc : peut-on trouver une généralisation de cet opérateur 3II qui soit configurée pour travailler sur des plages de données de séries temporelles ?

Dans cet article, nous tentons de répondre à cette dernière question. Notre étude propose une généralisation de l'opérateur 3II qui peut être utilisée sur des séries temporelles voire des

fonctions monodimensionnelles. La communication est organisée comme suit. Le deuxième paragraphe présente quelques rappels sur les opérateurs  $3\Pi$  et  $M3\Pi$ . La section 3 présente la généralisation de l'opérateur. Certaines des propriétés de cette généralisation sont présentées et démontrées. Dans la section 4, nous proposons quelques pistes en perspective pour des applications futures de cette généralisation.

## 2 Les opérateurs $3\Pi$ et $M3\Pi$

### 2.1 L'opérateur $3\Pi$

Avant de fournir la définition de l'opérateur  $3\Pi$ , rappelons deux propriétés importantes de cet opérateur :

- l'opérateur  $3\Pi$  est une uninorme (Yager et Rybalov, 1998).
- l'opérateur  $3\Pi$  est une généralisation des sommes symétriques (Waissman-Vilanova, 2000).

Les uninormes ont été proposées par (Yager et Rybalov, 1996) tandis que les sommes symétriques ont été proposées par (Silvert, 1979). Le triple  $\Pi$  est défini comme suit :

$$PI(x_1, \dots, x_n) = \frac{\prod_{j=1}^n x_j}{\prod_{j=1}^n x_j + \prod_{j=1}^n (1 - x_j)} \quad (1)$$

avec  $x_j \in [0, 1]$ .

On notera que dans le cas problématique où le dénominateur serait nul, Yager propose de définir arbitrairement le triple  $\Pi$  comme étant égal à 0 ou à 1 (Yager, 2002). Cet opérateur  $3\Pi$  a comme particularité d'être totalement renforcé c'est-à-dire qu'il est renforcé positivement et négativement. Rappelons également les définitions de ces notions de renforcement.

Un opérateur d'agrégation  $L$  dont les arguments sont dans l'intervalle  $[0, 1]$ , a la propriété de renforcement positif si lorsque tous ses attributs sont affirmatifs (i.e. supérieur ou égaux à 0,5) il vérifie :

$$L(x_1, \dots, x_n) \geq \max_i [L(x_i)] \quad (2)$$

De façon similaire, un opérateur d'agrégation  $L$  dont les arguments sont dans l'intervalle  $[0, 1]$ , a la propriété de renforcement négatif si lorsque tous ses attributs sont non-affirmatifs (i.e. inférieur ou égaux à 0,5), il vérifie :

$$L(x_1, \dots, x_n) \leq \min_i [L(x_i)] \quad (3)$$

Un opérateur qui possède les deux propriétés est défini comme étant *totalement renforcé (full reinforced)*.

Le renforcement total est une propriété intéressante car elle modélise le comportement humain (Elkan, 1994), (Yager et Rybalov, 1998).

### 2.2 La moyenne $M3\Pi$

L'opérateur  $M3\Pi$  se définit comme suit :

$$M3\Pi(x_1, \dots, x_n) = \frac{\prod_{j=1}^n (x_j)^{(1/n)}}{\prod_{j=1}^n (x_j)^{(1/n)} + \prod_{j=1}^n (1 - x_j)^{(1/n)}} \quad (4)$$

## Généralisation d'un opérateur d'aggrégation pour des séries temporelles

avec  $x_j \in [0, 1]$  Cet opérateur ne peut être totalement renforcé du fait de sa nature (c'est une moyenne) mais il possède une propriété ayant une certaine similitude avec celle de renforcement total du  $3\Pi$ . Cette propriété est présentée ci-dessous.

Soit  $M3\Pi$  la moyenne triple  $\Pi$ . On considère la moyenne arithmétique classique  $\frac{1}{n} \sum_{j=1}^n x_j$ . Alors :

Si  $\forall j \in 1, \dots, n$ , on a  $x_j \geq 0,5$ , on a :

$$M3\Pi(x_1, \dots, x_n) \geq \frac{1}{n} \sum_{j=1}^n x_j \quad (5)$$

Si  $\forall j \in 1, \dots, n$ , on a  $x_j \leq 0,5$ , on a :

$$M3\Pi(x_1, \dots, x_n) \leq \frac{1}{n} \sum_{j=1}^n x_j \quad (6)$$

Cette propriété a été baptisée *renforcement moyen* (Emilion et al., 2004; Doncescu et al., 2007) en référence au renforcement total du triple  $\Pi$ .

## 3 Généralisation de l'opérateur $3\Pi$ pour des séries temporelles

### 3.1 Liminaire

Définissons la généralisation d'une somme symétrique :

$$PI(x_1, \dots, x_n) = \frac{\prod_{j=1}^n G(x_j)}{\prod_{j=1}^n G(x_j) + \prod_{j=1}^n G(1-x_j)} \quad (7)$$

où  $G(x)$  est une fonction appelée *fonction génératrice* et qui est non négative et croissante (Waissman-Vilanova, 2000; Silvert, 1979). Il est clair que pour définir l'opérateur triple  $\Pi$  on prend la fonction  $G(x_j) = x_j$  et pour la moyenne triple  $\Pi$  on prend la fonction  $G(x_j) = x_j^{1/n}$ . A partir de l'équation 7, on cherche cette fois une généralisation de la fonction  $G$  pour différents cas : ensembles finis, cas discret et cas continu. On va donc définir une généralisation de l'opérateur que l'on pourra appliquer à des fonctions.

### 3.2 Définition

#### 3.2.1 Définition et inégalité

##### Définition

Soit  $f : \Omega \rightarrow [0, 1]$  une fonction mesurable définie sur un espace de probabilité  $(\Omega, \mathcal{F}, \mu)$  et prenant ses valeurs dans  $[0, 1]$ . Alors,

$$G3\Pi = \frac{e^{\int_{\Omega} \log(f) d\mu}}{e^{\int_{\Omega} \log(f) d\mu} + e^{\int_{\Omega} \log(1-f) d\mu}} \quad (8)$$

est appelée *généralisation de  $3\Pi$*  et est écrit  $G3\Pi$ .

Evidemment, le  $G3\Pi$  n'est pas un opérateur d'agrégation mais il possède des propriétés similaires à celles du  $3\Pi$  comme nous le voyons ci-dessous.

**Proposition 1**

Soit  $f : \Omega \rightarrow (0, \frac{1}{2}]$  une fonction mesurable définie sur un espace de probabilités  $(\Omega, \mathcal{F}, \mu)$  et prenant ses valeurs dans  $(0, \frac{1}{2}]$ . Alors

$$\frac{e^{\int_{\Omega} \log(f) d\mu}}{e^{\int_{\Omega} \log(f) d\mu} + e^{\int_{\Omega} \log(1-f) d\mu}} \leq \int_{\Omega} f d\mu \quad (9)$$

Si  $f$  prend ses valeurs dans  $[\frac{1}{2}, 1)$  alors l'inégalité inverse est vérifiée.

Notons que l'intégrale ci-dessus est bien définie puisque  $\log(f) < 0 \mu - a.e.$  et  $e^{-\infty} = 0$ .

Cette inégalité théorique et générale est la plus importante de cet article car toutes les autres inégalités en découlent. Le cas fini correspond à un opérateur d'agrégation ; c'est en fait la définition de la moyenne  $M3\Pi$  et de ses propriétés. Le cas discret et le cas continu permettent de considérer de nouvelles applications pour les séries temporelles (voire des flots de données), et montrent qu'on peut trouver une généralisation du  $3\Pi$  pour des séries temporelles. Nous détaillerons les 3 cas (fini, discret et continu) que nous venons de mentionner. Mais avant cela, il convient de donner la preuve de l'équation 9.

### 3.2.2 Preuve de l'inégalité (9)

Si  $\int_{\Omega} \log(f) d\mu = -\infty$  alors  $e^{\int_{\Omega} \log(f) d\mu} = 0$  et l'inégalité (9) est vérifiée de façon triviale.

On suppose maintenant que  $-\infty < \int_{\Omega} \log(f) d\mu \leq \frac{1}{2}$  de sorte que  $0 < e^{\int_{\Omega} \log(f) d\mu} < +\infty$ . Par ailleurs  $\int_{\Omega} f d\mu > 0$  puisque  $f > 0$ . Ainsi, l'inégalité (9) est équivalente à :

$$\begin{aligned} \frac{1}{\int_{\Omega} f d\mu} &\leq \frac{e^{\int_{\Omega} \log(f) d\mu} + e^{\int_{\Omega} \log(1-f) d\mu}}{e^{\int_{\Omega} \log(f) d\mu}} \\ \frac{1}{\int_{\Omega} f d\mu} &\leq 1 + e^{\int_{\Omega} (\log(1-f) - \log(f)) d\mu} \\ \frac{1}{\int_{\Omega} f d\mu} - 1 &\leq e^{\int_{\Omega} \log(\frac{1}{f} - 1) d\mu} \\ \log\left(\frac{1}{\int_{\Omega} f d\mu} - 1\right) &\leq \int_{\Omega} \log\left(\frac{1}{f} - 1\right) d\mu \\ \Phi\left(\int_{\Omega} f d\mu\right) &\leq \int_{\Omega} \Phi(f) d\mu \end{aligned}$$

avec  $\Phi(x) = \log(\frac{1}{x} - 1)$ . La dernière inégalité est vraie puisqu'elle représente une application de l'inégalité classique de Jensen (Jensen, 1906). En effet  $0 < f \leq \frac{1}{2}$  et la fonction  $\Phi$  est convexe sur l'intervalle  $(0, \frac{1}{2}]$  puisque  $\Phi'(x) = -\frac{1}{1-x} - \frac{1}{x} = \frac{1}{x^2-x}$  et  $\Phi''(x) = \frac{1-2x}{(x^2-x)^2} > 0$  pour  $0 < x \leq \frac{1}{2}$ .

### 3.3 Le cas fini

Soit  $\Omega = \{1, \dots, n\}$  et soit  $\mu$  une mesure de probabilité uniforme  $\mu\{i\} = \frac{1}{n}$  pour  $i = 1, \dots, n$ . Si  $f(i)$  est noté  $x_i$ , alors, avec la convention que  $\log(0) = -\infty$  et  $e^{-\infty} = 0$ , on a :

$$\int_{\Omega} f d\mu = \frac{1}{n} \sum_{i=1}^n x_i,$$

$$\int_{\Omega} \log(f) d\mu = \frac{1}{n} \sum_{i=1}^n \log(x_i) = \log((x_1 \dots x_n)^{\frac{1}{n}})$$

et

$$e^{\int_{\Omega} \log(f) d\mu} = (x_1 \dots x_n)^{\frac{1}{n}}$$

ainsi dans le cas fini l'inégalité (9) peut être écrite :

$$\frac{(x_1 \dots x_n)^{\frac{1}{n}}}{(x_1 \dots x_n)^{\frac{1}{n}} + ((1-x_1) \dots (1-x_n))^{\frac{1}{n}}} \leq \frac{1}{n} \sum_{i=1}^n x_i \quad (10)$$

quand  $0 \leq x_i \leq \frac{1}{2}$  pour tout  $i$  ; l'inégalité inverse est vérifiée quand  $\frac{1}{2} \leq x_i \leq 1$  pour tout  $i$ .

Comme nous l'avons indiqué plus haut, l'inégalité (10) a déjà été prouvée et elle représente la principale propriété de la Moyenne 3Π (Emilion et al., 2004). On peut trouver plus de détails dans (Doncescu et al., 2007), avec un exemple d'application dans les bioprocédés. L'intérêt majeur de M3Π est de fournir une meilleure discrimination que la moyenne arithmétique classique.

### 3.4 Le cas discret

Plus généralement, soit  $\Omega = \{0, 1, \dots, n, \dots\}$  et soit  $\mu$  une distribution de probabilité sur  $\Omega$ . Si  $f(i)$  est notée  $x_i$ , alors, avec les mêmes conventions que ci-dessus, on a :

$$\int_{\Omega} f d\mu = \sum_{i=0}^{+\infty} x_i \mu\{i\},$$

$$\int_{\Omega} \log(f) d\mu = \sum_{i=0}^{+\infty} \log(x_i) \mu\{i\} = \log(\prod_{i=0}^{+\infty} x_i^{\mu\{i\}})$$

et par conséquent

$$e^{\int_{\Omega} \log(f) d\mu} = \prod_{i=0}^{+\infty} x_i^{\mu\{i\}}$$

de sorte que dans le cas discret, l'inégalité (9) peut être réécrite comme suit :

$$\frac{\prod_{i=0}^{+\infty} x_i^{\mu\{i\}}}{\prod_{i=0}^{+\infty} x_i^{\mu\{i\}} + \prod_{i=0}^{+\infty} (1-x_i)^{\mu\{i\}}} \leq \sum_{i=0}^{+\infty} x_i \mu\{i\}. \quad (11)$$

Par exemple si  $\mu$  est une distribution géométrique ( $p$ ) avec le paramètre  $0 \leq p \leq 1$ , telle que  $\mu\{i\} = (1-p)p^i$ , on a :

$$\frac{\prod_{i=0}^{+\infty} x_i^{(1-p)p^i}}{\prod_{i=0}^{+\infty} x_i^{(1-p)p^i} + \prod_{i=0}^{+\infty} (1-x_i)^{(1-p)p^i}} \leq (1-p) \sum_{i=0}^{+\infty} x_i p^i \quad (12)$$

D'autre part si  $\mu$  est une distribution de Poisson( $\lambda$ ) avec le paramètre  $\lambda > 0$ , telle que  $\mu\{i\} = e^{-\lambda} \frac{\lambda^i}{i!}$ , on a :

$$\frac{\prod_{i=0}^{+\infty} x_i e^{-\lambda} \frac{\lambda^i}{i!}}{\prod_{i=0}^{+\infty} x_i e^{-\lambda} \frac{\lambda^i}{i!} + \prod_{i=0}^{+\infty} (1-x_i) e^{-\lambda} \frac{\lambda^i}{i!}} \leq e^{-\lambda} \sum_{i=0}^{+\infty} x_i \frac{\lambda^i}{i!}. \quad (13)$$

Bien sûr, (13) est un cas particulier de (11) quand on prend  $\mu\{i\} = \frac{1}{i}$  si  $i = 1, \dots, n$  et  $\mu\{i\} = 0$  sinon.

Les figures 1 et 2 donnent deux exemples simulés pour illustrer l'inégalité de la généralisation du 3II.

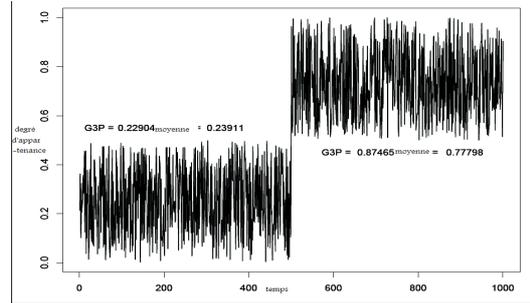


FIG. 1 – Exemple de l'utilisation de la propriété du G3II. Les valeurs de la courbe représentent des degrés d'appartenance à une classe donnée, allant de 0 (non appartenance à cette classe) à 1 (appartenance à cette classe). Cette série temporelle a été simulée uniformément dans l'intervalle de valeurs  $[0; 0.5]$  sur l'intervalle temporel  $[0; 500]$  et uniformément dans l'intervalle de valeurs  $[0.5; 1]$  sur l'intervalle temporel  $[501; 1000]$ . Comme attendu, l'opérateur G3II donne une valeur plus petite (respectivement plus grande) que la moyenne sur l'intervalle temporel  $[0; 500]$  (respectivement sur l'intervalle temporel  $[501; 1000]$ ).

Cette inégalité est certainement la plus prometteuse, car elle correspond à une fusion d'information sur une série temporelle, afin d'avoir un point de vue global sur un intervalle de temps donné. Ainsi, cette inégalité qui peut s'appliquer à des cas discrets, permettra une plus grande discrimination sur un intervalle temporel donné. Comme on l'a vu on peut utiliser cette généralisation d'opérateur sur des phénomènes connus comme par exemple les arrivées de type poissonien.

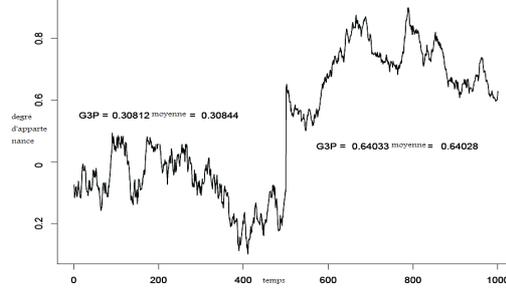


FIG. 2 – Autre exemple d'utilisation de la propriété du  $G3\Pi$ . Ici il s'agit d'une série temporelle simulée par un processus stochastique de type brownien dans un intervalle de valeurs de  $[0; 0.5]$  (respectivement dans un intervalle de valeurs  $[0.5; 1]$ ) sur un intervalle de temps  $[0; 500]$  (respectivement sur un intervalle de temps  $[501; 1000]$ ). Comme attendu, l'opérateur  $G3\Pi$  donne une valeur plus petite (respectivement plus grande) que la moyenne sur l'intervalle temporel  $[0; 500]$  (respectivement sur l'intervalle temporel  $[501; 1000]$ ). Dans ce cas la différence entre le  $G3\Pi$  et la moyenne est très faible.

### 3.5 Le cas continu

#### 3.5.1 Cas général

Soit  $\Omega = R$  l'ensemble des réels et soit  $\mu = g.\lambda$  une distribution de probabilité sur  $\Omega$  avec une densité  $g$  pour la mesure de Lebesgue  $\lambda$ . Alors l'inégalité (9) peut être écrite comme suit :

$$\frac{e^{\int_R \log(f(t))g(t)dt}}{e^{\int_R \log(f(t))g(t)dt} + e^{\int_R \log(1-f(t))g(t)dt}} \leq \int_R f(t)g(t)dt. \quad (14)$$

Par exemple si  $\mu$  est la distribution uniforme sur un intervalle fini  $[a, b]$ ,  $a < b$  dans  $R$ , alors l'inégalité (14) donne :

$$\frac{e^{\frac{1}{b-a} \int_a^b \log(f(t))dt}}{e^{\frac{1}{b-a} \int_a^b \log(f(t))dt} + e^{\frac{1}{b-a} \int_a^b \log(1-f(t))dt}} \leq \frac{1}{b-a} \int_a^b f(t)dt \quad (15)$$

D'autre part, si  $\mu$  est une distribution exponentielle sur  $[0, +\infty)$  avec le paramètre  $\theta > 0$ , on a :

$$\frac{e^{\theta \int_0^{+\infty} \log(f(t))e^{-\theta t} dt}}{e^{\theta \int_0^{+\infty} \log(f(t))e^{-\theta t} dt} + e^{\theta \int_0^{+\infty} \log(1-f(t))e^{-\theta t} dt}} \leq \theta \int_0^{+\infty} f(t)e^{-\theta t} dt. \quad (16)$$

On trouve une inégalité semblable avec une distribution standard comme la distribution de type normale  $(\mu, \sigma^2)$  ou une distribution de type exponentielle.

#### 3.5.2 Fonctions positives

Maintenant, on suppose que  $f > 0$ . Alors, comme on a  $0 < \frac{f}{2(1+f)} \leq \frac{1}{2}$ , l'inégalité (9) implique ce qui suit.

Soit  $f : \Omega \rightarrow (0, +\infty)$  une fonction positive mesurable sur un espace de probabilité alors,

$$\frac{e^{\int_{\Omega} \log(\frac{f}{2(1+f)}) d\mu}}{e^{\int_{\Omega} \log(\frac{f}{2(1+f)}) d\mu} + e^{\int_{\Omega} \log(\frac{2+f}{2(1+f)}) d\mu}} \leq \int_{\Omega} \frac{f}{2(1+f)} d\mu \quad (17)$$

De la même façon, dans l'équation ci-dessus,  $\frac{f}{2(1+f)}$  peut être remplacée par  $\phi(f)$ ; où  $\phi$  est une fonction mesurable projetant  $R_+$  sur l'intervalle  $(0, \frac{1}{2}]$ .

Ce dernier cas (cas continu) montre l'intérêt de cet opérateur et de l'inégalité sous-jacente pour la fusion de plages de données d'une série temporelle, quand bien même celle-ci serait le résultat d'une modélisation mathématique d'un phénomène expérimental.

## 4 Perspectives : applications possibles

Nous venons de présenter dans la section précédente une généralisation de l'opérateur 3II, nommée G3II et nous avons montré certaines propriétés et inégalités. L'exemple suivant est certainement la plus simple application. Supposons que l'on a une série temporelle représentant un phénomène quelconque, et supposons que l'on a deux classes possibles pour chaque élément de la série : la classe numéro 1 représente l'état "danger" et la classe numéro 2 représentant l'état "sans danger". On suppose que la figure3 représente le degré d'appartenance pour la classe numéro 1 (sur un intervalle de temps donné) des éléments de la série temporelle. Comme les degrés d'appartenance à la classe 1 sont supérieur à 0.5 sur l'intervalle de temps [26;50]; il y a danger dans cet intervalle. L'utilisation du G3II va renforcer le résultat "danger" comparativement à une moyenne classique (voir aussi les exemples des figures 1 et 2).

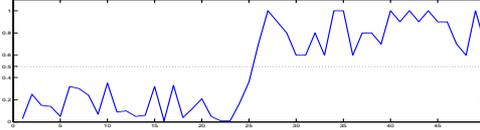


FIG. 3 – Courbe des degrés d'appartenance à la classe 1 (classe "danger") pour les éléments de la série temporelle sur un petit intervalle de temps [1 ; 50]. Les valeurs sont comprises entre 0 (non appartenance à la classe "danger") et 1 (appartenance à la classe "danger"). Sur l'intervalle [1 ; 25] les degrés d'appartenance ont des valeurs faibles (inférieures à 0.5). Cependant sur l'intervalle [26 ; 50], les degrés d'appartenance ont des valeurs supérieures à 0.5 montrant une appartenance forte à la classe "danger". Sur cet intervalle [26 ; 50], l'utilisation du G3II va renforcer l'appartenance à la classe "danger", comparativement à la moyenne géométrique par exemple.

Cependant, les applications potentielles pour cette généralisation restent à découvrir. Nous donnons deux options possibles pour l'utilisation de la généralisation de 3II, en fonction de l'objectif prioritaire que l'on a. Si l'analyse local des signaux est primordiale, alors l'approche suivante (illustrée par la figure 4) peut être appliquée :

## Généralisation d'un opérateur d'agrégation pour des séries temporelles

1. pour chaque élément de temps mesuré et pour chaque variable, on calcule le 3II : le résultat est un signal (à partir de l'agrégation des différentes variables à un instant donné) défini pour chaque instant donné (étape 1).
2. la valeur de la généralisation du 3II, c'est-à-dire le  $G3II$  est alors calculé à partir de ce signal "résultat" (étape 2).

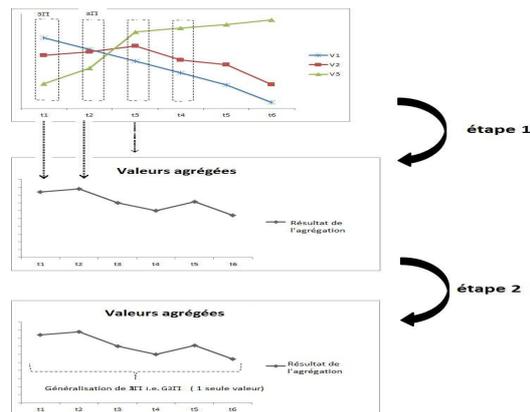


FIG. 4 – Cette approche favorise l'analyse locale : d'abord on applique le 3II, puis la généralisation du 3II, le  $G3II$ .

Si l'analyse global d'un signal est primordiale alors la méthode suivante peut être appliquée (voir figure 5) :

1. Pour chaque variable, on calcule la généralisation, le  $G3II$  : le résultat est une seule valeur par variable. (étape 1).
2. agréger toutes les valeurs avec le 3II (étape 2)

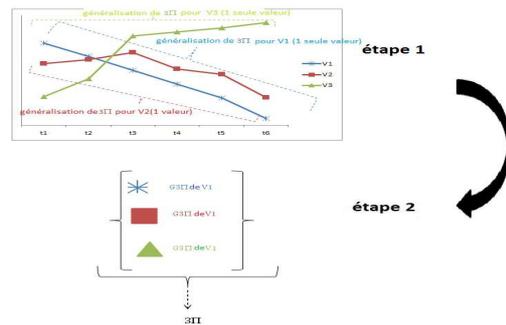


FIG. 5 – Cette approche favorise l'analyse globale : premièrement on applique le  $G3II$  puis le 3II.

Le choix d'une approche parmi les deux applications dépend de l'application et de son cadre. Quelque soit la méthode choisie, les deux approches sont configurées pour agréger des séries temporelles. Evidemment, pour des applications expérimentales, le cas discret semble être le plus adapté ; cependant le choix adéquate et l'utilisation d'une fonction  $g$  (voir équation 14) permettrait également l'utilisation du cas continu pour des applications expérimentales. Une fonction de pondération serait l'option la plus naturelle et semble être la perspective la plus évidente à creuser. Quoiqu'il en soit de nombreuses applications semblent possibles : analyse médicale de signaux physiologiques, analyse fonctionnelle, etc.

## 5 Conclusion

Dans cette étude nous avons mis en évidence une généralisation de l'opérateur  $3\Pi$  appelée  $G3\Pi$ . Cette généralisation permet d'agréger des informations à partir d'une série temporelle sur un intervalle temporel donné, et certaines propriétés-des inégalités- ont été démontrées. Ces inégalités montrent que pour des séries temporelles, la généralisation  $G3\Pi$  est plus discriminante qu'une moyenne classique. Agréger des informations d'une série temporelle sur un intervalle donné, permettra d'avoir une vision plus globale plutôt que d'avoir une vision locale du problème. Cette généralisation permettra le management d'un plus grand volume de données. Une meilleure compréhension, et donc, une meilleure prise de décision pourrait résulter de l'utilisation de la généralisation  $G3\Pi$ . A plus long terme, on envisage une approche d'agrégation multi-échelle : on part d'un point (avec le  $3\Pi$  ou la  $M3\Pi$ ) pour aller vers le plus global (en utilisant le  $G3\Pi$ ). A court terme, les prochains travaux concerneront l'application directe de cette généralisation sur des séries temporelles.

## Références

- Beliakov, G., T. Calvo, et A. Pradera (2007). Handling of neutral information by aggregation information. *Fuzzy sets and systems* 158, 861–880.
- Bloch, I. et A. Hunter (Eds 2001). Fusion : General concepts and characteristics. *International Journal of Intelligent Systems* 16, 1107–1134.
- Calvo, T., B. D. Baets, et J. Fodor (2001). The functional equations of franck and alsina for uninorms and nullnorms. *Fuzzy Sets and Systems* 120, 385–394.
- Doncescu, A., S. Régis, K. Inoue, et R. Emilion (2007). Analysis of new aggregation operators : Mean  $\hat{3}\tilde{a}$ . *J. of Advanced Computational Intelligence and Intelligent Informatics* 11(6).
- Dubois, D. et H. Prade (2004). On the use of aggregation operations in information fusion process. *Fuzzy Sets and Systems* 142, 143–161.
- Elkan, C. (1994). The paradoxical success of fuzzy logic. *IEEE Expert*, 3–8.
- Emilion, R., S. Régis, A. Ménil, et A. Doncescu (2004). Un nouvel opérateur de type moyenne pour la fusion de données. In *LFA2004*, Nantes.
- Jensen, J. (1906). Sur les fonctions convexes et les inégalités entre les valeurs moyennes. *Acta Mathematica* 30(1), 175–193.

- Jilani, T. A. et S. M. A. Burney (2008). Multivariate stochastic fuzzy forecasting models. *Expert Systems with Applications* 35, 691–700.
- Komornik, J., M. Komornikova, R. Mesiar, D. Szokeova, et J. Szolgay (2006). Comparison of forecasting performance of nonlinear models of hydrological time series. *Physics and Chemistry of the Earth* 31, 1127–1145.
- Martin, A. (2005). La fusion d'informations. Polycopié de cours ENSIETA 1484, Max Planck Institut für Informatik.
- Nelsen, R. (1999). *An introduction to Copulas*, Volume 139 of *Lecture Notes in Statistics*. New York : Springer.
- Piera-Carreté, N., J. Aguilar-Martin, et M. Sanchez (1988). Mixed connectives between min and max. In *8th Inter. Symp. on Multiple Valued Logic*.
- Pradera, A. et E. Trillas (2006). Aggregation, non-contradiction and excluded-middle. *Mathware and Soft Computing* 13, 189–201.
- Silvert, W. (1979). Symmetric summation : A class of operations on fuzzy sets. *IEEE Transactions on Systems, Man and Cybernetics* 9(10), 657–659.
- Waissman-Vilanova, J. (2000). *Construction d'un modèle comportemental pour la supervision de procédés : application à une station de traitement des eaux*. Thèse de Doctorat, LAAS - CNRS.
- Yager, R. (1988). On ordered weighted averaging operators in multi-criteria decision making. *IEEE Transaction on Systems, Man, and Cybernetics*, 183–190.
- Yager, R. (1994). Misrepresentations and challenges : A response to elkan. *IEEE Expert*, 41–42.
- Yager, R. (2002). Defending against strategic manipulation in uninorm-based multi-agent decision making. *European Journal of Operational Research* 141, 217–232.
- Yager, R. (2004). Owa aggregation over a continuous interval argument with applications to decision making. *IEEE Transactions on Systems, Man and Cybernetics-Part B : Cybernetics* 34(5), 1952–1963.
- Yager, R. (2008). Time series smoothing and owa aggregation. *IEEE Transactions on Fuzzy Systems* 16(4), 994–1007.
- Yager, R. et A. Rybalov (1996). Uninorm aggregation operators. *Fuzzy Sets and Systems* 80, 111–120.
- Yager, R. et A. Rybalov (1998). Full reinforcement operators in aggregation techniques. *IEEE Transactions on Systems, Man and Cybernetics-Part B : Cybernetics* 28(6).
- Zimmerman, H. et P. Zynso (1980). Latent connectives in human decision making. *Fuzzy Sets and Systems* 4, 37–51.

## Summary

In this article, we present a generalization of the 3II aggregation operator. This generalization is called  $G3II$ , and can be used to aggregate values of a time series on an interval. Some inequalities of this generalization are presented and demonstrated.

# Mesure de dépendance positive et négative de sources crédibilistes

Mouna Chebbah\*,\*\*, Arnaud Martin\*\*  
Boutheina Ben Yaghlane\*\*\*

\*LARODEC, ISG Tunis, 41 Rue de la Liberté, Cité Bouchoucha  
2000 Le Bardo, Tunisie  
Mouna.Chebbah@gnet.tn

\*\*UMR 6074 IRISA, Université de Rennes1 / IUT de Lannion, Rue Edouard Branly  
BP 3021, 22302 Lannion cedex  
Arnaud.Martin@univ-rennes1.fr

\*\*\*LARODEC, IHEC Carthage, Carthage Présidence 2016,  
Tunisie  
boutheina.yaghlane@ihec.rnu.tn

**Résumé.** La théorie des fonctions de croyance permet une bonne gestion et modélisation de données incertaines et imprécises. En particulier, la théorie des fonctions de croyance permet une fusion d'informations issues de différentes sources imparfaites. Afin de combiner au mieux les informations, il est important de savoir si les sources sont indépendantes ou dépendantes.

Dans cet article nous proposons d'étudier la corrélation de deux sources afin de déduire leurs éventuelles indépendances et ainsi choisir le type de règle de combinaison à utiliser. Nous définissons alors une mesure d'indépendance, dépendance positive et dépendance négative.

Les mesures proposées sont testées sur des données générées aléatoirement pour les différents cas de dépendance.

## 1 Introduction

Plusieurs théories de l'incertain telles que *la théorie des probabilités*, *la théorie des possibilités* et *la théorie des sous-ensembles flous* permettent de traiter et de modéliser les données imparfaites. *La théorie des fonctions de croyance* initialement proposée par Dempster (1967) et Shafer (1976) permet également une bonne modélisation des données imprécises et/ou incertaines. En plus de la modélisation, la combinaison des données incertaines est possible avec la théorie des fonctions de croyance. En effet, plusieurs règles de combinaison ont été proposées dans le cadre de la théorie des fonctions de croyance. Ces règles sont utilisées pour combiner plusieurs données imparfaites fournies par des sources différentes. La combinaison d'informations imparfaites permet de prendre des décisions plus fiables et plus précises.

Dans le cadre de la théorie des fonctions de croyance, plusieurs règles de combinaison ont été proposées. Certaines règles, comme celle de Dempster (1967), Smets (1990), Yager (1987),

Dubois et Prade (1988) ... combinent des fonctions de croyance dont les sources sont différentes et indépendantes. La dépendance cognitive est une hypothèse fondamentale pour ces règles de combinaison alors que la règle prudente de Dencœur (2006) est appliquée quand les sources des fonctions de croyance sont cognitivement dépendantes. Deux sources<sup>1</sup> sont cognitivement indépendantes quand la connaissance de la croyance de l'une n'affecte pas celle de l'autre. Autrement dit, une source est cognitivement indépendante d'une autre source lorsque sa croyance ne change pas avec le changement de la croyance de l'autre source. Les travaux de Smets (1993) et ceux de Shafer (1976) ont défini la dépendance cognitive des informations évidentielles.

La dépendance des sources est une information primordiale pour le choix de la règle de combinaison à utiliser lors de la combinaison des informations crédibilistes qu'elles fournissent. Certains travaux comme ceux de Ben Yaghlane et al. (2002a,b) étudient principalement la dépendance statistique des variables. Dans ce papier nous proposons une méthode d'estimation de la dépendance cognitive de deux sources. Nous proposons une méthode permettant d'étudier le comportement général de deux sources et de les comparer pour déceler toute dépendance pouvant exister entre ces deux sources. Dans le cas de sources dépendantes, nous proposons d'étudier le type de cette dépendance. C'est-à-dire analyser les données de sorte à voir si les sources sont dépendantes en disant la même chose (*dépendance positive*) ou si l'une dit toujours le contraire de l'autre (*dépendance négative*).

Dans la suite de cet article, nous commençons par rappeler quelques notions de base de la théorie des fonctions de croyance. Ensuite, nous présentons, dans la troisième section, notre méthode d'estimation de la dépendance cognitive. Lors de cette méthode, nous proposons d'appliquer un *algorithme de classification non supervisée* sur toutes les informations incertaines de chaque source. Un appariement des clusters est recherché après avoir appliqué l'algorithme de classification. La dépendance des sources est estimée alors à partir des poids attribués à chaque couple de clusters liés. Dans la quatrième section, dans le cas de sources dépendantes, une étude sur la nature de la dépendance est faite afin de voir si cette dépendance est positive ou négative. Pour finir, nous présentons quelques résultats de tests sur des données générées aléatoirement.

## 2 La théorie des fonctions de croyance

Soit  $\Omega = \{\omega_1, \omega_2, \dots, \omega_n\}$  l'ensemble de toutes les solutions possibles à un problème donné. Les  $\omega_i$  sont toutes les hypothèses exclusives et exhaustives. L'ensemble  $\Omega$  représente l'ensemble de discernement ou l'univers de discours du problème en question.

L'ensemble  $2^\Omega = \{A | A \subseteq \Omega\} = \{\emptyset, \omega_1, \omega_2, \dots, \omega_n, \omega_1 \cup \omega_2, \dots, \Omega\}$ , est un ensemble composé de tous les sous-ensembles possibles de  $\Omega$ .  $2^\Omega$  contient toutes les hypothèses de  $\Omega$  ainsi que leurs unions.

Une *fonction de masse* est une fonction de  $2^\Omega$  vers l'intervalle  $[0, 1]$  qui affecte à chaque sous-ensemble une valeur positive inférieure ou égale à 1 représentant sa *masse de croyance élémentaire*. Cette fonction de masse est une représentation des connaissances incertaines et imprécises fournies par une source (un expert, un classifieur, ...). Formellement, une fonction

---

1. Toute source d'information comme par exemple un expert humain, un classifieur, un capteur, ... etc.

de masse  $m^\Omega$  est définie comme suit :

$$m^\Omega : 2^\Omega \mapsto [0, 1] \quad (1)$$

tel que :

$$\sum_{A \subseteq \Omega} m^\Omega(A) = 1 \quad (2)$$

Un sous-ensemble ayant une masse de croyance non-nulle est un *élément focal*. La masse affectée à un élément focal  $A$  représente le degré de croyance élémentaire sur l'hypothèse  $A$ .

D'autres fonctions de croyances sont dérivées de la fonction de masse. La *fonction de croyance* (ou de *crédibilité*)  $bel^\Omega$  représente le degré de croyance minimale affecté à un sous-ensemble de  $2^\Omega$  justifié par les informations disponibles.  $bel^\Omega(A)$  est la somme de toutes les masses des éléments focaux  $B$  qui confortent l'hypothèse  $A$  ( $B \subseteq A$ ).

$$\begin{aligned} bel^\Omega : 2^\Omega &\rightarrow [0, 1] \\ A &\mapsto \sum_{B \subseteq A, B \neq \emptyset} m^\Omega(B) \end{aligned} \quad (3)$$

Pour des raisons calculatoires, nous considérons aussi la *fonction d'implicabilité*  $b^\Omega$  :

$$\begin{aligned} b^\Omega : 2^\Omega &\rightarrow [0, 1] \\ A &\mapsto \sum_{B \subseteq A} m^\Omega(B) = bel^\Omega(A) + m^\Omega(\emptyset) \end{aligned} \quad (4)$$

Dans le cadre de la théorie des fonctions de croyance, il existe un grand nombre de règles de combinaison. Ces règles sont utilisées pour la combinaison d'information, représentées par des fonctions de masse. Elles sont issues de différentes sources et sont définies sur le même ensemble de discernement. La combinaison permet de synthétiser ces différentes informations (avis d'experts, ...) afin d'en obtenir une seule en vue d'une prise de décision. Dans cet article, nous utilisons la moyenne comme règle de combinaison. Bien que cette règle soit idempotente et ne génère pas de conflit, d'autres règles de combinaison vérifiant ces critères peuvent être utilisées. Pour chaque élément focal  $A$  des  $M$  fonctions de masse, la masse combinée de  $A$ ,  $m^\Omega(A)$  est calculée à partir des  $M$  masses de croyance élémentaires  $m_i^\Omega(A)$  comme suit :

$$m_{Moyenne}^\Omega(A) = \frac{1}{M} \sum_{i=1}^M m_i^\Omega(A) \quad (5)$$

Il existe d'autres règles de combinaison, telle que la règle de Dempster (1967), Yager (1987), Martin et Osswald (2007), ...

La transformation pignistique est utilisée pour calculer les probabilités pignistiques à partir des fonctions de masse afin de pouvoir prendre une décision. Si un expert fournit une fonction de masse reflétant son avis sur la solution d'un problème précis, les probabilités pignistiques reflètent la solution la plus probable. La probabilité pignistique  $BetP$  d'un singleton  $A$  est calculée comme suit :

$$BetP(A) = \sum_{C \subseteq 2^\Omega, C \neq \emptyset} \frac{|A \cap C|}{|C|} \frac{m(C)}{1 - m(\emptyset)}. \quad (6)$$

Après l'acquisition d'une fonction de masse, une information certaine peut apparaître confirmant que l'hypothèse, solution du problème en question, est (ou n'est pas) dans l'un des sous ensemble de  $2^\Omega$ . Dans ce cas, la fonction de masse doit être mise à jour afin de prendre en considération cette nouvelle information certaine. Cette mise à jour est réalisée par l'opérateur de conditionnement qui consiste à affecter la masse attribuée à chaque élément focal à son union avec l'ensemble certain. Le conditionnement d'une fonction de masse  $m^\Omega$  par l'hypothèse  $A \subseteq \Omega$  revient à transférer les masses de croyance de tous les éléments focaux de  $m^\Omega$  à leur union avec  $A$ . La fonction de masse conditionnée  $m_{[A]}^\Omega$  est donnée par Smets et Kruse (1997) comme suit :

$$m_{[A]}^\Omega(C) = \begin{cases} 0 & \text{for } C \not\subseteq A \\ \sum_{B \subseteq A^c} m^\Omega(C \cup B) & \text{for } C \subseteq A \end{cases} \quad (7)$$

avec  $A^c$  est l'ensemble complémentaire de  $A$  ( $A^c = \Omega \setminus A$ ).

Le *Théorème de Bayes Généralisé (TBG)* a été proposé par Smets (1993) pour généraliser le théorème de Bayes en remplaçant les probabilités conditionnelles par des fonctions de masse jointes. Soit deux *variables dépendantes*  $X$  et  $Y$  définies sur deux cadres de discernement différents  $\Omega_X$  et  $\Omega_Y$ . Soit la fonction  $m_{[X]}(Y)$ , la fonction de masse conditionnelle de  $Y$  sachant  $X$ . Le théorème de bayes généralisé sera utilisé pour calculer la fonction de masse conditionnelle de  $X$  sachant  $Y$  comme suit :

$$\begin{aligned} bel_{[Y]}(X) &= b_{[Y]}(X) - b_{[Y]}(\emptyset) \\ bel_{[Y]}(X) &= \prod_{x_i \in \bar{X}} b_{[x_i]}(\bar{Y}) \end{aligned} \quad (8)$$

La fonction de crédibilité conditionnelle  $bel_{[X]}(Y)$  peut être étendue au cadre de discernement joint  $\Omega_X \times \Omega_Y$ , puis conditionné par  $y_i \in \Omega_Y$ , le résultat sera marginalisé par  $X$  pour obtenir la règle disjonctive de combinaison :

$$\begin{aligned} bel_{[X]}(Y) &= b_{[X]}(Y) - b_{[X]}(\emptyset) \\ bel_{[X]}(Y) &= \prod_{x_i \in X} b_{[x_i]}(Y) \end{aligned} \quad (9)$$

### 3 Indépendance

L'indépendance a été introduite en premier dans le cadre de la théorie des probabilités pour modéliser la dépendance statistique des variables. Avec les probabilités, deux variables  $A$  et  $B$  sont indépendantes si  $P(A \cap B) = P(A) \times P(B)$  ou encore si  $P(A|B) = P(A)$ . La dépendance statistique a été généralisée de la théorie des probabilités à la théorie des fonctions de croyance par BenYaghlane et al. (2002a) et BenYaghlane et al. (2002b).

Les fonctions de masse peuvent être considérées comme des probabilités subjectives fournies par des sources (ou experts), sauf que ces sources peuvent communiquer entre elles. Dans le cas d'une possible communication entre les sources, elles sont *cognitivement dépendantes*.

Shafer (1976) définit la dépendance cognitive de deux cadres de discernement par rapport à une croyance élémentaire comme étant le changement des croyances sur l'un si une nouvelle croyance élémentaire apparaît sur l'autre.

Smets (1993) définit la dépendance cognitive de deux variables comme étant le changement de la croyance sur l'une des variables si la croyance sur l'autre change.

Dans ce papier, nous nous intéressons à la dépendance des sources et non pas à la dépendance des variables.

**Définition 1.** Deux sources sont dépendantes s'il existe un quelconque lien entre toutes les fonctions de masse qu'elles fournissent.

Nous définissons la dépendance cognitive comme étant la ressemblance du comportement général de deux sources. Afin d'étudier la dépendance de deux sources, nous proposons d'étudier leur comportement général. Nous proposons, alors, de voir si les deux sources ont tendance à donner des réponses similaires ou plutôt des réponses différentes. Nous proposons aussi d'étudier la ressemblance des éléments focaux. Le but étant de choisir la règle de combinaison à utiliser pour combiner leurs fonctions de croyances. Nous introduisons d'abord la mesure d'indépendance de deux sources  $S_1$  et  $S_2$  notée  $I_d(S_1, S_2)$  comme étant l'indépendance de  $S_1$  à  $S_2$ . Cette mesure vérifie les axiomes ci-dessous :

1. Non- négative : L'indépendance d'une source  $S_1$  par rapport à une source  $S_2$ ,  $I_d(S_1, S_2)$  est une valeur qui est, soit nulle si  $S_1$  est complètement dépendante de  $S_2$ , soit strictement positive.
2. Normalisée :  $I_d(S_1, S_2) \in [0, 1]$ , si  $I_d$  est nulle alors  $S_1$  est complètement dépendante de  $S_2$ . Si  $I_d = 1$ , alors  $S_1$  est complètement indépendante de  $S_2$  autrement c'est un degré dans  $]0, 1[$ .
3. Non-symétrique : Si  $S_1$  est indépendante de  $S_2$ , cela n'implique pas que  $S_2$  est indépendante de  $S_1$ .  $S_1$  et  $S_2$  peuvent être simultanément indépendantes avec des degrés d'indépendance différents.
4. Identité :  $I_d(S_1, S_1) = 0$ , toute source est complètement dépendante d'elle même.

Si deux sources  $S_1$  et  $S_2$  sont dépendantes alors il existe sûrement un lien direct ou indirect entre leurs fonctions de masse. L'idée proposée dans ce papier étant de classer toutes les fonctions de masse des deux sources et de comparer les clusters. Le clustering est utilisé pour regrouper les cas où la source s'exprime d'une façon similaire. Si les clusters sont similaires, c'est à dire que les sources ont un comportement similaire dans des cas similaires, alors il est fort probable qu'elles soient dépendantes. Si les clusters sont fortement liés, alors les sources sont dépendantes. Dans le cas de sources dépendantes, si les clusters liés sont proches, alors les sources sont positivement dépendantes sinon elles sont négativement dépendantes.

### 3.1 Classification non-supervisée

Dans cet article, nous proposons d'utiliser un algorithme de classification non-supervisée de type  $C$ -moyenne, utilisant une distance sur les fonctions de masse définie par Jusselme et al. (2001) comme proposé par BenHariz et al. (2006), Chebbah et al. (2012a) et Chebbah et al. (2012b). Soit un ensemble  $T$  contenant  $n$  objets  $o_i : 1 \leq i \leq n$  à classer dans  $C$  clusters. Les valeurs des  $o_i$  sont des fonctions de masse  $m_i^\Omega$  définies sur le cadre de discernement  $\Omega$ . Le but est de classer les  $n$  fonctions de masse valeurs des objets  $o_i$ . Les fonctions de masse  $m_i^\Omega$  sont fournies par la même source, c'est à dire qu'une même source a attribué les valeurs des objets  $o_i$ . Appliquer un algorithme de classification non-supervisée sur ces fonctions de masse revient à étudier le comportement général de la source c'est à dire voir les cas

où elle s'exprime de la même façon. Une mesure de similarité  $D(o_i, Cl_k)$  permet de mesurer la dissimilarité d'un objet  $o_i$  et un cluster  $Cl_k$  comme suit :

$$D(o_i, Cl_k) = \frac{1}{n_k} \sum_{j=1}^{n_k} d(m_i^\Omega, m_j^\Omega) \quad (10)$$

$$d(m_1^\Omega, m_2^\Omega) = \sqrt{\frac{1}{2}(m_1^\Omega - m_2^\Omega)^t \underline{D}(m_1^\Omega - m_2^\Omega), \underline{D}(A, B) = \begin{cases} 1 & \text{si } A = B = \emptyset \\ \frac{|A \cap B|}{|A \cup B|} & \forall A, B \in 2^\Omega \end{cases}} \quad (11)$$

La dissimilarité d'un objet  $o_i$  et un cluster  $Cl_k$  est définie par la moyenne des distances entre la fonction de masse  $m_i^\Omega$  valeur de cet objet et toutes les  $n_k$  fonctions de masse valeurs des  $o_j : 1 \leq j \leq n_k$  objets contenus dans le cluster  $Cl_k$ . Chaque objet est affecté au cluster qui lui est le plus similaire (ayant une valeur de dissimilarité minimale) de manière itérative jusqu'à ce qu'une répartition stable soit obtenue. À la fin de la classification non-supervisée,  $C$  clusters contenant chacun un certain nombre d'objets sont obtenus. Dans cet article, nous supposons que le nombre de clusters  $C$  est égal à la cardinalité de l'ensemble de discernement ( $C = |\Omega|$ ) puisque  $\Omega$  représente les classes possibles dans un problème de classification. Il existe des approches permettant de définir automatiquement le nombre de clusters qui peuvent être appliquées.

### 3.2 Appariement des clusters

L'algorithme de classification non-supervisée est utilisé pour la classification des fonctions de masse fournies par une même source. Dans de nombreuses applications, plusieurs sources s'expriment sur la même problématique et fournissent différentes fonctions de masse comme valeurs aux mêmes objets. L'algorithme de classification non-supervisée est appliqué aux fonctions de masse de chaque source séparément et puis ces clusters doivent être comparés dans le but de voir s'il y a un lien entre ces clusters. Plus les liens entre ces clusters sont forts plus les sources ont tendance à être dépendantes du moment qu'elles disent la même chose dans des cas similaires. Supposons que nous avons deux sources  $S_1$  et  $S_2$ , chaque source fournit  $n$  fonctions de masse pour les mêmes objets. Ceci exprime le fait que la fonction de masse  $m_i^\Omega$  fournie par  $S_1$  et celle fournie par  $S_2$  se réfèrent au même objet  $o_i$ . Le nombre de clusters étant la cardinalité du cadre de discernement des fonctions de masse  $m_i^\Omega$  fournies par  $S_1$  et  $S_2$ . Après avoir classifié les fonctions de masse de  $S_1$  et  $S_2$ , la matrice de correspondance des clusters  $M$  est obtenue par :

$$M_1 = \begin{pmatrix} \alpha_{11}^1 & \alpha_{12}^1 & \dots & \alpha_{1C}^1 \\ \dots & \dots & \dots & \dots \\ \alpha_{k1}^1 & \alpha_{k2}^1 & \dots & \alpha_{kC}^1 \\ \dots & \dots & \dots & \dots \\ \alpha_{C1}^1 & \alpha_{C2}^1 & \dots & \alpha_{CC}^1 \end{pmatrix} \quad \text{and} \quad M_2 = \begin{pmatrix} \alpha_{11}^2 & \alpha_{12}^2 & \dots & \alpha_{1C}^2 \\ \dots & \dots & \dots & \dots \\ \alpha_{k1}^2 & \alpha_{k2}^2 & \dots & \alpha_{kC}^2 \\ \dots & \dots & \dots & \dots \\ \alpha_{C1}^2 & \alpha_{C2}^2 & \dots & \alpha_{CC}^2 \end{pmatrix} \quad (12)$$

Avec

$$\alpha_{k_1, k_2}^i = \frac{|Cl_{k_1}^i \cap Cl_{k_2}^j|}{|Cl_{k_1}^i|} \quad (13)$$

Notons  $\alpha_{k_1, k_2}^1$  est la similarité des clusters  $Cl_{k_1}^1$  de  $S_1$  et  $Cl_{k_2}^2$  de  $S_2$  par rapport à  $S_1$ . Cette similarité est le rapport entre la cardinalité de l'intersection des deux clusters  $Cl_{k_1}^1$  et  $Cl_{k_2}^2$  et la cardinalité du cluster  $Cl_{k_1}^1$ . Autrement dit, la similarité de deux clusters est la proportion des objets classés de la même manière par rapport à la cardinalité du cluster de la source référente. Si par exemple, le cluster  $Cl_1^1$  de  $S_1$  contient les 5 objets  $\{o_1, o_5, o_9, o_{12}, o_{15}\}$  et le cluster  $Cl_5^2$  de  $S_2$  contient les 3 objets  $\{o_1, o_2, o_9\}$ .  $\alpha_{1,5}^1$  est la similarité de  $Cl_5^2$  de  $S_2$  par rapport à  $Cl_1^1$  de  $S_1$  représentant la proportion des objets de  $Cl_5^2$  qui sont aussi classés dans  $Cl_1^1$  de  $S_1$  donc  $\alpha_{1,5}^1 = \frac{2}{3}$  et  $\alpha_{1,5}^2 = \frac{2}{5}$ . Notons que le clustering est appliqué sur les mêmes objets sauf que les valeurs de ces objets sont différentes puisqu'elles sont fournies par deux sources différentes  $S_1$  et  $S_2$ . Notons alors que les matrices  $M_1$  et  $M_2$  sont différentes puisque les clusters de  $S_1$  et ceux de  $S_2$  sont différents. Notons également que d'autres coefficients de similarité peuvent aussi être utilisés.

Une fois les deux matrices de correspondances  $M_1$  et  $M_2$  calculées, une correspondance entre les clusters peut être établie. Chaque cluster est lié au cluster qui lui est le plus similaire en vérifiant que chaque deux clusters de la même source ne peuvent pas être liés au même cluster de l'autre source. Par exemple, nous ne pouvons pas avoir les deux clusters  $Cl_1^1$  et  $Cl_3^1$  de  $S_1$  liés au même cluster  $Cl_9^2$  de  $S_2$ . Pour déterminer la correspondance des clusters pour chaque source, les deux clusters ayant la plus grande similarité sont reliés d'une façon unique. La recherche des correspondances des clusters est faite pour les deux sources. Deux différentes correspondances peuvent être obtenues pour les deux sources.

### 3.3 Indépendance des clusters

Une fois la correspondance des clusters établie une fonction de masse définissant l'indépendance de chaque couple de cluster est construite. Supposons que deux sources  $S_1$  et  $S_2$  fournissent  $n$  fonctions de masse comme valeurs aux  $n$  objets  $o_i$ . Les fonctions de masse des deux sources sont classifiées chacune à part en utilisant l'algorithme de classification détaillé ci-dessus.  $C$  clusters sont le résultat de classification des fonctions de masse de  $S_1$  et  $C$  clusters différents sont obtenus pour  $S_2$ . Après appariement de clusters, les clusters de  $S_1$  sont liés aux clusters de  $S_2$  qui leur sont similaires et ceux de  $S_2$  sont également liés aux clusters de  $S_1$  les plus similaires. Différents appariements sont obtenus pour  $S_1$  et  $S_2$ . Dans cette section, nous définissons l'indépendance de chaque couple de clusters liés  $(k_1, k_2)$  comme une fonction de masse définie sur le cadre de discernement  $\Omega_I = \{Dependant Dep, Independant Ind\}$  :

$$\begin{cases} m_{k_1, k_2}^{\Omega_I}(Dep) = \beta \alpha_{k_1 k_2}^1 \\ m_{k_1, k_2}^{\Omega_I}(Ind) = \beta (1 - \alpha_{k_1 k_2}^1) \\ m_{k_1, k_2}^{\Omega_I}(Dep \cup Ind) = 1 - \beta \end{cases} \quad (14)$$

Le coefficient  $\beta$  est un facteur d'affaiblissement utilisé pour prendre en compte le nombre d'objets contenus dans chaque clusters. Si deux clusters contenant très peu de fonctions de masse sont reliés et que deux autres clusters contenant beaucoup plus d'objets sont aussi reliés avec un même degré de similarité, les fonctions de masse des deux couples de clusters ne doivent pas avoir un même poids. Une fonction de masse est définie pour chaque couple de cluster relié pour chacune des sources. Pour avoir une information sur la dépendance globale

de chaque source, toutes ces fonctions de masse sont combinées avec la moyenne. C'est-à-dire que les  $C$  clusters de  $S_1$  sont reliés aux  $C$  clusters de  $S_2$ , une fonction de masse est ainsi obtenue pour chaque couple de clusters reliés afin de refléter son degré d'indépendance.  $C$  fonctions de masse sont alors obtenues pour chaque source. La combinaison de ces  $C$  fonctions de masse est une fonction de masse décrivant l'indépendance totale de la source  $S_1$  par rapport à  $S_2$ . Les probabilités pignistiques calculées à partir de la fonction de masse combinée permettent la prise de décision sur la dépendance des sources. l'indépendance de la source  $S_1$  de la source  $S_2$ ,  $I_d(S_1, S_2)$  n'est autre que la probabilité pignistique de  $Ind$ ,  $I_d(S_1, S_2) = Betp(Ind)$  et  $\bar{I}_d(S_1, S_2) = Betp(Dep)$ . Si  $I_d(S_1, S_2) < \bar{I}_d(S_1, S_2)$ , alors  $S_1$  est dépendante de  $S_2$ , dans le cas échéant  $S_1$  est indépendante de  $S_2$ .  $I_d$  est non symétrique c'est-à-dire  $I_d(S_1, S_2) \neq I_d(S_2, S_1)$  puisque la correspondance des clusters est différente pour  $S_1$  et  $S_2$ , ainsi les fonctions de masse d'indépendance des clusters liés sont aussi différentes. Cette propriété permet par exemple de mettre en évidence une indépendance de  $S_1$  par rapport à  $S_2$  et une dépendance de  $S_2$  par rapport à  $S_1$ .

## 4 Dépendance positive ou négative

La mesure  $I_d(S_1, S_2)$  informe sur l'indépendance ou *a contrario* la dépendance de la source  $S_1$  par rapport à la source  $S_2$  permettant par exemple de choisir la règle de combinaison à utiliser. Si les sources  $S_1$  et  $S_2$  sont dépendantes, il est alors préférable d'utiliser la règle de Denœux (2006) sinon utiliser les règles de Dubois et Prade (1988); Martin et Osswald (2007); Murphy (2000); Smets et Kennes (1994); Yager (1987). Quant au moins  $I_d(S_1, S_2) < \bar{I}_d(S_1, S_2)$  ou  $I_d(S_2, S_1) < \bar{I}_d(S_2, S_1)$ , l'une des sources est dépendante de l'autre. Dans le cas de sources dépendantes  $I_d$  n'est pas suffisante pour indiquer le type de dépendance.

Deux sources dépendantes disent soit la même chose soit des choses contradictoires. Si  $S_1$  est dépendante de  $S_2$ , elle peut soit l'imiter en disant exactement la même chose et dans ce cas là les fonctions de masse qu'elle fournit sont similaires à celles de l'autre source ce qui donne des clusters similaires. Elle peut aussi dire le contraire et être dépendante. En effet à chaque fois que la source s'exprime sur un objet l'autre dit totalement de contraire, dans ce cas là les fonctions de masse ne se ressemblent pas du tout mais la classification non-supervisée donne des clusters similaires. Les clusters liés seront très similaires mais ils seront distants.

Si par exemple  $I_d(S_1, S_2) < \bar{I}_d(S_1, S_2)$ , alors  $S_1$  est dépendante de  $S_2$  ce qui signifie que les clusters de  $S_1$  ressemblent aux clusters de  $S_2$ . Une étude de la distance entre les deux clusters de  $S_1$  et  $S_2$  peut nous renseigner sur cette dépendance que nous qualifions de positive ou négative. Si les clusters liés sont proches en terme de distance alors  $S_1$  est positivement dépendante de  $S_2$  sinon elle est négativement dépendante. Nous définissons alors la distance entre les deux clusters dépendants  $Cl_{k_1}^1$  et  $Cl_{k_2}^2$  comme étant la moyenne des distances entre les fonctions de masse des objets en commun :

$$Dist(Cl_{k_1}^1, Cl_{k_2}^2) = \frac{1}{|Cl_{k_1}^1 \cap Cl_{k_2}^2|} \sum_{j \in E(Cl_{k_1}^1, Cl_{k_2}^2)} d(m_j^{\Omega,1}, m_j^{\Omega,2}) \quad (15)$$

avec

$$E(Cl_{k_1}^1, Cl_{k_2}^2) = \{j \in [1, n], m_j^{\Omega,1} \in Cl_{k_1}^1 \text{ et } m_j^{\Omega,2} \in Cl_{k_2}^2\} \quad (16)$$

Une distance est associée à chaque couple de cluster lié. Alors si  $Cl_1^1$  de  $S_1$  est lié à  $Cl_5^2$  de  $S_2$  et que  $S_1$  est dépendante de  $S_2$  alors la distance entre  $Cl_1^1$  et  $Cl_5^2$  est la moyenne des distances des fonctions de masse relatives aux objets qui sont à la fois dans  $Cl_1^1$  et  $Cl_5^2$ . Une fonction de masse définie sur le cadre de discernement  $\Omega_P = \{\text{dépendance positive } P, \text{ dépendance négative } \bar{P}\}$  décrivant cette dépendance est obtenue pour chaque couple de cluster :

$$\begin{cases} m_{k_1, k_2}^{\Omega_P}(P|Dep) = 1 - Dist(Cl_{k_1}^i, Cl_{k_2}^j) \\ m_{k_1, k_2}^{\Omega_P}(\bar{P}|Dep) = Dist(Cl_{k_1}^i, Cl_{k_2}^j) \\ m_{k_1, k_2}^{\Omega_P}(P \cup \bar{P}|Dep) = 0 \end{cases} \quad (17)$$

Notons que ces fonctions de masse sont des fonctions de masse conditionnelles puisque la dépendance positive ou négative ne sera calculée que si les sources sont dépendantes. Pour avoir les fonctions de masse marginales, la règle disjonctive de Smets (1993) est utilisée. En combinant les fonctions de masse définies sur  $\Omega_I$  avec celles définies sur  $\Omega_P$  une fonction de masse marginale décrivant la dépendance positive et négative de chaque couple de cluster est alors obtenue. Les fonctions de masse marginales sont combinées avec la moyenne pour pouvoir conclure sur la dépendance positive ou négative de la source par rapport à l'autre, d'une façon générale.

Notons aussi qu'une source est complètement dépendante d'elle-même puisqu'en appliquant l'algorithme de classification nous obtiendrons exactement la même répartition de classe ce qui impliquera  $I_d(S, S) = 0$ .

## 5 Expérimentations

Pour tester la méthode précédemment décrite, nous avons généré des fonctions de masse aléatoirement. L'algorithme 1 est utilisé pour générer  $n$  fonctions de masse.

---

**Algorithm 1** Générer  $n$  fonction de masse

---

**Require:**  $|\Omega|, n$  : nombre des fonctions de masse à générer

- 1: **for**  $i = 1$  to  $n$  **do**
  - 2: Tirer aléatoirement  $F$ , le nombre d'éléments focaux dans l'intervalle  $[1, |2^\Omega|]$ .
  - 3: Partager l'intervalle  $[0, 1]$  en  $F$  sous-intervalles continus.
  - 4: Tirer aléatoirement une masse de chaque sous-intervalle.
  - 5: Affecter une masse à chaque élément focal.
  - 6: Affecter une masse à l'ignorance totale de sorte que la somme de toutes les masses des éléments focaux soit égale à 1.
  - 7: **end for**
  - 8: **return**  $n$  fonctions de masse
- 

Cet algorithme a été utilisé pour générer 100 fonctions de masse pour deux sources  $S_1$  et  $S_2$  avec les trois hypothèses sur la dépendance des sources (sources indépendantes, sources dépendantes positivement et sources dépendantes négativement).

1. Sources indépendantes : Supposons que deux sources  $S_1$  et  $S_2$  sont complètement indépendantes. Nous avons alors généré 100 fonctions de masse pour chaque source comme décrit dans l'algorithme 1 avec  $|\Omega| = 5$ . En appliquant la méthode proposée, nous avons

obtenu deux fonctions de masse définies sur le cadre de discernement  $\Omega_I$  décrivant l'indépendance de  $S_1$  et de  $S_2$ . La fonction de masse de l'indépendance de  $S_1$  par rapport à  $S_2$  est :  $m_1(Ind) = 0.7265$ ,  $m_1(Dep) = 0.2735$ . La fonction de masse de l'indépendance de  $S_2$  par rapport  $S_1$  est :  $m_2(Ind) = 0.6913$ ,  $m_2(Dep) = 0.3087$ . Après transformation pignistique, l'indépendance de  $S_1$  par rapport à  $S_2$  est  $I_d(S_1, S_2) = 0.7265$  celle de  $S_2$  par rapport à  $S_1$  est  $I_d(S_2, S_1) = 0.3087$  ce qui montre que les sources sont complètement indépendantes.

2. Sources dépendantes positivement : Lorsque deux sources  $S_1$  et  $S_2$  sont dépendantes positivement, les classes de décision (en terme de probabilité pignistique) calculées à partir des fonctions de masse qu'elles fournissent se ressemblent. Les deux sources  $S_1$  et  $S_2$  s'expriment de la même manière puisqu'elles sont dépendantes. Nous avons généré aléatoirement 100 fonctions de masse pour chaque source. Ces fonctions de masse sont modifiées par la suite suivant l'algorithme ci-dessous :

---

**Algorithm 2** Générer des fonctions de masse positivement dépendantes

---

**Require:**  $n$  fonctions de masse générées aléatoirement avec l'algorithme 1, Les classes de décision

- 1: **for**  $i = 1$  to  $n$  **do**
  - 2: Recherche des éléments focaux de chaque fonction de masse  $m_i$
  - 3: **for**  $j = 1$  to  $F$  **do**
  - 4: la masse affectée au  $j^{\text{ème}}$  élément focal est transférée à son union avec la classe de décision.
  - 5: **end for**
  - 6: **end for**
  - 7: **return**  $n$  fonctions de masse modifiées.
- 

En appliquant la méthode présentée dans cet article deux fonctions de masse définies sur le cadre de discernement  $\Omega_P = \{P, \bar{P}\}$  sont obtenues. La fonction de masse de la dépendance de  $S_1$  par rapport à  $S_2$  est :

$$m_1(P) = 0.7652, m_1(\bar{P}) = 0.204, m_1(\bar{P} \cup P) = 0.0308.$$

Après transformation pignistique, nous obtenons :

$BetP(P) = 0.7806$  et  $BetP(\bar{P}) = 0.2194$  ce qui signifie que  $S_1$  est positivement dépendante de  $S_2$  avec un degré de 0.7806. La fonction de masse de dépendance de  $S_2$  par rapport à  $S_1$  est :  $m_2(P) = 0.7653$ ,  $m_2(\bar{P}) = 0.2035$ ,  $m_2(\bar{P} \cup P) = 0.0312$ .

Après transformation pignistique, nous obtenons :

$BetP(P) = 0.7809$  et  $BetP(\bar{P}) = 0.2191$  ce qui signifie que  $S_2$  est aussi positivement dépendante de  $S_2$  avec un degré de 0.7809.

3. Sources dépendantes négativement : Lorsque deux sources  $S_1$  et  $S_2$  sont dépendantes négativement, les classes de décision (en terme de probabilité pignistique) calculées à partir des fonctions de masse qu'elles fournissent sont contradictoires mais d'une façon ordonnée. Les deux sources  $S_1$  et  $S_2$  sont dépendantes mais l'une des deux sources a tendance à dire l'opposé de l'autre. Nous avons généré aléatoirement 100 fonctions de masse pour chaque source. Ces fonctions de masse sont modifiées par la suite suivant l'algorithme 3.

**Algorithm 3** Générer des fonctions de masse négativement dépendantes

---

**Require:**  $n$  fonctions de masse générées aléatoirement avec l’algorithme 1 pour une source, Les classes de décision de l’autre source, La correspondance de classes contradictoires

- 1: **for**  $i = 1$  to  $n$  **do**
- 2: Recherche des éléments focaux de chaque fonction de masse  $m_i$
- 3: **for**  $j = 1$  to  $F$  **do**
- 4: la masse affectée au  $j^{\text{ème}}$  élément focal est transférée à son union avec la classe contradictoire à la classe de décision de  $m_i$ .
- 5: **end for**
- 6: **end for**
- 7: **return**  $n$  fonctions de masse modifiées.

---

En appliquant la méthode présentée dans cet article deux fonctions de masse définies sur le cadre de discernement  $\Omega_P = \{P, \bar{P}\}$  sont obtenues. La fonction de masse de la dépendance de  $S_1$  par rapport à  $S_2$  est :

$$m_1(P) = 0.2938, m_1(\bar{P}) = 0.4448, m_1(\bar{P} \cup P) = 0.2614.$$

Après transformation pignistique, nous obtenons :

$BetP(P) = 0.4245$  et  $BetP(\bar{P}) = 0.5755$  ce qui signifie que  $S_1$  est négativement dépendante de  $S_2$  avec un degré de 0.5755. La fonction de masse de dépendance de  $S_2$  par rapport à  $S_1$  est :

$$m_2(P) = 0.2805, m_2(\bar{P}) = 0.4195, m_2(\bar{P} \cup P) = 0.3.$$

Après transformation pignistique, nous obtenons :

$BetP(P) = 0.4305$  et  $BetP(\bar{P}) = 0.5695$  ce qui signifie que  $S_2$  est aussi négativement dépendante de  $S_2$  avec un degré de 0.5695.

Remarquons que les degrés de dépendance de  $S_1$  par rapport à  $S_2$  et celui de  $S_2$  par rapport à  $S_1$  sont différents ce qui montre  $I_d(S_1, S_2) \neq I_d(S_2, S_1)$ .

## 6 Conclusion

Dans cet article, nous avons proposé une méthode permettant de mesurer l’indépendance cognitive de deux sources. Cette mesure de dépendance guidera le choix de la règle de combinaison lors de la fusion des fonctions de masse fournies par ces deux sources. La méthode proposée comporte trois étapes ; dans la première étape un algorithme de classification non-supervisée est appliqué sur les fonctions de masse de chaque source. Ensuite, un alignement des clusters est recherché et un poids est attribué à chaque couple de clusters. Enfin, si les clusters sont globalement dépendants le type de cette dépendance pourra être déterminé. Dans les prochains travaux, une généralisation de cette méthode sur plus de deux sources sera envisagée.

## Références

BenHariz, S., Z. Elouedi, et K. Mellouli (2006). Clustering approach using belief function theory. In *AIMSA*, pp. 162–171.

- BenYaghlane, B., P. Smets, et K. Mellouli (2002a). Belief function independence : I. the marginal case. *Int. J. Approx. Reasoning* 29(1), 47–70.
- BenYaghlane, B., P. Smets, et K. Mellouli (2002b). Belief function independence : 2. the conditional case. *Int. J. Approx. Reasoning* 31(1-2), 31–75.
- Chebbah, M., A. Martin, et B. B. Yaghlane (2012a). About sources dependence in the theory of belief functions. In *Belief Functions*, pp. 239–246.
- Chebbah, M., A. Martin, et B. B. Yaghlane (2012b). Positive and negative dependence for evidential database enrichment. In *IPMU (3)*, pp. 575–584.
- Dempster, A. P. (1967). Upper and Lower probabilities induced by a multivalued mapping. *Annals of Mathematical Statistics* 38, 325–339.
- Denceux, T. (2006). The cautious rule of combination for belief functions and some extensions. In *International Conference on Information Fusion*, Florence, Italy.
- Dubois, D. et H. Prade (1988). Representation and combination of uncertainty with belief functions and possibility measures. *Computational Intelligence* 4, 244–264.
- Jousselme, A.-L., D. Grenier, et E. Bossé (2001). A new distance between two bodies of evidence. *Information Fusion* 2, 91–101.
- Martin, A. et C. Osswald (2007). Une nouvelle règle de combinaison répartissant le conflit - applications en imagerie sonar et classification de cibles radar. *Traitement du Signal* 24(2), 71–82.
- Murphy, C. (2000). Combining belief functions when evidence conflicts. *Decision Support Systems* 29, 1–9.
- Shafer, G. (1976). *A mathematical theory of evidence*. Princeton University Press.
- Smets, P. (1990). The Combination of Evidence in the Transferable Belief Model. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 12(5), 447–458.
- Smets, P. (1993). Belief Functions : the Disjunctive Rule of Combination and the Generalized Bayesian Theorem. *International Journal of Approximate Reasoning* 9, 1–35.
- Smets, P. et R. Kennes (1994). The Transferable Belief Model. *Artificial Intelligence* 66, 191–234.
- Smets, P. et R. Kruse (1997). *The transferable belief model for belief representation*, pp. 343–368. Boston : Kluwer Academic Publishers.
- Yager, R. R. (1987). On the Dempster-Shafer Framework and New Combination Rules. *Information Sciences* 41, 93–137.

## Summary

In the theory of belief functions many combination rules are proposed. Some of these combination rules are used with the assumption of their sources independence whereas others combine believes of dependent sources. Therefore, the choice of the combination rule depends on the dependency of sources involved in the combination. In this paper, we proposed an independence, positive and negative measures in order to inform about sources dependencies and guide the choice of the combination rule to be used in the combination step.

# Intégration d'une mesure d'indépendance pour la fusion d'informations

Mouloud Kharoune\*, Arnaud Martin\*

\*UMR 6074 IRISA, Université de Rennes 1 / IUT de Lannion, Rue Edouard Branly  
BP 3021, 22302 Lannion cedex

Mouloud.Kharoune@univ-rennes1.fr, Arnaud.Martin@univ-rennes1.fr

**Résumé.** La fusion d'informations fait intervenir plusieurs sources d'informations afin d'améliorer la décision en terme de certitude et de précision. Quelle que soit l'approche retenue pour réaliser la fusion d'informations, l'hypothèse d'indépendance est généralement une hypothèse forte et incontournable. Nous proposons dans cet article une approche permettant d'intégrer une mesure d'indépendance avant de réaliser la combinaison des informations dans le cadre de la théorie des fonctions de croyance.

## 1 Introduction

Tel que repris par Martin (2005), la fusion d'informations consiste à combiner des informations issues de plusieurs sources afin d'aider à la prise de décision. Les approches de fusion d'informations cherchent donc à tenir compte des redondances des informations issues des différentes sources. Les approches de fusion n'ont bien sûr d'intérêt que si les sources sont imparfaites et fournissent des informations peu sûres et précises qui se complètent. Ainsi il faut donc chercher à modéliser aux mieux les imperfections des sources et des données. Pour ce faire, différentes théories de l'incertain ont été sollicitées. Parmi elles, citons la théorie des probabilités, des sous-ensembles flous et des possibilités ou encore la théorie des fonctions de croyance. Quel que soit le cadre théorique retenu, lors de l'étape de combinaison de l'information, l'hypothèse d'indépendance des sources est généralement faite. Cette hypothèse est plus ou moins forte. Par exemple, l'indépendance statistique est généralement retenue pour appliquer plus aisément la combinaison bayésienne du cadre probabiliste. En effet, les estimations peuvent s'avérer très vite compliquées sans cette hypothèse. Dans le cas de la théorie des fonctions de croyance, il est question d'indépendance cognitive définie par Shafer (1976). Elle correspond à une absence de communication entre les sources sans que celles-ci soient pour autant indépendantes statistiquement. Malheureusement cette hypothèse d'indépendance est rarement vérifiée ou justifiée.

Dans ce travail nous nous intéressons plus particulièrement à la théorie des fonctions de croyance car elle offre un outil riche de modélisation et de gestion de l'information. En particulier, Chebbah et al. (2012, 2013) ont récemment proposé des approches pour mesurer l'indépendance des sources en distinguant de plus la dépendance positive et négative.

Nous poursuivons ainsi cet article en présentant les principes de base de la théorie des fonctions de croyance et en particulier la notion d'indépendance et d'affaiblissement. Nous ex-

posons ensuite l'approche proposée permettant de tenir compte d'une mesure d'indépendance avant la combinaison des fonctions de croyance. Nous illustrons ce principe à partir d'exemples générés.

## 2 Théorie des fonctions de croyance

La théorie des fonctions de croyance issue des travaux de Dempster (1967), repris par Shafer (1976) est depuis quelques années employée dans des applications de fusion d'informations. Nous présentons ci-dessous les principes de cette théorie.

### 2.1 Principes de base

Considérons le cadre de discernement  $\Omega = \{\omega_1, \omega_2, \dots, \omega_n\}$  correspondant à l'ensemble de toutes les hypothèses possibles de décision d'un problème donné. Les éléments  $\omega_i$  représentent ainsi toutes les hypothèses exclusives et exhaustives.

L'ensemble  $2^\Omega = \{A/A \subseteq \Omega\} = \{\emptyset, \omega_1, \omega_2, \dots, \omega_n, \omega_1 \cup \omega_2, \dots, \Omega\}$ , est composé de toutes les disjonctions de  $\Omega$ . L'espace puissance  $2^\Omega$  comporte  $2^{|\Omega|} = 2^n$  éléments.

Une fonction de masse est une fonction de  $2^\Omega$  vers l'intervalle  $[0, 1]$  qui affecte à chaque sous-ensemble de  $2^\Omega$  une valeur de l'intervalle  $[0, 1]$  représentant sa masse de croyance élémentaire. Elle s'écrit :

$$m^\Omega : 2^\Omega \mapsto [0, 1] \quad (1)$$

telle que :

$$\sum_{A \subseteq \Omega} m^\Omega(A) = 1 \quad (2)$$

Un sous-ensemble de  $2^\Omega$  de masse de croyance non-nulle est un élément focal. La masse affectée à un élément focal  $A$  représente le degré de croyance élémentaire de la source à ce que la solution du problème soit  $A$ . Une fonction de masse permet ainsi de représenter des connaissances incertaines et imprécises d'une source d'informations. En général, nous manipulons des fonctions de masse non dogmatiques (*i.e.* dont l'ignorance  $\Omega$  est élément focal), d'une part car elles permettent de systématiquement modéliser la part d'ignorance intrinsèque à toute source, mais également car toute fonction de masse non dogmatique est décomposable en fonctions de masse à support simple (*i.e.* qui ne comporte que deux éléments focaux dont  $\Omega$ ). Les fonctions de masse à support simple sont notées  $A^w$  telles que  $m(A) = 1 - w \forall A \neq \Omega$  et  $m(\Omega) = w$ . Ainsi une fonction de masse non dogmatique peut s'écrire :

$$m^\Omega = \odot_{A \subseteq \Omega} A^w(A) \quad (3)$$

où  $\odot$  est donnée par l'équation (5) ci-dessous.

La crédibilité *bel* et la plausibilité *pl* sont des fonctions duales définies à partir de la fonctions de masse et représentent respectivement une fonction de croyance minimale et maximale. Ainsi la fonction de plausibilité est donnée par :

$$pl(X) = \sum_{Y \subseteq \Omega, Y \cap X \neq \emptyset} m(Y) = bel(\Omega) - bel(X^c) = 1 - m(\emptyset) - bel(X^c), \quad (4)$$

où  $X^c$  est le complémentaire de  $X$ .

Une fois les fonctions de masse  $m_j^\Omega$  déterminées pour chaque source d'informations  $S_j$ , plusieurs opérateurs de combinaison sont envisageables en fonction des hypothèses initiales. Les opérateurs de type conjonctif peuvent être employés lorsque les sources sont fiables et indépendantes cognitivement. La combinaison conjonctive s'écrit pour deux fonctions de masse  $m_1^\Omega$  et  $m_2^\Omega$  et pour tout  $X \in 2^\Omega$  par :

$$m_{\text{Conj}}^\Omega(X) = m_1 \odot m_2 = \sum_{Y_1 \cap Y_2 = X} m_1^\Omega(Y_1) m_2^\Omega(Y_2). \quad (5)$$

Notons que l'élément neutre pour cette règle est la masse :  $m_\Omega^\Omega(X) = 1$  si  $X = \Omega$  et 0 sinon. Lorsque cette hypothèse de fiabilité est trop forte et que l'on ne peut supposer que seule une des sources est fiable, la combinaison disjonctive peut alors être employée toujours sous l'hypothèse d'indépendance cognitive :

$$m_{\text{Dis}}^\Omega(X) = \sum_{Y_1 \cup Y_2 = X} m_1^\Omega(Y_1) m_2^\Omega(Y_2). \quad (6)$$

Notons que l'élément neutre pour cette règle est la masse :  $m_\emptyset^\Omega(X) = 1$  si  $X = \emptyset$  et 0 sinon. La plupart des règles de combinaison issues des règles conjonctives et disjonctives, en particulier pour répartir le conflit, supposent que les sources sont indépendantes cognitivement. Martin (2010) en rappelle quelques unes.

Dencœux (2008) propose une famille de règles qui ne nécessitent pas l'hypothèse d'indépendance cognitive. Ainsi selon le comportement conjonctif ou disjonctif deux règles principales sont définies, la règle prudente et hardie. La règle prudente s'écrit pour les fonctions de masse non dogmatiques :

$$m_1^\Omega \otimes m_2^\Omega = \odot_{A \subset \Omega} A^{w_1(A) \wedge w_2(A)} \quad (7)$$

où  $\wedge$  est le maximum. La règle hardie s'écrit de même en considérant le minimum au lieu du maximum. Si ces règles sont efficaces lorsque les sources sont dépendantes, cette notion de dépendance ou d'indépendance n'est pas clairement définie par Dencœux (2008).

Lorsqu'une connaissance supplémentaire garantit un sous-ensemble  $A \subset \Omega$ , nous pouvons définir une fonction de masse conditionnelle par :

$$m^\Omega[A](X) = (m^\Omega \odot m_A^\Omega)(X) \quad (8)$$

où  $m_A^\Omega(A) = 1$  est la fonction de masse garantissant la réalisation de  $A$ .

## 2.2 Notion d'indépendance

L'indépendance statistique est définie pour deux variables  $A$  et  $B$  par  $P(A|B) = P(A)$  ou de façon équivalente par  $P(A \cap B) = P(A)P(B)$ . Cette indépendance est étendue par Shafer (1976) dans le cadre de la théorie des fonctions de croyance et est donnée par :  $pl(A \cap B) = pl(A)pl(B)$ . Ben Yaghlane et al. (2002a,b) définissent une indépendance doxatique entre des variables définies sur des cadres de discernement différents éventuellement.

Ces définitions de l'indépendance ne correspondent pas à la notion d'indépendance cognitive entre les sources d'informations. Cette dernière se révèle très difficile à mesurer. Chebbah

et al. (2012, 2013) proposent une définition d'une mesure d'indépendance entre deux sources d'informations étendues à une mesure de dépendance positive et négative. La mesure d'indépendance entre deux sources est définie comme une sorte de corrélation entre deux sources issues d'un *clustering* (classification non-supervisée) sur les fonctions de masse de chacune des sources en associant ensuite les clusters. Si  $|\Omega| = n$ , le clustering des fonctions de masse issues de la source  $S_1$  fourni  $n$  clusters, de même pour  $S_2$ . Les clusters des deux sources ( $Cl_{k_1}$ ,  $Cl_{k_2}$ ) sont associés de façon non symétrique en maximisant :

$$\alpha_{k_1, k_2}^i = \frac{|Cl_{k_1} \cap Cl_{k_2}|}{|Cl_{k_i}|}, \quad i = 1, 2 \quad (9)$$

Il est ensuite possible de définir une fonction de masse sur  $\Omega_I = \{I, \bar{I}\}$  représentant les deux possibilités : indépendant et dépendant ( $\bar{I}$ ) de la source  $S_1$  par rapport à la source  $S_2$  :

$$\begin{cases} m_{k_1 k_2}^{\Omega_I}(I) = \beta (1 - \alpha_{k_1 k_2}^1) \\ m_{k_1 k_2}^{\Omega_I}(\bar{I}) = \beta \alpha_{k_1 k_2}^1 \\ m_{k_1 k_2}^{\Omega_I}(I \cup \bar{I}) = 1 - \beta \end{cases} \quad (10)$$

où  $\beta$  est un facteur d'affaiblissement permettant de tenir compte du nombre d'observations dans chaque cluster. Ainsi la croyance élémentaire que la source  $S_1$  est indépendante de  $S_2$  est donnée par la masse :

$$m^{\Omega_I}(X) = \frac{1}{n} \left( \sum_{k_1=1}^n m_{k_1 k_2}^{\Omega_I} \right) (X) \quad (11)$$

où  $k_2$  est le cluster de la source  $S_2$  associé au cluster  $k_1$  de la source  $S_1$ . La moyenne est ici employée du fait de la dépendance des fonctions de masse.

Chebbah et al. (2012, 2013) proposent un prolongement pour différencier la dépendance positive (la source  $S_1$  suit les avis de la source  $S_2$ ) et la dépendance négative (la source  $S_1$  dit le contraire des avis de la source  $S_2$ ). Ainsi, une fonction de masse conditionnelle est construite sur le cadre de discernement  $\Omega_P = \{P, \bar{P}\}$  :

$$\begin{cases} m_{k_1 k_2}^{\Omega_P}[\bar{I}](P) = 1 - Dist(Cl_{k_1}, Cl_{k_2}) \\ m_{k_1 k_2}^{\Omega_P}[\bar{I}](\bar{P}) = Dist(Cl_{k_1}, Cl_{k_2}) \\ m_{k_1 k_2}^{\Omega_P}[\bar{I}](P \cup \bar{P}) = 0 \end{cases} \quad (12)$$

où  $Dist(Cl_{k_1}, Cl_{k_2})$  est la distance entre les deux clusters dépendants  $Cl_{k_1}$  et  $Cl_{k_2}$  liés comme étant la moyenne des distances entre les fonctions de masse des objets en commun :

$$Dist(Cl_{k_1}, Cl_{k_2}) = \frac{1}{|Cl_{k_1} \cap Cl_{k_2}|} \sum_{j=1}^{|Cl_{k_1} \cap Cl_{k_2}|} d(m_{1,j}^{\Omega}, m_{2,j}^{\Omega}) \quad (13)$$

où  $d$  est la distance proposée par Jousselme et al. (2001) entre les fonctions de masse de la source  $S_1$  et  $S_2$  respectivement.

Si nous considérons que  $\bar{I} = P \cup \bar{P}$ , nous pouvons réécrire les deux fonctions de masse précédentes dans le cadre de discernement  $\mathcal{I} = \{I, P, \bar{P}\}$ . Nous définissons ainsi la fonction

de masse entre deux clusters de  $S_1$  et  $S_2$  :

$$\begin{cases} m_{k_1 k_2}^{\mathcal{I}}(I) = m_{k_1 k_2}^{\Omega_I}(I) = \beta \alpha_{k_1 k_2}^1 \\ m_{k_1 k_2}^{\mathcal{I}}(P) = m_{k_1 k_2}^{\Omega_I}(\bar{I}) m_{k_1 k_2}^{\Omega_P}[\bar{I}](P) = \beta (1 - \alpha_{k_1 k_2}^1) (1 - \text{Dist}(Cl_{k_1}, Cl_{k_2})) \\ m_{k_1 k_2}^{\mathcal{I}}(\bar{P}) = m_{k_1 k_2}^{\Omega_I}(\bar{I}) m_{k_1 k_2}^{\Omega_P}[\bar{I}](\bar{P}) = \beta (1 - \alpha_{k_1 k_2}^1) \text{Dist}(Cl_{k_1}, Cl_{k_2}) \\ m_{k_1 k_2}^{\mathcal{I}}(P \cup \bar{P}) = m_{k_1 k_2}^{\Omega_I}(\bar{I}) m_{k_1 k_2}^{\Omega_P}[\bar{I}](P \cup \bar{P}) = 0 \\ m_{k_1 k_2}^{\mathcal{I}}(I \cup P \cup \bar{P}) = m_{k_1 k_2}^{\Omega_I}(I \cup \bar{I}) = 1 - \beta \end{cases} \quad (14)$$

La fonction de masse sur la dépendance de la source  $S_1$  par rapport à  $S_2$  est donnée par :

$$m^{\mathcal{I}}(X) = \frac{1}{n} \left( \sum_{k_1=1}^n m_{k_1 k_2}^{\mathcal{I}} \right) (X) \quad (15)$$

où  $k_2$  est le cluster de la source  $S_2$  associé au cluster  $k_1$  de la source  $S_1$ . Cette fonction de masse représente ainsi l'ensemble des croyances élémentaires sur l'indépendance et dépendance positive et négative de la source  $S_1$  face à la source  $S_2$ .

### 2.3 Notion d'affaiblissement

Shafer (1976) a proposé la procédure d'affaiblissement suivante :

$$\alpha m^{\Omega}(X) = \alpha m^{\Omega}(X) \quad \forall X \in 2^{\Omega} \setminus \Omega \quad (16)$$

$$\alpha m^{\Omega}(\Omega) = 1 - \alpha(1 - m^{\Omega}(\Omega)) \quad (17)$$

où  $\alpha$  est un facteur d'affaiblissement de  $[0, 1]$ . Cette procédure est généralement employée pour affaiblir les fonctions de masse par la fiabilité  $\alpha$  des sources d'informations. Cette procédure a pour effet d'augmenter la masse sur l'ignorance  $\Omega$ . Smets (1993) a justifié cette procédure en considérant que :

$$m^{\Omega}[F](X) = m^{\Omega}(X) \quad (18)$$

$$m^{\Omega}[\bar{F}](X) = m^{\Omega}(X) \quad (19)$$

où  $m_{\Omega}^{\Omega}(X) = 1$  si  $X = \Omega$  et 0 sinon,  $F$  et  $\bar{F}$  représentent la fiabilité et la non fiabilité et  $m^{\Omega}[F]$  est une fonction de masse conditionnellement à la fiabilité  $F$ . Soit  $\mathcal{F} = \{F, \bar{F}\}$  le cadre de discernement correspondant, et la fonction de masse représentant la connaissance sur la fiabilité de la source :

$$\begin{cases} m^{\mathcal{F}}(F) = \alpha \\ m^{\mathcal{F}}(\mathcal{F}) = 1 - \alpha. \end{cases} \quad (20)$$

Afin de combiner les deux sources d'informations fournissant les deux fonctions de masse  $m^{\Omega}[F]$  et  $m^{\mathcal{F}}$ , il faut pouvoir les représenter dans le même espace  $\Omega \times \mathcal{F}$ . Ainsi, nous devons effectuer une *extention à vide* sur la fonction de masse  $m^{\mathcal{F}}$ , opération que l'on note  $m^{\mathcal{F}\uparrow\Omega \times \mathcal{F}}$  :

$$m^{\mathcal{F}\uparrow\Omega \times \mathcal{F}}(Y) = \begin{cases} m^{\mathcal{F}}(X) & \text{si } Y = \Omega \times X, \quad X \subseteq \mathcal{F} \\ 0 & \text{sinon} \end{cases} \quad (21)$$

Dans le cas de la fonction de masse  $m^{\Omega}[F]$ , il faut déconditionner :

$$m^{\Omega}[F]^{\uparrow\Omega \times \mathcal{F}}((A \times F) \cup (\Omega \times \bar{F})) = m^{\Omega}[F](A), \quad A \subseteq \Omega \quad (22)$$

Il est ainsi possible d'effectuer la combinaison :

$$m_{\text{Conj}}^{\Omega \times \mathcal{F}}(Y) = m^{\mathcal{F} \uparrow \Omega \times \mathcal{F}} \odot m^{\Omega} [F]^{\uparrow \Omega \times \mathcal{F}}(Y), \quad \forall Y \subset \Omega \times \mathcal{F} \quad (23)$$

Ensuite il faut marginaliser la fonction de masse obtenue pour revenir dans l'espace  $\Omega$  :

$$m^{\Omega \times \mathcal{F} \downarrow \Omega}(X) = \sum_{\{Y \subseteq \Omega \times \mathcal{F} \mid \text{Proj}(Y \downarrow \Omega) = X\}} m_{\text{Conj}}^{\Omega \times \mathcal{F}}(Y) \quad (24)$$

où  $\text{Proj}(Y \downarrow \Omega)$  est la projection de  $Y$  sur  $\Omega$ . Nous retrouvons ainsi :

$${}^{\alpha} m^{\Omega}(X) = m^{\Omega \times \mathcal{F} \downarrow \Omega}(X) \quad (25)$$

Mercier (2006) a proposé une extension de cet affaiblissement en contextualisant le coefficient d'affaiblissement  $\alpha$  en fonction de sous-ensembles de  $\Omega$ .

### 3 Intégration de l'indépendance dans une fonction de masse

Nous avons vu que la notion de l'indépendance est généralement une information supplémentaire nécessaire à la fusion d'informations, mais non prise en compte dans le formalisme choisi. La section 2.2 propose une modélisation et estimation d'une mesure d'indépendance dans le cadre de la théorie des fonctions de croyance. Nous allons ici nous appuyer sur le principe de l'affaiblissement présenté dans la section 2.3 afin de tenir compte de l'indépendance dans les fonctions de masse en vue de la combinaison.

En effet, lors de la combinaison conjonctive par exemple l'hypothèse d'indépendance cognitive des sources d'informations est nécessaire. Si les sources ne sont pas indépendantes on peut penser qu'elles ne devraient pas être combinées par ce biais. Cependant, comme le montre la section 2.2 les sources peuvent avoir des degrés de dépendance et d'indépendance. L'information fournie sur l'indépendance n'est pas catégorique. Ainsi, combiner deux sources indépendantes fortement devraient tendre vers le résultat de la combinaison de deux sources indépendantes. Si une source est dépendante d'une autre source, nous pouvons considérer que cette première source ne doit pas influencer la combinaison avec une seconde source. Ainsi cette source doit représenter l'élément neutre de la combinaison.

Dans ce cas, il suffit d'appliquer la procédure d'affaiblissement de la section 2.3 sur la fonction de masse  $m^{\Omega}$  de la source  $S_1$  en considérant l'indépendance donnée par la fonction de masse de l'équation (11) au lieu de celle de l'équation (20) dans le cas de la fiabilité.

À présent, nous distinguons la dépendance positive de la dépendance négative. Si une source est dépendante positivement d'une autre source, il ne faut pas en tenir compte et donc tendre vers un résultat de combinaison qui prendrait cette première source comme un élément neutre. Enfin si une source est dépendante négativement d'une autre source, il peut être intéressant de marquer cette dépendance conflictuelle en augmentant la masse sur l'ensemble vide.

Pour réaliser ce schéma, nous proposons d'affaiblir les fonctions de masse d'une source  $S_1$  en fonction de sa mesure d'indépendance à une autre source  $S_2$ , donnée par la fonction de masse  $m^{\mathcal{F}}$  de l'équation (15). Nous réécrivons ici cette fonctions de masse de façon à simplifier

les notations :

$$\begin{cases} m^{\mathcal{I}}(I) = \alpha\beta \\ m^{\mathcal{I}}(P) = \alpha(1-\beta)\gamma \\ m^{\mathcal{I}}(\bar{P}) = \alpha(1-\beta)(1-\gamma) \\ m^{\mathcal{I}}(I \cup P \cup \bar{P}) = 1 - \alpha \end{cases} \quad (26)$$

Ainsi, la paramètre  $\alpha$  représente la fiabilité de la source  $S_1$ ,  $\beta$  l'indépendance de  $S_1$  face à  $S_2$  et  $\gamma$  la dépendance positive de  $S_1$  face à  $S_2$ . Ces trois paramètres,  $\alpha$ ,  $\beta$  et  $\gamma$  sont compris entre 0 et 1.

Nous considérons ici une fonction de masse d'une source  $m^\Omega$  en fonction de son indépendance ou dépendance à une autre source. Ainsi nous définissons :

$$\begin{cases} m^\Omega[I](X) = m^\Omega(X) \\ m^\Omega[\bar{P}](X) = m_\emptyset^\Omega(X) \\ m^\Omega[P](X) = m_\Omega^\Omega(X) \end{cases} \quad (27)$$

où  $m_\Omega^\Omega(X) = 1$  si  $X = \Omega$  et 0 sinon et  $m_\emptyset^\Omega(X) = 1$  si  $X = \emptyset$  et 0 sinon. Suivant la procédure d'affaiblissement, nous effectuons une extension à vide sur la fonction de masse  $m^{\mathcal{I}}$  :

$$m^{\mathcal{I}\uparrow\Omega\times\mathcal{I}}(Y) = \begin{cases} m^{\mathcal{I}}(X) & \text{si } Y = \Omega \times X, \quad X \subseteq \mathcal{I} \\ 0 & \text{sinon} \end{cases} \quad (28)$$

Le déconditionnement des fonctions de masse  $m^\Omega[I]$ ,  $m^\Omega[P]$  et  $m^\Omega[\bar{P}]$  est donné par :

$$m^\Omega[I]^{\uparrow\Omega\times\mathcal{I}}((A \times I) \cup (\Omega \times \bar{I})) = m^\Omega[I](A), \quad A \subseteq \Omega \quad (29)$$

où  $\bar{I} = P \cup \bar{P}$ .

$$m^\Omega[\bar{P}]^{\uparrow\Omega\times\mathcal{I}}((A \times \bar{P}) \cup (\Omega \times \{I \cup P\})) = m^\Omega[\bar{P}](A), \quad A \subseteq \Omega \quad (30)$$

$$m^\Omega[P]^{\uparrow\Omega\times\mathcal{I}}((A \times P) \cup (\Omega \times \{I \cup \bar{P}\})) = m^\Omega[P](A), \quad A \subseteq \Omega \quad (31)$$

Ce dernier déconditionnement mène en fait à la masse de l'ignorance et est l'élément neutre de la combinaison conjonctive.

Nous réalisons ensuite la combinaison conjonctive :

$$m_{\text{Conj}}^{\Omega\times\mathcal{I}}(Y) = m^{\mathcal{I}\uparrow\Omega\times\mathcal{I}} \odot m^\Omega[I]^{\uparrow\Omega\times\mathcal{I}} \odot m^\Omega[\bar{P}]^{\uparrow\Omega\times\mathcal{I}}(Y), \quad \forall Y \subset \Omega \times \mathcal{I} \quad (32)$$

La marginalisation de la fonction de masse permet ensuite de revenir dans l'espace  $\Omega$  :

$$m^{\Omega\times\mathcal{I}\downarrow\Omega}(X) = \sum_{\{Y \subseteq \Omega \times \mathcal{I} \mid \text{Proj}(Y \downarrow \Omega) = X\}} m_{\text{Conj}}^{\Omega\times\mathcal{I}}(Y) \quad (33)$$

Cette procédure réalisée pour la source  $S_1$  en rapport à la source  $S_2$  peut être réalisée pour la source  $S_2$  au regard de la source  $S_1$ . Ainsi les deux fonctions de masse obtenue peuvent être combinées par la règle de combinaison conjonctive qui suppose l'indépendance.

## 4 Illustration

### 4.1 Fonctionnement de l'affaiblissement par la mesure d'indépendance

Nous allons dans un premier temps illustrer le fonctionnement de l'affaiblissement par la mesure d'indépendance. Nous considérons ici un cadre de discernement  $\Omega = \{\omega_1, \omega_2, \omega_3\}$ . Supposons que nous ayons deux sources  $S_1$  et  $S_2$  donnant deux fonctions de masse :

$$m_1^\Omega(\omega_1) = 0.2, m_1^\Omega(\omega_1 \cup \omega_2) = 0.5, m_1^\Omega(\Omega) = 0.3, \quad (34)$$

$$m_2^\Omega(\omega_2) = 0.1, m_2^\Omega(\omega_1 \cup \omega_2) = 0.6, m_2^\Omega(\Omega) = 0.3 \quad (35)$$

La combinaison conjonctive donne :

$$m_{1 \cap 2}^\Omega(\emptyset) = 0.02, m_{1 \cap 2}^\Omega(\omega_1) = 0.18, m_{1 \cap 2}^\Omega(\omega_2) = 0.08, \\ m_{1 \cap 2}^\Omega(\omega_1 \cup \omega_2) = 0.63, m_{1 \cap 2}^\Omega(\Omega) = 0.09$$

Cette combinaison conjonctive est effectuée avec l'hypothèse d'indépendance cognitive des deux sources. Si une connaissance externe permet de mesurer la dépendance positive et négative de la source  $S_1$  par rapport à la source  $S_2$  telle que fournie par l'équation (36), nous devons en tenir compte avant la combinaison conjonctive. Supposons ainsi que  $\alpha = 0.95$ ,  $\beta = 0.05$  et  $\gamma = 0.95$  dans l'équation (36). Cette fonction de masse traduit donc une forte dépendance positive de  $S_1$  par rapport à  $S_2$ . Nous avons ainsi la fonction de masse :

$$\begin{cases} m^{\mathcal{I}}(I) = 0.0475 \\ m^{\mathcal{I}}(P) = 0.8574 \\ m^{\mathcal{I}}(\bar{P}) = 0.0451 \\ m^{\mathcal{I}}(I \cup P \cup \bar{P}) = 0.05 \end{cases} \quad (36)$$

Le tableau 1 présente les différentes étapes d'extension à vide, de déconditionnement et de combinaison dans l'espace  $\Omega \times \mathcal{I}$ . L'extension à vide et le déconditionnement transfèrent la masse sur les éléments focaux correspondant de l'espace  $\Omega \times \mathcal{I}$ . La combinaison des trois fonctions de masse dans cet espace fait apparaître la masse sur l'ensemble vide qui correspond à la part de dépendance négative.

Le tableau 2 présente ensuite la marginalisation et le résultat de combinaison avec la fonction de masse  $m_2$  non modifiée (*i.e.* que l'hypothèse d'indépendance totale de  $S_2$  par rapport à  $S_1$  est faite). Nous constatons que la masse transférée sur l'ignorance devient plus importante que lors de la combinaison conjonctive sans hypothèse sur la dépendance positive.

Afin de bien illustrer le transfert de masse sur l'ensemble vide et sur l'ignorance, les figures 1 et 2 représentent les masses en fonction des variations de  $\alpha$ ,  $\beta$  et  $\gamma$  pour une fonction de masse dogmatique quelconque. Ainsi sur la figure 1 représentant les variations de masse sur l'ensemble vide,  $\alpha$  est fixé à 1,  $\beta$  et  $\gamma$  variant, alors que sur la figure 2 représentant les variations de masse sur l'ignorance,  $\gamma$  est fixé à 1,  $\alpha$  et  $\beta$  variant.

La figure 1 montre ainsi que plus  $\beta$  et  $\gamma$  sont petits plus on obtient une masse importante sur l'ensemble vide et donc une dépendance négative. La quantité  $\beta$  représente la part d'indépendance et la quantité  $\gamma$  représente la part de dépendance positive.

La figure 2 présente quand à elle, la variation de la masse sur  $\Omega$ , l'ignorance. Cette masse est donnée directement par  $\alpha\beta$  qui contient donc la part d'indépendance  $\beta$  et la fiabilité  $\alpha$  de la source.

focal	$m^{I \uparrow \Omega \times \mathcal{I}}$	$m^\Omega [I] \uparrow \Omega \times \mathcal{I}$	$m^\Omega [\bar{P}] \uparrow \Omega \times \mathcal{I}$	$m_{\text{Conj}}^{\Omega \times \mathcal{I}}$
$\emptyset$				0.0451
$\omega_1 \times I$				0.0095
$(\omega_1 \cup \omega_2) \times I$				0.0237
$\Omega \times I$	0.0475			0.0142
$\Omega \times P$	0.8574			0.8574
$(\omega_1 \times I) \cup (\Omega \times P)$				0.01
$((\omega_1 \cup \omega_2) \times I) \cup (\Omega \times P)$				0.025
$\Omega \times \bar{P}$	0.0451			
$\Omega \times (P \cup \bar{P})$			1	
$(\omega_1 \times I) \cup (\Omega \times (P \cup \bar{P}))$		0.2		
$((\omega_1 \cup \omega_2) \times I) \cup (\Omega \times (P \cup \bar{P}))$		0.5		
$\Omega \times \mathcal{I}$	0.05	0.3		

TAB. 1 – Détails de l'affaiblissement de la mesure d'indépendance : fonctions de masse dans  $\Omega \times \mathcal{I}$ .

focal	$m_1^{\Omega \times \mathcal{I} \downarrow \Omega}$	$m_2^\Omega$	$m_1^{\Omega \times \mathcal{I} \downarrow \Omega} \odot m_2^\Omega$
$\emptyset$	0.0451		0.0461
$\omega_1$	0.0095		0.0085
$\omega_2$		0.1	0.0945
$\omega_1 \cup \omega_2$	0.0237	0.6	0.5743
$\Omega$	0.9216	0.3	0.2765

TAB. 2 – Détails de l'affaiblissement de la mesure d'indépendance : marginalisation et combinaison

Nous illustrons ainsi le résultat escompté de l'affaiblissement par la mesure d'indépendance, c'est-à-dire que nous retrouvons sur la masse de l'ensemble vide la quantité de dépendance négative et sur l'ignorance la quantité de fiabilité et de d'indépendance.

## 4.2 Influence sur le résultat de combinaison

Afin d'illustrer l'influence de la prise en compte de la mesure d'indépendance sur les fonctions de masse, nous allons considérer ici les deux sources précédentes  $S_1$  et  $S_2$  qui fournissent les fonctions de masse données par les équations (34) et (35). Nous allons considérer trois cas pour chaque source avec un cas où la source  $S_1$  est plutôt indépendante de  $S_2$  ( $\alpha = 0.95, \beta = 0.95, \gamma = 0.05$ ), un cas où elle est plutôt dépendante positivement ( $\alpha = 0.95, \beta = 0.05, \gamma = 0.95$ ) et un cas où elle est plutôt dépendante négativement ( $\alpha = 0.95, \beta = 0.05, \gamma = 0.05$ ).

Intégration d'une mesure d'indépendance pour la fusion d'informations

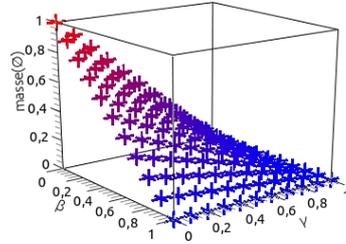


FIG. 1 – Variation de la masse sur l'ensemble vide pour un affaiblissement par la mesure d'indépendance d'une masse dogmatique.

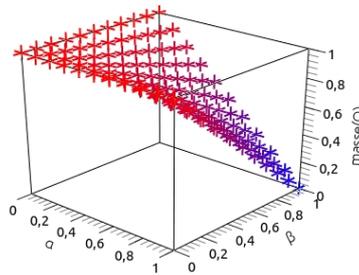


FIG. 2 – Variation de la masse sur l'ignorance  $\Omega$  pour un affaiblissement par la mesure d'indépendance d'une masse dogmatique.

$\gamma = 0.05$ ). Pour la source  $S_2$  nous considérons trois cas moins catégorique en fixant la fiabilité  $\alpha = 0.9$  : le cas plutôt indépendant ( $\beta = 0.9, \gamma = 0.1$ ), le cas plutôt dépendant positivement ( $\beta = 0.1, \gamma = 0.9$ ) et le cas plutôt dépendant négativement ( $\beta = 0.1, \gamma = 0.1$ ).

Ainsi, le tableau 3 présente les résultats de la combinaison des deux sources en fonction des hypothèses d'indépendance et de dépendance, positive ou négative des deux sources  $S_1$  et  $S_2$ . Nous constatons que lorsque les deux sources sont plutôt indépendantes l'une de l'autre, les résultats obtenus sont proches de ceux obtenus par la combinaison conjonctive directe sous l'hypothèse d'indépendance. Lorsqu'une des deux sources est dépendante négativement de l'autre, la masse transférée sur l'ensemble vide est importante. Lorsque l'une des deux sources est dépendante positivement la masse transférée sur l'ignorance mais de façon moins importante que pour la dépendance négative. En effet, l'ensemble vide est un élément absorbant

cas	élément focal	$S_2 : \alpha = 0.9$						
		$\beta = 0.9, \gamma = 0.1$		$\beta = 0.1, \gamma = 0.9$		$\beta = 0.1, \gamma = 0.1$		
		$m_1^{\Omega \times \mathcal{I} \downarrow \Omega}$	$m_2^{\Omega \times \mathcal{I} \downarrow \Omega}$	$m_{1 \cap 2}^{\Omega \times \mathcal{I} \downarrow \Omega}$	$m_2^{\Omega \times \mathcal{I} \downarrow \Omega}$	$m_{1 \cap 2}^{\Omega \times \mathcal{I} \downarrow \Omega}$	$m_2^{\Omega \times \mathcal{I} \downarrow \Omega}$	$m_{1 \cap 2}^{\Omega \times \mathcal{I} \downarrow \Omega}$
$S_1$ $\alpha = 0.95$ $\beta = 0.95$ $\gamma = 0.05$	$\emptyset$	0.0451	0.081	0.1371	0.081	0.1240	0.729	0.7428
	$\omega_1$	0.1805		0.1513		0.1643		0.0473
	$\omega_2$		0.081	0.0627	0.009	0.007	0.009	0.007
	$\omega_1 \cup \omega_2$	0.4513	0.486	0.5352	0.054	0.4281	0.054	0.1357
	$\Omega$	0.3231	0.352	0.1137	0.856	0.2766	0.208	0.0672
$S_1$ $\alpha = 0.95$ $\beta = 0.05$ $\gamma = 0.95$	$\emptyset$	0.0451	0.081	0.1232	0.081	0.1226	0.729	0.7413
	$\omega_1$	0.0095		0.008		0.0086		0.0025
	$\omega_2$		0.081	0.0766	0.009	0.0085	0.009	0.0085
	$\omega_1 \cup \omega_2$	0.0238	0.486	0.4678	0.0054	0.0714	0.054	0.056
	$\Omega$	0.9216	0.352	0.3244	0.856	0.7889	0.208	0.1917
$S_1$ $\alpha = 0.95$ $\beta = 0.05$ $\gamma = 0.05$	$\emptyset$	0.8574	0.081	0.8697	0.081	0.869	0.729	0.9614
	$\omega_1$	0.0095		0.008		0.0087		0.0025
	$\omega_2$		0.081	0.0108	0.009	0.0012	0.009	0.0012
	$\omega_1 \cup \omega_2$	0.0237	0.486	0.073	0.054	0.0275	0.054	0.0121
	$\Omega$	0.1094	0.352	0.0385	0.856	0.0936	0.208	0.0228

TAB. 3 – Résultats de combinaison selon les hypothèses de dépendance et d'indépendance des deux sources  $S_1$  et  $S_2$ .

pour la combinaison conjonctive. Cette masse sur l'ensemble vide à l'issue de la combinaison conjonctive peut ainsi jouer un rôle d'alerte sur la dépendance négative. Une autre alternative serait d'envisager une autre règle de combinaison lorsque la masse issue de la dépendance négative est trop importante.

## 5 Conclusion

Cet article souligne l'importance de mesurer la réelle indépendance des sources et d'en tenir compte en vue de la combinaison des informations issues de celles-ci. Nous nous restreignons ici à la théorie des fonctions de croyance qui représente un cadre assez général pour la fusion d'informations. De ce contexte théorique, nous avons montré une approche originale pour intégrer une mesure d'indépendance exprimée sous la forme d'une fonction de masse. Nous avons ensuite explicité les fonctions de masse conditionnellement à leur indépendance mutuelle par un procédé d'affaiblissement des masses initiales. Ce principe doit être réalisé en vue d'une combinaison de ces fonctions de masse qui nécessite l'hypothèse d'indépendance.

Une autre approche envisageable serait d'intégrer la mesure d'indépendance dans la combinaison des fonctions de masse.

## Références

- Ben Yaghlane, B., P. Smets, et K. Mellouli (2002a). Belief function independence : I. the marginal case. *International Journal of Approximate Reasoning* 29(1), 47–70.
- Ben Yaghlane, B., P. Smets, et K. Mellouli (2002b). Belief function independence : II. the conditional case. *International Journal of Approximate Reasoning* 29(1), 47–70.
- Chebbah, M., A. Martin, et B. Ben Yaghlane (2012). Positive and negative dependence for evidential database enrichment. In *IPMU*, Italy, pp. 575–584.
- Chebbah, M., A. Martin, et B. Ben Yaghlane (2013). Dépendance et indépendance des sources imparfaites. In *EGC*, Toulouse, France.
- Dempster, A. P. (1967). Upper and Lower probabilities induced by a multivalued mapping. *Annals of Mathematical Statistics* 38, 325–339.
- Denœux, T. (2008). Conjunctive and disjunctive combination of belief functions induced by nondistinct bodies of evidence. *Artificial Intelligence* 172, 234–264.
- Jousselme, A.-L., D. Grenier, et E. Bossé (2001). A new distance between two bodies of evidence. *Information Fusion* 2, 91–101.
- Martin, A. (2005). Fusion de classifieurs pour la classification d'images sonar. *RNTI Extraction des connaissances : Etat et perspectives E-5*, 259–268.
- Martin, A. (2010). Le conflit dans la théorie des fonctions de croyance. In *EGC*, Hammamet, Tunisie.
- Mercier, D. (2006). *Fusion d'informations pour la reconnaissance automatique d'adresses postales dans le cadre de la théorie des fonctions de croyance*. Ph. D. thesis, Université de Technologie de Compiègne.
- Shafer, G. (1976). *A mathematical theory of evidence*. Princeton University Press.
- Smets, P. (1993). Belief Functions : the Disjunctive Rule of Combination and the Generalized Bayesian Theorem. *International Journal of Approximate Reasoning* 9, 1–35.

# Etude comparative d'algorithmes filtres pour la sélection de motifs biologiques

Hanen Mhamdi\*, Faouzi Mhamdi\*

\*Laboratoire de Recherche en Technologies de l'Information et de la Communication & Génie Electrique  
ESTT- Université de Tunis, Tunisie  
Mhamdi.hanen@gmail.com, Faouzi.mhamdi@ensi.rnu.tn

**Résumé.** La sélection de motifs biologiques est une étape cruciale dans les systèmes de classification. Elle vise la réduction du nombre de motifs extraits des séquences protéiques, tout en essayant de préserver ou d'améliorer la performance du classifieur utilisé. En effet cette réduction permet d'éliminer les motifs non pertinents et/ou redondants ce qui augmente la capacité de la prédiction des algorithmes de classification, réduire le temps de traitement des données, faciliter leurs visualisation et leurs analyse. Dans ce papier quatre méthodes de type "filtres" pour la sélection de motifs sont comparées. Ces motifs sont utilisés pour la discrimination des séquences biologiques.

## 1 Introduction

La réduction de la dimensionnalité des données est un domaine qui suscite depuis ces dernières années de nombreux travaux de recherche. En effet, ce domaine touche à de nombreux autres domaines, tels que la fouille de données, l'apprentissage automatique, les statistiques, la catégorisation de texte, le traitement d'images, les bases de données et l'analyse de données en bioinformatique. Dans tous ces domaines, les applications nécessitent de traiter des données décrites par un très grand nombre d'attributs. Parmi ces applications pratiques, celles liées à la bioinformatique prennent une place de plus en plus importante. Ce papier est consacré à ce type d'application en traitant particulièrement le problème de sélection de motifs extraits pour la classification des structures primaires. Ce problème représente un véritable centre de préoccupation pour de nombreux chercheurs en bioinformatique.

La sélection de motifs constitue l'une des tâches préliminaire de prétraitement de données dans le processus d'extraction de connaissances à partir des données biologiques (ECDB). Elle consiste à réduire la dimension de l'espace des motifs en éliminant ceux qui sont non pertinents et/ou redondants.

Il existe trois dimensions pour catégoriser les méthodes de sélection. Les stratégies de recherche (complète, séquentielle et aléatoire), les critères d'évaluation (Filtre, Enveloppante, et Hybride) et les tâches de fouille de données (classification, regroupement, biregroupement,...).

Toute méthode de sélection passe généralement par quatre étapes Dash et Liu (1997) : (i) Génération de sous ensemble d'attribut. (ii) Évaluation. (iii) Critère d'arrêt. (iv) Validation des résultats. Durant la première étape, on génère un sous-ensemble de motifs, par la suite

## Algorithmes filtres pour la sélection de motifs

on évalue le sous-ensemble de motifs génère. Enfin si le sous-ensemble généré représente un meilleur sous-ensemble alors on arrête le processus de sélection sinon on revient à l'étape (i).

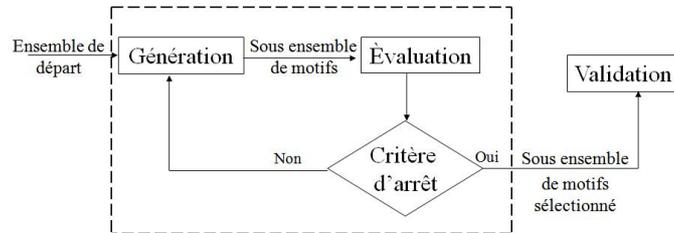


FIG. 1: *Processus général de sélection de motifs.*

Nous nous situons dans le cas où le principal problème est la sélection de motifs pertinents pour la classification des séquences protéiques. L'objectif principal est optimiser le classifieur appris, en fournissant moins de motifs à l'algorithme d'apprentissage on diminue les risques de sur-apprentissage, on entant par sur-apprentissage est le fait que lorsqu'on augmente la complexité d'un classifieur, celui-ci fini par ne plus s'améliorer, au contraire il se dégrade. Ainsi le classifieur appris est un meilleur classifieur au sens où son taux d'erreur sur des données inconnues sera moins important. Les phénomènes de corrélation entre motifs et les risques de sur-apprentissage sont très importants, car on se retrouve confronté à un très grand nombre de motifs pour peu d'exemples Saeys et al.(2007).

Pour que les algorithmes d'apprentissage puissent mieux fonctionner, il est donc nécessaire d'inclure des mécanismes pour sélectionner les motifs. Dans la littérature, il existe trois grandes familles d'algorithme de sélection de motifs : la famille des méthodes filtres, la famille des méthodes enveloppantes et la famille des méthodes hybrides Iñza et al. (2004). Cette distinction est basée sur leur dépendance ou indépendance par rapport à l'algorithme d'induction.

L'approche filtre utilise des critères ad hoc pour sélectionner le meilleur ensemble de motifs, et éliminer les motifs inutiles et redondants sans tenir compte des performances en apprentissage, concernant l'approche enveloppante qui cherche à optimiser le taux d'erreur en utilisant explicitement la méthode d'apprentissage, cette approche donne de meilleurs résultats que les algorithmes filtrants qui sont plus rapides et moins couteux. En effet dans les situations où le nombre de motifs est énorme les méthodes filtres sont indispensables pour obtenir un ensemble optimal de motifs qui par la suite peut être traité par plusieurs autre méthodes de sélection de motifs.

Les deux types d'algorithmes précédents sont duaux, dans le sens où l'avantage du premier type représente l'inconvénient du second et vice versa. C'est pour cette raison, un nouveau paradigme de sélection d'attributs, l'approche hybride, qui réunit ainsi l'avantage de l'approche filtre, i.e., rapidité à celui enveloppante, i.e., meilleurs résultats s'est apparue. En adoptant un algorithme hybride, on opère en deux étapes : Durant la première étape, on utilise un algorithme filtrant pour réduire la taille de l'ensemble de motifs. Puis, durant la seconde étape, on utilise un algorithme enveloppant pour sélectionner le meilleur sous-ensemble à partir de l'ensemble réduit.

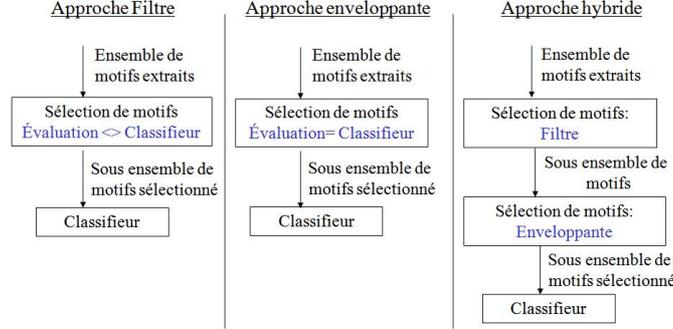


FIG. 2: Comparaison entre les trois approches de sélection de motifs.

## 2 Méthode filtrantes pour la sélection de motifs

Les méthodes de sélection de motifs adoptant une approche filtre, peuvent être groupées en deux catégories suivant le type d'évaluation : **algorithmes de recherche de sous-espace** et **algorithmes de classement**. La première catégorie consiste à identifier un sous-ensemble de nombre minimum de motifs qui satisfont un critère d'évaluation, en effet chaque sous-ensemble candidat est évalué par une certaine mesure d'évaluation et comparé avec le meilleur sous-ensemble obtenu précédemment par rapport à cette mesure.

On peut noter que les mesures d'évaluation utilisées par les algorithmes de recherche de sous-espace prennent en compte l'existence et l'effet de motifs redondants à la différence de la deuxième catégorie qui cherche à évaluer les motifs de manière individuelle (algorithmes de classement), ce qui peut permettre leur classement selon un critère de pertinence afin de choisir les plus discriminants et ainsi former un sous-espace pertinent de motifs.

### 2.1 Algorithme basé sur Chi-2

Le critère statistique indépendant  $\chi^2$ . Hastie et al. (2001) est utilisé pour mesurer l'écart à l'indépendance entre un motif et une famille de chaînes  $f_i$ . Cet écart sert à classer les motifs selon leurs degrés de corrélation avec la famille de protéines à discriminer. La mesure  $\chi^2$  est définie par :

$$\chi^2 = \sum_{i=1}^n \sum_{j=1}^q \frac{\left( n_{ij} - \frac{n_{i*} n_{*j}}{n_{**}} \right)^2}{\frac{n_{i*} n_{*j}}{n_{**}}} \quad (1)$$

Avec  $n$  est le nombre des familles de chaînes,  $q$  est le nombre de motifs,  $n_{i*}$  est la somme marginale des lignes :

$$n_{i*} = \sum_{j=1}^q n_{ij} \quad (2)$$

Algorithmes filtres pour la sélection de motifs

$n_{*j}$  est la somme marginale des colonnes :

$$n_{*j} = \sum_{i=1}^n n_{ij} \quad (3)$$

et

$$n_{**} = \sum_{i=1}^n n_{i*} = \sum_{j=1}^q n_{*j} \quad (4)$$

## 2.2 Algorithme basé sur la Fréquence

Cette méthode est souvent utilisée dans la fouille de texte Y. Xu et al. (2008). Elle correspond à la fréquence d'apparition des motifs extraits dans les différentes structures primaires. On calcule la fréquence pour chaque motif unique dans le corpus d'apprentissage et on enlève de l'espace de motifs ceux dont la fréquence est inférieure à un seuil prédéfini.

## 2.3 Algorithme Relief

L'algorithme Relief est basé sur l'analyse statistique pour sélectionner les motifs pertinents Kira et Rendell (1992), inspiré des algorithmes d'apprentissage à base d'exemple Dash et Liu, 1997. Il essaye indépendamment de tout algorithme d'apprentissage de deviner quels sont les motifs les plus utiles pour établir la tâche de fouille de données qui est la classification des protéines dans notre cas. Son processus de sélection est basé sur une stratégie de recherche séquentielle et a recours à la distance comme un critère d'évaluation. L'idée principale de Relief est d'estimer la pertinence des motifs selon le fait suivant : « à quel point leurs valeurs distinguent les séquences qui sont proches entre eux ? »

C'est un algorithme à processus aléatoire. En effet, il échantillonne les instances de façon aléatoire à partir de l'ensemble de données d'exemples et met à jour les valeurs de pertinences basées sur les différences entre l'instance sélectionnée et les deux plus proches instances de la même famille et d'une famille opposée "near-hit" et "near-miss". A l'origine, il a été réalisé pour l'algorithme d'induction k plus proches voisins. Notons qu'il s'agit d'une méthode nécessitant que les données soient étiquetées.

L'algorithme Relief opère de la manière suivante : Étant donné une séquence aléatoirement choisie  $X_i = \{x_1, \dots, x_n\}$  représentée par un vecteur composé de n valeurs de motif où  $x_j$  est la valeur du motif j de la séquence  $X_i$ . Relief recherche l'ensemble de données pour ses deux voisins les plus proches : un de la même famille, appelé "nearest hit" H, et l'autre d'une famille différente, appelé "nearest miss" M. Par la suite il met à jour le poids  $W[m_j]$  pour tous les motifs  $m_j$  basée sur les valeurs de la fonction de différence  $diff()$  pour X, H et M. Le processus est répété n fois, où p est un paramètre prédéfini par l'utilisateur

Pour les exemples  $X_1, X_2$ , la fonction  $diff(m_j, X_1, X_2)$  calcule la différence entre les valeurs  $(X_1, X_2)$  pour le motif  $m_j$  :

$$diff(m_j, X_1, X_2) = \begin{cases} |X_1 - X_2| & \text{si } m_j \text{ est numérique} \\ 0 & \text{si } m_j \text{ est nominale et } X_1 = X_2 \\ 1 & \text{si } m_j \text{ est nominale et } X_1 \neq X_2 \end{cases} \quad (5)$$

Le poids  $W[m_j]$  est calculé comme suit :

$$W[m_j] = W[m_j] - \frac{diff(m_j, X_i, H)}{p} + \frac{diff(m_j, X_i, M)}{p} \quad (6)$$

## 2.4 Correlation-Based feature selection :CFS

Correlation based Feature Selection (CFS) Hall (2000) est un simple algorithme filtrant qui permet de prendre en considération la dépendance d'un motif à l'ensemble des motifs déjà sélectionnés, en plus de la dépendance de ce motif à la famille à discriminer. Il consiste à trier les motifs selon une fonction d'évaluation heuristique *Merit*. La caractéristique principale de cette fonction est qu'elle à une grande valeur pour les sous-ensembles de motifs qui possèdent une forte corrélation motifs-familles et une faible valeur pour les sous-ensembles qui possèdent une faible corrélation motif-motif. Cette fonction permet de comparer des sous-ensembles de tailles différentes, et ainsi, elle permet d'évaluer la contribution d'un nouvel motif.

$$Merit(S) = \frac{k\bar{r}_{yx}}{\sqrt{k + k(k-1)\bar{r}_{xx}}} \quad (7)$$

Avec  $S$  est un sous ensemble de motifs,  $K=|S|$ ,  $\bar{r}_{yx}$  est la moyenne des corrélations motifs-familles ( $x \in S$ ) et  $\bar{r}_{xx}$  est la moyenne des corrélations motif-motif.

L'algorithme CFS opère de la manière suivante : Durant la première étape, on discrétise les motifs Hall (2000) en utilisant la technique supervisée Fayyad et Irani (1993). Ensuite, on construit une matrice de corrélation motifs-familles et motifs-motifs, en utilisant une des mesures de qualité de motifs pour estimer la corrélation décrite dans l'équation *Merit* à savoir l'incertitude symétrique *SU* Press et al. (1998), *MDL* (the minimum description length) ou *Relief* Kira et Rendell, (1992). Et, enfin, on explore et on évalue les sous-ensembles de motifs en faisant une recherche en avant (best-first search) Kohavi et John (1997). En adoptant cette méthode, on opère de la manière suivante : (i) D'abord, on génère les différents sous-ensembles composés d'un seul motif et on les évalue via la fonction *Merit* Ghiselli. (1964). (ii) Puis, on génère les sous-ensembles composés de deux motifs en commençant par le sous-ensemble qui maximise la fonction *Merit* dans l'étape (i). (iii) Enfin, on reitère l'étape (ii) jusqu'à ce que la fonction *Merit* régresse après un certain nombre extensions successives complètes. Le sous-ensemble de motifs ainsi obtenu représente l'ensemble des motifs sélectionnés.

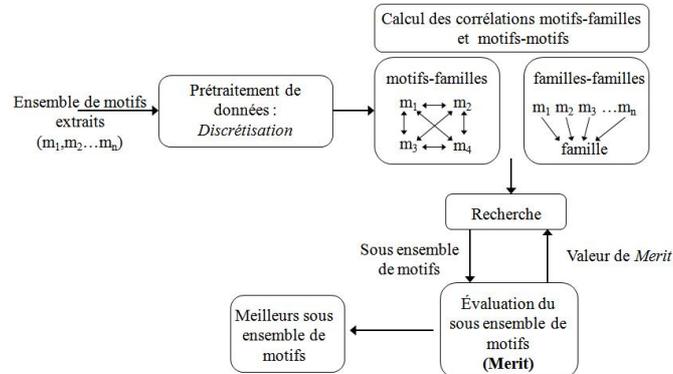


FIG. 3: Principe de l'algorithme CFS.

### 3 Présentation des données biologiques utilisées

Les données biologiques sont rassemblées dans des banques de données publiées sur internet, elles sont principalement représentées par des séquences d'ADN, d'ARN, de protéines,... Notre objectif est de trouver l'algorithme filtrant pour la sélection de motifs le mieux adapté pour minimiser le nombre de motifs extraits. Le sous ensemble de motifs sélectionné permet la mise en place d'un système de classification qui à partir de la description d'une séquence de protéine, lui affecte automatiquement sa famille d'appartenance. Or, l'utilisation des données sous leur forme native n'est pas possible, les méthodes de fouille de données ne peuvent pas être appliquées sur des séquences biologiques sous leur forme primaire.

Il est donc nécessaire de transformer ces données en tableau individus/motifs,  $T$ , avec, une ligne  $i$  représente une structure de protéines, une colonne  $j$  représente un descripteur extrait(motif), et la case  $T[i,j]$  représente alors la pondération du  $j^{me}$  motifs dans le  $i^{me}$  structure primaire (séquence), cette valeur indique l'importance du motif dans une séquence. La dernière colonne de  $T$  représente les numéros des familles auxquelles appartiennent les différentes structures primaires.

En fait, il existe plusieurs méthodes de pondération de motifs, vu les études réalisées au niveau de la classification automatique des textes, nous avons testé quatre types de pondération qui : Sebastiani (2005)

- Booléenne : indique si un motif extrait est présent ou non au sein d'une séquence, si  $T[i,j]=1$  alors le  $j^{me}$  motif est présent dans le  $i^{me}$  séquence sinon il est absent.
- Occurrences : nombre d'occurrence d'un motif dans une séquence,  $T[i,j]=k, k \geq 0$ , alors le  $j^{me}$  motif est présent  $k$  fois dans le  $i^{me}$  séquence.
- Fréquence : fréquence relative d'un motif par rapport au nombre de motifs composant une séquence.
- TF\*IDF : corrige la fréquence de motif en fonction de sa fréquence au sein du fichier.

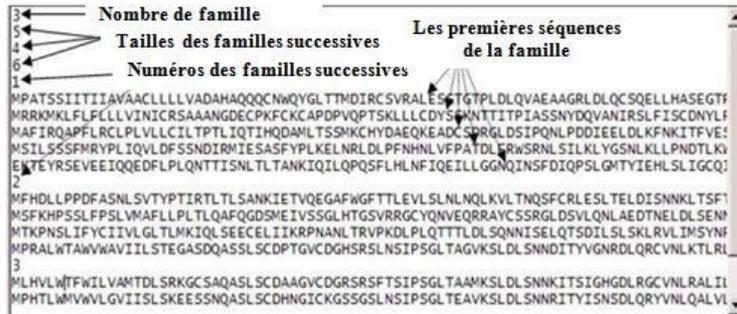


FIG. 4: Fichier de familles de protéines

Ensemble de séquences	Motif extrait												
	NPA	PAT	RES	TSS	SSI	SLI	ITI	III	IIA	IIV	AVA	VAA	AAC
Seq1	1	1	1	1	1	1	1	1	1	1	1	1	1
Seq2	0	0	0	0	0	0	1	1	0	0	1	1	0
Seq3	0	1	0	1	0	0	1	0	1	0	0	0	0
Seq4	0	0	0	0	1	0	0	0	0	0	0	0	0
Seq5	0	1	0	0	0	1	0	0	1	1	0	0	0
Seq6	0	0	0	0	0	0	0	0	0	0	0	0	0
Seq7	0	0	0	0	0	0	0	0	0	0	0	0	1
Seq8	0	0	0	1	0	0	0	0	0	0	0	0	0

FIG. 5: Fichier d'apprentissage booléen

La figure 5 illustre un exemple de tableau booléen construit après l'extraction de motifs. Dans ce travail nous nous situons dans le cas où l'extraction de motifs a été déjà faite via un algorithme hors ligne et où il faudrait faire une sélection parmi les motifs extraits. À partir de la représentation du tableau booléen construit après l'extraction des motifs, illustré dans la figure 4, nous constatons la pertinence de l'analogie entre la représentation des séquences et celle des données textuelles. En effet, une séquence de protéines est une suite d'acides aminés, il y a 20 acides aminés possibles : une protéine est donc décrite par une chaîne de caractères représentant des acides aminés. Nous montrons dans la figure 4, un exemple de fichier décrivant quelques protéines. Cependant, à la différence de la fouille de texte, il n'existe pas de séparation naturelle dans les séquences de caractères, il n'est donc pas possible de mettre en exergue des "mots" auxquelles nous pourrions rattacher aisément une sémantique.

Nous avons donc recours aux techniques d'extraction des k-grammes, des suites de caractères, pour produire les descripteurs discriminants. L'algorithme de construction de k-grammes est appliqué sur plusieurs familles de structures primaires de protéines. Ces familles sont extraites à partir de la banque de protéines SCOP [Murzin et al. 95]. Les expérimentations réalisées au niveau de cette étape montre que les k-grammes de taille 3 présentent un bon compromis sous ensemble de motifs de taille réduit avec une capacité de discrimination élevée Mhamdi et ELLoumi. (2005).

Algorithmes filtres pour la sélection de motifs

	Nombre de séquences	Longueur Moyenne	Taille Totale
$f_1$	40	884	35382
$f_2$	47	667	31363
$f_3$	54	601	32480
$f_4$	81	505	40907
$f_5$	68	462	31473

TAB. 1: Familles de structures primaires de protéines

## 4 Expérimentation et résultats

Nous présentons d'abord les résultats obtenus avec les différents types de codage des k-grammes détaillés dans la section précédente afin de déterminer lequel représente un bon compromis pour notre problématique.

Famille de protéine	Fichier booléen	Fichier occurrence	Fichier Fréquence	Fichier TF*IDF
$f_1 \cup f_2$	0.007	0.0256	0.04656	0.007
$f_1 \cup f_3$	0.0553	0.0830	0.1170	0.0553
$f_1 \cup f_4$	0.0217	0.05	0.0750	0.0217
$f_1 \cup f_5$	0.0593	0.0852	0.1	0.0593
$f_2 \cup f_3$	0.0420	0.044	0.1760	0.0420
$f_2 \cup f_4$	0.0125	0.0313	0.0578	0.0109
$f_2 \cup f_5$	0.0316	0.0333	0.0860	0.0316
$f_3 \cup f_4$	0.0731	0.109	0.1866	0.0731
$f_3 \cup f_5$	0.0475	0.0836	0.1131	0.0459
$f_4 \cup f_5$	0.0581	0.1068	0.1689	0.0581

TAB. 2: Taux d'erreur de classification obtenus avec les quatre type de pondération.

En analysant les résultats illustrés dans le tableau 2, nous constatons clairement que la pondération booléenne et TF\*IDF semblent mieux adaptées et performantes. Pour des raisons technique on peu privilégié la représentation booléenne pour différentes raisons : stockage en mé moire de données, elle convient autant aux technique utilisant des variables continues (exemple PPV) que celle de variables discrètes.

Nous avons opté pour une stratégie de discrimination entre familles de protéines : Les familles seront discriminées deux-à-deux. Ensuite, on extrait les motifs à partir des chaînes de chaque couple de familles et on construit les tableaux de données T correspondants.

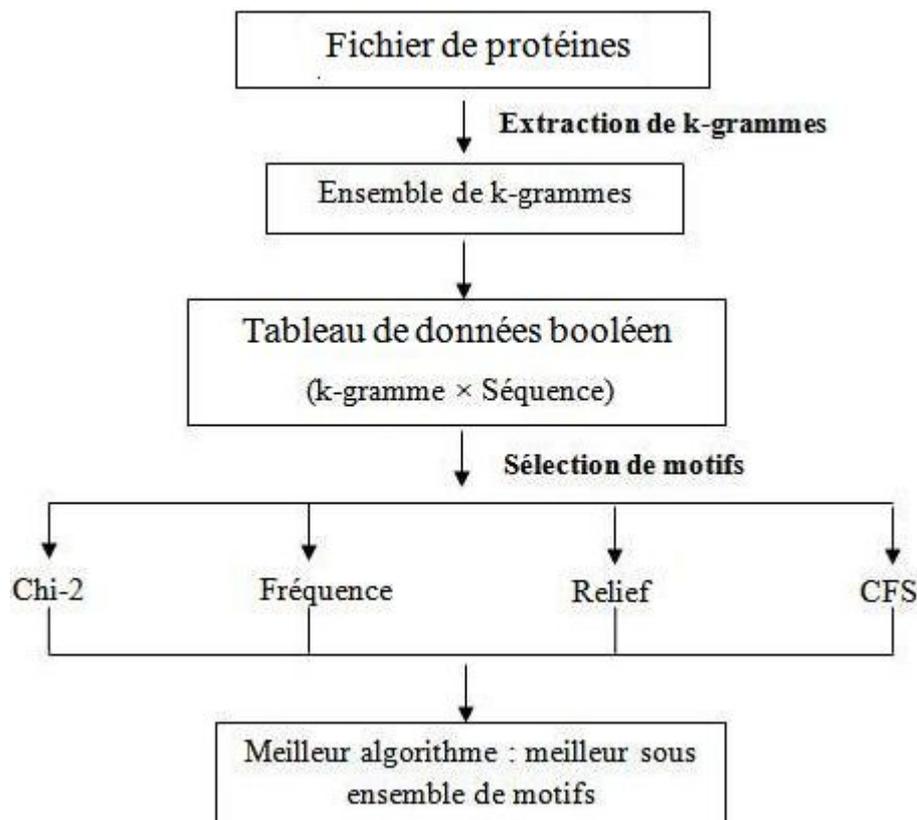


FIG. 6: Stratégie de sélection de motifs en adoptant une approche filtre.

Afin d'évaluer les sous ensembles de motifs sélectionnés avec ces algorithmes filtrants nous avons recours à une stratégie d'évaluation par classification qui repose sur le taux d'erreurs de classification comme mesure de performance. Cette stratégie consiste à utiliser les motifs sélectionnés pour essayer de classifier des structures primaires de macromolécules biologiques.

Nous avons choisi comme classifieur le k-Plus Proches Voisins (k-PPV) qui permet, à partir des données d'apprentissage dont on connaît le label, de classifier les données test. Afin d'être le plus simple et indépendant des règles de décision sur le vote majoritaire en cas d'égalité de votes, nous avons choisi de fixer  $k$  à 1. À chaque donnée test est alors attribuée la même classe que celle de la donnée d'apprentissage qui lui est la plus proche dans l'espace d'attributs considéré et cela en utilisant la distance Euclidienne.

Pour évaluer les taux d'erreur de classification de chaque ensemble de motifs, nous avons utilisé la méthode de validation croisée 5\*2 [Kohavi 95]. Cette méthode est basée sur le partitionnement des données en deux sous-ensembles complémentaires. Le modèle est appris sur un sous-ensemble dénommé ensemble d'apprentissage et la validation et l'analyse des prédictions opérées sur son complémentaire, dénommé ensemble de test. Le résultat final est obtenu

## Algorithmes filtres pour la sélection de motifs

Familles de Protéines	Avant la sélection		Après la sélection							
			Chi-2		Fréquence		Relief		CFS	
	Nb motifs	TEr	Nb motifs	TEr	Nb motifs	TEr	Nb motifs	TEr	Nb motifs	TEr
$f_1 \cup f_2$	6600	0.2163	180	0.0605	117	0.0115	1072	0.1116	44	0.0837
$f_1 \cup f_3$	6288	0.2043	189	0.1702	186	0.1064	1392	0.1851	42	0.1340
$f_1 \cup f_4$	6183	0.1917	271	0.0550	193	0.0496	1129	0.1	3	0.0317
$f_1 \cup f_5$	6004	0.2463	173	0.0852	132	0.0741	869	0.1130	38	0.1167
$f_2 \cup f_3$	7143	0.2060	278	0.0420	217	0.0495	1535	0.124	49	0.0380
$f_2 \cup f_4$	7098	0.1281	506	0.1141	213	0.0391	1098	0.1062	32	0.0500
$f_2 \cup f_5$	7011	0.1649	360	0.0632	156	0.0609	1208	0.1193	39	0.0474
$f_3 \cup f_4$	6851	0.2701	391	0.1060	277	0.1061	1239	0.1567	12	0.0746
$f_3 \cup f_5$	6740	0.3311	328	0.1052	204	0.1008	1151	0.1295	28	0.0803
$f_4 \cup f_5$	6659	0.1689	424	0.0946	213	0.0769	1208	0.1284	6	0.0662

TAB. 3: comparaison entre le nombre de motifs et le taux d'erreur de classification des structures primaires.

par la moyenne des différentes validations qui sont 5 validations dans notre cas .

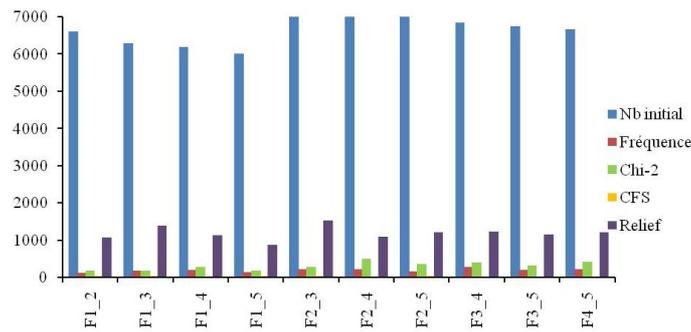


FIG. 7: Comparaison entre le nombre de motifs extraits et le nombre sélectionné avec les différents algorithmes.

Les résultats illustrés dans le tableau Tab3 montre que l'approche CFS atteint des meilleurs résultats par rapport aux autres approches. Elle fournit des sous ensembles de motifs de petites tailles et qui ont une meilleure capacité de discrimination des familles de protéines. La performance de cette approche peut être expliquée par le fait qu'elle prend en compte lors de la procédure de sélection la corrélation motifs-motifs ainsi que la corrélation motifs-famille.

Famille de protéine	Sous ensemble de motifs sélectionnée (CFS)
$f_1 \cup f_4$	{GGS,VGG,IGG}
$f_4 \cup f_5$	{GIG,GGS,VGG,FDQ,KTH,IGG}

TAB. 4: Exemple de Sous ensemble de motifs sélectionné avec CFS.

	Seq0	Seq1	Seq2	Seq3	Seq4	Seq5	Seq6	Seq7	Seq8	Seq9	Seq10	Seq41	Seq42	Seq43	Seq44	Seq45	Seq46	Seq47	Seq48	Seq49	Seq50
GGS	0	0	1	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1
VGS	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	1	1	1	1	1
IGG	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	1	1	1	1	1

FIG. 8: Valeurs des motifs sélectionnés avec CFS.

Cette figure illustre les valeurs des motifs dans les dix première séquences de la famille  $F_1$  et les dix première de la famille  $F_4$ . En analysant ces valeurs nous constatons que ces motif existent(prennent la valeur 1)dans la deuxième famille alors qu'ils n'existent presque pas dans la première familles,ce qui expliquent ces pouvoirs discriminants entre les familles des protéines.

## 5 Conclusion

Cet article présente, une étude comparative entre quatre algorithmes adoptants une approche filtre pour la sélection de motifs : La première méthode de sélection est fondée sur le calcul des corrélations motifs-familles, en utilisant le critère statistique chi-2. La seconde méthode est basée sur le critère statistique, fréquence. La troisième basée sur l'algorithme filtrant Relief.Enfin une méthode qui permet de traiter la problématique de la redondance des motifs sélectionnés. Cette méthode est basée sur l'algorithme filtrant Correlation-based Feature Selection (CFS). Ces algorithmes ont été évalués sur des séquences réelles de protéines issues de banques de données accessibles sur internet. Nous avons opté les k-grammes de taille 3 et la pondération booléenne (présence/absence). Ces méthodes sont restreintes sur les problèmes de discrimination entre deux familles de protéines.

Dans le cadre des expérimentations, l'étude comparative réalisée entre ces divers algorithmes nous permet de conclure que l'algorithme CFS atteint un bon équilibre entre le taux de classification et le nombre de motifs sélectionnés pour tous les jeux de données

## Référence

- Fayyad, Irani (1993). Multi-Interval Discretisation of Continuous-Valued Attributes. Proc. 30th International Joint Conference on Artificial Intelligence, Morgan Kaufmann (Publish.), p 1022-1027.
- M. Dash et H. Liu(1997). Feature Selection Methods for Classifications. Intelligent Data Analysis : An International Journal, Vol.1, N°3.
- M. Hall (2000). Correlation Based Feature Selection for Discrete and Numeric Class Machine Learning. The 17th International conference on machine learning, CML'00, Morgan kaufman (Publish.), (Berlin, Germany).
- A. Murzin, S. Brenner, T. Hubbard, C. Chothia, Scop(1995). A Structural Classification of Proteins Database for the Investigation of Sequences and Structures. Journal of Molecular Biology, p536-540.
- T. Hastie, R. Tibshirani, J. Friedman (2001). The Elements of Statistical Learning, Springer Verlag (Germany).

## Algorithmes filtres pour la sélection de motifs

- I. Inza, P. Larranaga, R. Blanco, J. Cerrolaza (2004). Filter versus wrapper gene selection approaches in DNA microarray domains. *Journal of Elsevier health*.
- K. Kira and L. Rendell (1992). A practical approach to feature selection. *Proceedings of the Ninth International Conference on Machine Learning*. San Francisco, Morgan Kaufmann, CA.
- R. Kohavi, G. H. John (1997). Wrappers for Feature Subset Selection. *Journal of Artificial Intelligence*, Vol. 97, N°1-2, p273-324.
- F. Mhamdi, M. Elloumi (2005). Extraction et Comparaison entre N-Grammes et Descripteurs Discriminants pour la Classification de Protéines, *Conférence d'Apprentissage Automatique*. (Nice, France).
- W. H. Press, S. A. Teukolsky, W. T. Vetterling, B. P. Flannery. (1988). *Numerical Recipes in C*, Cambridge University Press.
- Y. Saeys, I. Inza et P. A. Larran (2007). Review of feature selection techniques in bioinformatics. *The Bioinformatics oxford journals*, 2507-2517.
- Yan Xu, Bin Wang, JinTao Li, et Hongfang Jing (2008). An Extended Document Frequency Metric for Feature Selection in Text Categorization. *AIRS : Springer-Verlag Berlin Heidelberg*, p71-82.

## Summary

Motif selection from biological sequences is an important step in classification systems. It aims at reducing the number of motifs while trying to preserve or improve classifier performance. Motif selection has been very effective in reducing dimensionality, removing irrelevant motifs, increasing learning accuracy, and improving result comprehensibility. In this paper we compare four filter algorithms for motif selection. The subset of motifs selected is used to discriminate biological sequences.

# Une plateforme collaborative pour la recherche de consensus ou d'alternatives en classification non-supervisée

Jacques-Henri Sublemontier\*

\*LIFO - Université d'Orléans, ENSI de Bourges  
Bâtiment IIIA, rue Léonard de Vinci  
F-45067 ORLEANS Cedex 2  
jhs@univ-orleans.fr,

<http://www.univ-orleans.fr/lifo/Members/sublemontier/>

**Résumé.** Dans cet article nous proposons un environnement collaboratif pour résoudre les problématiques de *clustering* multi-vues et de *clustering* alternatifs en invoquant des principes issus du *clustering* d'ensemble et du *clustering* semi-supervisé. Nous proposons de contrôler, via un partage d'informations adapté entre différents algorithmes de *clustering*, la recherche d'une partition consensus ou d'un ensemble de partitions alternatives. L'intérêt de l'approche proposée est qu'il n'est pas nécessaire de connaître la nature de chaque algorithme employé et par extension leur paramétrage.

## 1 Introduction

Nous proposons dans cet article une plateforme unificatrice répondant aux problèmes des *clusterings* multi-vues et alternatif et basée sur du *clustering* collaboratif.

Le ***clustering multi-vues*** propose de déterminer une partition d'un ensemble d'individus dits multi-représentés *i.e.* pour lesquels chaque individu est décrit par plusieurs ensembles de variables (les vues des données). Cette problématique répond pleinement au problème classique du *clustering* pour lequel on cherche à intégrer plusieurs informations portant sur le même ensemble d'individus, par exemple, dans le contexte de la génomique, du marketing, et de la classification automatique de documents webs. Le *clustering* multi-vues a été introduit il y a quelques années, et se formalise traditionnellement par la recherche d'une partition réalisant un consensus entre les différentes partitions locales, adaptées à chacune des vues (Pedrycz, 2002; Bickel et Scheffer, 2005; Sublemontier et al., 2011a).

Le ***clustering alternatif*** propose de trouver un ensemble de partitions dissemblables et de bonne qualité étant donné un ensemble d'individus mono-représentés. Cette problématique est très récente et répond à un cadre particulier orienté utilisateur, lorsque celui-ci est confronté à une forte volumétrie de variables pour décrire ses individus. Ce problème de dimensionnalité induit naturellement la possibilité de trouver plusieurs solutions de *clusterings* intéressantes et dissimilaires, dans différents sous-espaces des données d'origine. Le *clustering* alternatif peut se résoudre *via* des techniques empruntées à l'apprentissage de distance et au *clustering* semi-supervisé (Davidson et Qi, 2008; Dang et Bailey, 2010).

Le *clustering* multi-vues généralise en un sens la problématique du *clustering* d'ensemble, qui vise à déterminer une partition d'un ensemble d'individus réalisant un consensus sur la base d'un ensemble ou profil de partitions (Strehl et Ghosh, 2003). En effet les différentes partitions du profil peuvent être obtenues par application de différents algorithmes de *clustering* sur chaque vue des données.

Nous proposons alors une plateforme collaborative au sein de laquelle plusieurs algorithmes de *clustering*, appliqués sur différentes vues des données, s'échangent des informations pour tendre vers la recherche d'une solution consensus (objectif du *clustering* multi-vues) ou la recherche de solutions différentes (objectif du *clustering* alternatif). L'intérêt de l'approche proposée est qu'il n'est pas nécessaire de connaître la nature de chaque algorithme employé et par extension leur paramétrage. Pour cela, notre plateforme se fonde sur une approche récente de *clustering* semi-supervisé permettant d'améliorer les performances de n'importe quel algorithme de *clustering* en ne faisant aucune hypothèse sur les propriétés de celui-ci (Sublemontier et al., 2011b). Nous proposons alors un mécanisme de collaboration fondé sur un partage d'informations permettant de guider chaque algorithme par des connaissances produites par l'ensemble des autres algorithmes de *clustering*. Dans un premier temps nous rappelons le contexte du *clustering* semi-supervisé ainsi que l'approche que nous étendons au contexte de la multiplicité. Dans un second temps nous établissons les principes de collaboration entre algorithmes de *clusterings* que nous préconisons pour atteindre chaque objectif. Enfin, nous concluons sur les premières observations expérimentales sur des données *benchmark* selon des évaluations internes et externes.

## 2 Notation

NOTATION	
$n$	: le nombre d'individus à regrouper.
$n_p^{(r)}$	: le nombre d'attributs décrivant les individus dans la vue $r$ .
$n_k$	: le nombre de groupes à identifier.
$n_c$	: le nombre de classes associé aux données.
$\mathcal{X} = \{x_1, \dots, x_n\}$	: l'ensemble des $n$ individus à partitionner.
$X^{(r)} \in \mathbb{R}^{n \times n_p^{(r)}}$	: la représentation matricielle de $\mathcal{X}$ dans la vue $r$ .
$x_i^{(r)} \in \mathbb{R}^{n_p^{(r)}}$	: la représentation vectorielle de l'individu $x_i$ dans la vue $r$ .
$C = \{C_1, \dots, C_{n_k}\}$	: la structure de <i>clustering</i> en $n_k$ groupes à construire.
$\Pi = \{C^{(1)}, \dots, C^{(n_r)}\}$	: l'ensemble des $n_r$ <i>clusterings</i> locaux dans chaque vue.
$C^{(r)} = \{C_1^{(r)}, \dots, C_{n_k}^{(r)}\}$	: l'ensemble des $n_k$ groupes du <i>clustering</i> dans la vue $r$ .
$d_{(r)}(x_i, x_j)$	: la distance au sens général entre deux individus $x_i$ et $x_j$ dans $r$ .
$d_{P^{(r)}}(x_i, x_j)$	: la distance entre $x_i$ et $x_j$ dans un sous-espace $P$ de $r$ .
$\mathcal{ML}^{(r)}$	: l'ensemble des $(x_i, x_j) \in \mathcal{X}^2$ devant être regroupés dans $r$ .
$\mathcal{CL}^{(r)}$	: les $(x_i, x_j) \in \mathcal{X}^2$ devant être séparés dans $r$ .
$A^{(r)}$	: l'algorithme de <i>clustering</i> employé pour obtenir $C^{(r)}$ .
$Link^{(r)}(x_i, x_j)$	: $x_i$ et $x_j$ sont regroupés par $A^{(r)}$ ou dans $C^{(r)}$ .
$\overline{Link}^{(r)}(x_i, x_j)$	: $x_i$ et $x_j$ sont séparés par $A^{(r)}$ ou dans $C^{(r)}$ .
$H^{(r)} \in \{0, 1\}^{n \times n}$	: la matrice de <i>clustering</i> associée à $C^{(r)}$ .

## 3 Clustering guidé par des connaissances

### 3.1 Généralités

Le *clustering* guidé par des connaissances ou *clustering* semi-supervisé, vise à déterminer une partition de l'ensemble d'individus, de sorte à respecter des connaissances exprimées sous forme de contraintes sur certaines paires d'individus. Les contraintes peuvent-être de la forme :

- *must-link* ( $\mathcal{ML}$ ) : si  $(x_i, x_j) \in \mathcal{ML}$  alors  $x_i$  et  $x_j$  doivent être regroupés
- *cannot-link* ( $\mathcal{CL}$ ) : si  $(x_i, x_j) \in \mathcal{CL}$  alors  $x_i$  et  $x_j$  doivent être séparés

Parmi les approches proposées pour résoudre ce problème, nous nous intéressons aux développements récents consistant à apprendre itérativement un espace de représentation des données adapté pour renforcer la performance de l'algorithme de *clustering* (considéré comme une boîte noire) sur la satisfaction des contraintes (Liu et al., 2007; Sublemontier et al., 2011b).

### 3.2 L'approche ADAUZABOC

L'approche ADAUZABOC propose d'apprendre simultanément (1) une métrique permettant de rendre plus proches les individus liés par une contrainte  $\mathcal{ML}$  et tenant éloignés les individus liés par une contrainte  $\mathcal{CL}$ , (2) la représentation (dans un espace de faible dimension) des individus selon cette métrique et (3) la partition obtenue par  $A$  à partir de cette représentation. Plus formellement, les auteurs posent le problème d'intégration des connaissances pour le *clustering* comme un problème d'apprentissage de représentation  $X^*$  paramétré par une variable exogène ( $\xi$ ) permettant d'introduire en ligne, dans le processus d'optimisation, le *feed-back* de  $A$  sur ses performances en classification selon la solution courante de la représentation :

$$\mathcal{P}_{\text{UZABOC}}(\xi) \left\{ \begin{array}{l} \max_P \quad \text{trace}(P^\top X^\top X P) \\ \text{s.c.} \quad P^\top P = Id_s \\ (cs1) \quad d_P^2(x_i, x_j) \leq \xi_{ij} \quad \forall (x_i, x_j) \in \mathcal{ML} \\ (cs2) \quad d_P^2(x_i, x_j) \geq \xi_{ij} \quad \forall (x_i, x_j) \in \mathcal{CL} \end{array} \right. \quad (1)$$

Une nouvelle représentation s'obtient alors par projection de  $X$  sur  $P$  :  $X^* = X P^*$ . Le problème d'optimisation présenté est alors résolu de manière approchée par une méthode itérative de type primal-dual *Uzawa*. Le paramètre  $\xi$  correspondant à une variable exogène est modifié à chaque itération de la résolution du problème, après obtention de la solution courante  $P'$ , de sorte à réduire systématiquement l'espace des solutions réalisables. La modification de  $\xi$  est réalisée afin de rendre les contraintes d'optimisation (*cs1*) et (*cs2*) plus difficiles à satisfaire si les connaissances correspondantes ne sont pas satisfaites par  $A$  appliqué sur la représentation  $X' = X P'$ . Les multiplicateurs de lagrange permettent de forcer automatiquement la recherche d'un sous-espace conforme aux connaissances, lorsque les valeurs de  $\xi$  sont apprises. Cette approche permet de converger vers un sous-espace  $P^*$  permettant l'apprentissage d'une représentation optimale  $X^*$  permettant le respect des contraintes  $\mathcal{ML}$  et  $\mathcal{CL}$  par  $A$ .

## 4 Plateforme collaborative non supervisée

### 4.1 Généralités

L'approche proposée s'inspire des méthodes de *clustering* d'ensembles et de *clustering* collaboratif CLUSTERING ENSEMBLE (Strehl et Ghosh, 2003) et SAMARAH (Wemert et al., 2000). Pour répondre à la problématique du *clustering* multi-vues, COBOC repose sur l'approche ADAUZABOC présentée précédemment. L'idée est d'appliquer sur chaque vue le *meta*-algorithme ADAUZABOC, ce qui permet d'utiliser n'importe quel objectif de *clustering* sous-jacent. La réponse au problème du *clustering* multi-vues ou du *clustering* alternatif est réalisée au moyen de la génération incrémentale d'un ensemble de contraintes que devront respecter au mieux l'ensemble des algorithmes de *clusterings* locaux. Ce mécanisme de génération incrémentale de contraintes est tiré d'un principe ayant fait ses preuves en apprentissage semi-supervisé : le co-apprentissage (Blum et Mitchell, 1998).

**Construction incrémentale de contraintes.** Le mécanisme de construction incrémentale des contraintes consiste à générer à chaque étape un ensemble de contraintes parmi les plus «évidentes», *i.e.* associées à une plus grande confiance selon partitions produites par les différents algorithmes de *clustering* de la plateforme. Les nouvelles contraintes sélectionnées à chaque étape sont ajoutées à l'ensemble des contraintes déjà considérées. Un ensemble final de connaissances  $\mathcal{ML}$  ou  $\mathcal{CL}$  jugées satisfaisantes sert alors de guide aux algorithmes de *clusterings* locaux qui cherchent une partition de  $\mathcal{X}$  dans un contexte alors semi-supervisé, guidés par ces connaissances. Les différents objectifs visés mènent à deux variantes de ce mécanisme de co-apprentissage pour le *clustering* dans un cadre de multiplicité : (1) **COBOC**, pour guider les algorithmes de *clustering* vers la production de partitions locales proches ; et (2) **ALTERBOC**, pour guider les algorithmes de *clustering* vers la production de partitions locales différentes. La différenciation entre les deux approches COBOC et ALTERBOC se fait alors *via* la génération des contraintes.

L'idée est alors de partir des ensembles  $\mathcal{ML}^{(r)} = \mathcal{CL}^{(r)} = \emptyset$  et d'alterner deux étapes qui sont :

- la recherche des *clusterings* optimaux locaux selon ADAUZABOC, pour  $\mathcal{ML}^{(r)}$  et  $\mathcal{CL}^{(r)}$  fixés ;
- l'augmentation de  $\mathcal{ML}^{(r)}$  et  $\mathcal{CL}^{(r)}$  selon les *clusterings* locaux obtenus.

La recherche d'un *clustering* local optimal étant indépendante de la recherche des *clusterings* dans les autres vues, cette étape est réalisée indépendamment dans chaque vue et correspond exactement à l'algorithme ADAUZABOC.

La seconde étape consiste à augmenter  $\mathcal{ML}^{(r)}$  et  $\mathcal{CL}^{(r)}$ . Cette augmentation est réalisée en sélectionnant à partir de l'ensemble des paires d'individus non présentes dans les contraintes, un ensemble de  $m$  paires candidates pour chaque *clustering* local. Les paires candidates sont associées à une confiance indiquant leur prédisposition à être une contrainte  $\mathcal{ML}$  ou  $\mathcal{CL}$ .

Soit  $H^{+(r)}$  et  $H^{-(r)}$  les matrices des hypothèses de *clustering* dans la vue  $r$  définies par :

$$H^{+(r)}_{ij} = \begin{cases} 1 & \text{si } i \neq j \text{ et } \text{Link}^{(r)}(x_i, x_j) \\ 0 & \text{sinon} \end{cases}$$

$$H^{-(r)}_{ij} = \begin{cases} -1 & \text{si } i \neq j \text{ et } \overline{\text{Link}}^{(r)}(x_i, x_j) \\ 0 & \text{sinon} \end{cases}$$

La matrice complète des hypothèses de *clustering* est alors donnée par :

$$H^{(r)} = H^{+(r)} + Id_n + H^{-(r)} \quad (2)$$

En particulier les paires d'individus  $(x_i, x_i) \in \mathcal{X}$  sont toujours classées ensemble par  $A^{(r)}$ , ainsi  $H^{(r)}_{ii} = 1$ .

Soit  $D^{(r)}$  la matrice des distances entre individus dans le dernier sous-espace optimal de la vue  $r$  définie par :

$$D^{(r)}_{ij} = d_{P^{(r)*}}^2(x_i, x_j) \quad (3)$$

On pose  $D_{max}^{(r)}$  et  $D_{min}^{(r)}$  tels que :

$$D_{max}^{(r)} = \begin{cases} \max_{(x_i, x_j) \in \mathcal{X}^2} \left( H_{ij}^{+(r)} D_{ij}^{(r)} \right) & \text{si } i \neq j \text{ et } Link^{(r)}(x_i, x_j) \\ \max_{(x_i, x_j) \in \mathcal{X}^2} \left( H_{ij}^{- (r)} D_{ij}^{(r)} \right) & \text{si } i \neq j \text{ et } \overline{Link}^{(r)}(x_i, x_j) \end{cases}$$

$$D_{min}^{(r)} = \begin{cases} \min_{(x_i, x_j) \in \mathcal{X}^2} \left( H_{ij}^{+(r)} D_{ij}^{(r)} \right) & \text{si } i \neq j \text{ et } Link^{(r)}(x_i, x_j) \\ \min_{(x_i, x_j) \in \mathcal{X}^2} \left( H_{ij}^{- (r)} D_{ij}^{(r)} \right) & \text{si } i \neq j \text{ et } \overline{Link}^{(r)}(x_i, x_j) \end{cases}$$

On pose  $\alpha_{ij}^{(r)}$  la confiance associée à la paire  $(x_i, x_j)$  dans la vue  $r$  qui s'exprime par :

$$\alpha_{ij}^{(r)} = \frac{H_{ij}^{(r)} (D_{max}^{(r)} - D_{ij}^{(r)})}{D_{max}^{(r)} - D_{min}^{(r)}} \quad (4)$$

Cette confiance est à la base des différentes déclinaisons de COBOC et ALTERBOC. Les hypothèses considérées pour les approches proposées sont alors les suivantes :

- plus une confiance  $\alpha_{ij}^{(r)} > 0$  est élevée, plus on a la certitude d'avoir  $Link^{(r)}(x_i, x_j)$  ;
- plus une confiance  $\alpha_{ij}^{(r)} < 0$  est faible, plus on a la certitude d'avoir  $\overline{Link}^{(r)}(x_i, x_j)$ .

En raisonnant en terme de distance, et non en terme de confiance, ces hypothèses reflètent les résultats obtenus à l'issue des travaux sur ADAUZABOC.

L'idée des différentes variantes de la plateforme collaborative est de calculer pour chaque vue (ou algorithme ou alternative) une nouvelle confiance  $\tilde{\alpha}_{ij}^{(r)}$  pour chaque paire d'individus. Ces nouvelles valeurs de confiances permettent d'observer pour chaque paire d'individus, leur tendance globale à être regroupés ou séparés. Ces observations nous permettent de décider pour chaque paire si elle est candidate à être une connaissance  $\mathcal{ML}^{(r)}$  ou une connaissance  $\mathcal{CL}^{(r)}$ . Nous noterons  $\zeta_{\mathcal{ML}}^{(r)}$  (resp.  $\zeta_{\mathcal{CL}}^{(r)}$ ) la liste ordonnée par les confiances  $\tilde{\alpha}_{ij}^{(r)} \geq 0$  (resp.  $< 0$ ) de candidats pour être  $\mathcal{ML}^{(r)}$  (resp.  $\mathcal{CL}^{(r)}$ ). Une stratégie  $\Gamma$  permet alors de sélectionner à chaque étape un ensemble de connaissances parmi les ensembles candidats. Pour toutes les variantes proposées, nous proposons d'employer les stratégies de sélection :

- aléatoire  $\Gamma_{Random}$  qui consiste à tirer  $m^+$  et  $m^-$  paires d'individus aléatoirement parmi  $\zeta_{\mathcal{ML}}^{(r)}$  et  $\zeta_{\mathcal{CL}}^{(r)}$  respectivement ;
- confiante  $\Gamma_{Max}$  qui consiste à sélectionner les  $m^+$  et  $m^-$  premières paires d'individus des listes ordonnées associées à  $\zeta_{\mathcal{ML}}^{(r)}$  et  $\zeta_{\mathcal{CL}}^{(r)}$  respectivement ;
- incertaine  $\Gamma_{Min}$  qui consiste à sélectionner les  $m^+$  et  $m^-$  dernières paires d'individus des listes ordonnées associées à  $\zeta_{\mathcal{ML}}^{(r)}$  et  $\zeta_{\mathcal{CL}}^{(r)}$  respectivement.

À l'issue de la sélection, les nouvelles connaissances sont ajoutées aux connaissances courantes par les règles d'augmentation :

$$\mathcal{ML}^{(r)} = \mathcal{ML}^{(r)} \cup \Gamma(\zeta_{\mathcal{ML}}^{(r)}) \quad (5)$$

$$\mathcal{CL}^{(r)} = \mathcal{CL}^{(r)} \cup \Gamma(\zeta_{\mathcal{CL}}^{(r)}) \quad (6)$$

L'algorithme générique de la plateforme collaborative peut alors être dressé (1)

---

**Algorithme 1** plateforme collaborative

---

**ENTRÉES :**  $\mathcal{X}, \{X^{(r)}\}_{r \in [1..n_r]}, n_k^{(r)}, \{A^{(r)}\}_{r \in [1..n_r]}, \Gamma, m^+, m^-, t_f$

**SORTIES :**  $C = \{C_1, \dots, C_{n_k}\}$

- 1 : Initialiser  $\mathcal{CL}^{(r)} = \mathcal{ML}^{(r)} = \emptyset, \forall r \in [1..n_r]$
  - 2 : Initialiser  $t = 0$
  - 3 : Appliquer ADAUZABOC sur  $X^{(r)}$  avec  $A^{(r)}, \mathcal{ML}^{(r)}$  et  $\mathcal{CL}^{(r)}$
  - 4 : Déterminer  $H^{(r)}$  selon (2),  $\forall r \in [1..n_r]$
  - 5 : Calculer  $\tilde{\alpha}_{ij}$  selon l'objectif de consensus ou d'alternatives
  - 6 : Augmenter  $\mathcal{ML}^{(r)}$  et  $\mathcal{CL}^{(r)}$  par (5) et (6)
  - 7 : Si  $t < t_f$  alors  $t = t + 1$  et aller en 3
  - 8 :  $C = Vote(\{H^{(r)}, X^{*(r)}\}_{r \in [1..n_r]})$
- 

## 4.2 L'approche COBOC pour la recherche de consensus

L'approche générique de recherche de consensus entre plusieurs vues d'un même jeu de données, ou entre plusieurs algorithmes de *clusterings* appliqués à un jeu de donnée mono-vue se décline en deux heuristiques : (1) **COBOC consensus**, où chaque vue participe à la construction d'un même ensemble de contraintes que tous les algorithmes devront satisfaire au mieux ; et (2) **COBOC complémentaire**, pour laquelle chaque vue  $\bar{r} \neq r$  participe à la construction d'un même ensemble de contraintes pour  $r$  que l'algorithme  $A^{(r)}$  ne parvient pas par lui-même à satisfaire *a priori*.

**COBOC consensus.** On se place dans le cadre où chaque vue participe à la construction du même ensemble de contraintes. Ainsi pour simplifier on notera  $\mathcal{ML} = \mathcal{ML}^{(r)}$  et  $\mathcal{CL} = \mathcal{CL}^{(r)} \forall r \in [1..n_r]$  (de la même manière  $\zeta_{\mathcal{ML}} = \zeta_{\mathcal{ML}}^{(r)}, \zeta_{\mathcal{CL}} = \zeta_{\mathcal{CL}}^{(r)}$  et  $\alpha_{ij} = \alpha_{ij}^{(r)} \forall r \in [1..n_r]$ ).

Partant du calcul de la confiance  $\alpha_{ij}^{(r)}$  (4), on calcul une confiance globale pour chaque paire d'individus comme une moyenne des confiances locales :

$$\alpha_{ij} = \frac{1}{n_r} \sum_{r=1}^{n_r} \alpha_{ij}^{(r)} \quad (7)$$

Une valeur positive et élevée de  $\alpha_{ij}$  indique que  $x_i$  et  $x_j$  ont majoritairement été classés ensemble par les algorithmes  $A^{(r)}$  et que ceux-ci sont dans chaque vue plus proches entre eux que des autres individus. Dans ce cas on est davantage certain que  $x_i$  et  $x_j$  devraient appartenir à un même groupe. Cette confiance permet de définir les ensembles  $\zeta_{\mathcal{ML}}$  et  $\zeta_{\mathcal{CL}}$  :

$$\zeta_{\mathcal{ML}} = \{(x_i, x_j) \in \mathcal{X}^2 \setminus (\mathcal{ML} \cup \mathcal{CL}) \mid \alpha_{ij} > 0\} \quad (8)$$

$$\zeta_{\mathcal{CL}} = \{(x_i, x_j) \in \mathcal{X}^2 \setminus (\mathcal{ML} \cup \mathcal{CL}) \mid \alpha_{ij} < 0\} \quad (9)$$

**COBOC complémentaire.** L'heuristique complémentaire se distingue de l'heuristique consensus par le fait que l'on maintient des ensembles de connaissances différents pour chaque algorithme.

Partant des confiances  $\alpha_{ij}^{(r)}$ , on calcul une confiance  $\tilde{\alpha}_{ij}^{(r)}$  comme moyenne sur les vues  $\bar{r}$  des confiances locales associées aux paires d'individus :

$$\tilde{\alpha}_{ij}^{(r)} = \frac{1}{n_r - 1} \sum_{\substack{\bar{r}=1 \\ \bar{r} \neq r}}^{n_r} \alpha_{ij}^{(\bar{r})} \quad (10)$$

Dans ce contexte complémentaire, l'idée est que si deux individus appartiennent à un même groupe dans les vues  $\bar{r} \neq r$ , et si ces individus sont séparés par  $A^{(r)}$ , alors il faut suggérer à  $A^{(r)}$  de les regrouper. Les ensembles de paires d'individus candidates  $\zeta_{\mathcal{ML}}^{(r)}$  et  $\zeta_{\mathcal{CL}}^{(r)}$  sont obtenus par :

$$\zeta_{\mathcal{ML}}^{(r)} = \{(x_i, x_j) \in \mathcal{X}^2 \setminus (\mathcal{ML}^{(r)} \cup \mathcal{CL}^{(r)}) \mid \tilde{\alpha}_{ij}^{(r)} > 0 \wedge \overline{Link}^{(r)}(x_i, x_j)\} \quad (11)$$

$$\zeta_{\mathcal{CL}}^{(r)} = \{(x_i, x_j) \in \mathcal{X}^2 \setminus (\mathcal{ML}^{(r)} \cup \mathcal{CL}^{(r)}) \mid \tilde{\alpha}_{ij}^{(r)} < 0 \wedge Link^{(r)}(x_i, x_j)\} \quad (12)$$

**Vote pour une partition unique.** Dans l'esprit des méthodes de *clustering* multi-vues auxquelles se confronte CoBOC, un unique *clustering* des individus est attendu. Dans ce contexte, une fusion finale est réalisée sous la forme d'un vote à la majorité entre les différents *clusterings* de chaque vue. Ces *clusterings* sont alors considérés comme des hypothèses et peuvent être combinées de différentes façons. À partir de l'ensemble  $\{H^{(r)}\}_{r \in [1..n_r]}$  des hypothèses de *clustering* sur les paires d'individus et l'ensemble  $\{X_{r \in [1..n_r]}^*\}$  des représentations optimales locales de  $\mathcal{X}$  obtenues par ADAUZABOC, un *clustering*  $C$  final peut être obtenu par :

1. La construction d'une matrice de similarité  $K_1$  à partir des hypothèses de *clustering* :

$$K_1 = \sum_{r=1}^{n_r} \tilde{H}^{(r)} \quad (13)$$

où  $\tilde{H}^{(r)} = \frac{1}{2}(H^{(r)} + 1)$ , ainsi  $H_{ij}^{(r)} \in \{0, 1\}$ .  $K_1$  est ensuite utilisé comme matrice de similarité, dans un algorithme de *clustering* classique mono-vue adapté (e.g. AGNES, KKM, KFKM, SC, etc.).

2. Selon le même principe de vote, mais en utilisant davantage les représentations finales optimales des individus en recalculant les confiances  $\alpha_{ij}$  pour chaque paire  $(x_i, x_j)$ . Soit  $\alpha_{min}$  (resp.  $\alpha_{max}$ ) quantité négative (resp. positive) correspondante à la plus faible (resp. forte) des confiances sur les paires d'individus :

$$\alpha_{min} = \min_{(x_i, x_j) \in \mathcal{X}^2} \alpha_{ij} ; \alpha_{max} = \max_{(x_i, x_j) \in \mathcal{X}^2} (\alpha_{ij} - \alpha_{min})$$

Un noyau normalisé peut alors être construit à partir de  $\alpha$  par :

$$K_{2ij} = \frac{\alpha_{ij} - \alpha_{min}}{\alpha_{max}} \quad (14)$$

### 4.3 L'approche ALTERBOC pour la recherche d'alternatives

L'approche générique de recherche de *clusterings* alternatifs d'un même jeu de données se décline également en : (1) **ALTERBOC global**, pour laquelle chaque vue ou alternative  $\bar{r} \neq r$  participe à la construction d'un même ensemble de contraintes pour  $r$  quelque soient les résultats de  $A^{(r)}$  ; et (2) **ALTERBOC complémentaire**, où chaque alternative  $\bar{r} \neq r$  participe à la construction d'un même ensemble de contraintes pour  $r$  que l'algorithme  $A^{(r)}$  ne parvenait pas à satisfaire *a priori*.

**ALTERBOC global.** Partant d'une représentation matricielle  $X$  de l'ensemble d'individu  $\mathcal{X}$ , l'idée est de construire des ensembles  $\mathcal{ML}^{(r)}$  et  $\mathcal{CL}^{(r)}$  permettant à un algorithme  $A^{(r)}$  d'obtenir un des  $n_r$  *clusterings* alternatifs. Soit la confiance  $\tilde{\alpha}_{ij}^{(r)}$  (10). Une valeur positive et élevée de  $\tilde{\alpha}_{ij}^{(r)}$  indique que  $x_i$  et  $x_j$  ont majoritairement été classés ensemble par les algorithmes  $A^{(\bar{r})}$  ( $\bar{r} \neq r$ ) et que ceux ci sont pour chaque vue autre que  $r$ , plus proches entre eux qu'aux autres individus. L'idée dans le cadre de la recherche d'un *clustering*  $C^{(r)}$  alternatif à  $\{C^{(\bar{r})}\}$  est de s'assurer que  $A^{(r)}$  ne regroupe pas  $x_i$  et  $x_j$ . Ainsi,  $(x_i, x_j)$  doit correspondre alors à une contrainte  $\mathcal{CL}$ .

La confiance (10) permet de définir pour chaque vue  $r$ , les ensembles de paires d'individus candidates  $\zeta_{\mathcal{ML}}^{(r)}$  et  $\zeta_{\mathcal{CL}}^{(r)}$  :

$$\zeta_{\mathcal{ML}}^{(r)} = \{(x_i, x_j) \in \mathcal{X}^2 \setminus (\mathcal{ML}^{(r)} \cup \mathcal{CL}^{(r)}) \mid \tilde{\alpha}_{ij}^{(r)} < 0\} \quad (15)$$

$$\zeta_{\mathcal{CL}}^{(r)} = \{(x_i, x_j) \in \mathcal{X}^2 \setminus (\mathcal{ML}^{(r)} \cup \mathcal{CL}^{(r)}) \mid \tilde{\alpha}_{ij}^{(r)} > 0\} \quad (16)$$

**ALTERBOC complémentaire.** L'heuristique complémentaire est essentiellement la même que la précédente, si ce n'est dans la construction explicite des ensembles de paires d'individus candidates  $\zeta_{\mathcal{ML}}^{(r)}$  et  $\zeta_{\mathcal{CL}}^{(r)}$  :

$$\zeta_{\mathcal{ML}}^{(r)} = \{(x_i, x_j) \in \mathcal{X}^2 \setminus (\mathcal{ML}^{(r)} \cup \mathcal{CL}^{(r)}) \mid \tilde{\alpha}_{ij}^{(r)} < 0 \wedge \overline{Link}^{(r)}(x_i, x_j)\}$$

$$\zeta_{\mathcal{CL}}^{(r)} = \{(x_i, x_j) \in \mathcal{X}^2 \setminus (\mathcal{ML}^{(r)} \cup \mathcal{CL}^{(r)}) \mid \tilde{\alpha}_{ij}^{(r)} > 0 \wedge Link^{(r)}(x_i, x_j)\}$$

Intuitivement, un bon exemple de paire candidate pour être une contrainte  $(x_i, x_j) \in \mathcal{ML}^{(r)}$  est un couple séparé dans les alternatives  $\bar{r}$  et aussi dans  $r$ . Ainsi une façon de forcer les algorithmes de *clustering* à se comporter différemment est d'insister pour que  $A^{(r)}$  regroupe  $x_i$  et  $x_j$ .

## 5 Évaluation

Les approches COBOC et ALTERBOC ont été testées expérimentalement selon une évaluation interne. Les jeux de données qui ont servi de base de test sont issus de la base UCI. L'approche COBOC a été testée dans les contextes applicatifs (1) multi-vues, où l'on cherche une partition consensus de l'ensemble  $\mathcal{X}$  où chaque individu est décrit simultanément par plusieurs représentations. Ce cadre applicatif est celui des approches de *clustering* multi-vues ; et (2) de la combinaison d'algorithmes où l'on

applique plusieurs algorithmes de *clustering* différents sur un jeu de donnée mono-vue. Ce cadre applicatif est typique des développements des approches de *clustering* d'ensemble, de *clustering* collaboratif ou de *clustering* alternatif. L'application au contexte multi-vues est observée sur le jeu de données *mfeat*, et l'application au contexte de la combinaison de modèles pour la recherche de consensus ou d'alternatives est observée sur le jeu de donnée *parkinson*.

La recherche d'une solution consensus par COBOC et de solutions alternatives par ALTERBOC sont caractérisées en termes d'évaluation interne, en observant l'évolution de la moyenne des informations mutuelles entre les différents *clusterings* locaux (avant l'étape de vote final pour COBOC).

Le nombre d'échanges de contraintes est fixé à 10. À chaque itération de COBOC ou ALTERBOC (lors de chaque échange de connaissances),  $m = p\% \times \binom{n(n-1)}{2n_k}$  contraintes sont générées, où  $p\%$  est un pourcentage prédéfini. le terme  $m$  correspond à un pourcentage de nombre de contraintes  $\mathcal{ML}$  pouvant être généré, sous hypothèse de groupes de tailles homogènes. Dans les expériences,  $p\% = 1$ , ainsi le nombre total de contraintes générées est de  $\frac{1}{10n_k} \times$  le nombre de paires d'individus différentes. Lorsqu'ils sont utilisés, les algorithmes de *clustering* sont paramétrés de manière classique. Si le nombre de groupes est nécessaire, celui-ci correspond au nombre de classes du jeu de données correspondant. Les paramètres de flou éventuels nécessaires sont tous fixés à  $\beta = 1, 25$ .

Les approches ADAUZABOC encapsulant les algorithmes précédents choisissent de façon heuristique la dimensionnalité du sous-espace à calculer à chaque étape pour chaque vue  $r$ , selon le nombre de valeurs propres positives de la matrice à diagonaliser. L'initialisation de l'algorithme boîte noire employé est invariante pour une recherche de sous-espace optimal donnée, mais différente entre les vues ou alternatives. L'évaluation interne est proposée dans TAB. 1.

## 6 Conclusion

La plateforme de *clustering* collaboratif proposée se décline en deux variantes selon l'objectif de recherche d'un ou plusieurs *clusterings* consensus ou de plusieurs *clusterings* alternatifs. Celles-ci peuvent être appliquées dans différents contextes comme la combinaison d'algorithmes ou le contexte multi-vues.

La collaboration proposée pour atteindre l'accord (consensus) ou le désaccord (alternatives) entre les différents algorithmes employés est basé sur l'échange de contraintes permettant localement de trouver simultanément un *clustering* atteignant l'objectif et un sous-espace de représentation des données menant à ces *clusterings*. Cette dernière facette n'est pas présente dans les différentes approches étudiées dans l'état de l'art, et permet des analyses d'un autre ordre. Par exemple, une question à laquelle la contribution proposée peut répondre est la suivante :

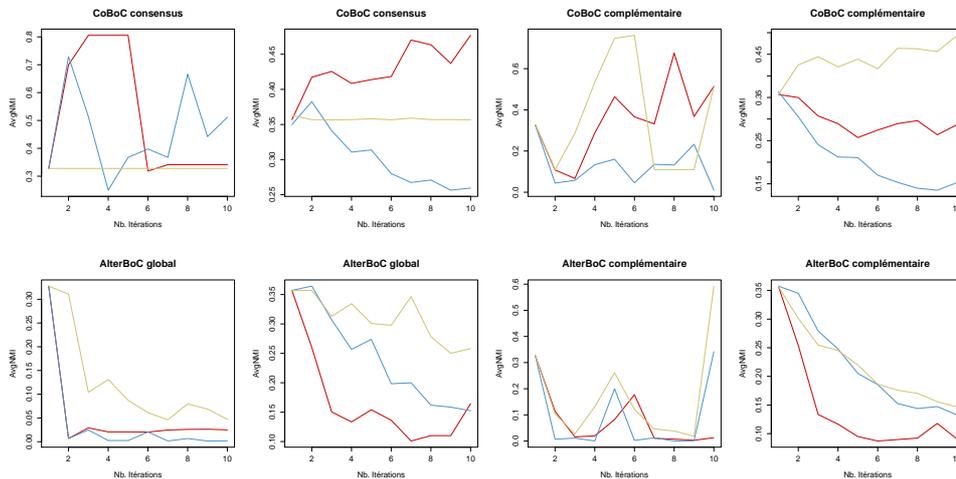
*quelles sont localement les sous-espaces de représentation qui permettent d'atteindre un consensus quelquesoient les algorithmes de clustering locaux employés ?*

La résolution de cette question peut permettre, pour des données multi-vues, d'identifier les attributs créant du bruit pour l'obtention d'un *clustering* cible. Ceux-ci sont

alors de faibles contributeurs à la définition du sous-espace permettant par exemple d’atteindre des solutions de *clustering* proches.

L’approche a été évaluée de manière interne afin d’observer son comportement. Celle-ci a le défaut d’être assez fortement paramétrée, notamment par le volume de contraintes échangées et le nombre d’échanges envisagé. Nous avons observé notamment que les solutions les plus intéressantes étaient obtenues lors des quelques premiers échanges. La variante ALTERBOC, elle, manque de procédure d’évaluation externe, mais ceci est normal par essence. En effet si un *clustering* de bonne qualité peut être obtenu sur des données, au sens de cette évaluation externe, alors une alternative sera de mauvaise qualité au sens de l’évaluation choisie et sera donc peu valorisable. En revanche, les techniques de recherche d’alternatives trouvent tout à fait leur place lors de la confrontation à de réelles données dont on ne connaît pas du tout la classification de départ, ou bien lorsque celle-ci est connue de l’analyste qui préfère alors découvrir quelque chose de *différent*.

Des développements récents, et des interrogations sur la diversification des problèmes autour du *clustering* (Kriegel et Zimek, 2010) ont guidé la recherche bibliographique autour notamment du *clustering* alternatif, et laisse entrevoir les liens entre toutes les problématiques, avec en suspens l’éventualité de voir des approches susceptibles de les unifier et de proposer un mécanisme de résolution adéquat. La plateforme proposée tend vers cet objectif de pouvoir gérer simultanément la recherche d’un ou plusieurs *clusterings*, consensus ou alternatifs, à travers un même mécanisme de collaborations entre plusieurs classificateurs non supervisés.



TAB. 1 – Évaluation de COBOC (première ligne) et ALTERBOC (seconde ligne) selon les deux heuristiques, sur parkinson (1) et (3) et mfeat (2) et (4). Les courbes rouge, jaune et bleu correspondent respectivement à  $\Gamma_{Random}$ ,  $\Gamma_{Max}$  et  $\Gamma_{Min}$ .

## Références

- Bickel, S. et T. Scheffer (2005). Estimation of mixture models using co-EM. In *16th European Conference on Machine Learning ECML 2001*, Volume 3720 of *Lecture Notes in Artificial Intelligence*, pp. 35–46. Springer.
- Blum, A. et T. Mitchell (1998). Combining labeled and unlabeled data with co-training. In *COLT : Proceedings of the Workshop on Computational Learning Theory*, Morgan Kaufmann Publishers.
- Dang, X. H. et J. Bailey (2010). Generation of alternative clusterings using the cami approach. In *SDM*, pp. 118–129. SIAM.
- Davidson, I. et Z. Qi (2008). Finding alternative clusterings using constraints. In *ICDM*, pp. 773–778. IEEE Computer Society.
- Kriegel, H.-P. et A. Zimek (2010). Subspace Clustering, Ensemble Clustering, Alternative Clustering, Multiview Clustering : What Can We Learn From Each Other? In *Proceedings of MultiClustKDD*.
- Liu, Y., R. Jin, et A. K. Jain (2007). Boostcluster : boosting clustering by pairwise constraints. In P. Berkhin, R. Caruana, et X. Wu (Eds.), *KDD*, pp. 450–459. ACM.
- Pedrycz, W. (2002). Collaborative fuzzy clustering. *Pattern Recogn. Lett.* 23(14), 1675–1686.
- Strehl, A. et J. Ghosh (2003). Cluster ensembles — a knowledge reuse framework for combining multiple partitions. *J. Mach. Learn. Res.* 3, 583–617.
- Sublemontier, J.-H., G. Cleuziou, M. Exbrayat, et L. Martin (2011a). Clustering multi-vues : une approche centralisée. *Revue des Nouvelles Technologies de l'Information, numéro spécial Fouille de Données Complexes : données multiples*.
- Sublemontier, J.-H., L. Martin, G. Cleuziou, et M. Exbrayat (2011b). Integrating pairwise constraints into clustering algorithms : optimization-based approaches. In *ICDMW 2011, The Eleventh IEEE International Conference on Data Mining Workshops*, Vancouver, Canada.
- Wemmert, C., P. Gançarski, et J. J. Korczak (2000). A collaborative approach to combine multiple learning methods. *International Journal on Artificial Intelligence Tools* 9(1), 59–78.

## Summary

In this paper, we propose a collaborative framework that is able to solve the multi-view clustering and alternative clustering problems, while using some clustering ensemble and semi-supervised clustering principles. We provide a mechanism to control, via a shared information model, different clustering algorithms to provide consensus or alternative clustering solutions. The strong point is that our model does not need to know which clustering algorithms to use nor their parameters.

# Apprentissage de règles floues pour caractériser des objets d'intérêt dans une image de télédétection

Bruno Belarte\*<sup>1</sup>, Cédric Wemmert\*  
Germain Forestier\*\*, Christiane Weber\*\*\*

\*Université de Strasbourg, LSIT, UMR CNRS 7005, Strasbourg, France  
{belarte,wemmert}@unistra.fr,

\*\*Université de Haute Alsace, MIPS EA 2332, Mulhouse, France  
germain.forestier@uha.fr

\*\*\*Université de Strasbourg, LIVE, ERL CNRS 7230, Strasbourg, France  
christiane.weber@live-cnrs.unistra.fr

**Résumé.** L'évolution des technologies nous donne accès à des images d'un très haut niveau de détail. Les algorithmes de segmentation permettent d'extraire des objets d'intérêt de ces images. L'expert du domaine sait décrire les objets d'intérêts présents dans l'image. Cependant les segments ne correspondent pas aux objets d'intérêt. Nous proposons dans cet article une méthode d'apprentissage de règles visant à apprendre comment sont décomposés les objets d'intérêt dans la segmentation. Les expériences en cours pour tester la validité de l'approche sont présentées.

## 1 Introduction

L'évolution croissante de la résolution d'acquisition des images dans tous les domaines (imagerie médicale, télédétection, etc.) permet d'obtenir des images d'un très haut niveau de détail. Pour pouvoir analyser ce type d'images, des méthodes dites *orientées régions* ont été développées. Ce type d'approches offre la possibilité de s'intéresser aux propriétés (radiométriques, structurelles, spatiales, etc.) de ces régions.

À partir d'une description faite par l'expert des objets d'intérêt, il est possible de caractériser les régions extraites. Cependant, le niveau de détail des images est désormais tel que les algorithmes de segmentation usuels ne sont pas capables de produire des régions correspondant à des objets d'intérêt pour l'expert, rendant caduc l'utilisation de connaissances expertes. En effet, les objets d'intérêt se retrouvent découpés en plusieurs segments.

C'est dans ce contexte que s'inscrivent nos travaux. Nous proposons dans cet article un algorithme d'apprentissage de règles qui vise à apprendre comment sont composés les objets d'intérêt dans une image sur-segmentée. Ces travaux poursuivent les travaux de (Belarte et Wemmert, 2012). Le reste de l'article est organisé de la manière suivante : la section 2 décrit le formalisme utilisé par la méthode proposée. La section 3 présente la méthode d'analyse d'images de télédétection guidée par la connaissance du domaine. La méthode d'apprentissage

---

1. Les travaux présentés ont été financés par le Centre National des Études Spatiales et la Région Alsace.

de règles est détaillée dans la section 4. Enfin, la section 5 conclut et évoque les expériences en cours pour valider notre approche.

## 2 Représentation de la connaissances

La méthode d'analyse d'images que nous proposons repose sur l'identification d'objets d'intérêt à partir de leur définition par un expert du domaine. Pour cela, il est nécessaire de formaliser la représentation des objets d'intérêt, et de la base de connaissances.

### 2.1 Objet d'intérêt

Un objet d'intérêt est un objet ayant une réalité sémantique dans le domaine d'expertise. Ils peuvent être de deux types. Les objets d'intérêt simples représentent la granularité la plus fine pour l'expert. Les objets d'intérêt composés sont des agrégats d'objets simples et/ou d'autres objets composés. Dans le contexte de ces travaux, un objet simple peut être par exemple une maison, un arbre, une route ou encore une étendue d'eau ; un objet composé peut être par exemple un îlot pavillonnaire, un parc, une zone d'activité ou encore un aéroport. Un objet d'intérêt est représenté par sa forme géométrique (le polygone englobant cet objet) et par trois types de propriétés numériques définies comme suit :

**Attributs.** Un attribut propre est une propriété qui ne dépend que de l'objet en lui-même. Ces propriétés décrivent notamment la signature radiométrique de l'objet (valeur moyenne des bandes, indice NDVI, ...) et sa forme (taille, élongation, indice de Miller, ...).

**Relations spatiales.** Une relation spatiale est une propriété qui lie l'objet courant à son voisinage. La relation d'adjacence à un objet de type A est définie par le ratio des côtés du polygone partagés avec un objet de type A sur le périmètre de l'objet courant. La relation de distance à un objet de type B est définie par la distance en mètres à l'objet de type B le plus proche de l'objet courant.

**Relations de composition.** Une relation de composition est une propriété qui lie l'objet courant à des objets de niveaux d'abstraction différents. La relation de composition d'un objet composé par des objets de type C est définie par le ratio de la surface de chacun des objets de type C composant l'objet courant sur la taille de l'objet courant.

### 2.2 Concept

À partir de cette définition d'un objet d'intérêt, nous pouvons définir la notion de concept. Un concept est une représentation abstraite d'un objet d'intérêt. La manière la plus simple de représenter un concept est de le représenter sous la forme d'une liste de contraintes sur les propriétés d'un objet. De là, émergent plusieurs questions. Comment formaliser ces contraintes pour qu'elles puissent prendre en compte l'imprécision inhérente au traitement d'images ? Quel formalisme utiliser pour que l'expert puisse produire des règles concises et réutilisables ?

Nous avons choisi de représenter nos contraintes par des ensembles flous car ils permettent de gérer l'imprécision tout en étant faciles à mettre en place. Ceux-ci sont présentés dans la section suivante.

### 2.3 Théorie des ensembles flous

La théorie des ensembles flous a été introduite par (Zadeh, 1965). C'est une théorie mathématique qui étend la théorie ensembliste classique en permettant la gestion de l'imprécision et de l'erreur. En effet, dans la théorie ensembliste classique, le degré d'appartenance d'un élément à un ensemble est donné par une valeur binaire : l'élément appartient oui ou non à l'ensemble. Le degré d'appartenance à un ensemble flou est décrit à l'aide d'une fonction d'appartenance retournant une valeur réelle dans l'intervalle  $[0, 1]$ , 1 signifiant que l'élément appartient totalement à l'ensemble, 0 signifiant que l'élément n'appartient pas du tout à l'ensemble. Toutes les valeurs intermédiaires permettent de nuancer le degré d'appartenance.

Une fonction d'appartenance est définie par deux ensembles, le noyau et le support. Le *noyau* est l'ensemble des éléments appartenant totalement à l'ensemble (dont le degré de vérité est égal à 1). Le *support* est l'ensemble des éléments appartenant au moins un peu à l'ensemble (dont le degré de vérité est strictement positif). Par conséquent, un élément peut appartenir à deux ensembles flous en même temps, mais à des degrés différents.

**Définition formelle.** Soit  $E$  l'espace des données,  $A \subset E$  un ensemble flou,  $n_A$  le noyau et  $s_A$  le support de l'ensemble  $A$ ,  $x$  un élément de  $E$ ,  $\mu_A$  la fonction d'appartenance modélisant l'ensemble flou  $A$ . Le noyau et le support sont définis dans les équations 1 et 2.

$$n_A = \{x \in E | \mu_A(x) = 1\} \quad (1)$$

$$s_A = \{x \in E | \mu_A(x) > 0\} \quad (2)$$

Un ensemble classique peut être modélisé par un ensemble flou dont la fonction d'appartenance ne peut prendre comme valeur que 0 et 1. Autrement dit  $n_A = s_A$ .

**Opérations.** Les opérateurs classiques d'union, d'intersection et de complément sont étendus aux ensembles flous. L'intersection est modélisée par une T-norme et l'union par une T-conorme. Dans son papier original, (Zadeh, 1965) propose comme T-norme l'opérateur minimum et comme T-conorme l'opérateur maximum. Soit  $E$  le domaine,  $A \subset E$  et  $B \subset E$  deux ensembles flous. Les équations 3, 4 et 5 illustrent ces opérations.

$$\mu_{A \cap B} = \min(\mu_A, \mu_B) \quad (3)$$

$$\mu_{A \cup B} = \max(\mu_A, \mu_B) \quad (4)$$

$$\mu_{\bar{A}} = 1 - \mu_A \quad (5)$$

Comme pour les opérations ensemblistes classiques, les opérations d'union et d'intersection sont commutatives et associatives, l'intersection est distributive par rapport à l'union. Cependant, le principe du tiers exclu est contredit. D'autre part, un élément peut appartenir à un ensemble et à son complément (mais à des degrés différents).

Un ensemble flou étant étroitement lié à la fonction d'appartenance le représentant. Nous utiliserons par la suite indifféremment les termes *ensemble flou* et *fonction d'appartenance*.

## 2.4 Logique floue

La logique floue (Zadeh, 1975) est une technique de raisonnement en intelligence artificielle. Elle repose sur la théorie des ensembles flous présentée dans la section précédente. Le principe de la logique floue est basé sur l'utilisation de variables linguistiques pour modéliser la connaissance. Plutôt que de définir une contrainte par un intervalle de valeurs, elle est définie par une fonction d'appartenance. Ainsi, une variable telle que la *taille* va pouvoir prendre comme valeur *petit* ou *grand*. L'expert peut alors associer des ensembles flous à des variables linguistiques du domaine qu'il sera possible d'utiliser dans un second temps pour former des règles. Le fait que ces variables linguistiques soient basées sur des ensembles flous permet de gérer l'imprécision des mesures à travers l'utilisation du degré d'appartenance.

**Combinaison de variables linguistiques.** Les variables linguistiques peuvent être combinées à l'aide d'opérateurs logiques **et**, **ou** et **non**. L'opérateur **et** est modélisé par une T-norme, l'opérateur **ou** est modélisé par une T-conorme, l'opérateur **non** est modélisé par le complément, ceci de manière similaire aux opérateurs de la théorie des ensembles flous. Parmi les T-normes les plus utilisées dans la littérature, nous pouvons citer l'intersection ensembliste de Zadeh (équation 3) ou l'intersection probabiliste (équation 6). Parmi les T-conormes les plus utilisées dans la littérature, nous pouvons citer l'union ensembliste de Zadeh (équation 4) ou l'union probabiliste (équation 7). L'opérateur **non** est toujours modélisé par le complément (équation 5). Il existe d'autres opérateurs dans la littérature (Lee, 1990a,b).

$$\mu_{A \cap B} = \mu_A \times \mu_B \quad (6)$$

$$\mu_{A \cup B} = \mu_A + \mu_B - \mu_A \times \mu_B \quad (7)$$

**Règles.** En combinant différentes variables linguistiques, l'expert peut former des règles cohérentes et complexes. De par leurs définitions linguistiques, ces règles sont compréhensibles par un être humain. Leurs traductions par des fonctions d'appartenance les rendent compréhensibles par un algorithme mathématique.

Soit *allongé*, *brillant*, *petit* et *moyen* quatre variables linguistiques. Un exemple de règle combinant ces fonctions à l'aide des différents opérateurs est donné par la formule suivante :

$$r = \textit{allongé et non brillant et (petit ou moyen)}$$

La transformation de cette règle en expression mathématique utilisant les opérateurs ensemblistes de Zadeh est donnée dans l'équation 8. La même règle en utilisant les opérateurs probabilistes est donnée dans l'équation 9.

$$\mu_r = \min(\mu_{\textit{allongé}}, \min(1 - \mu_{\textit{brillant}}, \max(\mu_{\textit{petit}}, \mu_{\textit{moyen}}))) \quad (8)$$

$$\mu_r = \mu_{\textit{allongé}} \times ((1 - \mu_{\textit{brillant}}) \times (\mu_{\textit{petit}} + \mu_{\textit{moyen}} - \mu_{\textit{petit}} \times \mu_{\textit{moyen}})) \quad (9)$$

À chaque règle est associée une valeur de sortie. Cette valeur peut être une variable numérique ou un ensemble flou. Il est alors possible d'exprimer une base de règles sous la forme d'une liste de règles d'inférence, où A, B et N sont des variables linguistiques du même domaine :

- Si règle 1 alors A
- Si règle 2 alors B
- ...
- Si règle n alors N

**Évaluation.** À partir de cette base de règles, il est maintenant possible d'évaluer le système à l'aide d'un moteur de raisonnement basé sur la logique floue. Les variables du système sont transformées en variables linguistiques (processus de fuzzyfication, ou floutage) pour pouvoir être évaluées par le moteur de raisonnement. Chaque règle est ensuite évaluée. Les résultats de chaque évaluation sont agrégés à l'aide de l'opérateur maximum. La dernière étape du processus consiste à attribuer au résultat la bonne variable linguistique (processus de defuzzyfication, ou défloutage). Pour cela, il existe un grand nombre de méthodes dans la littérature (Lee, 1990a,b). Les deux plus utilisées sont le centre de gravité (*CdG*, équation 10) et la moyenne des maximums (*MM*, équation 11, où  $S = \{y_x \in E, \mu(y_x) = \max_{y \in E}(\mu(y))\}$ ,  $E$  étant le domaine de définition de la variable de sortie). Soit  $d$  la variable de décision retournée par l'algorithme de raisonnement.

$$d_{CdG} = \frac{\int y \cdot \mu(u) \cdot dy}{\int \mu(u) \cdot dy} \quad (10)$$

$$d_{MM} = \frac{\int_S y \cdot dy}{\int_S dy} \quad (11)$$

Le processus de défloutage va donner la valeur de sortie de l'algorithme de raisonnement à partir de l'agrégat d'ensembles flous obtenu lors de l'évaluation. Cette valeur peut maintenant être utilisée dans un processus de décision.

## 2.5 Base de connaissance

À partir des définitions précédentes, nous pouvons construire notre base de connaissances. L'expert dispose d'un grand nombre de variables linguistiques dans son domaine d'expertise. Par exemple, la propriété *élongation* est définie par les variables linguistiques *compact*, *moyennement allongé* et *très allongé*. La première étape consiste à transformer ces variables linguistiques en ensembles flous. Autrement dit, à donner des intervalles numériques à ces variables. L'expert peut alors formuler des règles à partir des différentes variables linguistiques formalisées précédemment. Chaque règle décrit un concept. Par exemple, le concept *arbre* est défini par la règle *petit et compact et NDVI élevé*. L'expert peut définir un concept qui est un cas particulier d'un autre concept. Par exemple, le concept *étang* est un cas particulier du concept *étendue d'eau*, ou, le concept *végétation* est une généralisation du concept *pelouse*. Pour pouvoir ordonner les concepts en conséquent, ils sont placés dans une hiérarchie dans laquelle une feuille est un concept particulier et chaque noeud est une généralisation de ses sous-arbres. Un exemple de hiérarchie est donné dans la figure 1.

Dans un paysage urbain, l'expert peut définir des concepts de plus haut niveau d'abstraction. Un objet correspondant à un tel concept composé est un agrégat d'objets élémentaires ou d'autres objets composés. Un *concept composé* est défini par une règle portant sur tout type de propriétés (attributs, relations spatiales, relations de composition). Par exemple, le concept

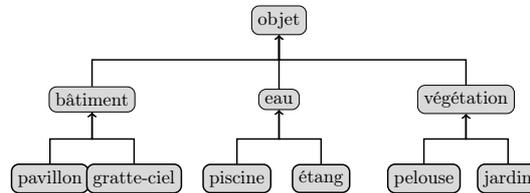


FIG. 1 – Exemple d’une hiérarchie de concepts

composé *ilot pavillonnaire* est défini par la règle *grand et composé de pavillons et composé de végétation et entouré par de la route*.

### 3 Analyse d’image guidée par la connaissance

À partir des notions que nous avons définies dans la section précédente, nous disposons de tous les outils nécessaires pour analyser des images de télédétection à l’aide de la connaissance de l’expert du domaine. Dans cette section, nous allons adapter ces outils pour traiter les images et les mettre en pratique.

#### 3.1 Modélisation des données

La donnée brute que nous voulons analyser est une image de paysage urbain. Pour pouvoir l’exploiter avec un moteur de raisonnement basé sur la logique floue, nous devons transformer cette donnée brute en donnée utilisable par l’algorithme. Nous nous intéressons à une analyse orientée objet de l’image. À l’aide d’un algorithme de segmentation, l’image est transformée en segments. Un segment correspond à un objet élémentaire.

Chaque objet est défini par un polygone (caractérisant son contour) et par un ensemble de propriétés (radiométriques et structurales) calculées à partir de l’image. Ces objets élémentaires sont ensuite placés dans un graphe d’adjacence. Un noeud correspond à un objet, un arc correspond à une relation d’adjacence entre deux objets. Deux objets sont adjacents s’ils partagent une arête commune dans l’image. Ainsi les données contenues dans le graphe correspondent à la définition des objets d’intérêt proposée en section 2.1.

L’objectif de l’analyse d’image est d’identifier les objets d’intérêt correspondant aux concepts définis par l’expert. Pour cela, nous allons adapter le moteur de raisonnement flou pour qu’il génère en sortie l’étiquette du concept à attribuer à l’objet évalué.

#### 3.2 Moteur d’inférence flou pour l’analyse d’objets d’intérêt

Dans cette section nous allons détailler comment nous avons adapté le moteur de raisonnement flou présenté en section 2.4 pour pouvoir fonctionner sur nos données.

La donnée de base est un objet d’intérêt. Les ensembles flous et les règles sont renseignés par l’expert du domaine et placés dans une base de connaissances. Cette base de connaissances contient la hiérarchie de concepts. Le moteur de raisonnement flou va servir à raffiner

le concept attribué à chaque objet d'intérêt. Toutes les règles ne sont pas évaluées en même temps. Seules les règles correspondant aux concepts raffinant directement le concept attribué à l'objet courant (descendants directs dans la hiérarchie) sont évaluées. Une fois la nouvelle étiquette trouvée et attribuée à l'objet courant, le processus itère tant que le concept attribué a des descendant dans la hiérarchie ou que la variable de sortie ne permet pas d'attribuer une nouvelle étiquette (*i.e.* la valeur de sortie est inférieure à un seuil et ne permet pas de prendre une décision).

Les opérateurs flous utilisés sont les opérateurs ensemblistes (*i.e.* minimum et maximum). L'opérateur de défloutage a du être adapté (notamment les opérateurs centre de gravité ou moyenne des maximum) car dans notre cas la variable de sortie n'est pas ordonnée. Nous avons choisi de prendre une variante de l'opérateur moyenne des maximums. Plutôt que de calculer la moyenne des maximums, nous gardons simplement le concept correspondant au score maximum. La formule utilisée est donnée par l'équation 12, où  $o$  est l'objet à étiqueter,  $x_j$  est le concept correspondant à la règle  $\mu_j$  et  $n$  est le nombre de concepts évalués.

$$d_{\max} = \{x_j | \mu_j = \max(\mu_i(o)), i \in [1, n]\} \quad (12)$$

### 3.3 Extraction d'objets composés

Une fois le graphe raffiné à partir des concepts définis par l'expert, nous obtenons un graphe dans lequel chaque objet est étiqueté. Il est alors possible de rechercher les objets composés. Un concept composé est défini par un ensemble de concept pouvant le composer et par une règle de composition. Par exemple, le concept composé *îlot pavillonnaire* est défini par la règle *grand et composé de pavillons et composé de végétation et entouré par de la route*, il est constitué d'objets de type *pavillon* et *végétation*. L'extraction des objets composés repose sur un algorithme de croissance de régions (Adams et Bischof, 1994) adapté aux graphes d'objets d'intérêt. Ce processus est décrit dans l'algorithme 1.

Dans la pratique, il n'est pas possible d'utiliser une telle approche pour analyser des images de télédétection. Des tests ont permis de mettre en avant le problème suivant : les objets analysés (produits par la segmentation) ne correspondent pas à des objets sémantiques (décrits par les concepts). Ceci est dû à un phénomène de sur-segmentation ou de sous-segmentation ; la classification des objets d'intérêt est très fortement biaisée par cette imprécision. Les résultats sont illustrés dans la figure 2. Nous cherchons à identifier les pavillons. Le pavillon entouré en rouge sur la vérité terrain n'est tout simplement pas identifié. Il s'avère que l'algorithme de segmentation l'a décomposé en plusieurs segments ce qui le rend non identifiable par la base de connaissances (contrainte de taille non satisfaite). Le second pavillon (à gauche) est partiellement identifié (seule une partie du toit l'est).

De ce fait, l'extraction des objets composés n'est pas réalisable puisqu'elle repose sur un étiquetage correct du graphe. La connaissance experte n'est de fait pas utilisable en l'état pour l'analyse d'image de télédétection à très haute résolution spatiale.

Nous avons pu voir dans cette section les limites de l'utilisation conjointe d'un algorithme de segmentation avec la connaissance de l'expert. En effet, bien que l'expert soit tout à fait capable de décrire les objets géographiques, les algorithmes de segmentation ne sont pas capables d'extraire correctement ces objets. Le résultat d'une segmentation étant soit sur- soit

**Algorithme 1:** Algorithme de croissance de régions adapté au graphes d'objets d'intérêt

soit  $C$  le concept composé à extraire,  $E_C$  l'ensemble des concepts composants  $C$ ,  $r_C$  la règle composée décrivant  $C$

**pour tous les objets d'intérêt  $O$  du graphe faire**

    soit  $c_O$  le concept étiquetant  $O$

**si  $c_O \in E_C$  et  $O$  non marqué alors**

        soit  $R$  la région candidate

        ajouter  $O$  dans  $R$

        marquer  $O$

**pour tous les objets  $O'$  voisin de  $R$  faire**

**si  $c_{O'} \in E_C$  et  $O'$  non marqué alors**

                ajouter  $O'$  dans  $R$

                marquer  $O'$

        évaluer  $r = \mu_{r_C}(R)$

**si  $r > seuil$  alors**

            ajouter  $R$  dans le graphe

sous-segmenté. Ceci est particulièrement le cas avec les images à très haute résolution spatiale de type PLEIADE ou QUICKBIRD. Cette approche est limitée par ce point. C'est pourquoi, nous avons proposé une nouvelle méthode pour lever ce verrou, qui est détaillée dans la section suivante.

## 4 Apprentissage de règles à partir d'exemples

### 4.1 Les limites de l'utilisation de connaissances experte

Nous avons vu dans la section 3 que la méthode de raisonnement basée sur la connaissance de l'expert n'est pas applicable en l'état sans de lourds pré-requis. En effet, l'utilisation de connaissances décrivant des objets sémantiques repose sur la bonne détection de ces objets dans l'image. Or les algorithmes de segmentation classiques ne sont pas capables de retrouver de tels objets. Un algorithme de segmentation repose sur l'identification de zones homogènes au niveau image (*i.e.* radiométrie, distance entre pixels, ...) et non sur l'identification d'objets sémantiques. Dans une image à très haute résolution spatiale (de type QUICKBIRD ou PLEIADE) les objets d'intérêt pour l'expert se retrouvent découpés en plusieurs segments.

La solution que nous proposons pour remédier à cette limitation est de baisser le niveau d'abstraction. En considérant les objets sémantiques non plus comme des objets élémentaires mais comme des objets composés, notre approche reste valide. Il suffit de décrire un objet d'intérêt par une règle de composition. Cependant, l'expert n'est pas capable de donner une telle règle de composition. Cette règle de composition va dépendre du type et de la résolution de l'image, de l'algorithme de segmentation utilisé, de l'orientation des objets ou encore du moment d'acquisition de l'image.

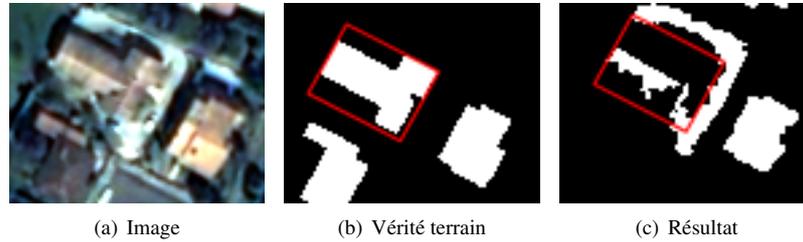


FIG. 2 – Illustration des résultats obtenus par l'algorithme. Nous pouvons constater que la maison de droite est correctement reconnue, que la maison en bas à gauche n'est pas reconnue et la maison entourée de rouge n'est pas reconnue intégralement.

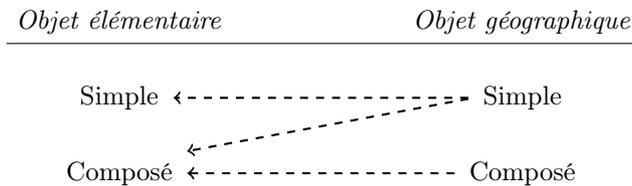


FIG. 3 – Relation entre les objets géographiques et les objets élémentaires.

Avec une telle combinaison de facteurs, il n'est pas possible pour l'expert de donner une règle générique décrivant un concept. Nous proposons d'utiliser un algorithme d'apprentissage pour apprendre une règle de composition à partir d'exemples donnés par l'expert. La méthode proposée repose sur plusieurs classifications hiérarchiques ascendantes consécutives pour apprendre les règles simples et de composition.

## 4.2 Algorithme d'apprentissage

**Extraction des exemples.** L'expert donne les exemples sous la forme d'une liste de polygones, un polygone par exemple. Les objets correspondant à ces exemples doivent être extraits. Le graphe est parcouru noeud par noeud et les objets correspondant à chacun des exemples sont conservés. Ce processus est décrit dans l'algorithme 2.

**Algorithme 2:** Extraction des exemples, avec  $S_i$  la taille de  $i$ ,  $Ex$  l'exemple courant,  $seuil$  un paramètre fixé par l'expert.

**pour tous les objets d'intérêt  $O$  du graphe faire**  
 [ si  $\frac{S_{O \cap Ex}}{S_O} > seuil$  alors  
 | ajouter  $O$  dans la liste des exemples

**Classification Hiérarchique Ascendante (CHA).** CHA est un algorithme de classification automatique (Lance et Williams, 1967). Cet algorithme repose sur une mesure de distance  $D$  entre classes qui est utilisée pour déterminer quelles sont les classes à fusionner. La distance que nous avons choisi est la distance euclidienne au carré, notée  $d_2^2$ . Au départ, les objets sont chacun répartis dans une classe. À chaque étape, les deux classes les plus proches sont fusionnées. L'algorithme 3 illustre ce procédé. La sortie de cet algorithme est un arbre de partition binaire. Le nombre de classes à conserver dépend du critère d'arrêt.

**Algorithme 3:** Classification hiérarchie ascendante

**pour tous les objets d'intérêt  $O$  du graphe faire**  
  └ générer une classe à un seul élément contenant uniquement  $O$   
**tant que le critère d'arrêt n'est pas atteint faire**  
  └ Trouver deux classes  $C_i$  et  $C_j$  dont la distance est la plus faible  
  └ Fusionner  $C_i$  et  $C_j$

**Critère d'arrêt.** Le critère d'arrêt peut être un nombre de classes prédéfini ou un critère automatique. Dans notre cas, nous ne connaissons pas le nombre de classes recherchées, nous choisissons donc une méthode automatique.

La mesure de dissimilarité de Ward (Ward, 1963) calcule la différence d'inertie intra-classe (ou inter-classe) avant et après fusion de deux classes. L'équation 13 donne la formule de la dissimilarité de Ward, où  $d_W$  est la mesure calculée,  $C_u$  et  $C_v$  sont deux classes,  $n_i = \text{card}(C_i)$  et  $D(C_u, C_v)$  est la distance entre deux classes. La distance utilisée est la même que celle de l'algorithme de classification hiérarchie ascendante, soit la distance euclidienne au carré  $d_2^2$ . Lorsque la dissimilarité de Ward est trop élevée (*i.e.* l'augmentation de l'inertie inter-classe ou la perte d'inertie intra-classe trop élevée), la fusion des deux classes produirait une nouvelle classe trop hétérogène et donc non cohérente.

$$d_W(C_u, C_v) = \frac{n_u \cdot n_v}{n_u + n_v} \cdot D(C_u, C_v) \quad (13)$$

**Itération.** L'apprentissage d'une règle de composition se fait en plusieurs étapes. L'expert spécifie les propriétés pertinentes pour décrire le concept composé. Pour chaque propriété, une classification est effectuée. Les variables à classer sont les valeurs de chaque objet pour la propriété donnée. Ceci produit une ou plusieurs fonctions d'appartenance par propriété. À partir des fonctions apprises, une deuxième classification est effectuée. Les variables à classer sont cette fois des vecteurs. Pour chaque objet un vecteur est créé. Chaque élément du vecteur est la variable de sortie d'une fonction d'appartenance apprise précédemment. Cette deuxième étape de la classification produit une ou plusieurs règles. Ces règles sont utilisées pour étiqueter les objets élémentaires composant les exemples. Pour finir, une dernière étape d'apprentissage permet d'apprendre une règle de composition unique décrivant l'ensemble des exemples. La figure 4 illustre le processus d'apprentissage dans son intégralité.

En pratique, l'apprentissage des règles sur les objets élémentaires est lui aussi décomposé en deux étapes. En effet, pour apprendre des règles sur les relations spatiales entre objets, ces

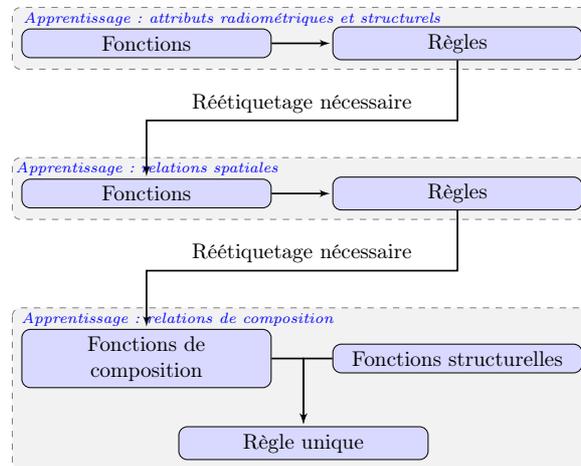


FIG. 4 – Organigramme résumant le processus d'apprentissage.

objets doivent être étiquetés. Pour cela, un premier jeu de règles est appris, celui-ci portant uniquement sur les attributs propres. Chacune des règles apprises est associée à un nouveau concept qui enrichit la hiérarchie. Ces concepts sont utilisés pour étiqueter les objets élémentaires composant les exemples. Un deuxième jeu de règles est appris, celui-ci portant uniquement sur les relations spatiales. Chacune de ces nouvelles règles permet de créer un ou plusieurs concepts dans la hiérarchie. En effet ces nouvelles règles permettent de différencier des objets étiquetés, les nouveaux concepts générés sont donc ajoutés plus bas dans la hiérarchie de concepts. Ceci est illustré dans la figure 5. Une fois l'apprentissage terminé, la nouvelle règle de composition est utilisée pour extraire les objets composés.

## 5 Conclusion

Cet article propose une méthodologie pour l'analyse d'images guidée par la connaissance du domaine. Lorsque la connaissance n'existe pas ou est incomplète, nous proposons une méthode d'apprentissage de règles floues pour caractériser les objets d'intérêt. Des expériences sont en cours pour analyser d'une part l'impact de la segmentation, d'autre part l'impact des paramètres d'apprentissage sur la qualité du résultat final. Les premiers résultats montrent que la qualité de la segmentation influe très largement sur l'algorithme d'apprentissage. C'est pourquoi d'autres expériences sont menées pour chercher des plages de paramètres acceptables pour les algorithmes de segmentation choisis.

La connaissance générée par l'algorithme d'apprentissage pourra ensuite être utilisée par l'algorithme de raisonnement présenté dans la section 3. L'image utilisée pour ces expériences est une image QUICKBIRD (2008) de Toulouse. D'autres expériences sont prévues sur une image de Strasbourg, prise par le capteur PLEIADE (2012).

## Apprentissage de règles floues

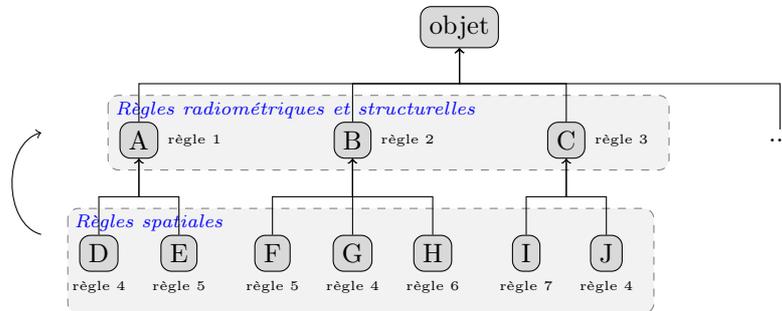


FIG. 5 – La nouvelle hiérarchie de concepts après apprentissage. Le premier niveau correspond à la sortie de la phase d'apprentissage sur les attributs propres, le second correspond à la sortie de la phase d'apprentissage sur les relations spatiales.

## Références

- Adams, R. et L. Bischof (1994). Seeded region growing. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 16(6), 641–647.
- Belarte, B. et C. Wemmert (2012). Apport des relations spatiales dans l'extraction automatique d'informations à partir d'image. In *Atelier Fouille de données complexes, Conférence EGC 2012*.
- Lance, G. N. et W. T. Williams (1967). A general theory of classificatory sorting strategies : 1. hierarchical systems. 9(4), 373–380.
- Lee, C. (1990a). Fuzzy logic in control systems : fuzzy logic controller. i. *Systems, Man and Cybernetics, IEEE Transactions on* 20(2), 404–418.
- Lee, C. (1990b). Fuzzy logic in control systems : fuzzy logic controller. II. *Systems, Man and Cybernetics, IEEE Transactions on* 20(2), 419–435.
- Ward, J. H. (1963). Hierarchical grouping to optimize an objective function. *Journal of the American Statistical Association* 58(301), 236–244.
- Zadeh, L. (1965). Fuzzy sets. *Information and Control* 8(3), 338–353.
- Zadeh, L. (1975). Fuzzy logic and approximate reasoning. *Synthese* 30(3), 407–428.

## Summary

The technological evolution gives us access to images with a high level of detail. Segmentation algorithms allow us to extract objects of interest from these images. An expert can describe the interest objects in the image. However, segments do not match interest objects. In this article we propose a method for learning rules which can describe how an interest object is split regarding a segmentation. Current experiments are presented.

# Un modèle générique pour les k-moyennes à recouvrements ajustables

Chiheb-Eddine ben N'Cir\*, Guillaume Cleuziou\*\*, Nadia Essoussi\*

\*LARODEC

ISGT, Université de Tunis - Tunis, TUNISIE

\*\*LIFO

Université d'Orléans - Orléans, FRANCE

**Résumé.** La classification non-supervisée recouvrante (ou *overlapping clustering*) consiste à organiser un ensemble d'individus non-étiquetés en classes, non nécessairement disjointes, d'individus similaires. La détection de structures recouvrantes est une problématique à part entière rencontrant de nouveaux défis théoriques mais offrant des perspectives de solutions pertinentes pour de nombreux domaines d'application. Cet article présente un nouveau modèle de classification généralisant l'approche des k-moyennes afin d'extraire des clusters recouvrants. Nous proposons différentes instanciations de ce modèle afin de générer différentes configurations de recouvrements entre clusters : fixes, paramétrables ou auto-ajustables. Les premières expérimentations réalisées montrent que le nouveau modèle proposé permet d'améliorer la qualité des clusters et la précision des recouvrements comparativement aux méthodes de la littérature.

## 1 Introduction

La classification non-supervisée recouvrante (ou *overlapping clustering*) consiste à organiser un ensemble d'individus non-étiquetés en classes, non nécessairement disjointes, d'individus similaires. Cette tâche de fouille de données trouve ses applications dans de nombreux domaines tels que le marketing, la biologie, la recherche d'informations, les réseaux sociaux, etc. Fondée sur l'hypothèse que chaque individu est susceptible d'appartenir totalement à plusieurs classes simultanément, les modèles utilisés en classification recouvrante s'affranchissent de tout coefficient d'appartenance probabiliste ou possibiliste conduisant à partager un individu sur plusieurs classes. Cependant, bien que les recouvrements entre classes soient tolérés voire encouragés par les modèles, il convient de contrôler leur importance afin de ne pas altérer l'objectif principal du clustering, à savoir l'extraction de classes d'individus séparés.

L'objectif de la présente étude est de considérer le modèle de classification recouvrante OKM (*Overlapping-k-means*) proposé par Cleuziou (2008), qui souffre justement d'un manque de contrôle dans les recouvrements générés. En nous appuyant sur le modèle OKM, nous proposons un nouveau modèle générique pour lequel nous présentons différentes instanciations afin de générer différentes configurations de recouvrements entre clusters : fixes, paramétrables ou auto-ajustables.

La suite de l'article est organisée ainsi : la Section 2 donne un aperçu des différentes approches de classification recouvrante et en particulier de l'algorithme OKM. La Section 3 présente l'idée du nouveau modèle proposé avant d'en donner une formalisation générique en Section 4. Les deux dernières sections 5 et 6 exposent respectivement les premiers résultats expérimentaux obtenus puis les perspectives de l'étude.

## 2 Contexte de l'étude

La tâche de classification (non-supervisée) recouvrante a été étudiée et partiellement résolue durant les trente dernières années par une série d'études et de propositions de deux types : des solutions *heuristiques* ou des études *théoriques*.

Nous appelons *heuristiques* les solutions qui consistent soit à modifier les sorties d'approches usuelles de clustering (e.g. k-moyennes ou k-moyennes flou) tel que proposé dans Lingras et West (2004) ou Zhang et al. (2007), soit à proposer un nouveau processus intuitif de construction de classes recouvrantes tel que l'algorithme CBC (*Clustering by Committee*) proposé par Pantel et Lin (2002) ou encore POBOC (*Pole-Based Overlapping Clustering*) proposé par Cleuziou et al. (2004). Ces deux types de contributions peuvent conduire à des résultats pertinents sans toutefois s'appuyer sur des modèles théoriques, limitant ainsi la possibilité de les améliorer ou de les généraliser.

Les études *théoriques* sont, en revanche, des extensions de modèles usuels de classification non-supervisée, tels que les approches hiérarchiques, génératives ou à base de graphes. Les variantes recouvrantes des hiérarchies sont les pyramides (Diday, 1984) et plus généralement des hiérarchies faibles (Bertrand et Janowitz, 2003) ; elles visent à améliorer la correspondance entre l'indice de distance induit par la structure et la mesure de dissimilarité initiale. Cependant les structures pseudo-hiérarchiques recouvrantes sont soit restrictives en terme de configuration des recouvrements, soit complexes à générer et à visualiser.

Plus récemment, les modèles de mélanges recouvrant ont été introduits (Banerjee et al., 2005; Heller et Ghahramani, 2007; Fu et Banerjee, 2008) ; ils sont motivés par la modélisation de processus biologiques et se fondent sur l'hypothèse que chaque observation est le résultat d'un mélange de lois et de combinaisons additives (Banerjee et al., 2005) ou multiplicatives (Heller et Ghahramani, 2007; Fu et Banerjee, 2008) de ces lois. Ce formalisme probabiliste permet de considérer non seulement des distributions gaussiennes mais peut se généraliser à toute loi exponentielle ; en revanche, les modèles génératifs ne sont pas paramétrables et n'autorisent pas le contrôle des recouvrements notamment.

Parce que nous nous plaçons dans le contexte où l'expert peut avoir des attentes légitimes sur l'importance des recouvrements qu'il souhaite autoriser, nous nous intéressons à un modèle plus "simple", l'approche des k-moyennes et en particulier à sa variante recouvrante OKM (Cleuziou, 2008).

La méthode OKM introduit la notion de recouvrement au niveau du critère objectif qui modélise une somme d'erreurs locales pour chaque individu à classer  $x$  ; l'erreur locale étant définie par la distance (Euclidienne et élevée au carré) entre l'individu  $x$  et son représentant (ou *image* notée  $im(x)$ ) dans la classification. Étant donné un ensemble  $X$  de  $N$  individus sur  $\mathbb{R}^P$  à organiser en  $K$  classes recouvrantes, OKM est guidé par la recherche d'une matrice binaire

d’affectations  $\Pi(N \times K)$  et d’un ensemble de représentants de classes  $C = \{c_1, \dots, c_K\}$  minimisant le critère objectif suivant :

$$J_{OKM}(\Pi, C) = \sum_{x_i \in X} \|x_i - im_{\Pi, C}(x_i)\|^2. \quad (1)$$

où  $im_{\Pi, C}(x_i)$  est défini par un combinaison (additive) de représentants :

$$im_{\Pi, C}(x_i) = \sum_{k \in \Pi_i} \frac{c_k}{|\Pi_i|}, \quad (2)$$

où  $\Pi_i$  est l’ensemble des  $|\Pi_i|$  (indices de) classes auxquelles  $x_i$  est affecté.

La minimisation du critère objectif (1) est réalisée par l’itération de deux étapes :

1. le calcul des représentants ( $C$ )
2. l’affectation des individus à une ou plusieurs classes ( $\Pi$ ).

Le calcul des représentants de classes peut être réalisé localement de manière optimale. En revanche, l’étape d’affectation nécessite le recours à une stratégie d’exploration de l’espace (exponentiel) des combinaisons possibles que nous évoquerons plus loin dans l’article (cf. Cleuziou (2008)).

### 3 OKM avec recouvrements restreints

Afin d’étudier les schémas recouvrants produits par OKM, nous visualisons les cellules de Voronoï obtenues pour trois classes en dimension deux. Dans l’exemple illustré en Figure 1.(a), l’espace de représentation est divisé en plusieurs zones correspondant chacune à un cluster ou une combinaison (ou intersection) de clusters. Chaque zone est pourvue d’un représentant ou combinaison de représentants. On peut alors observer l’importance des zones de recouvrement avec OKM : par exemple on note que les cellules de Voronoï associées au cluster 3 couvrent plus de 50% de l’espace de représentation<sup>1</sup>, sur cet exemple. Bien que, par nature, la problématique de classification recouvrante remette en cause l’une des propriétés fondamentale de la classification non-supervisée, à savoir la nécessité de détecter des clusters “bien séparés”, il n’est pas souhaitable pour autant d’autoriser des recouvrements trop larges entre les classes. En effet, dans la plupart des domaines d’applications, les classes attendues correspondent à des concepts spécifiques (sémantiques, thématiques, biologiques, etc.) qui se recouvrent modérément. Le fait que OKM génère de larges recouvrements est directement lié au critère objectif utilisé. En effet, pour une donnée, ce modèle définit comme “meilleure (multi)affectation” la combinaison de représentants la plus proche, quelque-soit le nombre de clusters mis en jeu dans cette combinaison. De cette façon, les clusters ainsi que leurs combinaisons tendent à couvrir des zones de tailles potentiellement comparables. L’idée directrice de notre étude est donc de gérer différemment les clusters et les combinaisons de clusters de manière à limiter les affectations multiples aux cas où l’amélioration induite est réellement significative.

Nous proposons une première modélisation consistant à introduire une pondération visant à réguler les erreurs locales en fonction du nombre d’affectations des individus. Ainsi, l’affectation  $\Pi_i$  d’un individu  $x_i$  à une seule classe ( $|\Pi_i| = 1$ ) sera délaissée au profit d’une affectation

1. La même remarque peut être faite pour les clusters 1 et 2.

## K-moyennes à recouvrements ajustables

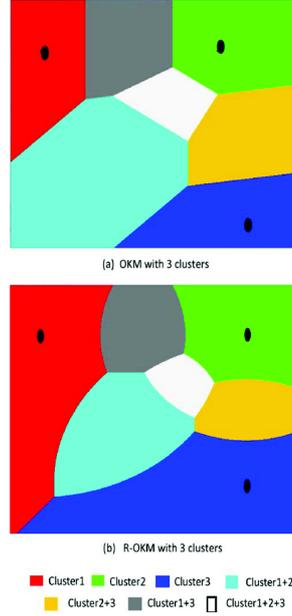


FIG. 1 – Cellules de Voronoï obtenues avec OKM et R-OKM pour 3 classes.

$\Pi'_i$  à deux classes ( $|\Pi'_i| = 2$ ) si et seulement si  $2.d^2(x_i, im_{\Pi', C}(x_i)) < d^2(x_i, im_{\Pi, C}(x_i))$ ; autrement dit, l'erreur induite par une double affectation devra être deux fois plus petite que l'erreur induite par une affectation simple.

De manière généralisée on appelle R-OKM (*Restricted-OKM*) le modèle suivant :

$$\begin{aligned}
 J(\Pi, C) &= \sum_{x_i \in X} |\Pi_i| \cdot \|x_i - im_{\Pi, C}(x_i)\|^2 \\
 &= \sum_{k=1}^K \sum_{x_i \in \pi_k} \|x_i - im_{\Pi, C}(x_i)\|^2 .
 \end{aligned} \tag{3}$$

et on montre en Figure 1.(b) une visualisation des recouvrements, plus réduits, obtenus à l'aide de ce nouveau modèle. R-OKM reste une généralisation de k-means : la restriction des affectations à une seule classe ( $\forall i, |\Pi_i| = 1$ ) se ramène à considérer exactement le critère des moindres carrés usuel.

Cette première proposition a le mérite de limiter la taille des recouvrements sans pour autant contrôler la pénalisation introduite sur les multi-affectations. L'objectif recherché dans la suite de l'article est de proposer deux nouvelles instanciations de ce modèle pénalisé afin de permettre un paramétrage externe puis interne d'ajustement des recouvrements.

## 4 Un modèle générique

Nous généralisons le modèle R-OKM précédent en introduisant une pondération (ou pénalisation)  $\omega_i$  associée à chaque individu  $x_i$  :

$$J(\Pi, C, \Omega) = \sum_{x_i \in X} \omega_i \cdot \|x_i - im_{\Pi, C}(x_i)\|^2. \quad (4)$$

Ce modèle peut être instancié de différentes manières, par exemple, en choisissant  $\forall i, \omega_i = |\Pi_i|$  on obtient le modèle R-OKM précédent et pour  $\forall i, \omega_i = 1$  on retrouve le critère objectif utilisé pour OKM.

Pour offrir à l'utilisateur la possibilité de réguler la taille des recouvrements, nous proposons une nouvelle instanciation  $\omega_i = |\Pi_i|^\alpha$  où  $\alpha \geq 0$  correspond au paramètre à ajuster. Les affectations multiples seront ainsi d'avantage pénalisées pour  $\alpha \rightarrow +\infty$  et au contraire, les recouvrements seront plus importants et tendront vers les recouvrements initialement produits par OKM lorsque  $\alpha \rightarrow 0$ . Dans ce qui suit, cette méthode sera appelée : R-OKM *paramétré*.

Nous proposons enfin une dernière instanciation de (4) visant à moduler le niveau de pénalisation en fonction des classes associées dans une combinaison. Nous définissons alors la nouvelle pondération  $\omega_i$  comme une somme de paramètres  $\lambda_k$  locaux aux clusters associés dans l'affectation  $\Pi_i$  :

$$\omega_i = \sum_{k \in \Pi_i} (\lambda_k)^\beta \text{ avec } \sum_{k=1}^K \lambda_k = 1 \text{ et } \beta \in ]0, 1[$$

Le critère objectif qui guidera ce nouveau modèle, que nous appellerons R-OKM *ajusté*, est donné par

$$J(\Pi, C, \Lambda) = \sum_{x_i \in X} \sum_{k \in \Pi_i} (\lambda_k)^\beta \cdot \|x_i - im_{\Pi, C}(x_i)\|^2, \quad (5)$$

le paramètre  $\beta$  est fixé et permettra de contrôler la variance sur l'ensemble des paramètres locaux  $\{\lambda_k\}_{k=1}^K$  : la variance sera faible pour  $\beta \rightarrow 0$  ( $\forall k, \lambda_k^\beta \approx \frac{1}{K}$ ) et plus élevée pour  $\beta \rightarrow 1$ .

La minimisation de la fonction objective générique (4) est réalisée par itération de deux ou trois étapes, selon l'instanciation choisie : (1) calcul des représentants de classes ( $C$ ), (2) affectation des individus à une ou plusieurs classes ( $\Pi$ ) et (3) calcul des pondérations ( $\Omega$ ). Ces trois étapes sont réalisées de façon indépendante, chacune assurant la diminution du critère objectif. Le Tableau 1 donne les règles de mises à jour pour chaque modèle considéré (R-OKM, R-OKM *paramétré* et R-OKM *ajusté*).

1. **Étape 1** : étant données les affectations ( $\Pi$ ) et les pondérations ( $\Omega$ ) fixées, les représentants de classes sont recalculés successivement et leur règle de mise à jour est obtenue par dérivation de la fonction objective (troisième colonne du Tableau 1<sup>2</sup>).

2. La notation  $\tilde{x}_i^k$  dans le Tableau 1 désigne le représentant de classe  $c_k$  idéal vis-à-vis de l'individu  $x_i$ , tel que  $\|x_i - im_{\Pi, C}(x_i)\|^2 = 0$ . Il est défini formellement par  $\tilde{x}_i^k = |\Pi_i| \cdot x_i - (|\Pi_i| - 1) \cdot im_{\Pi_i \setminus k, C}(x_i)$

## K-moyennes à recouvrements ajustables

TAB. 1 – *Mises à jour optimales pour les différentes instanciations de R-OKM*

Modèles	Instanciations $\omega_i$	Représentants de classes $c_k^*$	Paramètres des classes $\lambda_k^*$
OKM	1	$\frac{\sum_{x_i \in \pi_k} \frac{1}{ \Pi_i ^2} \cdot \tilde{x}_i^k}{\sum_{x_i \in \pi_k} \frac{1}{ \Pi_i ^2}}$	–
R-OKM	$ \Pi_i $	$\frac{\sum_{x_i \in \pi_k} \frac{1}{ \Pi_i } \cdot \tilde{x}_i^k}{\sum_{x_i \in \pi_k} \frac{1}{ \Pi_i }}$	–
R-OKM paramétré	$ \Pi_i ^\alpha$	$\frac{\sum_{x_i \in \pi_k} \frac{1}{ \Pi_i ^{2-\alpha}} \cdot \tilde{x}_i^k}{\sum_{x_i \in \pi_k} \frac{1}{ \Pi_i ^{2-\alpha}}}$	–
R-OKM ajusté	$\sum_{k \in \Pi_i} (\lambda_k)^\beta$	$\frac{\sum_{k \in \Pi_i} \frac{(\lambda_k)^\beta}{ \Pi_i ^2} \cdot \tilde{x}_i^k}{\sum_{x_i \in \pi_k} \frac{1}{ \Pi_i ^2}}$	$\frac{1}{\sum_{l=1}^K \left( \frac{\sum_{x_i \in \pi_k} \ x_i - im_{\Pi, C}(x_i)\ ^2}{\sum_{x_j \in \pi_l} \ x_j - im_{\Pi, C}(x_j)\ ^2} \right)^{\frac{1}{\beta-1}}}$

- Étape 2 : la phase d'affectation (pour  $C$  et  $\Omega$  fixés) correspond à un problème (combinatoire) d'optimisation discrète qui ne peut se résoudre directement. Nous proposons dans la suite d'adopter la stratégie d'exploration utilisée dans OKM en l'adaptant au nouveau modèle.
- Étape 3 : cette étape ne concerne que le modèle R-OKM *ajusté* ; la mise à jour des poids  $\Omega$  passe par le calcul des paramètres locaux  $\Lambda$  qui s'obtiennent de manière optimale indépendamment pour chaque cluster par résolution d'un problème classique d'optimisation sous-contraite (dernière colonne du Tableau 1).

On observera, par leur règle de mise à jour, que les paramètres de classes  $\Lambda$  quantifient l'erreur locale induite par chaque classe, que l'on peut voir également comme un critère de qualité associé à chaque classe. En choisissant  $\beta < 1$  la pénalisation  $\omega_i$  associée à l'affectation  $\Pi_i$  sera élevée pour des combinaisons composées de classes de haute qualité, limitant ainsi les affectations multiples vers ces classes.

L'algorithme 1<sup>3</sup> présente un pseudo-code générique pour le nouveau modèle proposé. Cet algorithme utilise la fonction *Restricted\_ASSIGN* qui définit la stratégie d'affectation utilisée. Cette stratégie consiste, pour chaque individu  $x_i$ , à ordonner les classes selon leur proximité en utilisant  $\|x_i - c_k\|^2$ , puis à affecter successivement  $x_i$  à chacune des classes tant que les nouvelles affectations améliorent le critère objectif (Algorithme 2).

3. L'étape 5 dans l'algorithme 1 est seulement utilisée pour le modèle R-OKM *ajusté*.

---

**Algorithm 1** *Restricted overlapping k-means*  $(X, t_{max}, \varepsilon, K) \rightarrow \Pi$

---

**Require:**  $X$  : a data set described over  $\mathbb{R}^d$ .

$t_{max}$  : maximum number of iterations.

$\varepsilon$  : minimal improvement in the objective function.

$K$  : number of clusters.

**Ensure:**  $\Pi$  : assignment of observations over  $K$  clusters.

- 1: Initialize the representatives ( $C^0$ ) of clusters randomly over  $X$ , initialize weights  $\Lambda^0$  ( $\forall k, \lambda_k^0 = 1/K$ ), initialize clusters memberships  $\Pi^0$  using "*Restricted\_ASSIGN*" and compute the objective function  $J(\Pi^0, C^0, \Lambda^0)$  at iteration 0.
  - 2:  $t = t + 1$
  - 3: Update cluster representatives  $C^t$  (using Table 1 column 3).
  - 4: Compute new assignments  $\Pi^t$  using "*Restricted\_ASSIGN*"( $x_i, C^t, \Pi_i^{t-1}$ ) for each  $x_i$ ".
  - 5: Update weights  $\Lambda^t$  (using Table 1 column 4).
  - 6: Compute objective function  $J(\Pi^t, C^t, \Lambda^t)$ .
  - 7: **if** ( $t < t_{max}$  and  $J(\Pi^{t-1}, C^{t-1}, \Lambda^{t-1}) - J(\Pi^t, C^t, \Lambda^t) > \varepsilon$ ) **then**
  - 8:     Go to step 2.
  - 9: **else**
  - 10:     Return  $\Pi^t$  the final cluster memberships matrix.
  - 11: **end if**
- 

---

**Algorithm 2** *Restricted\_ASSIGN* $(x_i, \{c_1, \dots, c_K\}, \Pi_i^{old}) \rightarrow \Pi_i$

---

**Require:**  $x_i$  : Vector in  $\mathbb{R}^d$ .

$\{c_1, \dots, c_K\}$  :  $K$  cluster representatives.

$\Pi_i^{old}$  : Old assignment for observation  $x_i$ .

**Ensure:**  $\Pi_i$  : New assignment for  $x_i$ .

- 1: Initialize  $\Pi_i = \{c^*\}$  the nearest cluster where  

$$c^* = \min_{c_k: k=1..K} \|x_i - c_k\|^2.$$
  - 2: Looking for the next nearest cluster  $c^*$  which is not included in  $\Pi_i$ .
  - 3: Compute  $im_{\Pi', C}(x_i)$  and  $\omega'_i$  with assignments  $\Pi'_i = \Pi_i \cup \{c^*\}$ .
  - 4: **if**  $\omega'_i \cdot \|x_i - im_{\Pi', C}(x_i)\|^2 < \omega_i \cdot \|x_i - im_{\Pi, C}(x_i)\|^2$  **then**
  - 5:      $\Pi_i \leftarrow \Pi'_i$  and go to step 2.
  - 6: **else**
  - 7:     **if**  $\omega_i \cdot \|x_i - im_{\Pi, C}(x_i)\|^2 \leq \omega_i^{old} \cdot \|x_i - im_{\Pi^{old}, C}(x_i)\|^2$  **then**
  - 8:         Return  $\Pi_i$ .
  - 9:     **else**
  - 10:         Return  $\Pi_i^{old}$ .
  - 11:     **end if**
  - 12: **end if**
-

TAB. 2 – Statistiques des jeux de données utilisés

Jeu de données	Domaine	Nb. observations	Dimensionnalité	Nb. labels	Taux de recouvrement
EachMovie	Video	75	3	3	1.14
Music emotion	Musique	593	72	6	1.86
Scene	Image	2407	192	6	1.07
Yeast	Biologie	2417	103	14	4.23

Le modèle générique des  $k$ -moyennes recouvrant pénalisé, introduit par le critère (4), nous a permis de proposer trois instanciations donnant lieu chacune à un algorithme : R-OKM, R-OKM *paramétré* et R-OKM *ajusté*. Chaque méthode offre des possibilités différentes en matière de recouvrements, ce sont ces caractéristiques que nous cherchons à observer et confirmer dans la section suivante.

## 5 Expérimentations

L'évaluation des méthodes de classification non-supervisée est réputée difficile puisqu'il n'existe pas, par définition, de classification exacte sur laquelle se fonder pour mesurer la qualité d'un schéma de classification obtenu. Ainsi trois stratégies complémentaires d'évaluation peuvent être envisagées : l'évaluation interne qui cherchera à évaluer dans quelle mesure certaines propriétés attendues d'une méthode ont effectivement été vérifiées, l'évaluation externe qui supposera l'existence d'une classification de référence (avec toutes les incertitudes que ce type d'évaluation entraîne) et enfin l'évaluation experte qui permet de confronter les résultats d'une méthode aux attentes d'experts sur des applications spécifiques.

Nous proposons une brève évaluation interne suivie d'une évaluation externe plus complète.

La Figure 2 permet d'illustrer et de valider visuellement le paramétrage des recouvrements introduits dans la méthode R-OKM *paramétré*. Sur l'exemple déjà utilisé en Figure 1 on constate que l'augmentation du paramètre  $\alpha$  dans  $]0, +\infty[$  permet d'évoluer de la classification générée par OKM ( $\alpha \rightarrow 0$ ) vers celle générée par  $k$ -means ( $\alpha \rightarrow +\infty$ ), en passant par des schémas aux recouvrements limités.

Nous proposons ensuite une évaluation sur des jeux de données "benchmarks" multi-classes traditionnellement utilisés en classification supervisée. Pour chaque jeu de données (cf. Tableau 2), chaque observation à classer est étiquetée par un ou plusieurs labels. Les classifications de référence sont donc recouvrantes et nous cherchons à savoir si les méthodes proposées permettent de retrouver des organisations naturelles qui s'apparenteraient à ces classifications de références.

Dans ce tableau, le taux de recouvrement correspond au nombre moyen de labels par observation ; dans les schémas générés par la suite, ce taux correspondra au nombre moyen d'affectations par observation. Nous reportons dans le Tableau 3, les taux de recouvrements obtenus par les méthodes proposées dans cette étude, sur les 4 jeux de données présentés précédemment

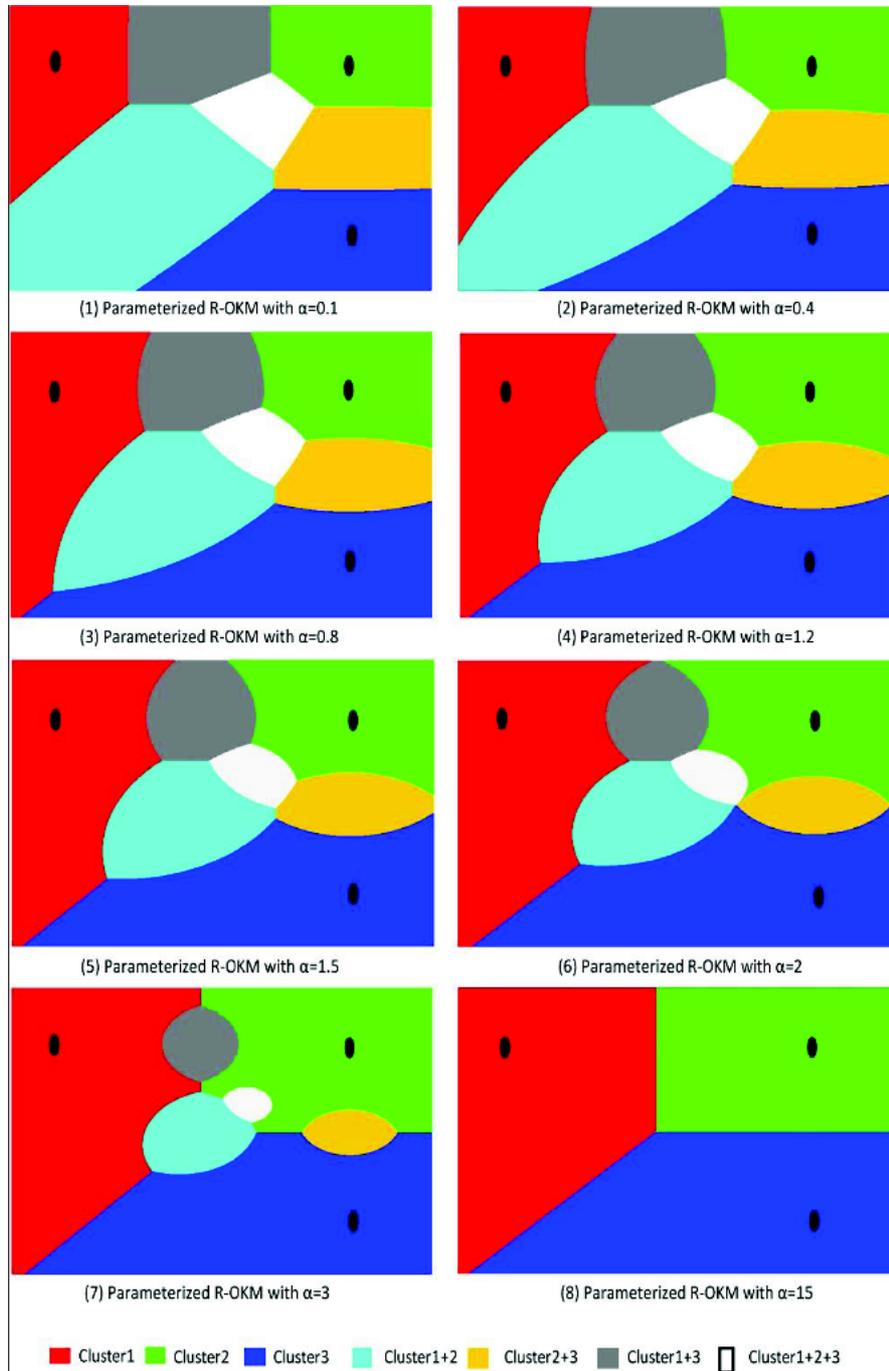


FIG. 2 – Cellules de Voronoï obtenues avec la méthode R-OKM paramétré pour différentes valeurs du paramètre  $\alpha$  et pour 3 classes (en dimension 2).

## K-moyennes à recouvrements ajustables

TAB. 3 – Taux de recouvrement obtenus comparativement à OKM et  $k$ -moyennes flou.

Jeu de données	Iris	Eachmovie	Music Emotion	Scene	Yeast
Taux de recouvrement	(1)	(1.14)	(1.81)	(1.07)	(4.23)
$k$ -moyennes	<b>1.00</b> ± 0.00	1.00 ± 0.00	1.00 ± 0.00	1.00 ± 0.00	1.00 ± 0.00
$k$ -moyennes flou ( $seuil = 0.4$ )	1.04 ± 0.01	0.96 ± 0.00	0.89 ± 0.00	0.00 ± 0.00	0.00 ± 0.00
$k$ -moyennes flou*	1.22 ± 0.02	1.52 ± 0.00	1.43 ± 0.00	5.96 ± 0.03	12.78 ± 0.16
OKM	1.34 ± 0.12	1.76 ± 0.22	2.35 ± 0.20	2.33 ± 0.08	<b>4.75</b> ± 0.23
R-OKM	1.16 ± 0.03	1.21 ± 0.11	1.31 ± 0.04	<b>1.03</b> ± 0.04	1.00 ± 0.00
Parameterized R-OKM ( $\alpha = 0.01$ )	1.72 ± 0.06	1.70 ± 0.07	2.11 ± 0.05	2.34 ± 0.11	<b>4.48</b> ± 0.22
R-OKM paramétré ( $\alpha = 0.1$ )	1.31 ± 0.04	1.68 ± 0.07	<b>1.84</b> ± 0.05	1.66 ± 0.09	2.75 ± 0.10
R-OKM paramétré ( $\alpha = 0.4$ )	1.25 ± 0.02	1.49 ± 0.03	1.69 ± 0.02	<b>1.03</b> ± 0.02	1.03 ± 0.04
R-OKM paramétré ( $\alpha = 0.6$ )	1.24 ± 0.02	1.29 ± 0.02	1.57 ± 0.04	1.01 ± 0.01	1.00 ± 0.00
R-OKM paramétré ( $\alpha = 0.8$ )	1.21 ± 0.03	1.28 ± 0.10	1.39 ± 0.08	1.01 ± 0.01	1.00 ± 0.00
R-OKM paramétré ( $\alpha = 1.2$ )	1.13 ± 0.03	<b>1.14</b> ± 0.07	1.22 ± 0.05	1.00 ± 0.00	1.00 ± 0.00
R-OKM paramétré ( $\alpha = 1.5$ )	1.10 ± 0.02	<b>1.11</b> ± 0.08	1.20 ± 0.02	1.00 ± 0.00	1.00 ± 0.00
R-OKM paramétré ( $\alpha = 2$ )	1.05 ± 0.01	1.05 ± 0.08	1.14 ± 0.04	1.00 ± 0.00	1.00 ± 0.00
R-OKM paramétré ( $\alpha = 3$ )	<b>1.01</b> ± 0.00	1.03 ± 0.03	1.05 ± 0.03	1.00 ± 0.00	1.00 ± 0.00
R-OKM paramétré ( $\alpha = 5$ )	<b>1.00</b> ± 0.00	1.00 ± 0.00	1.00 ± 0.01	1.00 ± 0.00	1.00 ± 0.00
R-OKM ajusté ( $\beta = 0.6$ )	1.18 ± 0.01	<b>1.16</b> ± 0.08	1.30 ± 0.10	1.00 ± 0.00	1.00 ± 0.00
R-OKM ajusté ( $\beta = 0.4$ )	1.17 ± 0.02	<b>1.18</b> ± 0.10	1.30 ± 0.08	1.00 ± 0.00	1.00 ± 0.00
R-OKM ajusté ( $\beta = 0.1$ )	1.16 ± 0.01	1.21 ± 0.12	1.27 ± 0.07	1.01 ± 0.01	1.00 ± 0.00
R-OKM ajusté ( $\beta = 0.05$ )	1.16 ± 0.03	1.21 ± 0.10	1.27 ± 0.03	1.01 ± 0.01	1.00 ± 0.00
R-OKM ajusté ( $\beta = 0.01$ )	1.16 ± 0.03	1.20 ± 0.10	1.28 ± 0.03	1.01 ± 0.01	1.00 ± 0.00

ainsi que sur les données Iris dont la classification de référence n'est pas recouvrante mais dont on sait que deux classes sont difficiles à distinguer. Les performances obtenues sont comparées avec des algorithmes couramment utilisés :  $k$ -moyennes (sans recouvrement),  $k$ -moyennes flou complété d'une étape d'affectation seuillée ( $k$ -moyennes flou\* correspond à un seuil optimal ajusté empiriquement) et OKM.

Nous terminons cette évaluation par une comparaison qualitative des schémas obtenus en utilisant les indices de Rand et du F-Score (combinant rappel et précision) qui quantifient chacun la correspondance entre les paires d'observations associées dans la classification obtenue et la classification de référence. Le Tableau 4 présente les résultats obtenus sur des moyennes de 10 exécutions dans des conditions semblables (mêmes initialisations).

## 6 Conclusion

Nous avons proposé dans cette étude un modèle générique de classification recouvrante avec contrôle des recouvrements. Ce modèle peut être instancié de manière fixe, paramétré ou auto-ajustée donnant lieu à trois nouvelles méthodes (R-OKM, R-OKM paramétré et R-OKM ajusté) vues comme autant de généralisations recouvrantes des  $k$ -moyennes.

Pour compléter cette étude nous envisageons, dans une version plus étendue, de proposer une analyse plus détaillée et discutée des résultats expérimentaux présentés ici ainsi qu'une évaluation experte sur un jeu de données réel lié au domaine de la recherche d'information.

TAB. 4 – Évaluation externe des trois méthodes proposées, comparativement aux algorithmes  $k$ -moyennes,  $k$ -moyennes flou et OKM

Data set	Method	Precision	Recall	F-measure	Rand Index	
Eachmovie	$k$ -moyennes	0.644 ± 0.100	0.587 ± 0.036	0.610 ± 0.070	0.689 ± 0.060	
	$k$ -moyennes flou ( $\sigma = 0.2$ )	0.523 ± 0.003	0.847 ± 0.003	0.647 ± 0.003	0.620 ± 0.001	
	$k$ -moyennes flou ( $\sigma = 0.3$ )	0.610 ± 0.001	0.734 ± 0.001	<b>0.666</b> ± 0.001	<b>0.696</b> ± 0.001	
	$k$ -moyennes flou ( $\sigma = 0.4$ )	<b>0.691</b> ± 0.001	0.523 ± 0.012	0.596 ± 0.007	<b>0.706</b> ± 0.003	
	$k$ -moyennes flou ( $\sigma = 0.6$ )	0.648 ± 0.001	0.265 ± 0.001	0.377 ± 0.001	0.637 ± 0.001	
	OKM	0.465 ± 0.020	<b>0.921</b> ± 0.055	0.618 ± 0.030	0.532 ± 0.032	
	R-OKM	0.532 ± 0.010	0.764 ± 0.003	0.627 ± 0.006	0.625 ± 0.011	
	R-OKM paramétré( $\alpha = 0.1$ )	0.474 ± 0.016	<b>0.901</b> ± 0.043	0.621 ± 0.024	0.547 ± 0.024	
	R-OKM paramétré( $\alpha = 0.4$ )	0.498 ± 0.029	0.846 ± 0.062	0.627 ± 0.023	0.584 ± 0.042	
	R-OKM paramétré( $\alpha = 0.9$ )	0.535 ± 0.013	0.763 ± 0.020	0.629 ± 0.016	0.629 ± 0.015	
	R-OKM paramétré( $\alpha = 1.5$ )	0.591 ± 0.071	0.700 ± 0.008	0.639 ± 0.044	0.672 ± 0.062	
	R-OKM paramétré( $\alpha = 3$ )	0.627 ± 0.136	0.601 ± 0.063	0.605 ± 0.083	0.672 ± 0.094	
	R-OKM paramétré( $\alpha = 5$ )	<b>0.645</b> ± 0.101	0.587 ± 0.037	0.610 ± 0.089	<b>0.689</b> ± 0.060	
	R-OKM ajusté ( $\beta = 0.4$ )	0.534 ± 0.039	0.639 ± 0.134	0.580 ± 0.080	0.624 ± 0.040	
	R-OKM ajusté ( $\beta = 0.1$ )	0.573 ± 0.058	0.735 ± 0.038	0.642 ± 0.034	<b>0.661</b> ± 0.051	
	R-OKM ajusté ( $\beta = 0.05$ )	0.568 ± 0.028	0.757 ± 0.015	0.649 ± 0.014	<b>0.661</b> ± 0.014	
	R-OKM ajusté ( $\beta = 0.01$ )	0.564 ± 0.030	0.745 ± 0.027	0.641 ± 0.028	<b>0.656</b> ± 0.026	
	Emotion	$k$ -moyennes	0.507 ± 0.003	0.206 ± 0.012	0.293 ± 0.010	<b>0.532</b> ± 0.002
		$k$ -moyennes flou ( $\sigma = 0.2$ )	0.493 ± 0.003	0.357 ± 0.001	0.414 ± 0.002	0.524 ± 0.002
		$k$ -moyennes flou ( $\sigma = 0.3$ )	<b>0.505</b> ± 0.001	0.246 ± 0.001	0.331 ± 0.001	<b>0.531</b> ± 0.002
$k$ -moyennes flou ( $\sigma = 0.4$ )		0.501 ± 0.001	0.164 ± 0.001	0.247 ± 0.001	0.529 ± 0.002	
$k$ -moyennes flou ( $\sigma = 0.6$ )		0.503 ± 0.002	0.082 ± 0.002	0.146 ± 0.002	0.529 ± 0.001	
OKM		0.483 ± 0.001	<b>0.646</b> ± 0.031	<b>0.552</b> ± 0.012	0.508 ± 0.001	
R-OKM		<b>0.505</b> ± 0.004	0.335 ± 0.058	0.401 ± 0.040	<b>0.532</b> ± 0.002	
R-OKM paramétré( $\alpha = 0.1$ )		0.480 ± 0.002	0.581 ± 0.087	<b>0.524</b> ± 0.036	0.506 ± 0.006	
R-OKM paramétré( $\alpha = 0.4$ )		0.484 ± 0.012	0.525 ± 0.079	0.503 ± 0.041	0.514 ± 0.010	
R-OKM paramétré( $\alpha = 0.9$ )		0.491 ± 0.012	0.378 ± 0.050	0.426 ± 0.031	0.522 ± 0.009	
R-OKM paramétré( $\alpha = 1.5$ )		0.500 ± 0.002	0.286 ± 0.019	0.363 ± 0.015	0.529 ± 0.001	
R-OKM paramétré( $\alpha = 3$ )		0.501 ± 0.007	0.218 ± 0.013	0.304 ± 0.010	0.529 ± 0.003	
R-OKM paramétré( $\alpha = 5$ )		<b>0.507</b> ± 0.003	0.207 ± 0.014	0.294 ± 0.012	<b>0.532</b> ± 0.002	
R-OKM ajusté ( $\beta = 0.4$ )		0.484 ± 0.004	0.280 ± 0.071	0.352 ± 0.054	0.509 ± 0.054	
R-OKM ajusté ( $\beta = 0.1$ )		0.483 ± 0.016	0.558 ± 0.044	0.479 ± 0.030	0.504 ± 0.034	
R-OKM ajusté ( $\beta = 0.05$ )		0.495 ± 0.022	0.329 ± 0.050	0.393 ± 0.028	0.524 ± 0.015	
R-OKM ajusté ( $\beta = 0.01$ )		<b>0.503</b> ± 0.006	0.322 ± 0.019	0.392 ± 0.016	<b>0.531</b> ± 0.004	
Scene		$k$ -moyennes	0.441 ± 0.010	0.406 ± 0.012	0.429 ± 0.010	0.786 ± 0.004
		$k$ -moyennes flou ( $\sigma = 0.2$ )	0.000 ± 0.000	0.000 ± 0.000	0.000 ± 0.000	0.000 ± 0.000
		$k$ -moyennes flou ( $\sigma = 0.1667$ )	0.324 ± 0.004	0.482 ± 0.022	0.388 ± 0.005	0.706 ± 0.008
	$k$ -moyennes flou ( $\sigma = 0.1665$ )	0.193 ± 0.001	<b>0.996</b> ± 0.001	0.323 ± 0.001	0.193 ± 0.001	
	OKM	0.233 ± 0.006	<b>0.928</b> ± 0.013	0.372 ± 0.008	0.397 ± 0.019	
	R-OKM	0.448 ± 0.004	0.413 ± 0.058	0.430 ± 0.040	<b>0.788</b> ± 0.002	
	R-OKM paramétré( $\alpha = 0.1$ )	0.290 ± 0.001	0.765 ± 0.027	0.421 ± 0.006	0.593 ± 0.006	
	R-OKM paramétré( $\alpha = 0.4$ )	0.444 ± 0.013	0.426 ± 0.014	0.435 ± 0.013	0.786 ± 0.001	
	R-OKM paramétré( $\alpha = 0.9$ )	<b>0.448</b> ± 0.005	0.413 ± 0.008	<b>0.430</b> ± 0.007	<b>0.788</b> ± 0.002	
	R-OKM paramétré( $\alpha = 1.5$ )	<b>0.448</b> ± 0.007	0.413 ± 0.009	<b>0.430</b> ± 0.007	<b>0.788</b> ± 0.007	
	R-OKM paramétré( $\alpha = 3$ )	<b>0.448</b> ± 0.007	0.413 ± 0.009	<b>0.430</b> ± 0.007	<b>0.788</b> ± 0.007	
	R-OKM paramétré( $\alpha = 5$ )	<b>0.448</b> ± 0.007	0.413 ± 0.009	<b>0.430</b> ± 0.007	<b>0.788</b> ± 0.007	
	R-OKM ajusté ( $\beta = 0.4$ )	0.286 ± 0.093	0.582 ± 0.002	0.344 ± 0.023	0.544 ± 0.245	
	R-OKM ajusté ( $\beta = 0.1$ )	0.416 ± 0.039	0.409 ± 0.026	0.412 ± 0.023	0.773 ± 0.020	
	R-OKM ajusté ( $\beta = 0.05$ )	<b>0.454</b> ± 0.014	0.415 ± 0.012	<b>0.434</b> ± 0.012	<b>0.790</b> ± 0.005	
	R-OKM ajusté ( $\beta = 0.01$ )	<b>0.454</b> ± 0.010	0.415 ± 0.009	<b>0.433</b> ± 0.010	<b>0.790</b> ± 0.004	
	Yeast	$k$ -moyennes	<b>0.801</b> ± 0.006	0.075 ± 0.001	0.137 ± 0.002	0.260 ± 0.001
		$k$ -moyennes flou ( $\sigma = 0.2$ )	0.000 ± 0.000	0.000 ± 0.000	0.000 ± 0.000	0.000 ± 0.000
		$k$ -moyennes flou ( $\sigma = 0.0715$ )	<b>0.913</b> ± 0.087	0.009 ± 0.001	0.017 ± 0.009	0.219 ± 0.004
		$k$ -moyennes flou ( $\sigma = 0.0714$ )	0.784 ± 0.000	<b>1.000</b> ± 0.000	<b>0.879</b> ± 0.000	<b>0.784</b> ± 0.001
$k$ -moyennes flou ( $\sigma = 0.01$ )		0.784 ± 0.000	<b>1.000</b> ± 0.000	<b>0.879</b> ± 0.000	<b>0.784</b> ± 0.001	
OKM		0.783 ± 0.001	<b>0.855</b> ± 0.051	<b>0.817</b> ± 0.024	<b>0.701</b> ± 0.029	
R-OKM		<b>0.801</b> ± 0.006	0.107 ± 0.067	0.137 ± 0.002	0.260 ± 0.001	
R-OKM paramétré( $\alpha = 0.01$ )		0.783 ± 0.002	<b>0.830</b> ± 0.050	<b>0.806</b> ± 0.024	0.687 ± 0.029	
R-OKM paramétré( $\alpha = 0.05$ )		0.783 ± 0.001	0.710 ± 0.055	0.744 ± 0.031	0.618 ± 0.030	
R-OKM paramétré( $\alpha = 0.1$ )		0.786 ± 0.003	0.493 ± 0.033	0.606 ± 0.027	0.426 ± 0.161	
R-OKM paramétré( $\alpha = 0.8$ )		<b>0.801</b> ± 0.005	0.075 ± 0.001	0.137 ± 0.002	0.260 ± 0.002	
R-OKM paramétré( $\alpha = 1.5$ )		<b>0.801</b> ± 0.006	0.107 ± 0.067	0.137 ± 0.002	0.260 ± 0.002	
R-OKM paramétré( $\alpha = 3$ )		<b>0.801</b> ± 0.006	0.107 ± 0.067	0.137 ± 0.002	0.260 ± 0.002	
R-OKM ajusté ( $\beta = 0.8$ )		0.782 ± 0.002	0.736 ± 0.389	0.735 ± 0.229	0.638 ± 0.001	
R-OKM ajusté ( $\beta = 0.4$ )		0.790 ± 0.011	0.163 ± 0.023	0.270 ± 0.016	0.309 ± 0.005	
R-OKM ajusté ( $\beta = 0.1$ )		<b>0.804</b> ± 0.003	0.072 ± 0.001	0.134 ± 0.002	0.259 ± 0.160	

## Références

- Banerjee, A., C. Krumpelman, S. Basu, R. J. Mooney, et J. Ghosh (2005). Model based overlapping clustering. In *International Conference on Knowledge Discovery and Data Mining*, Chicago, USA. SciTePress.
- Bertrand, P. et M. F. Janowitz (2003). The k-weak hierarchical representations : an extension of the indexed closed weak hierarchies. *Discrete Applied Mathematics* 127(2), 199–220.
- Cleuziou, G. (2008). An extended version of the k-means method for overlapping clustering. In *International Conference on Pattern Recognition ICPR*, Florida, USA, pp. 1–4. IEEE.
- Cleuziou, G., L. Martin, et C. Vrain (2004). PoBOC : an Overlapping Clustering Algorithm. Application to Rule-Based Classification and Textual Data. In R. López de Mántaras and L. Saitta, IOS Press (Ed.), *Proceedings of the 16th European Conf. on Artificial Intelligence*, Valencia, Spain, pp. 440–444.
- Diday, E. (1984). Orders and overlapping clusters by pyramids. Technical Report 730, INRIA, France.
- Fu, Q. et A. Banerjee (2008). Multiplicative mixture models for overlapping clustering. In *Proceedings of the 8th IEEE International Conference on Data Mining*, Washington, DC, USA, pp. 791–796.
- Heller, K. et Z. Ghahramani (2007). A nonparametric bayesian approach to modeling overlapping clusters. In *AISTATS*, Puerto Ric.
- Lingras, P. et C. West (2004). Interval set clustering of web users with rough k-means. *J. Intell. Inf. Syst.* 23(1), 5–16.
- Pantel, P. et D. Lin (2002). Discovering word senses from text. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, Edmonton, Alberta, Canada, pp. 613–619. ACM Press.
- Zhang, S., R.-S. Wang, et X.-S. Zhang (2007). Identification of overlapping community structure in complex networks using fuzzy c-means clustering. *Physica A : Statistical Mechanics and its Applications* 374(1), 483–490.

## Summary

Clustering is an unsupervised learning method that enables to fit structures in unlabeled data sets. Detecting overlapping structures is a specific challenge involving its own theoretical issues but offering relevant solutions for many application domains. This paper presents a new clustering model that generalizes the k-means approach to detect overlapping clusters. We propose different instantiations of the proposed model which produce different layouts for the overlapping boundaries between clusters. These instantiations are well adapted for the parameterization of the sizes of the overlaps. The experiments show that the proposed model outperforms existing overlapping methods in terms of clustering precision and produces clusters with relevant overlapping boundaries.







**Partenaires :**

