Fouille de données complexes et logique floue : extraction de motifs à partir de bases de données multidimensionnelles

Anne Laurent

UNIVERSITÉ MONTPELLIER 2

HABILITATION À DIRIGER DES RECHERCHES

Discipline	:	Informatique
Spécialité Doctorale	:	Informatique
École Doctorale	:	Informatique, Structure, Systèmes (I2S)

Fouille de données complexes et logique floue : extraction de motifs à partir de bases de données multidimensionnelles

Présentée et soutenue publiquement par

Anne LAURENT

le 27 avril 2009

DEVANT LE JURY COMPOSE DE :

B. Bouchon-Meunier (Directrice de Recherche), CNRS, Université Paris 6, Examinatrice
C. Collet (Professeur), Institut Polytechnique de Grenoble, Présidente
E. Hüllermeier (Professeur), Philipps-Universität Marburg, Examinateur
T. Martin (Professeur), University of Bristol, Rapporteur
J. Pei (Professeur), Simon Fraser University , Rapporteur
P. Poncelet (Professeur), Université Montpellier 2, Examinateur
M. Scholl (Professeur), CNAM Paris, Rapporteur
M. Teisseire (Directrice de Recherche), CEMAGREF, Examinatrice

Table des matières

Résumé

Ce mémoire décrit mes activités de recherche et d'animation de recherche depuis ma thèse, soutenue en 2002. Les travaux décrits ici ont été principalement menés au LIRMM (Université Montpellier 2, CNRS UMR 5506), au sein de l'équipe TATOO. Dans ce contexte, je me suis attachée à concilier des visions trop souvent vues comme divergentes au sein des communautés liées à la fouille de données complexes : gérer l'approximation (à la fois dans les données et dans les résultats produits), la fouille de données *et* les bases de données complexes et volumineuses, notamment les entrepôts de données. Plus précisément, mes travaux visent à montrer qu'il est possible de relever le défi jusqu'à présent non totalement solutionné d'extraire des connaissances exploitables par les experts non informaticiens à partir d'entrepôts de données, en prenant en compte au mieux les particularités de ce domaine. En particulier, j'ai porté d'une part une grande attention à exploiter la dimension temporelle des entrepôts et d'autre part à montrer autant que faire se peut que *flou* et *passage à l'échelle* ne sont pas des notions antagonistes. Dans cet objectif, j'ai mené, dirigé, encadré et valorisé à travers des collaborations scientifiques et industrielles des travaux dont je rapporte ici une synthèse.

Summary

This report describes my research activities I have been conducting for the last six years. This work has been mainly led at the LIRMM lab (Univ. Montpellier 2, CNRS UMR 5506), within the TATOO group. In this framework, I have put the emphasis on putting together research fields that were seen as antogonistic : managing the imperfection (on both data and discovered patterns) on the one hand, data mining on the second hand, and complex and huge databases on the other hand. More precisely, my research work aims at studying the use of fuzzy logic to mine more valuable patterns from data warehouses, while remaining scalable. To this aim, I have led, conducted and supervised research and industrial work that I discuss here by providing a synthetical view.

Remerciements

Je tiens tout d'abord à remercier Trevor Martin, Jian Pei et Michel Scholl pour avoir accepté d'être les rapporteurs de ce travail. Je suis honorée qu'ils aient consacré de leur temps précieux à cet effet. La renommée de ces chercheurs et leur connaissance des domaines associés à mon travail ont permis de nombreux échanges fructueux, et je les en remercie vivement.

Je tiens également à exprimer mes remerciements aux autres membres du jury. En particulier, je tiens à remercier Bernadette Bouchon-Meunier d'avoir accepté de revenir m'écouter, après ces quelques années qui nous séparent de ma soutenance de thèse. Sa présence au cours de ces dernières années et son accueil toujours aussi chaleureux dans sa belle équipe du LIP6 m'ont permis de maintenir des liens solides avec mes thématiques de cœur liées au traitement de données imparfaites.

La présence de Eyke Hüllermeier dans ce jury est également un honneur dont je suis très heureuse. Les discussions fructueuses que nous avons eues lors de nos rencontres sont le reflet de sa grande connaissance de tous les sujets traités ici et ont permis de faire avancer de nombreux aspects de mon travail.

Enfin, la présence de Christine Collet est pour moi un privilège et je tiens à la remercier d'avoir accepté de participer à ce jury et de le présider.

Je n'oublie bien sûr pas Maguelonne et Pascal qui m'ont offert un environnement de travail riche et dynamique qui m'a permis d'apprendre beaucoup et de découvrir de très nombreuses facettes de notre métier d'enseignant-chercheur. Les (anciens et nouveaux) doctorants de l'équipe m'ont apporté énormément et je tiens à les en remercier également. Merci donc à Cécile, Céline, Chedy, Federico, Hassan, Haoyuan, Julien, Lisa, Marc, Paola, Sarah, Yoann.

Je tiens également à remercier mes collègues du LIRMM et de Polytech'Montpellier, pour l'ambiance sympathique et dynamique qu'ils savent insuffler et la force que cela donne tous les jours pour avancer. Merci donc à Christophe, Mathieu, Olivier, Sandra, et les autres et tous ceux qui m'ont fait confiance en me confiant des missions au cours desquelles j'apprends tant de choses.

De même, je tiens à remercier les partenaires industriels sans qui notre métier ne serait pas tout à fait le même et auprès de qui nous apprenons également tant. Merci donc en particulier à Bénédicte, Cédrine, François, Françoise, Michel, Nicolas, Olivier, Rachel et Stéphane.

Certains travaux présentés ici sont issus de collaborations nationales ou internationales et c'est toujours un plaisir de s'inscrire dans cette démarche d'échanges, je tiens donc à remercier chaleureusement mes partenaires dans ces aventures : Denis, Marie-Jeanne, Maria, Nicolas, Putri, Saifullah, Sadok, Sophie, Yeow Wei.

Enfin, je tiens à remercier ma famille pour sa présence précieuse et en particulier Patrice et Salomé pour avoir accepté (et accepter encore maintenant) de sacrifier un peu des week-ends, vacances et soirées en famille.

Première partie

Introduction

Ancrées à l'intersection de plusieurs disciplines informatiques (bases de données, entrepôts de données, théorie des sous-ensembles flous, et fouille de données), mes activités de recherche consistent depuis 1999 à étudier comment rendre les méthodes de fouille de données robustes face à des données complexes : multidimensionnelles, hiérarchisées, arborescentes, numériques, etc. tout en conservant un grand potentiel explicatif aux résultats présentés à l'utilisateur, celui-ci étant souvent non informaticien.

Initiés au cours de mes travaux de thèse, ces travaux se poursuivent depuis 2002 au Laboratoire d'Informatique, de Robotique et de Micro-électronique de Montpellier (Université Montpellier 2). Dans le cadre de ma thèse, j'avais montré qu'il était pertinent de concilier les domaines liés aux entrepôts de données à la fouille de données et à la théorie des sous-ensembles flous. Il s'agissait alors de représenter l'information potentiellement imparfaite du monde réel au sein d'entrepôts de données et de définir des méthodes efficaces et pertinentes pour extraire des règles utiles aux utilisateurs.

Par la suite, ces travaux ont été étendus pour répondre aux nouveaux défis liés à la fouille de données complexes, notamment pour la prise en compte de la temporalité, l'extraction efficace d'exceptions, et la gestion de données en flots au sein des entrepôts de données.

Entrepôts de données : représentation et fouille

Les entreprises, qu'elles soient grandes ou moyennes, voire de petite taille, sont maintenant couramment dotées d'outils d'entreposage de leurs données. Véritables garants de la mémoire de l'entreprise, ces entrepôts sont souvent au cœur des outils de pilotage de l'activité tant au niveau de la production, de la gestion des prix ou des actions marketing etc, qu'au niveau de la gestion interne (*e.g.* resources humaines).

Des outils d'exploitation de ces entrepôts sont disponibles (e.g. navigation OLAP, reporting). Cependant il reste difficile de doter les entreprises et leurs utisateurs d'outils permettant de les guider automatiquement vers les connaissances cachées susceptibles d'éclairer leur décision et de guider leurs choix, que ce soit par la détection automatique de tendances ou au contraire d'exceptions. À la suite de l'article fondateur de J. Han proposant de coupler les approches OLAP et les méthodes de fouille de données, définissant ainsi l'*OLAP Mining* [28], de nombreuses recherches avaient débuté pour répondre aux nombreux challenges qui restaient et restent à relever, notamment en raison de la difficulté de réaliser un couplage performant, en raison de l'explosion du volume des données et de la rapidité de leur arrivée dans l'entrepôt, en raison de la complexité des données maintenant intégrées (e.g. données non structurées), et en raison de la nécessité de prendre en compte l'imperfection des données.

Au cours de ma thèse, j'avais choisi d'ajouter à la vision OLAP Mining classique la prise en compte de l'imperfection des données réelles. Nous nous

étions alors intéressés à l'intégration d'outils avancés au cœur même des entrepôts de données. Ces outils permettaient de prendre en compte les imperfections du monde réel en s'appuyant notamment sur la théorie des sous-ensembles flous : représentation de données imparfaites (notamment imprécises), interrogation flexible. D'autre part, un ensemble de méthodes de fouille de ces entrepôts avaient été proposées à la recherche de tendances mais aussi d'inattendus, en respectant là encore le caractère souvent imparfait des données sous-jacentes et les besoins d'agrégation et d'approximation nécessaire à un rendu pertinent pour les experts.

La recherche de tendances avait alors pris la forme de résumés multidimensionnels flous. Les approches proposées visaient principalement : d'une part à doter les approches floues d'outils de fouille de données puissants (algorithmes de recherche avec propriétés de coupure de type APriori) et d'un autre côté à doter les outils de fouille de données de sémantique plus proche de l'utilisateur avec des résumés flous : résumés inter-dimensions , intra-dimensions (*e.g.* la plupart des ventes de l'EST sont effectuées à Boston), ou raffinement de résumés (*e.g.* production d'un résumé à niveau de granularité élevé : peu de ventes au deuxième trimestre 1995 concernent les produits de camping puis raffinement souhaité par l'utilisateur sur une ou plusieurs dimensions : peu de ventes au deuxième trimestre 1995 concernent des tentes). Plusieurs méthodes avaient également été proposées pour déceler les cellules anormalement vides.

Ces travaux ont été étendus pour faire face aux enjeux des nouvelles formes de bases de données et aux besoins des utilisateurs de plus en plus émergents.

Entrepôts de données et fouille de données : les nouveaux défis

Comme vu précédemment, mes travaux de thèse (Université Paris 6) ont permis de montrer qu'il était non seulement possible mais aussi prometteur de coupler les entrepôts de données et les méthodes de fouille de données (en particulier fouille de données floue). Ainsi, les aspects multidimensionnels avaient été pris en compte, la spécificité de la mesure avait également été considérée pour définir plusieurs types de comptage du support des résumés. Les éléments inattendus avaient été étudiés (cellules anormalement vides). Cependant, la nature des données complexes liées aux entrepôts de données a levé de nouveaux défis.

En particulier, le fait que les données d'entrepôts sont historisées n'avait pas été exploité. Mon arrivée au LIRMM a donc été capitale pour prendre pleinement en compte les spécificités des entrepôts grâce à leur expertise sur l'extraction de motifs séquentiels [1]. De manière duale, j'ai pu orienter les recherches

1. On appelle motif séquentiel un motif de la forme $< \{a,b\}\{a,d\}\{e\} > x\%$ où a, b, d et e sont des items, $\{a,b\}$, $\{a,d\}$ et $\{e\}$ sont des itemsets (ou ensembles d'items) et x est le support. On lit alors "x% des transactions de la base de données contiennent a et b puis a et d puis e. Par exemple "20% des clients achètent du beurre et de la moutarde puis du beurre et des chips puis du pain" est un tel motif.

menées au sein de l'équipe vers la prise en compte d'entrepôts de données d'une part, et de données et règles imprécises d'autre part. De plus, l'avénement de nouvelles structures de bases de données complexes constitue de nouveaux défis qu'il s'agissait de relever : données dites *en flot*, données arborescentes, données numériques, etc. Enfin, il devenait crucial, au vu des demandes des entreprises et scientifiques avec lesquels nous collaborions, de proposer aux utilisateurs des outils leur permettant non seulement d'extraire des connaissances générales (tendances), mais aussi des exceptions.

C'est donc tout naturellement vers ces sujets que j'ai orienté mes travaux après mon arrivée au LIRMM.

Contributions et organisation du mémoire

Dans l'objectif de concilier d'une part l'approche prometteuse développée précédemment liant fouille de données d'entrepôts et logique floue, et d'autre part la nouvelle donne (nécessité de mieux prendre en compte les données séquentielles, les données complexes, et la gestion des exceptions), j'ai mis en place et mené différentes actions lors des cinq dernières années au LIRMM, dont j'ai assuré la responsabilité ou la co-responsabilité.

- des encadrements de thèse (huit théses co-encadrées dont trois déjà soutenues, et une en cours de finalisation),
- des collaborations scientifiques nationales (EMA, INSERM, INRIA Sophia-Antipolis, Université Montpellier 3, Orsay, Cergy-Pontoise, Paris 6, Tours, 1 projet supporté par l'ANR) et internationales (Allemagne, Canada, Indonésie, Malaisie, Pakistan, Tunisie),
- des collaborations industrielles (IBM, EDF R&D, et sociétés incubées régionales).

Dans le cadre de notre collaboration avec la Malaisie, nous nous sommes intéressés à la découverte de blocs de données homogènes au sein de cubes de données, travaux que nous rapportons dans la partie II. Il s'agissait alors d'être capables de détecter automatiquement des zones (par exemple une zone correspondant à des ventes assez fortes) afin de guider l'utilisateur dans sa navigation au sein de données OLAP. Ces travaux sont rapportés dans la partie II. Ils intègrent non seulement la prise en compte des spécificités des entrepôts de données (hiérarchies), mais aussi la définition souple de la notion de valeur du bloc. Ainsi, nous étudions des blocs contenant exactement la même valeur de mesure (par exemple une zone correspondant à des ventes de 500 unités), contenant une valeur comprise dans un intervalle (par exemple une zone correspondant à des ventes comprises entre 320 et 512 unités), ou contenant une valeur comprise dans un intervalle flou (par exemple une zone correspondant à des ventes *assez fortes*, c'est-à-dire autour de 400 unités). Les propriétés exhibées par nos travaux nous ont permis de définir des algorithmes efficaces. Notons que par souci de synthèse, nous ne rapportons pas ici les résultats expérimentaux présents dans les publications.

Cependant ces approches ne permettent pas de prendre en compte la dimension temporelle des entrepôts alors que l'historisation est l'une des caractéristiques centrales de ce type de bases de données. Les travaux réalisés au sein du LIRMM ont donc été centrés sur cette problématique.

Afin de permettre l'extraction de motifs séquentiels à partir de bases de données séquentielles numériques jusqu'alors impossibles à fouiller, j'ai donc proposé la définition d'algorithmes d'extraction de motifs séquentiels flous. Ces travaux ont été menés dans le cadre de la thèse de C. Fiot. Ils ont permis de poser les premiers jalons de l'intégration de méthodes floues dans le processus de fouille de données intégrant la dimension temporelle. Il s'agissait alors non plus de trouver des motifs séquentiels tels que "20% des clients achètent du beurre et de la moutarde puis du beurre et des chips puis du pain" mais plutôt des motifs du type "20% des clients achètent un peu de beurre et un peu de moutarde puis beaucoup de beurre et un peu de chips puis beaucoup de pain".

Cependant, ces motifs n'intégrent pas la multi-dimensionnalité qui est elle aussi au cœur de l'approche entrepôts de données et qui n'avait pas été traitée de manière satisfaisante dans la littérature. Nous avons donc proposé de définir les motifs séquentiels multidimensionnels dans le cadre de notre collaboration avec la Malaisie, et dans le cadre de la thèse de Marc Plantevit. Ces travaux, présentés dans la partie III, permettent alors d'extraire des motifs de la forme "23 % des clients ont achété une planche de surf et un sac à New York puis une combinaison à San Francisco". Ce motif permet de mettre en valeur des corrélations entre plusieurs dimensions (ville et produit) et extrait les différentes combinaisons de valeurs au cours du temps, ce qu'aucune autre méthode ne permettait de réaliser jusqu'alors. Les algorithmes proposés permettent également de combiner des niveaux de hiérarchie et s'interrogent sur la façon de prendre en compte la mesure des cubes de données dans ce contexte.

Cependant, il est apparu que de nombreux utilisateurs (et notamment les partenaires industriels de EDF R&D avec lesquels nous collaborions) souhaitaient non seulement extraire de telles tendances à partir de leurs entrepôts de données, mais aussi des comportements atypiques, afin de mettre en valeur les dysfonctionnements de leurs organisations.

Nous avons donc défini des méthodes originales de recherche d'exceptions au sein de données d'entrepôts. Deux méthodes principales ont été proposées, que nous rapportons dans la partie IV. La première méthode permet d'extraire des règles multidimensionnelles inattendues. Il est ainsi possible de découvrir des règles du type "les clients du sud de la France achètent des bottes puis des lunettes de soleil" alors que "les clients du sud de la France *qui sont à la retraite* achètent des bottes puis des parapluies". La deuxième méthode quant à elle propose une aide à la navigation. Par exemple, si l'utilisateur est désireux de naviguer dans ses données en fonction de la localisation, il pourra, pour chaque niveau de granularité (par exemple la région), repérer la valeur correspondant aux données historisées les plus atypiques en comparaison des autres, puis choisir de se focaliser sur un sous-ensemble de données (e.g. une région particulière) pour poursuivre son investigation à un niveau plus fin (e.g. vers les villes).

Les méthodes que nous avons proposées sont donc sémantiquement très riches et permettent de couvrir de nombreuses utilisations. Cependant, un type de règles est malheureusement trop souvent oublié : les règles graduelles (par exemple “Plus le mur est proche, plus le train doit freiner fort”).

Nous avons donc proposé de définir des algorithmes efficaces pour faire face à ce problème. Peu étudiée en raison de la complexité du problème associé, l'extraction de telles règles est cependant cruciale puisqu'il existe de nombreuses applications, notamment dans le domaine scientifique (données liées à la santé). Or aucun algorithme efficace n'existait. Le sujet de post-doctorat de C. Fiot, effectué en collaboration avec l'INRIA Sophia-Antipolis, a donc été proposé pour répondre à ce défi, et a été suivi par la mise en place de la thèse de L. Di Jorio. Il s'agit de définir des algorithmes efficaces tant en terme de temps de calcul qu'en termes d'utilisation mémoire, raison pour laquelle nous travaillons sur des structures de données optimisées et recherchons des propriétés permettant de réduire la complexité des calculs (voir partie V). Notamment appliquées à des données issues du domaine de la santé, ces algorithmes doivent en particulier être capables de gérer des bases de données contenant peu de lignes et beaucoup de colonnes. Différentes formes de règles sont recherchées, selon qu'elles prennent en compte la temporalité (“Plus un client achète de beurre, moins il achètera du lait plus tard”), ou non (“Plus un client achète de beurre, plus il achète du lait et moins il achète de chips”).

Notons que ce mémoire ne se veut pas exhaustif et ne présente pas l'ensemble de mes activiés en détail. En particulier, nous n'abordons pas ici les travaux menés sur la fouille de données arborescentes réalisés dans le cadre de la thèse de Federico Del Razo Lopez. Notons que ces travaux sont très liés au contexte des entrepôts de données puisque la fouille de telles données peut être utilisée dans un processus de médiation au moment d'interroger des sources de données dispersées et hétérogènes pour la construction d'un entrepôt. De même, ne sont pas rapportés ici les travaux encadrés dans le cadre de la thèse de Dong (Haoyuan) Li qui s'intéresse à la recherche de motifs séquentiels exceptionnels dans le cadre de données non issues d'entrepôts (données textuelles notamment).

Chapitre 1

Fouille de données floue

La théorie des sous-ensembles flous a été introduite par L. Zadeh en 1965 afin de permettre la représentation des connaissances imparfaites [62]. Cette théorie offre un cadre formel pour manipuler des données imprécises et/ou incertaines. Par exemple, il est possible de modéliser mathématiquement des données du type *jeune* où un individu appartient *plus ou moins* (de manière graduelle) au concept *jeune*.

De manière générale, un sous-ensemble flou de l'univers X est représenté par sa fonction d'appartenance prenant ses valeurs dans l'intervalle $[0, 1]$. Pour un sous-ensemble flou A de l'univers X, on note μ_A la fonction d'appartenance de A, avec $\mu_A : X \rightarrow [0, 1]$. Pour $x \in X$, $\mu_A(x)$ représente le degré d'appartenance de x au sous-ensemble flou A.

On appelle *support* l'ensemble des valeurs de $x \in X$ telles que $\mu_A(x) > 0$ et *noyau* l'ensemble des valeurs de $x \in X$ telles que $\mu_A(x) = 1$.

La Figure 1.1 illustre un exemple de sous-ensemble flou avec la fonction d'appartenance associée.

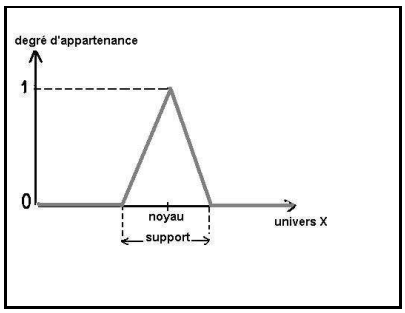

FIGURE 1.1 – Exemple de sous-ensemble flou

On note que tout ensemble classique est un sous-ensemble flou particulier,

pour lequel le degré d'appartenance vaut soit 0 soit 1 et non pas toute valeur entre 0 et 1.

La théorie des sous-ensembles flous a été très longtemps utilisée dans des systèmes déductifs, reproduisant le raisonnement humain dans le cadre de systèmes disposant de bases de connaisssances (commande floue). De nombreux succès scientifiques et commerciaux ont émergé de telles applications. Depuis de nombreuses années, les systèmes inductifs se sont pourtant également développés, et plus récemment, les méthodes de fouille de données floue sont apparues. Ces dernières ont pour particularité de pouvoir traiter des données imparfaites et/ou de produire des règles intégrant l'imprécision inhérente à toute extraction de tendance. Nous présentons brièvement ci-après les domaines les plus proches des travaux que nous avons effectués.

1.1 Résumés et règles d'association flous

Les résumés flous ont été étudiés depuis le début des années 1980. Un tel résumé est par exemple donné par la phrase *"la plupart des experts importants sont jeunes"*.

De manière plus formelle, soit Q un quantificateur (e.g. *la plupart*), S un terme de résumé (e.g. *jeune*), y le nom de la relation contenant les n données à résumer $y_1, \ldots, y_n$ (e.g. *experts*), B une valeur d'attribut de y (e.g. *important*) et τ le degré de vérité du résumé, les résumés générés sont alors de la forme [36, 35, 55] :

$$\text{“}Q\ B\ y\ sont\ S : \tau\text{”}$$

On considère un ensemble de quantificateurs et de termes de résumés connus *a priori* (donnés par le système et/ou par l'utilisateur). Le système calcule le degré de vérité pour chacune des combinaisons possibles de Q et S. Les quantificateurs et les termes de résumé sont des sous-ensembles flous. Les premiers sont définis sur l'intervalle $[0, 1]$ et les seconds sur l'intervalle de définition de y. L'introduction de sous-ensembles flous apporte plus de souplesse que des quantificateurs et termes classiques. Ils sont représentés par leurs fonctions d'appartenance ; Q et S sont ainsi respectivement représentés par μ_Q et μ_S.

Le degré τ est alors calculé de la manière suivante :

$$\tau = \mu_Q \left(\frac{1}{n} \sum_{i=1}^{n} \mu_S(y_i) \right) \tag{1.1}$$

Dans les cas où un sous-ensemble de données est extrait correspondant aux

données vérifiant le critère (flou ou non) B, on calcule ce degré de la manière suivante :

$$\tau = \mu_Q \left(\frac{1}{n} \sum_{i=1}^{n} \top(\mu_S(y_i), \mu_B(y_i)) \right) \tag{1.2}$$

où $\top$ est un opérateur de type t-norme.

Dans la continuité de ces travaux, j'avais développé au cours de ma thèse des méthodes d'extraction de résumés flous à l'aide d'algorithmes par niveau afin de permettre l'application sur de grands volumes et le passage à l'échelle.

De plus, des extensions aux règles d'association floues existent, afin de prendre en compte au mieux les attributs numériques en les partitionnant non pas à l'aide de seuils stricts, mais à l'aide d'intervalles flous [41]. Les règles trouvées sont alors de la forme *"Si l'âge est moyen Alors le salaire est élevé"* où *moyen* et *élevé* sont des sous-ensembles flous. De telles règles permettent de mieux appréhender les bases de données contenant des attributs numériques, puisque les algorithmes classiques transforment ces bases en bases de données binaires (présence/absence), oubliant alors l'information pourtant importante de la quantité. Il est par exemple très différent de considérer un client ayant acheté 1 bouteille et un client ayant acheté 3500 bouteilles. Or les méthodes classiques considèrent ces deux clients comme totalement similaires, puisqu'il y a présence d'au moins 1 bouteille dans leurs achats. Notons que ces méthodes constituent une extension de l'approche de [56] proposant une discrétisation stricte par intervalles pour trouver des règles de la forme "10% des personnes mariées ayant entre 50 et 60 ans ont au moins 2 voitures".

Lors de l'extraction de telles règles d'association floues, les défis soulevés, outre la définition des sous-ensembles flous eux-mêmes (e.g. comment définir qu'un âge est *moyen*?), sont alors la définition du comptage [18] et l'étude de propriétés intéressantes pour la mise en place d'un algorithme procédant le plus possible à des coupures (sur le principe de l'anti-monotonie).

Ces travaux ont été étendus au contexte des motifs séquentiels flous dans le cadre du travail de C. Fiot (thèse co-encadrée avec M. Teisseire) présenté ci-dessous.

1.2 Motifs séquentiels flous

Les motifs séquentiels flous permettent la prise en compte de données numériques, extrayant des informations de la forme *60% des clients achètent beaucoup de pain puis peu de gâteaux* où peu et beaucoup sont des sous-ensembles flous définis sur l'univers des quantités de produits achetés.

Dans sa thèse, C. Fiot a proposé des définitions d'item, itemset et séquence flous intégrant la prise en compte de sous-ensembles flous. Trois méthodes d'extraction de tels motifs séquentiels flous (Speedy Fuzzy, Mini Fuzzy et Totally Fuzzy) ont été proposées, permettant de moduler le degré d'approximation du support et la rapidité de son calcul. Ces travaux ont permis de compléter les premières approches présentées dans [30] qui ne permettaient pas de bien distinguer entre dates et ne définissaient pas d'algorithmes efficaces. De plus, ils ont été appliqués avec succès au problème difficile du traitement de bases de données incomplètes.

Dans la continuité de ces règles et motifs flous exprimant des tendances telles que *"Si l'âge est moyen Alors le salaire est élevé"*, nous nous sommes intéressés à l'extraction de règles graduelles permettant d'exprimer des tendances du type *"Plus l'âge est moyen, plus le salaire est élevé"*. L'extraction de telles règles étant rendue très difficile par la complexité des traitements à mettre en œuvre (explosion combinatoire), il n'existait que très peu de travaux. Nous les rapportons ci-dessous.

1.3 Règles graduelles

L'ordonnancement de données est un problème connu en informatique qui a donné lieu à de nombreux travaux. En fouille de données, on peut citer par exemple les travaux liés à la fouille de données de préférences, à l'extraction de top-k, ainsi qu'à la recherche d'ordonnancements de valeurs de mesure au sein de cubes multidimensionnels [14] qui est un problème np-difficile.

Dans notre approche, nous nous intéressons au problème de la recherche de règles graduelles dans des données multidimensionnelles contenant plusieurs attributs munis d'un ordre (attributs numériques par exemple). De telles règles ont la forme générale "*Plus (moins)* A_1 *et ... plus (moins)* A_n *alors plus (moins)* B_1 *et ... plus (moins)* B_n".

De nombreux travaux ont été proposés pour la recherche de règles graduelles. La notion de gradualité, et plus particulièrement de règles graduelles, a majoritairement été étudiée dans la *communauté floue*. Celles-ci étaient utilisées dans le but de modéliser des systèmes experts. L'accent n'est alors pas mis sur la manière de les extraire, mais plutôt sur leur rôle au sein de systèmes à base de règles (par exemple, "Plus le mur est proche, plus le train doit actionner le frein", voir par exemple [22]). [20] proposent un cadre théorique complet pour la formalisation des règles graduelles, et comparent diverses implications floues pour mesurer les dépendances graduelles. L'implication la plus utilisée reste Resher-Gaines ($A(X)$ est de degré d'appartenance de X au sous ensemble flou A) :

$$X \rightarrow_{RG} Y = \begin{cases} 1 & if\ A(X) \leq B(Y) \\ 0 & else \end{cases} \tag{1.3}$$

L'équation (1.3) assure que le degré d'implication de X est contraint par le degré d'implication de Y. Ainsi, si la valeur de Y augmente, alors celle de X peut augmenter, assurant que "*plus Y est B, plus X est A*". Cependant, l'implication de Resher-Gaines est restrictive et rend la conjonction difficile à implémenter.

[31] remplace les *tables de contingence* représentant des règles d'associations par des *diagrammes de contingence*. Puis, les corrélations entre variations sont extraites à l'aide d'une régression linéaire directement appliquée sur les diagrammes. Les coefficients de pente et de qualité de la régression peuvent être utilisés afin de décider de la validité d'une règle. Cependant, cette méthode ne peut être appliquée sur des jeux contenant un grand nombre d'attributs, car la régression linéaire peut s'avérer trop coûteuse en terme de temps.

Afin d'éviter des jointures trop coûteuses, [4] proposent l'utilisation de l'algorithme Apriori. Ainsi, les itemsets graduels sont définis à l'aide des opérateurs $\{<, >\}$, et la base de données est transformée en une base de couples d'objets. La fouille s'effectue alors directement à partir des couples. Le support est redéfini comme la proportion de couples supportant une variation parmi tous les couples de la base. Cette méthode est la première permettant de prendre en compte des conjonctions de variations, aussi bien croissantes que décroissantes, dans la condition et la conclusion de la règle. Cependant, la base de couples associée aux sous-ensembles flous rend la méthode complexe d'un point de vue calcul, ce qui empêche le passage à l'échelle. Les expérimentations sur jeux de données réelles, bien que prometteuses de par l'intérêt des règles extraites, sont menées sur une base contenant seulement 6 attributs.

Comme nous l'avons vu dans ce chapitre, la théorie des sous-ensembles flous permet d'extraire, à partir de bases de données (notamment numériques), des règles et motifs intéressants. Si la définition d'algorithmes efficaces face à de très gros volumes de données n'est pas aisée, en raison de l'espace de recherche souvent augmenté par la prise en compte de l'imperfection possible, il existe tout de même de nombreuses propriétés sur lesquelles il est possible de s'appuyer pour résoudre ces problèmes.

Il est donc tout à fait raisonnable d'envisager l'application de telles méthodes de fouille de données floue à des bases de données complexes telles que les entrepôts de données. Mon objectif a donc été de concilier ces domaines pour proposer des méthodes originales de couplage entre entrepôts de données et fouille de données floue.

Afin de mieux comprendre les défis associés à ces données complexes, nous présentons brièvement, dans le chapitre suivant, les principales caractéristiques des cubes de données et introduisons ensuite de manière plus détaillée l'ensemble de la problématique qui a été au cœur de nos préoccupations ces dernières années.

Chapitre 2

Entrepôts de données

Les entrepôts de données sont apparus sous l'implusion de besoins industriels forts pour stocker des données historisées à des fins d'analyse. Comme décrit par [32], "A data warehouse is a subject-oriented, integrated, non-volatile and time-variant collection of data in support of management's decision making process".

L'entrepôsage de données réfère donc au processus de construction et d'exploitation de gros volumes de données à partir de sources hétérogènes en un schéma unifié. La construction de telles bases inclut donc l'intégration, le nettoyage, la consolidation et les processus d'analyse, dits OLAP (On-Line Analytical Processing) [9, 16, 27].

Souvent opposés aux processus OLTP (On-Line Transactional Processing) liés aux bases de données classiques, les sytèmes OLAP en sont le complément et viennent se greffer au-dessus des systèmes opérationnels classiques. Longtemps très séparés de ces derniers systèmes pour ne pas en handicaper le fonctionnement, les systèmes décisionnels deviennent maintenant directement interactifs pour des résultats de plus en plus "temps réel".

Si les systèmes dits de *Business Intelligence* ne contiennent pas à proprement parler d'outils très intelligents, la fouille de données peut être considérée comme une extension naturelle de ces systèmes [27].

2.1 Modélisation multidimensionnelle

Les entrepôts de données sont modélisés de manière multidimensionnelle, et contiennent des cubes de données décrivant des *indicateurs* ou *mesures* selon un ensemble de *dimensions* qui peuvent être organisées en hiérarchies.

L'exemple souvent repris pour présenter les cubes de données permet d'ana-

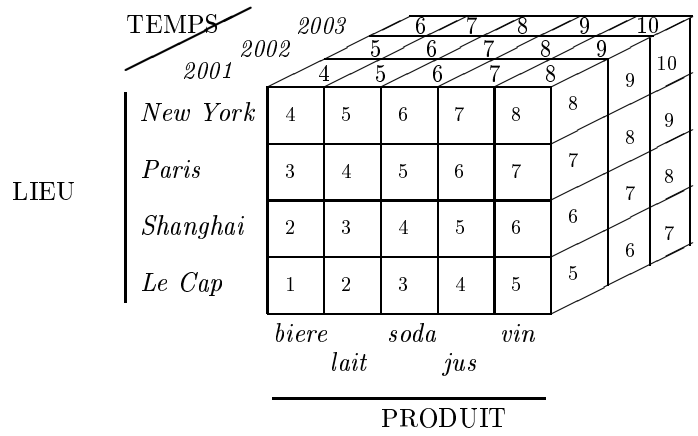

FIGURE 2.1 – Cube de données à 3 dimensions

lyser le volume de ventes réalisées en fonction de trois dimensions : localisation géographique, produit vendu, et moment de la vente. (cf. Figure 2.1). Une cellule d'un tel cube correspondant alors à la valeur du volume de vente pour un produit donné, une localisation données et une date donnée. D'autres mesures peuvent être considérées (e.g. prix, bénéfice).

Il est possible de poser des requêtes sur les cubes de données, on parle alors d'analyse OLAP.

On appelle base de données multidimensionnelles un ensemble de dimensions, de mesures (indicateurs) et de cubes de données définis à partir de ces dimensions et mesures. Des hiérarchies peuvent être définies sur les dimensions. Nous décrivons ci-dessous les définitions plus formelles.

Définition 1 - Cube. *Un cube de dimension k est défini par $\langle C, dom_1, \ldots, dom_k, dom_m, m_C \rangle$ où*

- *C est le nom du cube,*
- *$dom_1, \ldots, dom_k$ sont k ensembles finis de symboles correspondant respectivement aux membres des dimensions $1, \ldots, k$,*
- *$dom_m = dom_{mes} \cup \{\perp\}$, dom_{mes} est un ensemble totalement ordonné de valeurs possibles de mesure et $\perp$ est une constante non incluse dans dom_{mes} indiquant la valeur nulle.*
- *m_C est une application de $dom_1 \times \ldots \times dom_k$ vers dom_m.*

Une cellule c d'un cube C de dimension k est un $(k+1)$-uplet $\langle v_1, \ldots, v_k, m \rangle$ tel que, pour tout $i = 1, \ldots, k$, v_i appartient à dom_i et $m = m_C(v_1, \ldots, v_k)$. De plus, m est appelé *contenu* de c et c est dite *m-cellule.*

Comme expliqué par [14], un cube peut être associé à plusieurs *représentations,* selon la façon d'ordonner les ensembles des domaines de dimensions dom_i $(i = 1, \ldots, k)$. Par exemple, la figure 2.2 représente deux représentations diffé-

rentes du même cube. Selon ses choix de présentation, l'utilisateur peut alors être confronté à une représentation meilleure qu'une autre au sens où elle lui permet de tirer automatiquement des conclusions. Or il est impossible pour un utilisateur de visualiser toutes les représentations possibles, leur nombre étant très grand.

Définition 2 - Représentation. *Une représentation d'un cube C est un ensemble $R = \{rep_1, \ldots, rep_k\}$ où pour tout $i = 1, \ldots, k$, rep_i est une application 1-1 de dom_i vers $\{1, \ldots, |dom_i|\}$.*

Dans notre approche, nous considérons une représentation donnée $R = \{rep_1, \ldots, rep_k\}$.

PRODUIT

P1	6	6	8	5	5	2	
P2	6	8	5	5	6	75	
P3	8	5	5	2	2	8	
P4	8	8	8	2	2	2	
	V1	V2	V3	V4	V5	V6	**VILLE**

(a)

PRODUIT

P4	8	8	8	2	2	2	
P2	5	6	8	5	6	75	
P1	8	6	6	5	5	2	
P3	5	8	5	2	2	8	
	V3	V1	V2	V4	V5	V6	**VILLE**

(b)

FIGURE 2.2 – Deux représentations d'un même cube.

2.2 Opérations OLAP

Comme mentionné précédemment, dans le modèle multidimensionnel, les données sont organisées selon plusieurs dimensions, et chaque dimension contient plusieurs niveaux de granularité définis à partir des hiérarchies, permettant à l'utilisateur d'analyser les données du cube à différents niveaux de détail. Pour cela, des opérations sont disponibles pour naviguer dans les données. La navigation est un processus dirigé par les requêtes utilisateurs. Si de nombreux modèles et langages de requêtes ont été proposés [27, 45, 61] il n'existe à l'heure actuelle aucun langage faisant consensus.

Cependant, les opérations OLAP sont principalement regroupées autour des opérations suivantes :

- *Généralisation (Roll-up).* Cette opération calcule l'agrégation d'un ensemble de cellules quand il s'agit de passer d'un niveau de granularité de hiérarchie à un autre plus général (par exemple analyser les ventes par

régions plutôt que ville par ville). L'agrégation totalle (jusqu'au niveau dit ALL) revient à éliminer la dimension et donc à réduire le nombre de dimensions du cube.

- *Spécialisation (Drill-down).* Cette opération est l'inverse de la précédente, et permet de retrouver plus de détails, par exemple pour repasser du niveau des régions au niveau des villes. Notons que cette opération n'est possible que si le détail du niveau précédent est connu. Elle requiert parfois d'interroger les sources de données si le cube de données a été construit à un niveau plus agrégé.
- *Sélection (Slice and dice).* L'opération *slice* permet de sélectionner certains membres d'une dimension (par exemple Avril, Mai et Juin sur la dimension temporelle) pour obtenir un sous-cube (hypertranche). L'opération *dice* permet de sélectionner des valeurs de mesures.
- *Rotation (Pivot).* Cette opération permet d'inverser les dimensions visibles d'un hypercube, par exemple pour passer d'une visualisation où les dimensions temporelle et spatiale sont mises au premier plan à une visualisation dans laquelle la dimension spatiale s'efface au profit de la dimension produit. Notons que cette opération est nécessitée par le fait que les cubes de données ne peuvent être visualisés qu'en deux dimensions. Les $k-2$ dimensions restantes sont alors soit imbriquées dans chaque cellule de la représentation, soit fixées à une valeur définie par l'utilisateur.
- *Inversion (Switch).* Cette opération liée à la représentation du cube permet d'interchanger les positions de deux membres d'une dimension, par exemple pour afficher la ville $V3$ avant la ville $V1$ (voir figures 2.2(a) et 2.2(b)).

Il existe de nombreuses autres opérations OLAP (e.g. *push*, *pull*, *join* et *merge*). Une liste plus détaillée de ces opérations pourra être trouvée dans [7, 23, 25, 32, 33, 38].

Comme nous l'avons vu dans ce chapitre, les entrepôts de données présentent des caractéristiques propres qui rendent difficile l'application de méthodes de fouille de données de manière directe, comme nous le détaillons dans le chapitre suivant.

Chapitre 3

Fouille de données floue et entrepôts de données : problématique et défis

La fouille de données complexes (arborescentes, en flots, issues d'entrepôts etc) est une tâche difficile nécessitant de nouveaux algorithmes. Dans le cadre des données d'entrepôts, les données à traiter sont souvent très volumineuses et les espaces de recherche ne permettent pas une exploration systématique et exhaustive. Les problèmes sous-jacents (liés aux entrepôts et à la fouille de données) sont exponentiels et il a été démontré que la plupart sont np-complets. Notons que dans les problèmes concernés par nos thématiques, la réponse de l'algorithme est celle fournie à l'utilisateur et qu'il ne s'agit donc pas seulement de décider s'il existe ou non une solution. De plus, les utilisateurs de tels systèmes, experts des données, ne sont pas informaticiens et ne disposent donc pas des connaissances nécessaires pour poser des requêtes et bénéficier de leurs résultats. Si des outils de *reporting* ou d'analyse existent, il reste toujours difficile de visualiser les données pour en dégager des informations pertinentes et potentiellement utiles. Typiquement, le nombre de dimensions varie selon les cubes construits, mais il est presque toujours supérieur à 4, et le volume des données présentes dans les cubes est très important, ce qui rend difficile la visualisation de telles données.

Face à de telles données, des méthodes de fouille de données doivent donc être mises en œuvre afin d'extraire les informations inconnues auparavant. Cependant, prétraiter les données pour utiliser les algorithmes déjà existants sans les modifier n'est pas possible puisque les bases de données multidimensionnelles ont des caractéristiques propres, décrites ci-dessous.

– La présence de mesures est l'une des principales caractéristiques des entrepôts. Numériques, ces mesures constituent un objet d'étude particulier car elles sont construites en fonction d'un ensemble de dimensions et ont un

domaine actif souvent très conséquent (beaucoup de valeurs différentes).
- De plus, ces mesures correspondent à des données agrégées. Il est par exemple le plus souvent impossible de retrouver l'identifiant individuel de clients au niveau d'un entrepôt.
- Pour expliquer ces mesures, de nombreuses dimensions sont présentes, ce qui diffère des approches classiques de fouille de données où seule une dimension est souvent considérée (e.g. nombre de ventes).
- Ces dimensions sont elles-mêmes souvent décrites à différents niveaux de granularité à l'aide de hiérarchies.
- De par la restitution souvent faite des bases de données multidimensionnelles, l'ordre défini sur les domaines des dimensions est important.
- Enfin, les cubes de données multidimensionnelles sont très souvent denses.

Dans ce contexte, nous nous intéressons à la découverte de tendances (motifs séquentiels multidimensionnels, découverte de règles graduelles) mais aussi à l'extraction d'exceptions, thématiques très importantes pour les utilisateurs.

Même si la fouille d'entrepôts n'est pas une thématique nouvelle [26] (1997), il n'en reste pas moins que les solutions présentes dans la littérature ne permettent toujours pas une implémentation directe dans les outils de *Business Intelligence* du marché.

De manière plus détaillée, les pincipales contributions rapportées dans les parties qui suivent concernent :

- la recherche de blocs au sein de données multidimensionnelles (par exemple pour retrouver automatiquement les zones du cube correspondant au même niveau de ventes),
- l'extraction de motifs séquentiels multidimensionnels et flous (par exemple pour extraire des motifs du type "23 % des clients ont achété une planche de surf et un sac à New York puis une combinaison à San Francisco" et "20% des clients achètent un peu de beurre et un peu de moutarde puis beaucoup de beurre et un peu de chips puis beaucoup de pain"),
- la recherche d'exceptions au sein de données complexes (par exemple pour extraire des connaissances du type "les clients du sud de la France achètent des bottes puis des lunettes de soleil" alors que "les clients du sud de la France *qui sont à la retraite* achètent des bottes puis des parapluies"),
- l'extraction de règles et motifs graduels (par exemple pour extraire des règles de la forme "Plus le mur est proche, plus le train doit freiner fort"),

Notons que le but de nos travaux est de nous préoccuper au mieux des attentes des experts, utilisateurs finaux des résultats.

Deuxième partie

Recherche de blocs au sein de données multidimensionnelles

Dans le contexte des bases de données multidimensionnelles, les outils OLAP permettent de naviguer dans les données dans le but de découvrir des information pertinentes. Cependant, en raison de la taille des ensembles de données, il est impossible d'adopter un parcours systèmatique et exhaustif des données. Pour cette raison, il est nécessaire de fournir aux utilisateurs des outils les guidant vers les parties des données les plus pertinentes leur permettant d'identifier des connaissances nouvelles.

Dans nos travaux, nous avons donc proposé des outils permettant de découvrir automatiquement des blocs de données homogènes, ce qui permet non seulement de résumer les données complexes, mais aussi de découvrir des exceptions par rapport à ces blocs.

Dans la littérature, il existe de nombreuses approches de résumés de cubes. Initialement motivées par la taille très volumineuse des cubes de données alors même qu'ils étaient "creux", ainsi que par leur volume, les méthodes de compression se sont également révélées intéressantes pour résumer sémantiquement les cubes de données. La principale approche de la littérature adoptant cette stratégie est celle de [42]. Cependant cette approche ne permet pas de prendre en compte des zones floues et ne gère pas de manière avancée les hiérarchies.

Notre approche est fondée sur les algorithmes classiques par niveaux, et nous avons défini plusieurs types de blocs, selon que la valeur majoritaire qu'ils contiennent est unique, ou appartient à un intervalle classique, ou à un intervalle flou.

Réalisés dans le cadre de notre collaboration avec la Malaisie (HELP University College) et l'Université Cergy-Pontoise, ces travaux ont reçu le soutien de l'Ambassade de France en Malaisie et plus généralement du Ministère des Affaires Etrangères (projet STIC-Asia EXPEDO). De nombreux étudiants (dont Anselme Beaud dont j'ai assuré l'encadrement de mémoire ingénieur CNAM) ont été impliqués dans cette thématique. Des tests ont été menés dans le cadre de la collaboration avec la société Namae Concept sur les données issues de l'INPI (Institut National de la Propriété Industrielle) concernant les noms déposés en France et leur typologie et ont démontré la pertinence de notre approche (intérêt des partenaires industriels pour les blocs trouvés).

Notons que nous avons également initié des travaux de visualisation de ces blocs qui ne sont pas rapportés ici (voir les publications associées).

Thèmes abordés	Entrepôts de données, hiérarchie, sous-ensembles flous, blocs de données homogènes
Encadrement d'étudiant	A. Beaud (ingénieur CNAM. 2006.)
Collaborations	HELP University College Univ. Cergy-Pontoise INPI (noms déposés) Société Namae Concept

Chapitre 4

Découverte de blocs flous à partir d'entrepôts de données : Définitions, Propriétés et Algorithmes

4.1 Définitions préliminaires

Habitués à naviguer au sein des cubes de données à l'aide des opérateurs OLAP (e.g. switch, roll-up), les décideurs utilisateurs des entrepôts de données sont pourtant souvent confrontés aux mêmes interrogations : comment faire pour retrouver rapidement les données correspondant aux ventes fortes ou au contraire aux ventes faibles. Si cette question semble simple dans le cadre de données tabulaires classiques (il semble qu'il suffirait d'appliquer une sélection sur la table des faits et d'afficher les n-uplets résultats), elle devient plus complexe dans le cadre de la navigation dans des cubes de données où chaque n-uplet est une cellule dont le voisinage (n-uplets suivants, précédents) est contraint par le caractère multi-dimensionnel et la visualisation *cubique*.

Nous nous sommes donc intéressés à définir une méthode originale permettant de construire et d'identifier de manière automatique et efficace des blocs de données similaires présents dans les cubes de données. Chaque bloc est en fait un sous-ensemble des données prenant la forme d'un sous-hypercube, les blocs irréguliers n'étant pas autorisés. Sur l'exemple décrit par la figure 4.1, le sous-ensemble de données correspondant aux produits $P1$, $P2$ et à la ville $V1$ constitue un bloc de valeur de mesure homogène (6). La Figure 4.2 présente les blocs découverts étiquetés avec la valeur de mesure associée.

Chaque bloc de données peut être exprimé sous la forme d'une règle pour en

faciliter l'analyse. Par exemple, la règle associée au bloc précédemment présenté est :

Si PRODUIT $= P1$ **ou** $P2$ **et** VILLE $= V1$ **Alors** Ventes $= 6$.

Si dans ce cas la valeur de mesure est la même pour toutes les cellules du bloc, cela n'est pas toujours le cas. Par exemple, il existe un bloc associé à la valeur de mesure 5 correspondant aux produits $P1$, $P2$, $P3$ et aux villes $V3$ et $V4$ qui ne contient pas uniquement la valeur 5. Ce bloc est néanmoins considéré comme intéressant puisque *la plupart* des cellules qui le composent contiennent la même valeur. De même, il existe un bloc de valeurs 2 pour la zone correspondant aux produits $P3$, $P4$ et aux villes $V4$, $V5$, $V6$. Ces deux blocs se recouvrent puisqu'ils ont en commun la cellule correspondant au produit $P3$ et à la ville $V4$.

Il se produit donc des cas de recouvrement entre les blocs découverts, qu'il s'agit de traduire lors de la génération des règles. Pour ce faire, nous utilisons la théorie des sous-ensembles flous. Ce formalisme nous permet de représenter des informations du type : pour le produit $P2$ et dans une moindre mesure pour le produit $P3$.

Le but de notre travail est d'identifier le plus rapidement possible les blocs de données représentés sur la figure 4.1, d'en définir les recouvrements, et d'y associer des règles, floues ou non.

Notre méthode est fondée sur l'utilisation combinée des algorithmes par niveaux (fondés sur l'algorithme APriori) et de la théorie des sous-ensembles flous. L'utilisation de tels algorithmes est rendue nécessaire par la volonté de proposer des méthodes efficaces passant à l'échelle.

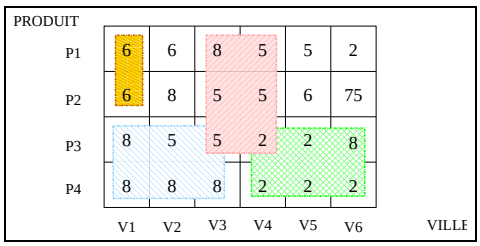

FIGURE 4.1 – Exemple d'un cube et des blocs associés

Nous considérons ici un cube à k dimensions C fixé et une de ses représentations, également fixée. On appelle alors bloc de données un sous-ensemble de cellules du cube formant un sous-cube :

Définition 3 - Bloc de données. *Un bloc de données b est un ensemble de cellules défini sur un cube C à k dimensions par $b = \delta_1 \times \ldots \times \delta_k$ où les δ_i sont des intervalles de valeurs contigües du domaine $dom(d_i)$ de la dimension d_i : $\delta_i \subseteq dom(d_i)$ pour $i = 1, \ldots, k$.*

On notera que, dans le cas où l'on ne spécifie pas un intervalle pour chacune

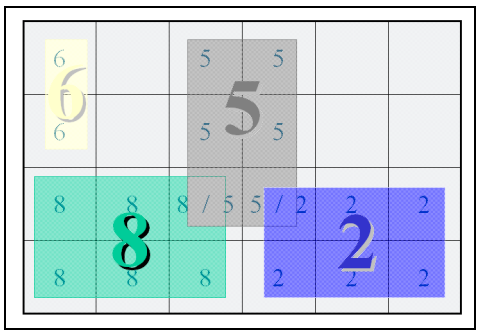

FIGURE 4.2 – Tous les blocs

des dimensions du cube, on se ramène à la definition ci-dessus en posant $\delta_i = ALL = dom(d_i)$ pour toute dimension d_i absente de la spécification.

Les blocs peuvent se recouvrir, ce recouvrement étant plus ou moins important. Dans notre approche, nous considérons qu'il y a recouvrement dès lors que deux blocs ont au moins une cellule commune.

Définition 4 - Recouvrement de blocs. *Deux blocs se recouvrent s'ils ont au moins une cellule en commun.*

Il est facile de voir que deux blocs $b = \delta_1 \times \ldots \times \delta_k$ et $b' = \delta'_1 \times \ldots \times \delta'_k$ du même cube *se recouvrent* si et seulement si pour toute dimension d_i $\delta_i \cap \delta'_i \neq \emptyset$.

La notion de tranche, qui peut être vue comme un bloc particulier, permet de considérer l'ensemble des cellules associées à une valeur de dimension, et sera importante dans le cadre de la définition des algorithmes. Nous l'introduisons donc ci-dessous.

Définition 5 - Tranche d'un cube. *Soit v_i une valeur de la dimension d_i. On appelle* tranche *(ou* slice*) de C associée à v_i, notée $\mathcal{T}_{v_i}$, le bloc $\delta_1 \times \ldots \times \delta_k$ tel que pour tout $j \neq i$, $\delta_j = ALL$ et $\delta_i = \{v_i\}$.*

Une tranche est donc un hyperplan, réduit à une ligne ou une colonne dans le cas particulier d'un cube à deux dimensions. Les notions de support et de confiance associées à un bloc et une valeur de mesure sont définies comme suit :

Définition 6 - Support. *On définit le support d'un bloc de données b dans C pour une valeur de mesure m comme :*

$$supp(b, m) = \frac{\#\ occurrences\ de\ m\ dans\ b}{\#\ cellules\ de\ C}$$

Étant donné un seuil de support σ fixé par l'utilisateur et une valeur de mesure m, un bloc b tel que $supp(b, m) > \sigma$ est appelé σ-fréquent pour m.

On note que le support est anti-monotone, c'est-à-dire que pour tous blocs b, b' et pour tout m :

$$b \subseteq b' \Rightarrow support(b, m) \leq support(b', m)$$

Définition 7 - Confiance. *On définit la confiance d'un bloc de données b pour une valeur de mesure m comme :*

$$conf(b, m) = \frac{\#\ occurrences\ de\ m\ dans\ b}{\#\ cellules\ de\ b}$$

Nous considérons des blocs maximalement spécifiques, c'est-à-dire les blocs définis à partir d'un nombre maximal de dimensions.

Définition 8 - Bloc maximalement spécifique. *Soit σ un seuil de support, m une valeur de mesure, et b un bloc σ-fréquent pour m, b est dit* maximalement spécifique pour m et un seuil de confiance γ *si*

- $conf(b, m) > \gamma$
- *il n'existe pas de bloc b' tel que :*
  - *b' est σ-fréquent pour m*
  - *$\exists j \in [1, k]$ tel que $\delta'_j = ALL$ et $\delta_j \neq ALL$*
  - *$\forall j' \in [1, k]$, $j' \neq j \Rightarrow \delta'_j = \delta_j$*
  - *$conf(b', m) > \gamma$.*

4.2 Génération des blocs

Dans ce travail, nous recherchons les blocs ayant une proportion de cellules de même valeur suffisante pour l'utilisateur. La recherche est fondée sur l'utilisation d'un algorithme par niveaux dérivé des travaux sur APriori [1], ce type d'algorithmes permettant de proposer des outils efficaces passant à l'échelle. Le but de l'algorithme proposé (voir Algorithme 2) est de construire les règles pour lesquelles la valeur de mesure est déterminée par un maximum de dimensions. Cet algorithme permet également de construire les blocs de taille maximale, en considérant non plus les règles les plus spécifiques mais les règles les plus générales.

On note que le seuil de support détermine la taille minimale des blocs tandis que le seuil de confiance détermine l'homogénéité à l'intérieur des blocs. En effet, pour une valeur de seuil de support donnée σ, si on note N le nombre de cellules du cube, un bloc ne peut être fréquent que s'il contient au moins $\sigma * N$ cellules. D'autre part, pour une valeur de seuil de confiance donnée γ, un bloc de cardinalité M n'est retenu que s'il contient au moins $\gamma * M$ cellules contenant la valeur de mesure m par rapport à laquelle les calculs sont effectués.

Data : C cube de données défini sur k dimensions, σ seuil de support minimum et γ seuil de confiance minimale.
Result : $\mathcal{B}$ l'ensemble des blocs associés au cube C
foreach *valeur de mesure m du cube* C [1] **do**
  foreach *dimension* d_i *(i=1,...,k)* **do**
    $\mathcal{L}_1^i \leftarrow \{v(d_i) \in dom(d_i) | supp(\mathcal{T}_{v(d_i)}, m) > \sigma\}$ où $\mathcal{T}_{v(d_i)}$ est la tranche associée à la valeur $v(d_i)$;
    Construire les intervalles maximaux $\delta_{i_j} = [\alpha_{i_j}, \beta_{i_j}]$ tels que pour toute valeur $v(d_i)$ située sur d_i entre α_{i_j} et β_{i_j} on a $v(d_i) \in \mathcal{L}_1^i$;
  for $l = 2$ *à* k **do**
    Générer les candidats à partir des fréquents de taille $l-1$. Étant dans le cas de données non binaires, les candidats devront regrouper des intervalles de valeurs sur des dimensions différentes.
    Pour chaque candidat $\delta_{i_1} \times \ldots \times \delta_{i_l}$, considerer le bloc $\delta_1 \times \ldots \times \delta_k$ où $\delta_p = \delta_{p_j}$ si la dimension d_p a été traitée et $\delta_p = ALL$ sinon ;
    Coupure : Supprimer tous les candidats $\delta_{i_1} \times \ldots \times \delta_{i_l}$ tels qu'il existe $p \in \{1, \ldots, l\}$ tel que $\delta_{i_1} \times \ldots \times \delta_{i_{p-1}} \times \delta_{i_{p+1}} \times \ldots \times \delta_{i_l}$ n'est pas fréquent ;
    Évaluer les supports des blocs candidats et supprimer les candidats non fréquents (support $\leq \sigma$) ;
  Supprimer les blocs b tels que $conf(b, m) \leq \gamma$;
$\mathcal{B} \leftarrow$ { ens. des blocs engendrés } ;

Algorithme 1: Algorithme de recherche des blocs maximalement spécifiques

L'algorithme ci-dessus peut facilement être adapté pour construire les *blocs maximaux* (règles les moins spécifiques). Il suffit pour celà de calculer à chaque étape la confiance associée aux blocs et de stopper le parcours des dimensions pour un bloc dès qu'il atteint un niveau de confiance suffisant.

Notre méthode peut être vue comme une méthode de segmentation. Nous sommes conscients que la méthode que nous proposons ne permet pas toujours de retrouver tous les blocs de données. Cependant, cette méthode est efficace pour détecter les blocs de données homogènes les plus pertinents.

Il est alors possible de produire des règles décrivant les blocs de données.

Quand un bloc b_j ne recouvre aucun autre bloc ($B_j = \emptyset$), la règle produite est du type : *Si* $d_1 = \delta_{1,j}$ *et* ... *et* $d_k = \delta_{k,j}$ *Alors* m_j où m_j est la valeur de mesure associée et où les ensembles $\delta_{i,j}$ sont exprimés à l'aide de clauses disjonctives. Par exemple, sur la Fig. 4.2, la règle produite pour le bloc b_1 correspondant à la valeur 6 est la suivante :

Si la ville est $V1$ *et le produit est* $P1$ *ou* $P2$ *Alors la valeur des cellules est* 6

En cas de recouvrement, notre méthode a recours à des règles floues pour exprimer l'imprécision de la définition des blocs. Les fonctions d'appartenance des sous-ensembles flous sont construites de manière automatique [15]. Les règles produites deviennent alors de la forme :

Si la ville est $V1$ *et le produit est* $P1$ *ou* $P2$ dans une moindre mesure *Alors la valeur des cellules est* 6

qui permet de dire que quand le produit $P2$ est concerné, alors plusieurs blocs cohabitent.

Rappelons que ces blocs sont construits à partir d'une représentation donnée dont il est alors possible d'estimer la qualité au sens du regroupement de valeurs de cellules.

4.3 Qualité des représentations

Dans [14], différentes manières de représenter un cube ont été étudiées. Il est en particulier montré dans cet article que certaines représentations des données sont plus pertinentes que d'autres puisqu'elles permettent de rapprocher des informations et de déduire ainsi des connaissances sur les données. Dans l'approche rapportée dans ce chapitre, nous considérons comme intéressants les rapprochements consistant à regrouper les valeurs de mesure identiques. Il existe d'autres possibilités d'organisations intéressantes, par exemple décrites dans [13], où les données sont organisées de telle sorte que la mesure est rangée en ordre croissant le long de toutes les dimensions. Cependant, il est très difficile d'organiser

automatiquement les cubes de données d'une manière pertinente. Des méthodes existent, issues des statistiques notamment, mais leur complexité ne permet pas d'envisager leur application sur les données issues des entrepôts de données ayant de nombreuses dimensions.

Dans l'approche présentée ici, l'organisation des données n'est pas modifiée avant la construction des règles. Il serait bien sûr intéressant d'organiser le cube afin que les blocs de données soient les plus grands possibles et se recouvrent le moins possible. Mais cette tâche ne constitue pas le but de nos travaux présents. Cependant, il est également intéressant de considérer le problème inverse et d'évaluer la qualité de la représentation à partir des règles construites. Par qualité de la représentation, on entendra *représentation groupée selon les valeurs de cellule*. Cette qualité s'exprime donc en fonction :

- de la proportion de cellules incluses dans des blocs (plus cette proportion est importante, moins il y aura de données non concernées par les règles construites),
- du nombre de blocs construits (plus il y a de blocs, plus les données sont hétérogènes),
- du nombre de blocs par rapport au nombre de valeurs de mesure (retrouver plusieurs blocs correspondant à la même valeur signifie que cette valeur n'est pas bien rangée de manière contigüe),
- du nombre de recouvrements entre blocs et de leur taille (plus les blocs se recouvrent, plus les données sont mélangées).

Dans ce chapitre, nous avons présenté nos travaux menés pour la découverte automatique de blocs de données flous au sein d'entrepôts de données, menés dans le cadre de collaborations avec la Malaisie et des partenaires industriels. Cependant, un aspect fondamental des entrepôts reste à étudier : l'exploitation des hiérarchies présentes sur les dimensions. De plus, notre méthode a été améliorée afin de découvrir une majorité des blocs présents au sein de l'entrepôts en gérant les voisinages de cellules, comme décrit dans le chapitre suivant.

Chapitre 5

Extension des approches : prise en compte des voisinages de cellules et blocs multi-niveaux

5.1 Raffinement du calcul des blocs

Dans cette section, nous étudions comment prendre en compte le *voisinage* des cellules afin d'améliorer la complétude de notre méthode.

Une cellule est considérée comme voisine d'une autre si elle partage au moins une valeur sur l'une des dimensions dans la représentation. Par exemple, les cellules $\langle P2, C3, 5\rangle$ et $\langle P2, C4, 8\rangle$ sont voisines.

Définition 9 - Voisinage de Cellule. *Deux cellules* $c = \langle v_1, \ldots, v_k, m\rangle$ *et* $c' = \langle v'_1, \ldots, v'_k, m'\rangle$ *(*$c \neq c'$*) sont dites* voisines *s'il existe un unique* $i_0 \in \{1, \ldots, k\}$ *tel que :*

- $|rep_{i_0}(v_{i_0}) - rep_{i_0}(v'_{i_0})| = 1$ *et*
- *pour chaque* $i = 1, \ldots, k$ *tel que* $i \neq i_0$, $v_i = v'_i$.

Notons que dans un cube à k dimensions, une cellule a au plus $2 \,.\, k$ voisins. De plus, si l'on considère une tranche $\mathcal{T}(v)$ avec v appartient au domaine dom_i de la dimension i, soit v^- et v^+ les membres de dom_i tels que $rep_i(v^-) = rep_i(v) - 1$ et $rep_i(v^+) = rep_i(v) + 1$, respectivement.

Ici, chaque cellule c de $\mathcal{T}(v)$ a *exactement un* voisin dans chacune des tranches $\mathcal{T}(v^-)$ et $\mathcal{T}(v^+)$. Si l'on considère une valeur de mesure m, on note $n(v^-, m)$,

respectivement $n(v^+, m)$, le nombre de m-cells de $\mathcal{T}(v)$ dont le voisin dans $\mathcal{T}(v^-)$, et respectivement dans $\mathcal{T}(v^+)$, est aussi une m-cellule. Alors, on définit $neighbors(v^-, m)$ et $neighbors(v^+, m)$:

$$neighbors(v^-, m) = \frac{n(v^-,m)}{Count(\mathcal{T}(v),m)} \quad \text{et} \quad neighbors(v^-, m) = \frac{n(v^+,m)}{Count(\mathcal{T}(v),m)}.$$

Intuitivement, $neighbors(v^-, m)$ et $neighbors(v^+, m)$ sont respectivement les ratios de m-cellules dans une tranche donnée ayant une m-cellule comme voisine dans la tranche précédente (respectivement suivante).

A partir de ces définitions, notre méthode fonctionne comme suit : nous considérons un seuil additionnel au support et confiance, nommé *seuil de voisinage*, noté ν. Alors lorsqu'une dimension i est scannée pour une valeur de mesure m, si v est la valeur de dom_i dont la tranche est en cours d'examen, et qu'un intervalle $[V, NIL]$ est en cours de construction (avec $V \neq NIL$) et que le support de $\mathcal{T}(v)$ est supérieur ou égal au seuil de support, alors :

- Si $neighbors(v^-, m) < \nu$, alors l'intervalle $[V, v^-]$ est produit et le calcul du nouvel intervalle $[v, NIL]$ est considéré.
- Si $neighbors(v^+, m) < \nu$, alors l'intervalle $[V, v]$ est produit et le calcul du nouvel intervalle $[v^+, NIL]$ est considéré.
- Sinon, la tranche suivante, *i.e.,* la tranche défine par v^+, est considérée pour lintervalle $[V, NIL]$.

Afin de prendre en compte cette notion de voisinage dans le calcul des blocs considérant des intervalles (flous ou non) sur la valeur de mesure, nous redéfinissons les comptages, comme décrit ci-dessous.

Soit v une valeur de dimension et un intervalle sur la valeur de mesure $[m_1, m_2]$, on note $i\_n(v^-, [m_1, m_2])$ (respectivement $i\_n(v^+, [m_1, m_2])$) le nombre de cellules de $\mathcal{T}(v)$ dont le contenu est compris dans $[m_1, m_2]$ et dont le voisin dans $\mathcal{T}(v^-)$, (respectivement dans $\mathcal{T}(v^+)$) est une cellule dont le contenu est dans $[m_1, m_2]$. Alors, $i\_neighbors(v^-, [m_1, m_2])$ et $i\_neighbors(v^+, [m_1, m_2])$ sont définis comme suit :

$$i\_neighbors(v^-, [m_1, m_2]) = \frac{i\_n(v^-,[m_1,m_2])}{iCount(\mathcal{T}(v),[m_1,m_2])} \quad \text{et}$$

$$i\_neighbors(v^+, [m_1, m_2]) = \frac{i\_n(v^+,[m_1,m_2])}{iCount(\mathcal{T}(v),[m_1,m_2])}.$$

Dans le cas d'intervalles flous, nous considérons une t-norm notée $\otimes$ afin de calculer à quel point deux cellules c and c' appartiennent au sous-ensemble flou φ. On note alors $\mu(c, \varphi) \otimes \mu(c', \varphi)$.

On a alors

$$f\_neighbors(v^-, \varphi) = \frac{\Sigma^f_{c \in \mathcal{T}(v)} \mu(c,\varphi) \otimes \mu(c^-,\varphi)}{fCount(\mathcal{T}(v),\varphi)}, \text{ et}$$

$$f\_neighbors(v^+, \varphi) = \frac{\Sigma^f_{c \in \mathcal{T}(v)} \mu(c,\varphi) \otimes \mu(c^+,\varphi)}{fCount(\mathcal{T}(v),\varphi)}$$

avec c^- and c^+ sont les voisins de c dans $\mathcal{T}(v^-)$ et $\mathcal{T}(v^+)$, respectivement.

Il est alors possible de démontrer que notre approche est complète (retrouve tous les blocs présents dans le cube) pour une valeur de mesure donnée si le cube peut être partitionné en blocs non recouvrants [10].

Nous avons présenté ici une méthode d'extraction de sous-cubes de données homogènes à partir de cubes de données. Cette méthode ne considère pourtant pas les hiérarchies potentiellement définies sur les dimensions. Nous présentons donc ci-après une méthode étendue à la gestion de blocs multi-niveaux.

5.2 Blocs multi-niveaux

Nous considérons ici les hiérarchies $H_1, \ldots, H_k$ potentiellement définies sur les k dimensions d'un cube C. Les ensembles de blocs sont alors munis de relations de spécificités, comme défini ci-dessous.

Définition 10 - Relation de Spécificité. *Soit* $b = \delta_1 \times \ldots \times \delta_k$ *et* $b' = \delta'_1 \times \ldots \times \delta'_k$ *deux blocs. On dit que* b' *est* plus spécifique *que* b*, et l'on note* $b \sqsubseteq b'$*, si pour chaque* $i = 1, \ldots, k$*,*

$$\delta_i \neq \delta'_i \Rightarrow (j > j') \wedge (h_i^j(\delta'_i) \subseteq \delta_i)$$

tel que j *et* j' *sont des niveaux de la hiérarchie* H_i *sur laquelle* δ_i *et* δ'_i *sont définis,* j *et* j' *étant donc des entiers de* $\{0, \ldots, h_i\}$ *tels que* $\delta_i \subseteq dom_i^j$ *et* $\delta'_i \subseteq dom_i^{j'}$.

Par exemple, sur le cube de la figure 4.1, pour $b = [\top_{PRODUIT}] \times [V1, V3]$ et $b' = [P1] \times [V1, V3]$, nous avons $b \sqsubseteq b'$ puisque les intervalles définissant b et b' satisfont la définition ci-dessus.

Il est alors possible de monter que $\sqsubseteq$ définit un ordre partiel sur l'ensemble des blocs du cube C.

On montre également que le support est anti-monotone vis-à-vis de cet ordre partiel et que l'espace de recherche pour construire les blocs fréquents à partir d'un algorithme par niveaux est un treillis [11].

Mise en œuvre

Etant donné un cube C à k dimensions muni des hiérarchies $H_1, \ldots, H_k$, un seuil de support σ et un seuil de confiance γ, notre méthode fonctionne selon les étapes décrites ci-dessous, pour un intervalle flou φ défini sur les valeurs de mesure :

1. *étape 1.* Calculer tous les intervalles de valeurs définissant une tranche σ-fréquente pour φ.
2. *étape 2.* Calculer tous les blocs σ-frequents pour φ, à partir des intervalles obtenus à l'étape 1.
3. *étape 3.* Parmi tous les blocs de l'étape 2, supprimer tous les blocs non maximalement spécifiques.

Notons également que notre approche a été étendue au cas des hiérarchies floues [11].

Discussion

Les travaux liés aux blocs de données, présentés dans cette partie, ont pour but de guider l'utilisateur vers les sous-cubes de données correspondant à des zones homogènes (au sens de la valeur de mesure contenue dans les cellules).

Ils permettent ainsi de retrouver facilement et rapidement, en fonction des valeurs de dimension, les zones correspondant à des caractéristiques de la mesure qui est au centre des préoccupations de l'analyste.

Définis à partir d'une valeur simple, d'un intervalle de valeurs, ou d'un intervalle flou, les blocs rendent ainsi très bien compte du contenu du cube.

En plus de cette capacité à pointer les zones homogènes, notre approche permet de déceler les exceptions qui sont les cellules contenant des valeurs très éloignées de la valeur associée au bloc. Cela a été notamment mis en valeur dans notre proposition de visualisation non rapportée ici [12].

Ces travaux ont donc permis de mieux cerner l'influence de la relation de hiérarchie sur la définition du support dans le contexte des entrepôts. Réalisés en partenariat avec la Malaisie, ils ont renforcé notre certitude que le couplage des méthodes de fouille de données et des entrepôts de données était prometteur, pour peu qu'il intègre de la souplesse (ici dans la définition de ce qu'est une valeur de bloc homogène avec les autres). Notons qu'ils ont fait l'objet de nombreuses expérimentations, rapportées notamment dans les articles [11, 10, 15, 12].

Cependant, de nombreuses perspectives restent ouvertes suite à cette approche, notamment pour mieux exploiter les ensembles de blocs découverts (par exemple à l'aide de règles graduelles, comme nous le verrons dans les perspectives de ce mémoire), ou encore en associant la découverte de blocs à la recherche efficace de représentations pertinentes.

En outre, ces travaux ne tiennent pas compte du fait que les données d'entrepôts sont historiées. Dans la suite de ce mémoire, nous nous focalisons donc plus précisément sur la définition de méthodes intégrant pleinement la dimension temporelle.

Troisième partie

Extraction de motifs séquentiels à partir d'entrepôts de données

Comme nous l'avons vu précédemment, les bases de données multidimensionnelles présentent des caractéristiques propres qui rendent difficile l'application directe des algorithmes de fouille de données. La découverte de blocs de données est l'une des méthodes de fouille de données possibles, cependant elle ne permet pas d'exhiber des règles et motifs dans lesquels la notion de temporalité est présente. Or la présence de la dimension temporelle est l'une des caractéristiques fortes des entrepôts de données. Nous nous intéressons donc ici à l'extraction de motifs séquentiels multidimensionnels.

Ces travaux ont été initiés dans le cadre de la collaboration avec la Malaisie et du projet STIC-Asia EXPEDO (financement du Ministère des Affaires Étrangères) destiné à étudier des méthodes capables d'exploiter intelligemment des entrepôts de données.

Ces travaux ont été également réalisés dans le cadre de la thèse de M.Plantevit (co-encadrée avec M. Teisseire, soutenue en juillet 2008) et du stage de Master de Delphine Jouve (soutenu en juillet 2007). Ils ont fait l'objet d'un transfert technologique dans le cadre de la collaboration avec EDF R&D.

Thèmes abordés	Entrepôts de données, hiérarchie, mesure, dimensions, motif séquentiel
Étudiants	D. Jouve (M2R - co-encadrement M. Teisseire) M. Plantevit (thèse - co-encadrement M. Teisseire)
Collaborations	EDF R&D Projet STIC-Asia EXPEDO (Malaisie notamment)

Dans la suite de ce chapitre, nous présentons tout d'abord les travaux liés à la définition de motifs séquentiels multidimensionnels puis nous exposons comment la dimension particulière qu'est la mesure peut être prise en compte dans ce contexte.

Chapitre 6

Motifs séquentiels multidimensionnels

6.1 M^{3}SP (*M*ining *M*ultidimensional and *M*ulti-Level *S*equential *P*atterns)

Combiner plusieurs dimensions d'analyse permet d'extraire des connaissances qui décrivent mieux les données. Dans [49] les auteurs sont les premiers à rechercher des motifs séquentiels multidimensionnels pour extraire des connaissances de la forme « *les consommateurs de la catégorie socio-professionnelle X achètent fréquemment le produit a puis les jeunes consommateurs achètent fréquemment le produit c* ». Cependant les corrélations restent extraites sur la seule dimension *Produit* au cours du temps.

Nous avons donc défini M^{3}SP (*M*ining *M*ultidimensional and *M*ulti-Level *S*equential *P*atterns).

Nous détaillons ici les concepts issus de notre collaboration avec la Malaisie et associés à la thèse de M. Plantevit [53, 50]. Considérons une base de données DB définie sur un ensemble de dimensions $\mathcal{D}$. Afin de permettre à l'utilisateur une plus grande liberté dans le choix des différents paramètres de l'extraction, nous proposons de réaliser une partition de $\mathcal{D}$ en quatre sous-ensembles :

- D_T pour les dimensions *temporelles* [1], l'ensemble des dimensions permettant d'introduire une relation d'ordre entre les événements (*e.g. temps*) ;
- D_A pour les dimensions d'*analyse*, l'ensemble des dimensions sur lesquelles les corrélations sont extraites (les dimensions décrivant les motifs

1. Notons que la relation d'ordre peut être introduite par des dimensions autres que des dimensions temporelles. Nous utilisons le terme « temporel » car c'est sur cette dimension que les motifs extraits sont le plus facilement interprétables, mais la relation d'ordre pourrait très bien être introduite par des dimensions géographiques par exemple.

extraits) ;
- D_R pour les dimensions de *référence*, l'ensemble des dimensions permettant de calculer le support d'une séquence et donc de déterminer si elle est fréquente ou non ;
- D_I pour les dimensions *ignorées*, l'ensemble des dimensions qui ne sont pas prises en compte durant l'extraction des motifs séquentiels multidimensionnels.

Chaque nuplet $c=(d_1, \ldots, d_n)$ peut ainsi s'écrire $c=(i, r, a, t)$ où i est la restriction sur D_I de c, r sa restriction sur D_R, a sa restriction sur D_A, et t sa restriction sur D_T.

Etant donnée une base de données DB, on appelle *bloc* l'ensemble des n-uplets ayant la même valeur r sur D_R. L'ensemble des blocs de DB est noté B_{DB,D_R}. Ainsi chaque bloc B_r de B_{DB,D_R} est décrit pas le n-uplet r qui le définit.

Notons que cette notion de bloc diffère de celle introduite dans la partie II.

Chaque bloc défini sur D_R identifie une séquence multidimensionnelle de données. La base exemple du Tableau 6.1 contient trois blocs différents identifiés par $CID{=}C_1, CID{=}C_2$ et $CID{=}C_3$.

Un bloc *supporte* une séquence s si on peut retrouver dans la séquence de données identifiée par ce bloc tous les items de tous les itemsets de s tout en respectant la relation d'ordre définie sur D_T.

Le but de l'extraction des motifs séquentiels multidimensionnels est de découvrir l'ensemble complet des séquences fréquentes, étant donnés une base de données DB et un seuil de support minimum σ. Durant l'extraction des motifs séquentiels multidimensionnels, l'ensemble D_R identifie les blocs de la base de données qui doivent être considérés pour calculer le support d'une séquence. C'est pour cette raison que cet ensemble est nommé *référence*. Remarquons que pour les motifs séquentiels « classiques » l'ensemble D_A décrit les dimensions d'*analyse*, ainsi les motifs définis sur ces dimensions seront découverts par un algorithme d'extraction de motifs séquentiels multidimensionnels. Pour les motifs séquentiels classiques, une seule dimension d'analyse est considérée, correspondant par exemple aux produits vendus ou aux pages web visitées. Enfin, l'ensemble D_I décrit les dimensions *ignorées* qui ne sont ni requises pour définir la relation d'ordre ou les motifs extraits, ni pour identifier les blocs.

Afin d'illustrer les différentes définitions, nous considérons une société de vente en ligne stockant les opérations de ses clients dans une base de données. Le tableau 6.1 représente un morceau de cette base de données. La partition des dimensions est la suivante :

$$D_I{=}\emptyset, D_R{=}\{\text{CID}\}, D_T{=}\{Date\} \text{ et } D_A{=}\{City, Cust\text{-}Grp, A\text{-}Grp, Product\}$$

Nous pouvons alors définir les concepts d'item, itemset et séquence multidimensionnels dans ce cadre.

CID	Date	City	Customer Informations		Product
			Cust-Grp	*Cust-Age*	
C_1	1	*NY*	*Educ.*	*Middle*	*A*
C_1	1	*NY*	*Educ.*	*Middle*	*B*
C_1	2	*LA*	*Educ*	*Middle*	*C*
C_2	1	*SF*	*Prof.*	*Middle*	*A*
C_2	2	*SF*	*Prof.*	*Middle*	*C*
C_3	1	*DC*	*Business*	*Retired*	*A*
C_3	1	*LA*	*Business*	*Retired*	*B*

TABLE 6.1 – DB : Base de Données « Exemple »

Définition 11 (Item multidimensionnel) *Un item multidimensionnel* $e = (d_1, \ldots, d_m)$ *est un n-uplet défini sur* D_A. *Ainsi, pour chaque* $i = 1, \ldots, m$, *nous avons* $d_i \in dom(D_i)$ *et* $D_i \in D_A$.

Notons qu'un tel item peut être défini à partir des valeurs présentes à tous les niveaux des hiérarchies associées aux dimensions d'analyse. Il est alors possible de comparer des items multidimensionnels, et nous définissons une relation de spécificité :

Définition 12 (Relation de spécificité) *Soient* e *et* e' *deux items multidimensionnels, avec* $e = (d_1, \ldots, d_m)$ *et* $e' = (d'_1, \ldots, d'_m)$. *On dit que* e *est* plus général *que* e', *et on note* $e \geq_h e'$, *si pour chaque* $i = 1, \ldots, m$, *on a* $d_i = d'_i$ *ou* $d_i \in d'_i \uparrow$. *De manière duale, on dit que* e *est* plus spécifique *que* e', *et on note* $e \leq_h e'$, *si pour chaque* $i = 1, \ldots, m$, *on a* $d_i = d'_i$ *ou* $d_i \in d'_i \downarrow$.

Un itemset multdimensionnel est alors défini comme un ensemble d'items multdimensionnels :

Définition 13 (Itemset multidimensionnel) *Un itemset multidimensionnel* $i = \{e_1, \ldots, e_k\}$ *est un ensemble non vide d'items multidimensionnels tels que pour chaque couple* i, j *de* $\{1, \ldots, k\}$, e_i *et* e_j *ne sont pas comparables par* $\leq_h$.

Définition 14 (Séquence multidimensionnelle) *Une séquence multidimensionnelle* $s = \langle i_1, \ldots, i_j \rangle$ *est une liste ordonnée non vide d'itemsets multidimensionnels.*

Définition 15 (Inclusion d'itemset) *Soient* a *et* a' *deux itemsets. On dit que* a *est un sous-itemset de* a', *et on note* $a \sqsubseteq a'$, *si pour chaque item* i *de* a, *il existe un item* i' *de* a' *tel que* $i \leq_h i'$.

La notion d'inclusion de séquences peut alors être définie.

Définition 16 (Inclusion de Séquence) *Une séquence* $\varsigma = \langle a_1, \ldots, a_l \rangle$ *est dite incluse dans la séquence* $\varsigma' = \langle b_1, \ldots, b_{l'} \rangle$ *s'il existe des entiers* $1 \leq j_1 \leq j_2 \leq \ldots \leq j_l \leq l'$ *tels que* $a_1 \sqsubseteq b_{j_1}, a_2 \sqsubseteq b_{j_2}, \ldots, a_l \sqsubseteq b_{j_l}$.

Calculer le support d'une séquence revient alors à compter le nombre de blocs définis sur les dimensions de référence D_R qui supportent la séquence.

Définition 17 *Un bloc supporte une séquence* $\langle i_1, \ldots, i_l \rangle$ *si pour chaque* $(j = 1 \ldots l)$*, il existe* d_j *dans* $Dom(D_t)$ *tel que pour chaque* e *de* i_j*, il existe* $t = (r, e, d_j)$ *de* T *qui supporte* e *et respecte* $d_1 < d_2 < \ldots < d_l$.

Le support *d'une séquence* ς*, noté* $sup(\varsigma)$*, est le nombre de blocs qui supporttent la séquence. Soit un seuil* min-sup*, une séquence est fréquente si son support est supérieur à* min-sup.

Il faut noter que la propriété d'anti-monotonie du support est toujours vérifiée pour l'inclusion de séquence.

Proposition 1 *Si* $\varsigma \sqsubseteq \varsigma'$ *alors* $sup(\varsigma') \leq sup(\varsigma)$.

Cette propriété nous permet de proposer des algorithmes efficaces pour extraire les motifs séquentiels fréquents.

6.2 Algorithmes

Nous détaillons ici les algorithmes proposés pour la découverte de motifs séquentiels à différents niveaux de granularité. Ces algorithmes font partie d'un processus en deux étapes :

1. Dans un premier temps, les items multidimensionnels fréquents sont extraits au niveau de hiérarchie le plus adapté.
2. Dans un deuxième temps, ces items sont combinés afin d'extraire les itemsets et séquences fréquents.

Extraction des items multidimensionnels fréquents

La recherche des items multidimensionnels fréquents pourrait se résumer à l'extraction de n-uplets de D_A fréquents et pourrait donc s'effectuer en une passe sur la base de données. Cependant, une valeur particulière $(*)$ peut être utilisée lors de ce processus et apparaître dans les items, signifiant que la dimension n'est pas spécifiée. L'espace de recherche est alors constitué d'un treillis similaire à celui exploré dans les approches de fouille de données classiques. Nous proposons

donc une approche par niveaux depuis la borne inférieure $(*, \ldots, *)$ jusqu'aux items contenant le moins de symboles $*$ possible. Au niveau i, i valeurs sont spécifiées, et les items de ce niveau sont combinés pour construire les candidats du niveau $i+1$. Deux items fréquents sont combinés s'ils sont compatibles au sens de la jointure. On les dit alors $\bowtie$-compatibles au sens de la définition ci-dessous :

Définition 18 ($\bowtie$-compatibilité) *Soient $e_1 = (d_1, \ldots, d_n)$ et $e_2 = (d'_1, \ldots, d'_n)$ deux items multidimensionnels avec d_i et $d'_i \in dom(D_i) \cup \{*\}$. On dit que e_1 et e_2 sont $\bowtie$-compatibles s'il existe $\Delta = \{D_{i_1}, \ldots, D_{i_{n-2}}\} \subset \{D_1, \ldots, D_n\}$ tel que pour chaque $j \in [1, n-2]$, $d_{i_j} = d'_{i_j} \neq *$ avec $d_{i_{n-1}} = *$ et $d'_{i_{n-1}} \neq *$ et $d_{i_n} \neq *$ et $d'_{i_n} = *$.*

Deux items sont alors joints de la manière suivante :

Définition 19 (Jointure) *Soient $e_1 = (d_1, \ldots, d_n)$ et $e_2 = (d'_1, \ldots, d'_n)$ deux items multidimensionnels $\bowtie$-compatibles. On définit $e_1 \bowtie e_2 = (v_1, \ldots, v_n)$ avec $v_i = d_i$ if $d_i = d'_i$, $v_i = d_i$ if $d'_i = *$ et $v_i = d'_i$ if $d_i = *$.*

Soit E et E' deux ensembles d'items multidimensionnels de taille n, on définit

$$E \bowtie E' = \{e \bowtie e' \mid (e, e') \in E \times E' \wedge \ e \text{ et } e' \text{ sont } \bowtie\text{-compatibles}\}$$

Ainsi si F_1^i est l'ensemble des items fréquents ayant i dimensions spécifiées (différentes de $*$), les items candidats ayant taille $i+1$ dimensions spécifiées sont obtenus par auto-jointure : $Cand_1^{i+1} = F_1^i \bowtie F_1^i$.

Par exemple, $(a, *, c)$ et $(*, b, c)$ sont $\bowtie$-compatibles et la jointure vaut (a, b, c) alors que les items $(a, b, *)$ et $(a, b, *)$ ne sont pas compatibles.

Fonction supportcount
Data : $\varsigma, T, D_R, counting$ //counting indicates if joker values are considered or not
Result : support of ς
Integer support $\longleftarrow 0$; *Boolean seqSupported*;
$B_{T,D_R} \longleftarrow \{blocks\ of\ T\ identified\ over\ D_R\}$;
foreach $B \in B_{T,D_R}$ **do**
  $seqSupported \longleftarrow supportTable(\varsigma, B, counting)$;
  if *seqSupported* **then** $support \longleftarrow support + 1$;
return $\left(\frac{support}{|B_{T,D_R}|}\right)$

Algorithme 2: Support d'une séquence (supportcount)

```
Fonction supportTable
Data   : ς, T, counting
Result : Boolean
ItemSetFound ⟵ false ; seq ⟵ ς ; itset ⟵ seq.first() ;
it ⟵ itset.first()
if ς = ∅ then return (true) // End of Recursivity
while t ⟵ T.next ≠ ∅ do
    if supports(t, it, counting) then
        if (NextItem ⟵ itset.second()) = ∅ then
        ItemSetFound ⟵ true
        // Look for all the items from the itemset
        else
            // Anchoring on the item (date)
            T' ⟵ σ_{date=t.date}(T)
            while t' ⟵ T'.next() ≠ ∅ ∧ ItemSetFound = false do
                if supports(t', NextItem, counting) then
                NextItem ⟵ itset.next()
                if NextItem = ∅ then ItemSetFound ⟵ true
        if ItemSetFound = true then
            // Anchoring on the current itemset succeeded ; test the
            other itemsets in seq
            return (supportTable(seq.tail(), σ_{date>t.date}(T), counting))
        else
            // Anchoring failure : try anchoring with the next dates
            itset ⟵ seq.first()
            T ⟵ σ_{date>t.date}(T) // Skip to next dates
return(false) // Not found
```

Algorithme 3: supportTable *(vérifie si une séquence ς est supportée par une table T)*

Extraction des séquences multidimensionnelles

L'algorithme précédemment présenté renvoie l'ensemble des séquences multidimensionnelles de taille 1 (un seul item). Nous appliquons ensuite un algorithme classique (par exemple PSP [46]) adapté au traitement de la valeur $*$.

Le calcul du support d'une séquence ς par rapport aux dimensions de référence D_R est donné par l'algorithme 2. Cet algorithme vérifie pour chaque bloc de la partition s'il supporte ou non la séquence (fonction *supportTable* de l'algorithme 3).

Grâce aux méthodes présentées ci-dessus, il est donc possible de trouver des règles de la forme $< \{(NY, planchesurf, *)(NY, housse, *)\}\{(LA, combinaison, *)\} >$ (les clients ont acheté des planches en même temps que des housses à New York puis des combinaisons à Los Angeles) qu'aucune autre approche de la littérature n'était capable d'extraire auparavant. Cette approche permet une prise en compte de la dimension multidimensionnelle et historique des entrepôts de données. De plus, les hiérarchies sont prises en compte. L'approche est donc originale, et a permis de faire face à des problèmes algorithmiques complexes liés à la nature des données manipulées.

Cependant, les tableaux de données gérés dans le cadre des bases de données multidimensionnelles sont plus complexes que ceux décrits dans le tableau 6.1, et contiennent une dimension particulière : la mesure, comme décrit dans le tableau 6.2.

La mesure est intégrée dans les dimensions d'analyse ($M \in D_A$). Par rapport aux définitions définies dans les chapitres précédents, il est assez intuitif de traiter cette dimension comme une dimension d'analyse et considérer seulement les cellules qui ont une mesure associée non vide.

Date	City	Customer Informations		Product	Mesure
		Cust-Grp	Age-Grp		
1	*NY*	**Educ.**	*Middle*	*A*	123
1	*NY*	**Educ.**	*Middle*	*B*	234
2	*LA*	**Educ.**	*Middle*	*C*	120
1	*SF*	**Prof.**	*Middle*	*A*	125
2	*SF*	**Prof.**	*Middle*	*C*	115
1	*DC*	**Business**	*Retired*	*A*	1
1	*LA*	**Business**	*Retired*	*B*	24

TABLE 6.2 – Partition en blocs en fonction de D_R=$\{Cust\text{-}Grp\}$

L'extraction de motifs séquentiels multidimensionnels s'appuie sur une gestion symbolique des données qu'elle traite. Ainsi, étant donnée la partition précédente, l'extraction de motifs séquentiels multidimensionnels a pour objectif de

découvrir des corrélations entre la ville, l'âge des consommateurs, les produits vendus et la mesure associée au cours du temps. Cependant, les motifs extraits présentent des limites non négligeables dues à la gestion symbolique de la mesure. En effet, en se basant sur les définitions précédentes, nous pouvons obtenir les situations suivantes :

Support de la séquence $\langle\{(*, M, A, 125)\}\rangle$ Le support absolu de la séquence $\langle\{(*, M, A, 125)\}\rangle$ est égal à 1. En effet, seul le bloc $B_{Prof.}$ supporte cette séquence. Le bloc $B_{Educ.}$ contient une séquence relativement similaire $\langle\{(*, M, A, 123)\}\rangle$. Toutefois, la gestion symbolique de la mesure (dimension numérique) implique que les valeurs 123 et 125 sont considérées comme totalement différentes.

Support de la séquence $\langle\{(*, *, A, *)\}\rangle$ Le support absolu de la séquence $\langle\{(*, *, A, *)\}\rangle$ est égal à 3. Les trois blocs supportent donc la séquence. Plus précisément, les items des séquences de données qui supportent la séquence (l'item) sont $(*, *, A, 123)$ pour $B_{Educ.}$, $(*, *, A, 125)$ pour $B_{Prof.}$ et $(*, *, A, 1)$ pour $B_{Business}$. Nous omettons les valeurs instanciées sur la ville, et l'âge afin de mettre en évidence l'observation suivante. $(*, *, A, 125)$ et $(*, *, A, 1)$ ont le *même impact* dans le calcul du support de la séquence $\langle\{(*, *, A, *)\}\rangle$.

Les deux points précédents soulignent les limites d'une gestion symbolique de la mesure dans l'extraction de motifs séquentiels multidimensionnels quand celle-ci est incluse dans les dimensions d'analyse. Il est donc nécessaire de prendre en compte la spécificité de cette dimension : son caractère numérique.

Nous proposons donc dans le chapitre suivant une étude des différents moyens d'appréhender une dimension numérique en général, et la mesure en particulier.

Chapitre 7

Motifs séquentiels multidimensionnels flous et prise en compte de la mesure

Au cours de ma thèse, je m'étais intéressée à la prise en compte de la mesure dans le cas de la construction de résumés flous et règles floues à partir de bases de données multidimensionnelles (potentiellement floues). Dans le cadre de mes travaux postérieurs, notamment pour la thèse de M. Plantevit et le stage de M2 Recherche de Dephine Jouve et en lien avec C. Fiot, nous nous sommes intéressés à la prise en compte de la mesure pour la construction de motifs séquentiels multidimensionnels. Ces travaux ont fait partie des études menées dans le cadre de la collaboration avec EDF R&D.

Même si les approches décrites précédemment s'attaquent à certaines spécifités inhérentes à OLAP comme la multidimensionnalité et la présence de hiérarchies, il n'y a pas de proposition qui tente de prendre directement en compte le caractère numérique de la mesure dans le cadre des motifs séquentiels, et peu dans le cadre des règles d'association [47]. Il existe de nombreux travaux permettant de discrétiser cette dimension numérique. Toutefois, ces travaux nécessitent un prétraitement des données, les algorithmes d'extraction de motifs séquentiels multidimensionnels étant exécutés sur les données discrétisées.

Dans mes travaux de thèse, je m'étais intéressée à la génération de résumés de données à partir de bases de données multidimensionnelles (floues ou non). Diverses possibilités ont alors été étudiées pour compter le suppport de ces réumés : prise en compte de la mesure comme support ou partitionnement en sous-ensembles flous. Plusieurs types de résumés sont considérés : intra-dimensions (avec prise en compte possible des hiérarchies), inter-dimensions (multidimensionnels), aide à la navigation par raffinement de résumés. Cependant, l'information temporelle n'est pas considérée comme une dimension particulière telle qu'elle l'est dans les motifs séquentiels.

A notre connaissance, il n'existe donc pas d'approche permettant la prise en compte de la mesure et son caractère numérique dans l'extraction de motifs séquentiels multidimensionnels. Nous proposons donc ici différentes façons de prendre en compte cette dimension :

- Nous utilisons des travaux de l'état de l'art sur la discrétisation des dimensions numériques à l'aide de partitions strictes ou floues afin de profiter de la *puissance informationnelle* de cette dimension particulière en l'intégrant dans les dimensions d'analyse et en extrayant ainsi des corrélations au sein de cette dimension.
- Nous proposons également de prendre en compte la mesure pour calculer le support des séquences multidimensionnelles (la mesure n'étant plus une dimension d'analyse). La mesure va donc nous permettre de déterminer la fréquence des séquences multidimensionnelles, ce qui représente une étape clé de l'extraction de motifs puisqu'une séquence est considérée comme fréquente si sa fréquence (support) est supérieure à un seuil minimal de fréquence appelé *support minimal*. La valeur d'agrégat d'une cellule peut être ainsi vue comme un « pré-calcul » du support d'une séquence. Nous proposons deux nouvelles définitions du support d'une séquence multidimensionnelle qui s'appuient sur la valeur des agrégats.

7.1 Discrétisation du domaine de la mesure

Dans cette section, nous proposons de discrétiser la mesure afin de bénéficier des informations présentes sur cette dimension. Lors de l'extraction de motifs séquentiels multidimensionnels, cette dimension peut alors être considérée de la même façon que les autres. La discrétisation d'un domaine de valeurs numériques peut se faire de plusieurs façons. Nous étudions ici différentes partitions possibles et comparons les motifs séquentiels multidimensionnels extraits selon la discrétisation opérée (partition de la mesure en intervalles stricts ou en sous-ensembles flous) et le comptage utilisé (normal ou flou).

Partition en intervalles stricts

Dans le cadre de l'extraction de connaissances par des techniques *symboliques* sur des données numériques, plusieurs approches ont été proposées afin de discrétiser les domaines de définition des attributs numériques en intervalles distincts. Il s'agit, la plupart du temps, de définir les bornes des intervalles de façon automatique. Plusieurs types de partitions sont couramment utilisés :

- Découpage *equi-width* où les intervalles ont tous la même largeur.
- Découpage *equi-depth* qui assure une équi-répartition des enregistrements dans chaque intervalle.
- Découpage selon la connaissance d'un expert ou le résultats de calculs statistiques.

La plupart des propositions qui s'attaquent à la découverte de motifs dans des données numériques à l'aide d'une partition en intervalles stricts [37, 56] soulignent la difficulté de déterminer les bornes optimales et le nombre d'intervalles. Des intervalles mal définis ont des conséquences sur la qualité des données extraites.

Dans le cas des intervalles stricts, la mesure est discrétisée et la base de donnée transformée en conséquence. Par exemple, par rapport au cube de données exemple (Tableau 6.2), nous choisissons la partition du domaine de la mesure en trois intervalles distincts :

- $Peu = [0, 99]$
- $Moyen = [100, 199]$
- $Beaucoup = [200, 300]$

Ainsi, chaque valeur m de mesure d'une cellule est associée à un unique intervalle parmi les trois définis. Le tableau Tab. 7.1 illustre le cube de données exemple après discrétisation de la mesure.

Le support absolu de la séquence $\langle\{(*, Middle, A, Moyen)\}\rangle$ est égal à 2. Avec la discrétisation des valeurs de mesure, les valeurs 123 et 125 appartiennent au même intervalle ($Moyen$) et sont donc considérées comme similaires lors de l'extraction de motifs séquentiels multidimensionnels. C'est ainsi que le bloc $B_{Educ.}$ supporte désormais la séquence $\langle\{(*, Middle, A, Moyen)\}\rangle$.

Date	City	Customer Informations		Product	Mesure		
					Peu	*Moyen*	*Beaucoup*
1	*NY*	**Educ.**	*Middle*	*A*	0	1	0
1	*NY*	**Educ.**	*Middle*	*B*	0	0	1
2	*LA*	**Educ.**	*Middle*	*C*	0	1	0
1	*SF*	**Prof.**	*Middle*	*A*	0	1	0
2	*SF*	**Prof.**	*Middle*	*C*	0	1	0
1	*DC*	**Business**	*Retired*	*A*	1	0	0
1	*LA*	**Business**	*Retired*	*B*	1	0	0

TABLE 7.1 – Partitions strictes des valeurs de la mesure

Partition en sous-ensembles flous

De nombreuses propositions ont été formulées afin d'utiliser des partitions floues d'attributs numériques en vue d'une extraction de connaissances symboliques [21, 8]. Ces travaux utilisent différentes techniques afin de réaliser le partitionnement flou (savoir d'un expert, equi-depth, equi-width, algorithmes génétiques, clustering).

L'utilisation d'une partition floue permet d'utiliser différentes méthodes de calcul du support d'une séquence comme développé dans [21]. On peut ainsi *pondérer* la présence d'un item multidimensionnel par le degré d'appartenance de la mesure à la valeur symbolique considérée. Le support d'une séquence multidi-

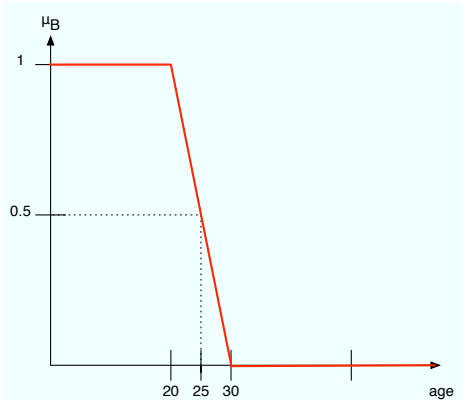

FIGURE 7.1 – Exemple : sous-ensemble flou *jeune* sur l'attribut *âge*

mensionnelle correspond à la moyenne pour tous les blocs de B_{DB,D_R} du degré d'appartenance de la séquence à chaque bloc. Ce degré est calculé en considérant l'intersection des sous-ensembles flous pour chaque item chaque itemset (utilisation d'une t-norme). Pour chaque bloc, la meilleure représentation sera renvoyée (utilisation d'une t-conorme).

Par rapport au cube de données exemple, la figure 7.2 illustre la partition floue en trois sous-ensembles *Peu, Moyen et Beaucoup* du domaine de la mesure. A partir de ces sous-ensembles flous, l'extraction des motifs séquentiels multidimensionnels s'effectue maintenant sur la base illustrée Tableau 7.2.

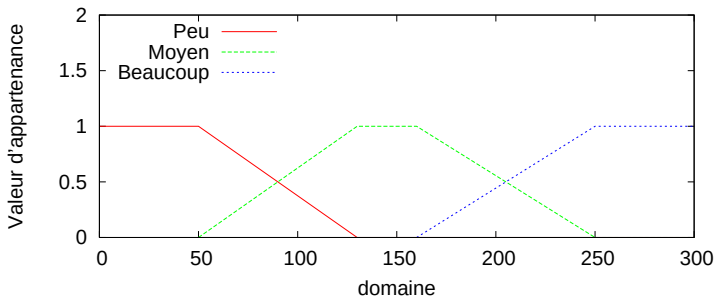

FIGURE 7.2 – Partitionnement flou de la mesure

Dans cette section, nous avons vu comment considérer la mesure comme l'une des dimensions d'analyse particulière. Dans la section suivante, nous nous intéressons au traitement de la mesure en la considérant comme un pré-calcul du support.

Date	City	Customer Informations		Product	Mesure		
					Peu	*Moyen*	*Beaucoup*
1	*NY*	**Educ.**	*Middle*	*A*	0.0875	0.925	0
1	*NY*	**Educ.**	*Middle*	*B*	0	0.18	0.82
2	*LA*	**Educ.**	*Middle*	*C*	0.125	0.875	0
1	*SF*	**Prof.**	*Middle*	*A*	0.0625	0.9375	0
2	*SF*	**Prof.**	*Middle*	*C*	0.1875	0.8125	0
1	*DC*	**Business**	*Retired*	*A*	1	0	0
1	*LA*	**Business**	*Retired*	*B*	1	0	0

TABLE 7.2 – Sous-ensembles flous sur les valeurs de la mesure

7.2 La mesure pour calculer le support

Dans la plupart des cas, les valeurs des agrégats d'un cube de données peuvent être vus comme un pré-calcul du support de certaines séquences. En effet, de nombreux cubes sont construits à l'aide d'une requête de type *Group By* et la mesure correspond alors à l'agrégation effectuée, qui est souvent à un comptage, sur le modèle du calcul du support.

Dans le cadre des motifs séquentiels, il est nécessaire de maintenir l'ordre d'apparition des événements dans la séquence ainsi que l'une des propriétés fondamentales inhérentes à l'extraction de motifs (multidimensionnels ou non) : *l'antimonotonie du support*. Soit un motif p, quel que soit P, un super motif (motif plus spécifique) de p, on a :

$$support(p) \geq support(P)$$

Tous les algorithmes d'extraction de motifs se basent sur cette propriété afin de parcourir efficacement l'espace de recherche pour extraire tous les motifs fréquents. Ainsi, ils partent de la séquence vide $\langle\rangle$ et essaient d'extraire des séquences plus longues, soit par un parcours niveau par niveau (*a priori*, [2, 46]), soit en profondeur d'abord (classes d'équivalence [63], Pattern-growth [48]). L'extraction de motifs séquentiels (multidimensionnels) a pour objectif d'établir des corrélations entre des événements suivant leur chronologie d'apparition. Il est ainsi nécessaire de maintenir l'ordre entre les éléments d'une séquence.

Pour préserver l'ordre d'apparition des événements dans la séquence ainsi que l'antimonotonie du support, nous utilisons une *t-norme* $\otimes$ qui est une généralisation de la conjonction logique. Une t-norme est un opérateur $[0,1]\times[0,1] \rightarrow [0,1]$ qui est associatif, commutatif, monotone et qui satisfait les conditions s $\alpha\otimes 0 = 0$ et $\alpha \otimes 1 = \alpha$. Les exemples les plus connus de t-norme sont le minimum $(\alpha, \beta) \mapsto \min(\alpha, \beta)$, le produit $(\alpha, \beta) \mapsto \alpha\beta$ et la t-norme de Lukasiewicz $(\alpha, \beta) \mapsto \max(\alpha + \beta - 1, 0)$.

Nous utilisons également une *t-conorme* $\oplus$ qui correspond à une disjonction logique. $\oplus$ est un opérateur $[0,1] \times [0,1] \rightarrow [0,1]$ qui est associatif, commutatif,

monotone et qui satisfait les conditions $\alpha \oplus 1 = 1$ et $\alpha \oplus 0 = \alpha$. Les exemples les plus connus de t-conorme sont le maximum $(\alpha, \beta) \mapsto \max(\alpha, \beta)$, la somme probabiliste $(\alpha, \beta) \mapsto \alpha + \beta - \alpha\beta$, la somme bornée $(\alpha, \beta) \mapsto \min(\alpha + \beta, 1)$, etc.

Etant donné qu'une séquence peut apparaître plusieurs fois dans une séquence de données identifiée par un bloc [1], il est nécessaire d'exhiber la combinaison qui « supporte le mieux » la séquence. Plus précisément, il faut exhiber les cellules qui ont la plus forte valeur de mesure et qui permettent de supporter la séquence.

Comme nous l'avons souligné précédemment, la mesure est souvent issue d'une agrégation et des dimensions décrivant très finement les données sont oubliées. Par exemple, l'identifiant individuel des clients n'est pas conservé quand on considère le nombre de ventes de chaque produit dans chaque ville. Quand on considère par exemple qu'il y a eu 123 unités du produit A chez les personnes d'âge moyen étudiantes habitant New York, la taille de la population sous-jacente des étudiants (ici la dimension de référence est la catégorie socio-professionnelle) est oubliée. Or elle a une signification forte pour les règles et motifs extraits. Nous proposons donc la possibilité de prendre en compte cette taille sous-jacente pour calculer ce que nous nommons le support relatif d'une séquence multidimensionnelle. De manière générale, nous définissons donc deux possibilités :

1. L'utilisateur peut considérer que l'importance des blocs doit s'exprimer dans le calcul du support d'une séquence. Ainsi, les blocs ont des poids différents en fonction de leur **effectif** ou **population**. L'importance d'un bloc intervient dans la valeur du support de la séquence. Par exemple, un bloc important à un impact plus important dans le support d'une séquence qu'un bloc de poids faible.
2. Comme pour les motifs séquentiels classiques où les dimensions de références D_R sont réduites à un singleton représentant l'identifiant d'une séquence de données (*e.g.* l'identifiant du client dans le contexte de l'analyse du panier de la ménagère), les blocs peuvent avoir des impacts égaux dans le support d'une séquence, et ceci quel que soit leur effectif sous-jacent.

Nous définissons ainsi deux façons de calculer le support relatif d'une séquence dans un cube de données suivant les deux points décrits précédemment.

Micro count prend en compte l'importance de chaque bloc lors du calcul du support d'une séquence. Ainsi, la mesure des cellules d'un bloc qui participent à supporter la séquence ($m[B_r, s_{i_j}]$) est divisée par la mesure totale ($m[cell(*, *, \ldots, *)]$).

Définition 20 (Micro Count)
Soit une g-k-séquence $s=\langle s_1, s_2, \ldots, s_g \rangle$, le support relatif de s dans un cube de données DB avec la technique micro count est égal à :

1. Nous rappelons ici que cette notion de bloc est liée à la partition des dimensions, et au partitionnement selon les dimensions de référence, et non aux blocs de données homogènes de la partie II.

$$\textit{Relative support}(s)=\sum_{\mathcal{B}_r \in B_{DB,D_R}} \bigoplus \bigotimes_{s_i \in s} \bigotimes_{s_{i_j} \in s_i} \frac{(m[B_r, s_{i_j}])}{m[(*,*,\ldots,*)]}$$

Pour chaque séquence apparaissant dans un bloc (tous les items de tous les itemsets doivent être présents en respectant la relation d'ordre), nous prenons la valeur minimale (t-norme) de la valeur de la mesure parmi les cellules supportant les items de la séquence (permet de garantir l'antimonotonie du support).

Puisqu'une séquence peut apparaître plusieurs fois dans la séquence de données pointée par la bloc, il faut considérer la meilleure solution, c'est-à-dire la combinaison la plus prometteuse. C'est pour cela que le support maximum (t-conorme) de cette séquence dans le bloc est retenu.

Macro count vise à calculer le support relatif d'une séquence en considérant que chaque bloc du cube de données doit avoir le même impact dans le support d'une séquence. Ainsi, la mesure des cellules d'un bloc B_r permettant à B_r de supporter la séquence recherchée ($m[B_r, s_{i_j}]$) est divisée par la valeur de mesure associée à B_r ($m[r, *, *, \ldots, *]$).

Définition 21 (Macro Count)
Soit une g-k-séquence $s=\langle s_1, s_2, \ldots, s_g \rangle$*, le support relatif de s dans un cube de données DB avec la technique macro count est égale à :*

$$\textit{Relative support}(s)=\frac{1}{|B_{DB,D_R}|} \times \sum_{\mathcal{B}_r \in B_{DB,D_R}} \bigoplus \bigotimes_{s_i \in s} \bigotimes_{s_{i_j} \in s_i} \frac{(m[B_r, s_{i_j}])}{m[(r,*,\ldots,*)]}$$

Comme pour la définition 20, il faut rechercher la meilleure combinaison de cellules (t-conorme) afin que le support de la séquence dans le bloc soit maximal. Pour chaque combinaison, il faut garantir l'antimonotonie du support, la valeur de mesure la plus faible (t-norme) des cellules de la combinaison est retenue.

Le calcul du support des items contenant une ou plusieurs valeurs jokers est assez simple. En effet, puisque nous considérons les mesures associées des cellules qui contiennent au plus un item de la séquence pour un bloc donné afin de calculer le support de la séquence. Ainsi, lorsqu'une valeur joker est présente dans un item de le séquence, il faut récupérer la mesure maximale parmi les cellules qui supportent cet item (exhiber la date où la mesure est maximale).

Mise en œuvre

Ces nouveaux types de comptage du support d'une séquence multidimensionnelle peuvent s'appliquer dans n'importe quelle approche d'extraction de motifs séquentiels multidimensionnels. En effet, ils respectent la propriété d'antimonotonie du support, ce qui permet aux algorithmes d'extraire l'ensemble

complet des séquences fréquentes. Toutefois, il est nécessaire d'adapter les algorithmes afin de permettre la recherche de la « meilleure » combinaison retrouvée dans la séquence de données d'un bloc. En effet, dans les autres approches, « la meilleure solution est la première découverte », dès que la séquence est trouvée dans le bloc, le support de la séquence est incrémenté et le calcul du support de la séquence se poursuit avec l'analyse du bloc suivant. On peut voir ces approches comme évoluant dans un contexte particulier où il n'y a pas de meilleure solution lorsqu'une séquence est supportée plusieurs fois dans un bloc, elles sont toutes équivalentes. Ainsi, chaque fois qu'une séquence est supportée par un bloc, on ajoute 1 au support de la séquence. Dans notre contexte, si un bloc supporte une séquence, on ajoute une valeur comprise dans l'intervalle $]0, 1]$ au support global de la séquence.

Nous avons adapté l'algorithme d'extraction de motifs séquentiels multidimensionnels fermés $CMSP\_Free$ [52]. Cet algorithme permet de parcourir efficacement l'espace de recherche en évitant d'extraire des connaissances redondantes. En effet, $CMSP\_Free$ extrait des motifs séquentiels multidimensionnels *fermés*. Un motif multidimensionnel est fermé ou clos s'il n'existe pas de séquence plus spécifique ayant le même support. Les motifs fermés offrent ainsi une représentation condensée des connaissances sans perte d'information, et introduisent des propriétés efficaces d'élagage de l'espace de recherche.

Discussion

Dans cette partie, nous avons défini la notion de motifs séquentiels multidimensionnels et avons étudié comment prendre en compte les dimensions qui constituent un cube de données et la dimension particulière qu'est la mesure (ou plus généralement n'importe quelle dimension numérique).

La découverte de motifs séquentiels multidimensionnels permet d'exhiber des motifs de la forme $< \{(NY, planchesurf, *)(NY, housse, *)\}\{(LA, combinaison, *)\} >$ (les clients ont acheté des planches en même temps que des housses à New York puis des combinaisons à Los Angeles) qu'aucune autre approche de la littérature n'était capable d'extraire auparavant. Ces règles sont extraites le long de plusieurs dimensions et plusieurs niveaux de hiérarchies. La gestion de ces hiérarchies rend la tâche d'extraction très complexe puisqu'il est impossible de considérer l'ensemble des items constitués des combinaisons de valeurs à tous les niveaux de granularité. L'approche proposée ici permet de ne considérer que les items fréquents les plus spécifiques. Pourtant il reste que nous n'avons pas encore établi de méthode permettant de retrouver les motifs séquentiels les plus spécifiques. Par exemple il se pourrait que (NY, vin) soit un item fréquent et qu'il soit donc considéré dans notre approche, mais qu'un motif séquentiel de niveau supérieur (par exemple $(NY, boisson)$ où *boisson* est un ancêtre de *vin*) permette de retrouver des motifs séquentiels plus longs.

Des travaux ont cependant été effectués pour exhiber des règles de différentes natures pour la gestion de la granularité. Par exemple nous avons défini des règles dites *convergentes* et *divergentes* selon qu'elles exhibent des items de plus en plus spécifiques (au sens des hiérarchies) ou au contraire *généraux* et avons proposé des algorithmes efficaces pour les extraire [51].

Nous avons également proposé deux approches différentes pour prendre en compte une dimension numérique (e.g. la mesure) dans l'extraction de motifs séquentiels multidimensionnels. Cette étude est rendue indispensable par la nature même des données d'entrepôts et des données multidimensionnelles que nous sommes amenés à traiter. La discrétisation de la mesure à l'aide de partions strictes ou floues permet de prendre en compte le potentiel informationnel de la mesure en l'intégrant dans les dimensions d'analyse. La définition de deux méthodes de comptage (macrocount et microcount) permet d'utiliser directement la mesure pour calculer le support des séquences de données multidimen-

sionnelles, ces deux propositions étant complémentaires.

La discrétisation de la mesure est très intéressante car elle permet de conserver l'information caractérisant la cellule (l'importance de la cellule). Il est donc nécessaire de s'appuyer sur les approches de la littérature qui permettent d'établir les meilleures partitions de manières automatiques tout en restant vigilant sur la complexité de l'extraction des motifs.

Réalisés dans le cadre de notre collaboration avec la Malaisie, et dans le cadre des thèses de Marc Plantevit et C. Fiot, ces travaux ont été appliqués sur des données réelles (collaboration EDF R&D) et ont montré qu'ils étaient pertinents. Ces approches sont difficiles, puisqu'elles font face à de gros volumes de données rendues complexes par les caractères multidimensionnel, hiérarchisé, et agrégé des données, comme nous l'avons décrit précédemment.

Au-delà de ces approches, il est apparu que de nombreux utilisateurs (et notamment les partenaires industriels de EDF R&D avec lesquels nous collaborions) souhaitaient non seulement extraire de telles tendances à partir de leurs entrepôts de données, mais aussi des comportements atypiques, afin de mettre en valeur les dysfonctionnements de leurs organisations. La prochaine partie est donc dédiée à l'extraction d'exceptions à partir d'entrepôts de données.

Quatrième partie

Fouille d'entrepôts de données et exceptions

Le traitement des exceptions est un problème difficile, ne serait-ce que parce que la notion d'exception est délicate à définir. Étudiées depuis longtemps, les exceptions ont en effet plusieurs facettes. Les premiers travaux sur la détection d'outliers proviennent du monde des statistiques où de nombreuses approches ont été développées comme les tests de discordances [29, 3]. En pratique, une règle 3σ est généralement adoptée. Cependant, il reste que ces approches ont été développées pour extraire des outliers dans un ensemble univarié où les éléments sont supposés suivre une distribution standard (Normale, Poisson) alors que l'essentiel des données issues du monde réel sont multivariées et qu'il est difficile de définir la distribution qui les régit.

De nombreux travaux proposent différentes méthodes pour détecter des outliers dans des données multivariées sans connaissance a priori de la distribution. Knorr et Ng donnent leur propre définition d'outlier basée sur la distance ([39, 40]). Dans le contexte OLAP, [39, 40, 58, 44] proposent d'extraire des cellules outliers.

Même si de nombreuses approches d'extraction d'outliers ont été proposées dans différents contextes, il n'existe pas d'approche permettant de caractériser des *séquences* outliers dans un contexte *multidimensionnel* (plusieurs dimensions et une mesure) où les données sont définies à différents niveaux d'agrégation.

Nous nous sommes donc intéressés à l'extraction d'exceptions dans le cadre de telles données complexes, et rapportons ici les travaux menés principalement dans le cadre de la thèse de M. Plantevit qui se sont inscrits dans le cadre de la collaboration avec EDF R&D, qui se sont intéressés à l'extraction de données atypiques à partir de cubes de données.

Thèmes abordés	Entrepôts de données, hiérarchie, motif séquentiel, exception, atypicité
Étudiant	M. Plantevit (thèse 2005-2008. co-encadrement M. Teisseire)
Collaborations	EDF R&D

Chapitre 8

Règles multidimensionnelles inattendues

Initialement proposées par Suzuki, les règles inattendues permettent de mettre en avant des règles très informatives pour les utilisateurs, telles que "*seat belt* and child → *danger*" alors que "*seat belt* → *safe*" [59, 5] qui vont à l'encontre de la tendance.

Dans des entrepôts de données, il est difficile d'appliquer directement les algorithmes de découverte de telles règles, la nature séquentielle et multidimensionnelle des données, ainsi que la présence de la mesure sont en effet autant d'obstacles à définir ce qu'est une règle ou partie de règle contradictoire par rapport à une autre.

Nous avons donc défini une nouvelle approche, dans le cadre de la thèse de M. Plantevit et de la collaboration avec EDF R&D. Cette approche permet de traiter des données issues de cubes de données. Considérons par exemple le cube DC du tableau 6.2 calculé à partir de transactions de clients contenant pour mesure M le nombre de ventes selon les dimensions suivantes : la date D de la vente (comprise entre 1 et 12), la ville C où a eu lieu la vente, l'âge A du client (discrétisé en intervalles, avec Y : young, M : middle-age et O : old), le loisir préféré CH des clients (e.g. surf, golf), le produit P (voir Figure 8.1).

De cette base de données, il est par exemple possible d'extraire que *Si le client est jeune et qu'il a récemment obtenu son permis de conduire ALORS il achètera une petite voiture de tourisme.* Cependant, il est très intéresant de noter que cette règle est contredite dans le cas où le client aime le surf. On a alors la règle suivante : *si le client est jeune, qu'il aime le surf et qu'il a récemment obtenu son permis de conduire ALORS il achètera un gros utilitaire.* Si cette dernière n'a pas un support très important, elle a pourtant une forte confiance et sa connaissance est très importante pour un décideur qui devrait par exemple programmer la prochaine campagne publicitaire associée aux voitures et ciblée

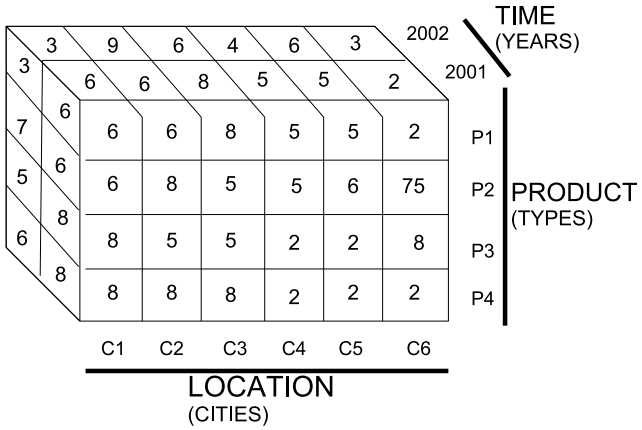

FIGURE 8.1 – Cube DC

vers les jeunes conducteurs.

Dans la suite de ce chapitre, nous détaillons donc les définitions et algorithmes associées à ces règles inattendues.

8.1 Règles multidimensionnelles inattendues : définitions

Notre but est d'extraire des règles souvent cachées car ayant un support faible. L'utilisation des jokers dans notre approche d'extraction de motifs séquentiels multidimensionnels est alors au centre de notre approche, en comparant le cas général (quand une étoile est présente dans un motif, e.g. $loisir = *$) qui mène à une conclusion (e.g. petite voiture), et un cas particulier (quand l'étoile est remplacée par l'une des valeurs de la dimension, e.g. $loisir = surf$) qui mène à une autre conclusion (e.g. gros utilitaire).

Instanciation

On nomme l'opération de remplacement d'une étoile par l'une des valeurs de la dimension *instanciation*.

On parle alors d'item instance d'un autre item, comme défini ci-dessous.

Définition 22 (Instance) *Soient C_1 et C_2 deux items multidimensionnels définis sur D_A, on dit que C_1 est une instance de C_2 si :*

- *il existe* $D_{A'} \neq \emptyset$ *t.q.* $D_{A'} \subseteq D_A$ *et pour* $\forall d_j \in D_{A'}$ *on a* $C_2[d_j] = *$ *et* $C_2[d_j] \neq *$,
- *et si* $\forall d_j \in D_A - D_{A'}$ *on a* $C_1[d_j] = C_2[d_j]$

Par exemple, $C_1 = (a, b, c)$ est une instance de $C_2 = (a, *, c)$.

Nous pouvons alors appliquer ces définitions dans le cadre des itemsets. Afin de permettre de ne conserver qu'une seule partie de l'itemset, nous considérons la notion de pseudo-instance et non d'instance. Par exemple, on considèrera $\{(a, b, c)\}$ comme une pseudo-instance de $\{(a, *, c), (*, b, b)\}$ tandis que $\{(a, b, c), (*, b, b)\}$ sera considéré comme une instance de $\{(a, *, c), (*, b, b)\}$.

Définition 23 (Pseudo-instance d'un itemset)
Soient $i = \{e_1, e_2, \ldots, e_m\}$ *et* $i' = \{e'_1, e'_2, \ldots, e'_{m'}\}$ *deux itemsets, on dit que* i *est une instance de* i' *s'il existe des entiers* $1 \leq k_1 \leq k_2 \leq \ldots \leq k_m \leq m'$ *tels que* $\forall e_j \in i$, e_j *est une instance de* e'_{k_j}.

De même, il est possible de définir la pseudo-instanciation dans le cas d'une séquence.

Définition 24 (Pseudo-instance d'une séquence)
Soient $s = \langle i_1, i_2, \ldots i_m \rangle$ *et* $s' = \langle i'_1, i'_2, \ldots i'_{m'} \rangle$ *deux séquences telles que* $m \leq m'$, *on dit que* s *est une pseudo-instance de* s' *s'il existe des entiers* $1 \leq k_1 \leq k_2 \leq \ldots \leq k_m \leq m'$ *tels que* $\forall i_j \in s$, i_j *est une pseudo-instance de* i'_{k_j}.

Par exemple, $\langle\{(a, b, c)(a, *, d)\}\{(a, b, *)(b, *, *)\}\rangle$ est une pseudo-instance de $\langle\{(a, b, c)(a, *, d), (d, e, f)\}\{(a, *, *)(b, *, *)\}\rangle$.

La découverte de règles multidimensionnelles séquentielles inattendues revient alors à rechercher les règles multidimensionnelles contenant au moins une étoile dont le remplacement de cette étoile change la conclusion de la règle. On nomme ce remplacement *instanciation*.

Définition 25 (instanciation) *Soient* s' *et* s *deux séquences multidimensionnelles telles que* s *est une pseudo-instance de* s', *la fonction* $\iota(s', s)$ *permet de construire l'ensemble des séquences résultant d'une substitution d'au moins un item* e'_i *dans* s' *avec une instance de* e'_i *dans* s.

ι : *sequence* $\times$ *sequence* $\rightarrow$ *ensemble de sequences*
$\iota(s', s) \mapsto \{s''$ *t.q.* $:s''$*est une instance de* $s'\exists$*des items* $e''_i \in s'', e'_i \in s'$ *et* $e_i \in s$ *tels que*e_i*est une instance de* e'_i *et*

Outre la notion d'instanciation, la recherche de règles séquentielles inattendues nécessite la définition de la notion de règle séquentielle (si jeune conducteur *ALORS* petite voiture).

Règle séquentielle et règle séquentielle inattendue

Très peu présente dans la littérature où les motifs séquentiels (extension des itemsets) sont étudiés en fonction de leur seul support, cette notion peut revêtir plusieurs sens. Dans ce chapitre, nous ne discutons pas cet aspect, qui l'a été dans d'autres travaux (voir thèse de H. Li).

Définition 26 (Règle séquentielle) *Soit* $\alpha = \langle i_1, i_2, \ldots i_k, i_{k+1}, \ldots, i_n \rangle$ *une séquence multidimensionnelle dans laquelle chaque* i_j *représente un itemset multidimensionnel. Une* règle séquentielle R *est de la forme :*

$$R : \langle i_1, i_2, \ldots i_k \rangle \rightarrow \langle i_{k+1}, \ldots, i_n \rangle$$

La qualité de cette règle est déterminée par son support *et sa* confidence. *Le support de* R *est égal au support de* α *:* $support(R) = support(\langle i_1, i_2, \ldots i_k, i_{k+1}, \ldots, i_n \rangle)$. *La confiance de* R *est égale à :*

$$Conf(R) = \frac{support(\langle i_1, i_2, \ldots i_k, i_{k+1}, \ldots, i_n \rangle)}{support(\langle i_1, i_2, \ldots i_k \rangle)}$$

Dans le contexte de la recherche de règles séquentielles inattendues, on considère une règle CR dite *commune* qui est fréquente et dont la confiance est au-delà du seuil fixé par l'utilisateur, telle que :

$$CR : P \rightarrow Q$$

où P et Q sont des séquences multidimensionnelles.

Une règle UR (non fréquente mais de forte confiance) est alors dite inattendue par rapport à CR si :

$$UR : P_{specialized} \rightarrow Q'$$

avec Q' différent de Q et $P_{specialized}$ une *instanciation* de P.

Il est donc nécessaire de définir en détail ce que signifie que Q est différent de Q'.

Règles séquentielles et différence

Afin de découvrir des règles inattendues par rapport à des règles communes, il est nécessaire de trouver une règle ayant une conclusion différente de celle de la règle commune. Or la conclusion de règles multidimensionnelles séquentielles est elle-même une séquence multidimensionnelle. Le problème revient donc à déterminer comment décrire qu'une séquence multidimensionnelle est différente d'une autre.

Définition 27 (Différence de séquences multidimensionnelles) *Soient* $s = \langle i_1, i_2, \ldots, i_l \rangle$ *et* $s' = \langle i'_1, i'_2, \ldots, i'_{l'} \rangle$ *deux séquences,* s *et* s' *sont dites différentes* $(s \neq s')$ *si* $s \not\subseteq s'$ *et* $s' \not\subseteq s$.

Notons que deux séquences s et s' sont alors dites *non comparables au sens de la différence* si s est plus spécifique ou plus générale que s'.

Ces définitions étant posées, il nous faut maintenant définir comment extraire les règles séquentielles inattendues. Nous décrivons ce processus ci-dessous.

8.2 Processus d'extraction

L'extraction de telles règles inattendues nécessite la définition de seuils auxquels seront effectuées les recherches. En effet, une règle UR est inattendue par rapport à une règle commune CR. Il faut donc déterminer les seuils de support et confiance minimaux pour ces deux types de règles. Les seuils de confiance sont par défaut égaux. Les seuils de support doivent au contraire être différents, puisque les règles communes sont par essence très fréquentes et les règles inattendues très exceptionnelles. Ce caractère exceptionnel sera par ailleurs garanti par un seuil de support maximal dans le cas des règles inattendues. Nous notons :

- $minCR$ le seuil de support minimal pour les règles communes,
- $maxUR$ le seuil de support maximal pour les règles inattendues,
- $minUR$ le seuil de support minimal pour les règles inattendues,
- $minConf$ le seuil de confiance minimal pour les toutes règles, qu'elles soient communes ou inattendues.

La Figure 8.2 illustre l'utilisation de ces seuils pour ce qui concerne le support.

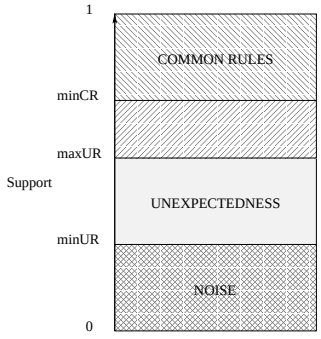

FIGURE 8.2 – Seuils de Support

Une règle commune CR est donc une règle à forte confiance et fort support :

$$CR : s_\alpha \rightarrow s_\beta \; t.q. \; supp(CR) > minCR \; et \; conf(RC) > minConf \qquad (8.1)$$

Nous considérons alors les instances de la prémisse de CR ayant un support supérieur à $minCR$:

$$s' \; t.q. \; supp(s') > minCR \; et \; s' \; est \; une \; instance \; de \; s_\alpha \tag{8.2}$$

Une règle séquentielle multidimensionnelle UR inattendue par rapport à une règle commune $CR : s_\alpha \rightarrow s_\beta$ est une règle dont la prémisse est une instanciation de s_α (prémisse de CR) et dont la conclusion est différente de s_β (conclusion de CR) :

$$UR : \iota(s_\alpha, s') \rightarrow s_c \; t.q. \; minUR \leq supp(UR) \leq maxUR \; et \; s_c \neq s_\beta \tag{8.3}$$

Il faut cependant vérifier qu'il n'existe pas de règle commune VR contredisant le caractère exceptionnel de UR

$$VR : s_\alpha \rightarrow s_c \; t.q. \; conf(VR) < minConf \; et/ou \; supp(RV) < maxUR \tag{8.4}$$

Afin de découvrir l'ensemble des règles communes et leurs exceptions (règles inattendues), nous devons trouver l'ensemble S vérifiant les propriétés des équations (8.2), (8.3) et (8.4).

L'algorithme 4 décrit ce processus découpé en trois étapes. Nous cherchons d'abord les *séquences* multidimensionnelles (fonction $getFreqSet()$), stockées dans une structure arborescente (arbre prefixé $freqTree$). Puis les *règles* multidimensionnelles communes sont établies (fonction $getCR()$) et stockées, avant de rechercher les règles inattendues en considérant les instanciations possibles des prémisses de règles communes.

Data : $DC, D_A, D_T, D_R, minCR, maxUR, minUR, minConf$
Result : The set URS of unexpected multidimensional sequential rules
begin
  $FreqTree \leftarrow getFreqSeq(DC, D_A, D_R, D_T, minUR)$
  $CRS \leftarrow getCR(FreqTree, minCR, minConf)$
  $URS \leftarrow \emptyset$
  foreach *rule* $r : p \rightarrow q \in freqTree$ *s.t.* $minUR \leq supp(r) \leq maxUR$ *and* $conf(r) \leq minConf$ **do**
    if $\exists$ *premise* $p' \in CRS$ *s.t.* p *is an instance of* p' **then**
      if $\exists seq \; x$ *s.t.* $\iota(p', x) \rightarrow p$ *and* $supp(x) \geq minCR$ **then**
        if $\nexists p' \rightarrow q \in CRS$ **then**
          $URS \leftarrow URS \cup \{r\}$
end

Algorithme 4: Extraction de règles séquentielles multidimensionnelles inattendues

Dans ce chapitre, nous avons vu comment générer des règles multidimensionnelles inattendues. Ce processus permet d'exhiber un ensemble de règles

intéressantes pour l'utilisateur. Une autre stratégie pour pointer les données atypiques dans les entrepôts consiste à fournir une aide à la navigation, méthode présentée dans le chapitre suivant.

Chapitre 9

Données inattendues et entrepôts de données : une aide à la navigation

Les utilisateurs étant souvent démunis face à la complexité de leurs données d'entrepôts de données, il est indispensable de les guider vers les zones les plus intéressantes pour eux, au sens où elles contiennent des informations nouvelles, ou des informations qu'ils recherchent. Dans la partie II, nous avons vu comment exploiter la notion de *bloc* pour permettre la distinction rapide et automatique de zones homogènes. Ces blocs pouvaient alors servir à la détection de comportements atypiques quand une cellule était très différente des autres cellules d'un même bloc. Cependant ces techniques ne permettaient pas la prise en compte de la dimension *temporelle* des données (données historisées). Nous nous sommes donc intéressés à un autre moyen de détecter des comportements atypiques, en proposant un nouveau mode de navigation fondé sur la recherche de *séquences outliers*. Nous travaillons à partir d'une dimension donnée par l'utilisateur. Pour un niveau donné, cette dimension comprend donc un ensemble de valeurs (par exemple $\{Paris, Montpellier, Marseille\}$) pour le niveau $Ville$ de la dimension $Lieu$), et à chacune de ces valeurs correspond un sous-cube contenant l'ensemble des cellules associées à la ville Paris. Ces sous-cubes peuvent alors être vus comme une séquence de données, si l'on imbrique la dimension temporelle. Nous proposons alors d'identifier les n séquences qui diffèrent le plus des autres, correspondant alors aux n villes présentant des résultats atypiques. Ce processus est ensuite réitéré aux niveaux inférieurs, afin que l'utilisateur découvre les raisons pour lesquelles ces villes n'ont pas eu le même comportement que les autres.

La définition d'atypicité de séquence est donc au centre de ce processus. Or il est difficile de comparer des séquences, tant sur le point sémantique que sur le point algorithmique. La distance la plus connue entre deux séquences est l'*edit distance*. L'edit distance correspond au nombre d'opérations d'édition (insertion,

suppression, remplacement) nécessaires pour transformer une séquence en une autre. Dans le contexte des bases de données multidimensionnelles historisées, il faut donc définir ce que sont ces opérations. En effet, la distance d'édition de deux séquences peut être très faible (1 opération d'insertion ou de suppression) alors que les séquences sont en total décalage. Nous avons donc également considéré d'autres distances très classiquement utilisées : la distance euclidienne, la distance de Manhattan et une mesure de similarité basée sur le cosinus. Nous présentons ci-dessous ces distances puis les algorithmes associés.

9.1 Comparaison de séquence par rapport à un ensemble de séquences

Pour pouvoir établir des distances ou des mesures de similarité entre deux séquences, nous introduisons la notion de cellules comparables selon un ensemble de dimensions.

Définition 28 (Cellules comparables) *Deux cellules* $c_1 = \langle (d_1, \ldots, d_n), \mu \rangle$ *et* $c_2 = \langle (d'_1, \ldots, d'_n), \mu' \rangle$ *sont comparables sur un ensemble de dimensions* D *si et seulement si* $c_1.D = c_2.D$.

Notons que nous nous situons dans un contexte de cube de données dense et nous supposons qu'il existe très peu de cellules vides [1]. Pour calculer la distance entre deux blocs, nous regroupons les cellules comparables sur D_A pour construire des vecteurs de mesure. L'algorithme 5 décrit comment deux blocs sont transformés en deux vecteurs contenant les valeurs de mesures des cellules comparables.

Data : b_1 et b_2 blocs, D un ensemble de dimensions
Result : Construction de deux vecteurs v_1 et v_2
begin
  $v_1 \leftarrow ()$
  $v_2 \leftarrow ()$
  foreach *cellule* $c_i \in b_1$ **do**
    if $\exists c_j \in b_2 \mid c_i$ *et* c_j *sont comparables sur* D **then**
      $v_1.add(mesure(c_i))$
      $v_2.add(mesure(c_j))$
  return v_1, v_2
end

Algorithme 5: (TransBlocVec) Construction des vecteurs représentant les blocs

1. Cette hypothèse émane du fait que cette proposition a été effectuée dans le cadre de la collaboration avec EDF R&D qui disposait de cubes très denses. Cependant, notre méthode peut être adaptée aux cubes creux, comme nous le verrons ultérieurement

La représentation vectorielle de deux blocs permet d'appliquer les mesures de distance et de similarité telles que la distance euclidienne et le cosinus. Le calcul de la distance entre deux blocs nous permet de calculer la distance entre deux séquences :

Définition 29 (Distance entre 2 séquences) *Soient* $s_1 = \langle b_1, b_2, \ldots, b_k \rangle$ *et* $s_2 = \langle b'_1, b'_2, \ldots, b'_k \rangle$ *deux séquences multidimensionnelles, dist une mesure de distance et Op un opérateur d'agrégation. La distance entre* s_1 *et* s_2 *se définit de la façon suivante :*

$$d(s_1, s_2) = Op(dist(b_j, b'_j)) \text{ pour } j = 1 \ldots k$$

Nous utilisons ici les distances de Manhattan et euclidienne définies ci-dessous. Nous utilisons aussi la mesure de similarité basée sur le cosinus.

Distance de Manhattan : $Man(v_1, v_2) = \sum_{k=0}^{m} |v_{1_k} - v_{2_k}|$

Distance euclidienne : $Euclid(v_1, v_2) = \sqrt{\sum_{k=0}^{m} (v_{1_k} - v_{2_k})^2}$

Cosinus : $cos(v_1, v_2) = \frac{v_1 . v_2}{||v_1|| ||v_2||} = \frac{\sum_{k=0}^{m} (v_{1_k} v_{2_k})}{\sqrt{\sum_{k=0}^{m} v_{1_k}^2 \sum_{k=0}^{m} v_{2_k}^2}}$

La définition 29 est suffisament générique pour appliquer n'importe quel opérateur d'agrégation pour calculer une distance entre deux séquences. La distance entre deux séquences peut être, par exemple, la moyenne des distances entre chaque bloc, la médiane, le min ou le max.

Pour déterminer si une séquence est un outlier, il est nécessaire de connaître sa similarité par rapport à toutes les autres séquences de la base. Nous établissons donc une matrice de distance représentant les distance entre chaque séquence de la base.

Sequence_Id	1	2	...	l
1	1	$sim(1,2)$	...	$sim(1,l)$
2	*	1	...	$sim(2,l)$
...	*	*	1	...
l	*	*	*	1

FIGURE 9.1 – Comparaison d'une séquence par rapport aux autres : Matrice de similarité

Sequence_Id	1	2	...	l
1	0	$d(1,2)$	...	$d(1,l)$
2	*	0	...	$d2,l)$
...	*	*	0	...
l	*	*	*	0

FIGURE 9.2 – Comparaison d'une séquence par rapport aux autres : Matrice de distance

Nous définissons la distance (resp. la similarité) d'une séquence par rapport à un ensemble de séquences, comme la moyenne des distances (resp. similarités) entre la séquence et les autres séquences. La distance d'une séquence s_α par rapport à un ensemble de séquences S est couramment définie dans la littérature de la façon suivante :

$$d(s_\alpha, S) = \frac{\Sigma_{i=1}^{i<\alpha} d(s_\alpha, s_i) + \Sigma_{j=\alpha+1}^{|s|} d(s_j, s_\alpha)}{|S| - 1}$$

Le calcul de la distance d'une séquence par rapport à un ensemble de séquences est primordial pour savoir si une séquence est un outlier ou non. Il est possible de définir un outlier par rapport à un seuil de distance fixé a priori par l'utlisateur. Définir ce seuil est très fastidieux et dépend fortement des séquences examinées. Il est, en conséquent, plus aisé pour l'utilisateur de définir un entier k qui correspond au nombre de séquences outliers qu'il souhaite étudier.

Définition 30 (top n outlier) *Une séquence s_α est un top n outlier ($n \geq 1$) s'il n'existe pas plus de $n-1$ séquences telles que $d(s_i, C_{D_R}) > d(s_\alpha, C_{D_R})$*

Une fois ces définitions posées, il s'agit de définir des algorithmes permettant d'extraire efficacement les top n outliers.

9.2 Algorithmes

Il s'agit ici de fournir les méthodes et outils à l'utilisateur pour qu'il soit capable, face à une séquence identifiée comme un outlier à un haut niveau de granularité, d'étudier plus en détail les sous-données associées à un niveau plus fin. Cette méthodologie permet de le guider dans sa recherche afin qu'il cible le plus directement possible les données susceptibles de l'intéresser.

Dans cette section, on notera :
- $S_{v_{R_i}}$ la séquence identifiée par $D_R = v_{R_i}$;
- $C_{D_R = v_R}$ le sous-cube relatif à v_R.

Chaque valeur v_R sur D_R identifie une séquence. Ainsi si une séquence est un top n outlier, alors ce sont les actions sur v_R qui sont anormales par rapport aux actions relatives aux autres valeurs sur D_R. Comme v_R n'est pas le niveau le plus fin dans la hiérarchie, il est toujours possible de se demander pourquoi v_R est outlier. Nous pouvons donc nous placer dans le sous-cube identifié par v_R et rechercher les top n outliers.

L'algorithme 6 permet d'extraire les top n outliers à un niveau d'agrégation donnée. Pour chaque séquence top n outlier identifiée par sa valeur v_R sur D_R, le processus est réitéré sur les sous-cubes identifiés par chaque valeur v_R jusqu'à arrivé au niveau d'agrégation le plus fin.

Data : C_{v_R} Cube de données, n entier, L ensemble, *dist* une mesure de distance
Result : Séquences outliers à chaque niveau de granularité
begin
  Calculer la matrice de distance
  foreach *séquence* $S_{v_{R_i}} \in C_{v_R}$ *top* n *outlier* **do**
    $add(v_{R_i}, L)$
    if v_{R_i} *is not leaf* **then**
      $RechTopn(C_{DrillDown(v_{R_i})}, n, L, dist)$
  return v_1, v_2
end

Algorithme 6: RechTopn

Pour $k = 1$, cet algorithme permet de proposer à l'utilisateur un chemin de navigation dans le cube afin d'identifier des séquences anormales par rapport à l'ensemble des données. Pour $k = 1$, le chemin regroupe les valeurs v_R dont les séquences associées sont des top 1 outliers à un niveau donné. Ce chemin part d'un niveau d'agrégation élevé et se termine au niveau d'agrégation le plus fin. Grâce à ce chemin, l'utilisateur peut directement aller sur la valeur v_R la plus fine, ou avancer pas à pas.

Pour $k \geq 1$, l'algorithme propose un arbre de navigation. En effet, il n'existe plus un seul chemin, mais plusieurs chemins. L'utilisateur peut ainsi visualiser les séquences anormales par l'intermédiaire de cet arbre. Il peut directement situer au niveau d'agrégation le plus fin (les feuilles), ou naviguer à travers les différents nœuds de l'arbre.

Notons qu'une séquence peut être un top n outlier pour plusieurs raisons :
- Une séquence à un niveau inférieur est sensiblement différente des autres. La séquence a ainsi une importance dans le fait que la séquence agrégée du niveau supérieur est outlier. Dans ce cas là, l'algorithme 6 permet d'extraire ces différents outliers pour chaque niveau.
- Une grande partie des séquences du niveau inférieur sont sensiblement différentes du comportement général du niveau supérieur des séquences non outliers. Ainsi, une séquence qui suit le comportement général peut être considérée comme un top n outlier. Nous proposons donc de calculer la distance de cette séquence avec les autres séquences non outliers afin de voir si cette séquence suit le comportement général (bien le seul). Comme nous ne nous situons pas au même niveau d'agrégation, il est nécessaire de normaliser les séquences afin de calculer la distance entre deux séquences de niveaux d'agrégation différents.

Discussion

Dans cette partie, nous avons montré l'íntérêt, mais aussi la complexité, de définir des méthodes de recherche d'exceptions. Appliqués à des données réelles issues de notre collaboration avec EDF R&D, nos algorithmes ont prouvé leur efficacité et leur pertinence. Deux approches principales ont été proposées.

La première approche permet de déceler, niveau de hiérarchie par niveau de hiérarchie, les valeurs correspondant à des données qui ne sont pas similaires au comportement général. Elle s'inscrit dans la continuité de notre approche développée dans la thèse basée sur la découverte de cellules anormalement vides dans le contexte des entrepôts de données et de notre approche de recherche de blocs permettant une approche accompagnant l'utilisateur dans sa navigation au sein de cubes de données. Pour répondre au mieux au contexte complexe des entrepôts de données stockant des données historiées, cette première approche considère la dimension temporelle afin de prendre en compte des séquences de données.

La deuxième approche quant à elle est liée à la découverte de règles inattendues, contredisant la connaissance exprimée par les règles de type *tendance* (fort support). Il est alors possible de découvrir que les jeunes qui viennent d'obtenir leur permis de conduire achètent plutôt des petites voitures de tourisme alors que ces mêmes jeunes achèteront plutôt des utilitaires s'ils aiment le surf.

Dans les deux cas, ces approches doivent faire face à une difficulté majeure qui est la définition de la similarité et de la différence dans le cas de données multidimensionnelles séquentielles. En effet, comment dire qu'une séquence est différente d'une autre ? L'ajout d'un item ou d'un itemset suffit-il à changer le sens d'une séquence ? Faut-il plutôt qu'un item soit remplacé par un autre pour que la différence soit avérée ? Telles sont les questions auxquelles nous tentons d'apporter des premières réponses qui ne sont forcément que partielles puisqu'elles dépendent du contexte sémantique.

Notons que ce problème a également été étudié dans le cadre du stage de M2R de H. Saneifar [57] et qu'il également largement considéré dans le cadre de la thèse d'H. Li.

Une fois la définition posée, il faut alors mettre en œuvre des algorithmes les plus efficaces possibles afin de faire face à la complexité de l'espace de recherche.

En effet, si les propriétés de coupure sont facilement exploitables dans le cas de la recherche de tendances, cela n'est plus le cas pour la recherche d'exceptions, qui devient alors sujette au bruit.

Dans la suite de ce mémoire, nous nous intéressons à un autre type de règles tout aussi difficile à définir et à extraire efficacement : les règles graduelles.

Cinquième partie

Règles et motifs graduels

On appelle communément règle graduelle une règle de la forme *"Plus le salaire est élevé, moins le nombre de crédits à la consommation est élevé"*.

Si les règles graduelles ont été utilisées et décrites depuis de nombreuses années, notamment pour faire fonctionner des systèmes à base de règles (e.g. commande floue), l'extraction *automatique* de règles graduelles est un problème assez nouveau dans la littérature. Rendu difficile par l'espace de recherche à parcourir, le problème de l'extraction automatique peut pourtant être vu comme un problème trouvant des solutions avec les algorithmes par niveaux issus du domaine de la fouille de données, ce que nous nous proposons d'étudier.

Dans nos travaux, nous nous intéressons à diverses formes de gradualité et nous focalisons sur la recherche d'algorithmes performants :
- l'extraction de règles graduelles à l'aide d'une heuristique,
- l'extraction exhaustive de règles graduelles à l'aide d'une représentation binaire des données,
- l'extraction de règles graduelles avec prise en compte de la temporalité.

Ces travaux sont menés dans le cadre de la thèse de L. Di Jorio et du post-doctorat de C. Fiot. Ils sont notamment appliqués dans le cadre de notre collaboration avec l'IRCM (données biologiques liées au cancer du sein), dans le cadre de notre collaboration avec l'INSERM pour une application à la base de données PAQUID décrivant le suivi de personnes âgées au fil du temps, et dans le cadre dune collaboration avec l'Université Montpellier 3 (projet PEPS financé par le CNRS) pour l'application à des données issues de tests psychologiques pour l'étude du vieillissement de la maladie d'Alzheimer.

Thèmes abordés	données numériques, règle graduelle, motif séquentiel graduel
Étudiante	L. Di Jorio (thèse 2007-2010. co-encadrement M. Teisseire)
Collaborations	IRCM (Institut de Recherche sur le Cancer de Montpellier) INSERM (Alzheimer, étude PAQUID) Université Montpellier 3 (Alzheimer)

Nous rapportons ici les travaux menés pour l'extraction efficace de règles graduelles avec deux approches principales, la première s'appuyant sur une heuristique et la seconde proposant une extraction complète.

Chapitre 10

Extraction de règles graduelles à l'aide d'une heuristique

10.1 Définitions préliminaires

Les règles d'association graduelles décrivent des co-variations entre attributs. Deux types de variations peuvent être considérées : soit la valeur d'un attribut augmente d'un objet à l'autre, soit elle diminue.

Définition 31 *(item graduel) Soit $\mathcal{I}$ un ensemble d'items, $i \in \mathcal{I}$ un item et $* \in \{\geq, \leq\}$ un opérateur de comparaison. Un item graduel i^* est défini comme un item i associé à un opérateur $*$.*

On note

$$c(*) = \begin{cases} \geq & si \quad * = \leq \\ \leq & si \quad * = \geq \end{cases}$$

Par exemple, à partir du tableau 10.1, six items graduels peuvent être considérés : $\{A^{\geq}, A^{\leq}, S^{\geq}, S^{\leq}, C^{\geq}, C^{\leq}\}$. Le premier item, l'âge, nous amène à considérer deux items graduels : $\{A^{\geq}, A^{\leq}\}$ signifiant respectivement "*l'âge augmente*" et "*l'âge diminue*". Un itemset graduel est alors défini par :

Définition 32 *(itemset graduel) Un itemset graduel $(i_1^{*_1}...i_n^{*_n})$ est un ensemble non vide d'items graduels. Un k-itemset est un itemset graduel contenant k items graduels.*

Personne	Âge (A)	Salaire (S)	Crédit (C)
p_1	22	1200	4
p_2	28	1850	2
p_3	24	1200	3
p_4	35	2200	2
p_5	38	2000	0
p_6	44	3400	1
p_7	52	3400	5
p_8	41	5000	5

TABLE 10.1 – Base exemple $\mathcal{BD}$

Par exemple, $S_1 = (A^{\geq}S^{\geq}C^{\leq})$ est un itemset graduel du tableau 10.1. Cette règle est obtenue par comparaison entre les propriétés de chaque objet : nous avons comparé les variations entre les attributs d'un objet à l'autre. Habituellement, l'intérêt d'une règle est mesuré par son support, qui reflète la proportion d'objets de la base contenant cette règle. Cette notion est différente dans le cas de la gradualité, car il ne s'agit plus de comptabiliser un nombre d'objets supportant l'itemset, mais le nombre d'objets respectant la variation décrite par l'itemset. A partir de la base de la table 10.1, nous pouvons donner deux ensembles d'objets respectant S_1 : $\{p_1p_3p_2p_4p_6\}$ et $\{p_1p_3p_2p_5\}$. La fréquence d'un itemset graduel est calculée à partir de l'ensemble le plus représentatif, c'est-à-dire l'ensemble ayant le plus d'éléments :

Définition 33 *Soit $s = (i_1^{*_1}...i_n^{*_n})$ un itemset graduel et G_s l'ensemble des objets respectant s. La fréquence (ou support) de s est donnée par $Freq(s) = \frac{max(|G_s^i|)}{|\mathcal{O}|}$ où $G_s^i \subseteq G_s$ et $\mathcal{O}$ est l'ensemble des objets décrivant la base de données.*

Nous obtenons $Freq(S_1) = \frac{5}{8} = 0.65$, ce qui signifie que S_1 est supporté par 62.5% de toutes les personnes.

Le problème que nous posons est alors de retrouver le plus efficacement possible ce type de règles graduelles. En effet, considérer toutes les comparaisons possibles entre objets est très coûteux. Le support de la règle correspond au nombre de données qu'il est possible d'ordonner pour qu'elles respectent la règle graduelle. Or il existe plusieurs manières d'ordonner les n-uplets de la base de données, et il n'est pas raisonnable d'explorer tous ces ordonnancements. Nous proposons donc ici d'utiliser une heuristique afin de calculer le support et de se positionner de manière itérative dans la sous-base pertinente (celle sur laquelle un ordonnancement respectant la règle peut être trouvé). Un algorithme par niveaux peut alors être couplé à cette heuristique afin de construire les règles graduelles contenant un grand nombre d'attributs dont les variations sont corrélées. Nous avons en effet montré que les propriétés classiques d'anti-monotonie sont conservées, ce qui nous permet d'utiliser un algorithme *a la* APriori.

Nous notons $G^{\mathcal{D}}$ la sous-base de données supportant une règle graduelle rg. Le but est alors de retrouver la sous-base $G^{\mathcal{D}}$ de taille maximale afin de calculer

le support de rg. Pour ce faire, nous introduisons la notion d'*ensemble de conflit.*

Définition 34 *On considère $\mathcal{O}$ un ensemble de n-uplets de la base de données de $\mathcal{DB}$ ordonnés sur les attributs $a_1 \dots a_n$ selon les relations d'ordre $*_1, \dots, *_n$. Pour chaque n-uplet $o_i \in \mathcal{O}$, l'ensemble de conflit associé, noté $\mathcal{C}_i$ est tel que $\exists a_k$ tel que $\forall o_j \in \mathcal{C}_i, t_{o_i}[a_k] c(*_i) t_{o_j}[a_k]$.*

Nous cherchons par exemple à construire un ordonnancement $G^{\mathcal{D}}$ pour la règle graduelle (A^+S^+). Ordonner la base du tableau **??** pour A^+ et pour S^+ revient à considérer la base illustrée par le tableau 10.2. Nous avons calculé, pour tous les objets, les ensembles de conflit correspondants (voir troisième colonne). Par exemple si l'on voulait conserver o_8 alors il faudrait supprimer o_6 et o_7, et de manière symétrique si l'on voulait conserver o_6 et o_7 alors il faudrait supprimer o_8.

Identifiant	**A** (âge)	**S** (salaire)	$\mathcal{C}_i$
o_1	22	1200	$\emptyset$
o_3	24	1200	$\emptyset$
o_2	28	1850	$\emptyset$
o_4	35	2200	$\{o_5\}$
o_5	38	2000	$\{o_4\}$
o_8	41	5000	$\{o_6, o_7\}$
o_6	44	3400	$\{o_8\}$
o_7	52	3400	$\{o_8\}$

TABLE 10.2 – Ordonnancement pour A^+ et S^+

Objet	**A**	**S**	$\mathcal{O}_{\mid i}$
o_1	22	1200	$\emptyset$
o_3	24	1200	$\emptyset$
o_2	28	1850	$\emptyset$
o_4	35	2200	$\{o_5\}$
o_5	38	2000	$\{o_4\}$
~~o_8~~	~~41~~	~~5000~~	~~$\{o_6, o_7\}$~~
o_6	44	3400	~~$\{o_8\}$~~ $= \emptyset$
o_7	52	3400	~~$\{o_8\}$~~ $= \emptyset$

TABLE 10.3 – Operation t_{01}

Ces ensembles de conflits sont à la base de l'heuristique que nous proposons.

10.2 Heuristique

L'heuristique alors utilisée est la suivante : le n-uplet dont l'ensemble de conflits est le plus grand est supprimé jusqu'à ce qu'aucun conflit ne subsiste. Cette heuristique est dite *gloutonne* puisqu'elle maximise le résultat de manière

locale, à chacune des étapes. Cependant, il peut se produire des cas pour lesquels choisir de supprimer plus de n-uplets à une étape permet de retrouver un support plus grand (donc plus proche de la vraie valeur de support) à une étape ultérieure. Si plusieurs ensembles de conflit ont la même cardinalité, un choix aléatoire est opéré.

Dans notre exemple précédent, o_8 a l'ensemble de conflit maximal. On le supprime donc, conduisant à $G^{\mathcal{D}} \leftarrow f_{emp}(\mathcal{O} \setminus o_8) \equiv G^{\mathcal{D}} \leftarrow \{o_1, o_3, o_2, o_6, o_7\}$ (voir tableau 10.3). Notons que la suppression de o_8 produit la mise à jour des ensembles de conflits de o_6 et o_7. Les ensembles de conflits sont alors vides.

Les n-uplets o_4 et o_5 sont alors considérés et o_4 est supprimé, conduisant à $G^{\mathcal{D}} = \{o_1, o_3, o_2, o_5, o_6, o_7\}$. Le support est alors calculé par $support = \frac{|G^{\mathcal{D}}|}{|\mathcal{D}|} = \frac{6}{8} = 0.75$

Comme dit précédemment, un algorithme par niveau est appliqué, ce qui implique que les règles graduelles de taille k fréquentes soient combinées pour générer des règles graduelles candidates de taille $k+1$ (voir algorithme 7).

Data : Un itemset graduel $s = (i_1^{*_1} ... i_n^{*_n})$,
Ensemble de n-uplets $\mathcal{O}$ ordonnés selon $n-1$ items,
Ensembles de conflits $\mathcal{C}^n$ et $\mathcal{C}^{n-1}$
Result : Base représentative $G^{\mathcal{D}}$ for s

$G^{\mathcal{D}} \longleftarrow \emptyset$
while $\mathcal{O} \neq \emptyset$ **do**
  $o = f_{max}(\mathcal{C}^{i_n}, \mathcal{C}^{i_{n-1}})$
  $\mathcal{O} \longleftarrow \mathcal{O} \setminus \{o\}$
  foreach $o_j \in \mathcal{O}$ **do**
    $f_{cnf}(o_j, \mathcal{C}^{i_n}) \longleftarrow f_{cnf}(o_i, \mathcal{C}^{i_n}) \setminus \{o\}$
    $f_{cnf}(o_j, \mathcal{C}_{i^{n-1}}) \longleftarrow f_{cnf}(o_i, \mathcal{C}^{i_{n-1}}) \setminus \{o\}$
    if $f_{cnf}(o_j, \mathcal{C}_{i_n}) = \emptyset$ *and* $f_{cnf}(o_j, \mathcal{C}^{i_{n-1}}) = \emptyset$ **then**
      $G^{\mathcal{D}} \longleftarrow G^{\mathcal{D}} + \{o_j\}$
      $\mathcal{O} \longleftarrow \mathcal{O} \setminus o_j$
    end
  end
end
return $\mathcal{O}_R$

Algorithme 7: n-SupportCount

Nous avons montré que des propriétés sont valides et permettent d'optimiser notre méthode. En premier lieu, il convient de rappeler que le support d'une règle graduelle est le même que le support de son complément, où le complément est défini en remplaçant chaque variation ($\leq$ ou $\geq$) par son inverse.

Ainsi, il suffit de ne générer que la moitié des règles graduelles pour déduire automatiquement leurs complémentaires.

De plus, nous notons que pour chaque i tel $i \in \mathcal{I}$, nous avons $support(i^+) = support(i^-) = 1$, et nous en déduisons que :

- $Conf(i_1^{*1} \Rightarrow i_2^{*2}) = \frac{Freq(i_1^{*1} i_2^{*2})}{Freq(i_1^{*1})}$
- $Conf(i_2^{*2} \Rightarrow i_1^{*1}) = \frac{Freq(i_1^{*1} i_2^{*2})}{Freq(i_2^{*2})}$

Comme $Freq(i_1^{*1}) = Freq(i_2^{*2})$, nous avons $Conf(i_1^{*1} \Rightarrow i_2^{*2}) = Conf(i_2^{*2} \Rightarrow i_1^{*1}) = Freq(i_1^{*1} i_2^{*2})$.

Comme nous l'avons vu dans ce chapitre, il est possible d'extraire de manière efficace des règles graduelles. L'heuristique proposée permet en effet de retrouver la plupart des règles graduelles tout en conservant des temps de calcul et des utilisations de mémoire raisonnables. Cette approche permet notamment de fouiller des bases de données qu'il n'aurait pas été possible de considérer avec les approches exitant précédemment. De plus, ces ensembles de conflits permettent facilement de retrouver les exceptions aux règles graduelles découvertes. Cependant, cette approche reste approximative, l'appliction d'une heuristique pouvant empêcher certaines règles d'être découvertes. Nous proposons donc ci-après une méthode exhaustive fondée sur l'utilisation de treillis. Associée à l'utilisation d'une représentation binaire, cette proposition est très efficace et très peu coûteuse en terme de consommation mémoire.

Chapitre 11

Une approche exhaustive

Dans ce chapitre, nous reprenons les définitions du chapitre précédent et proposons une approche exhaustive pour découvrir les règles graduelles.

11.1 Représentation de la gradualité

Dans notre approche, les items et itemsets graduels permettent de munir l'ensemble des attributs d'une relation d'ordre, qui peut être représentée sous la forme d'un graphe [34, 17].

Par exemple, la figure 11.1 représente les données issues du tableau 10.1 en considérant la relation issue de l'itemset graduel S_1 : "*l'âge augmente ET le salaire augmente ET le nombre de crédits diminue*". Chaque nœud correspond à une ligne (un objet) de la base de données, tandis qu'une flèche entre deux nœuds symbolise la validité de la relation considérée.

La recherche de l'ensemble le plus représentatif s'exprime alors différemment en utilisant ces représentations graphiques. En effet, celui-ci est directement lié à la recherche de la *chaîne* de longueur maximale.

La représentation associée à l'itemset graduel S_1 (figure 11.1) est constitué des deux chaînes représentant l'ensemble des solutions : $\{(p_1p_3p_2p_4p_6), (p_1p_3p_2p_5)\}$. Nous appelons la chaîne composée du plus grand nombre d'éléments *chaîne maximale*. La notion de fréquence est alors directement liée à la longueur des chaînes maximales.

Définition 35 *Soit* $s = (i_1^{*_1}...i_n^{*_n})$ *un itemset graduel, et* $\mathcal{L}_s$ *sa représentation. Soit* $\mathcal{C} = \{C_1, ..., C_p\}$ *l'ensemble des chaînes composant* $\mathcal{L}_s$*. Alors* $Freq(s) = \frac{max(|C_i|)}{|\mathcal{O}|}$*, où* $C_i \in [1, p]$.

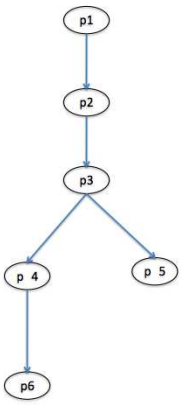

FIGURE 11.1 – (c) $\mathcal{L}_{S_1}$: objets ordonnés pour S_1

Proposition 2 *(Antimonotonie des itemsets graduels) Soit s et s' deux itemsets graduels, nous avons : $s \subseteq s' \Rightarrow Freq(s) \geq Freq(s')$.*

Un itemset graduel est fréquent si son support dépasse un seuil minimal défini par l'utilisateur. L'extraction d'itemsets graduels consiste donc en la recherche de l'ensemble des itemsets graduels fréquents à partir d'une base de données contenant des attributs numériques. Les approches d'extraction de connaissances sont soumises au problèmes de l'explosion combinatoire. Il en va de même pour les itemsets graduels. C'est pourquoi nous utilisons les itemsets graduels complémentaires afin de diminuer l'espace de recherche.

11.2 Algorithmes d'extraction

Nous proposons d'adopter des méthodes d'extraction classiques à la problématique d'extraction graduelle. Afin de faciliter la génération, un arbre de recherche peut être utilisé. Dans ce type d'arbre, appelé *arbre des préfixes*, un nœud représente un item, et le chemin de la racine à une feuille décrit un itemset. Ainsi, nous proposons de générer deux items graduels $i^{\leq}$ et $i^{\geq}$.

Une propriété importante des règles graduelles est leur complémentarité. Cette notion, initialement décrite dans [4], se comprend intuitivement par le fait que chaque ordre total peut trouver son équivalent. Par exemple, "Plus l'âge augmente, plus le salaire augmente" est l'équivalent de "plus l'âge diminue, plus le salaire diminue". Ceci est formalisé par la définition suivante :

Définition 36 *(itemset graduel complémentaire) Soit $s = (i_1^{*_1}...i_n^{*_n})$ un itemset graduel. Son itemset graduel complémentaire est $c(s) = (i_1^{'*_1}...i_n^{'*_n})$ si $\forall j \in [1,n] i_j = i_j^{'}$ et $*_j = c_*(*_j^{'})$, où $c_*(\geq) = \leq$ et $c_*(\leq) = \geq$.*

Proposition 3 *Soit s et s' deux itemsets graduels tels que $c(s) = s'$. Alors l'ensemble des chaînes composant $\mathcal{L}_s$ sont les mêmes que celles composant $\mathcal{L}_{s'}$.*

Corollary 1 $Freq(s) = Freq(c(s))$

Grâce au corollaire 1, la moitié seulement des itemsets seront générés, puisque les autres sont déduits automatiquement. Cette optimisation réduit suffisamment l'espace de recherche afin de passer à l'échelle. Il est maintenant nécessaire de redéfinir l'opération de jointure pour les itemsets graduels. Dans notre contexte, cette opération revient à définir la jointure entre deux graphes.

Nous présentons ici l'algorithme GRITE (GRadual ITemset Extractor). Dans un premier temps, nous expliquons comment joindre deux itemsets de longueur k afin de générer un itemset de longueur $k+1$. Dans un second temps, nous détaillons notre algorithme de calcul de support.

L'opération de jointure sera effectuée de très nombreuses fois. Il est donc indispensable d'utiliser une structure de modélisation adaptée. C'est pourquoi nous utilisons les représentations binaires. En effet, ce type de représentation présente le double avantage d'être peu consommateur de mémoire (un octet représente huit objets) et d'utiliser des opérations binaires très performantes en terme de temps d'exécution. Ainsi, une représentation contenant n sommets peut être projeté en mémoire à l'aide d'une matrice binaire de taille $n \times n$. S'il existe une relation entre un sommet v et un sommet v', alors le bit correspondant à la ligne v et à la colonne v' vaut 1, et 0 sinon. Par exemple, le tableau 11.2 est associé au graphe $\mathcal{L}_{A \geq S \geq C \leq}$. Pour faciliter l'opération d'intersection, la fermeture transitive est donc représentée.

	p_1	p_2	p_3	p_4	p_5	p_6	p_7	p_8
p_1	0	1	1	1	1	1	0	0
p_2	0	0	0	1	1	1	0	0
p_3	0	1	0	1	1	1	0	0
p_4	0	0	0	0	0	1	0	0
p_5	0	0	0	0	0	0	0	0
p_6	0	0	0	0	0	0	0	0
p_7	0	0	0	0	0	0	0	0
p_8	0	0	0	0	0	0	0	0

TABLE 11.1 – Matrice binaire $\mathcal{L}_{A \geq S \geq C \leq}$

À partir des matrices binaires, les sous-séquences communes sont celles dont les bits sont à 1 pour chacune. Ceci est réalisé par l'opération binaire ET entre chaque élément de la matrice :

Theorem 1 *La matrice représentant $\mathcal{L}_{ss'}$ est notée $M_{\mathcal{L}_{ss'}}$. Nous avons alors la relation suivante : $M_{\mathcal{L}_{ss'}} = M_{\mathcal{L}_s}$ **ET** $M_{\mathcal{L}_{s'}}$*

Le théorème 1 rend possible l'utilisation des méthodes générer-élaguer dans

	p_1	p_2	p_3	p_4	p_5	p_6
p_1	0	1	1	1	1	1
p_2	0	0	0	1	1	1
p_3	0	1	0	1	1	1
p_4	0	0	0	0	0	1
p_5	0	0	0	0	0	0
p_6	0	0	0	0	0	0

TABLE 11.2 – Matrice réduite pour $\mathcal{L}_{A^{\geq}S^{\geq}C^{\leq}}$

un temps efficace. En effet, les opérations binaires sont, d'un point de vue processeur, parmi les plus performantes. Le tableau 11.2 montre la représentation obtenue après la jointure entre les itemsets $A^{\geq}S^{\geq}$ et $A^{\geq}C^{\leq}$. Notons que les sommets p_7 et p_8 sont isolés : leurs lignes et leurs colonnes respectives sont à 0. Cela signifie que ces deux sommets n'ont de relation sur les items $A^{\geq}$, $S^{\geq}$ et $C^{\leq}$. La gradualité se mesure d'un objet à l'autre. Ainsi, de tels sommets ne participeront jamais à la chaîne maximale (de par la proposition 2). A l'issue de la jointure, de tels sommets sont élagués, ce qui permet de gagner d'une part de l'espace mémoire, et d'autre part du temps, puisque ceux-ci ne seront pas considérés lors de jointures ultérieures. Le tableau 11.2 montre la matrice $M_{\mathcal{L}_{A^{\geq}S^{\geq}C^{\leq}}}$ élaguée.

L'algorithme réalisant l'opération de jointure opère en deux étapes. Tout d'abord, une nouvelle matrice est initialisée avec les sommets (objets) communs au deux matrices M_s et $M_{s'}$ à joindre. Ensuite, l'opération binaire ET est effectuée entre les sommets communs à M_s et $M_{s'}$.

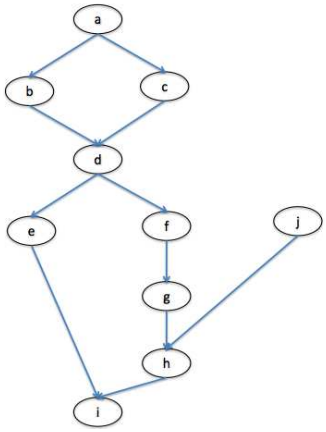

FIGURE 11.2 – Graphe $\mathcal{L}_s$

La fréquence d'un itemset graduel s est la longueur de l'une des chaînes maximales de $\mathcal{L}_{ss'}$ associé. Le calcul du plus long chemin est un problème difficile. Or, tout comme la jointure, le calcul du support est effectué un très grand nombre de fois. C'est pourquoi nous posons la contrainte suivante : chaque sommet ne devra être considéré qu'une seule fois.

chain	length
$\{abdei\}$	5
$\{abdfghi\}$	7
$\{abcei\}$	5
$\{abdfghi\}$	7
$\{kghi\}$	4

FIGURE 11.3 – Chaînes composant le graphe

Un sommet peut avoir plusieurs niveaux. Par exemple, considérons le graphe de la figure 11.2. Celui-ci est composé de 5 chaînes énumérées dans le tableau 11.2. Certains des sommets ont plusieurs pères, et participent donc à plusieurs chaînes. C'est le cas du sommet g, qui a pour pères les sommets f et i. Ainsi, selon le chemin emprunté, on peut considérer le sommet g comme ayant un niveau de 5 ou de 2. Notre but est de maximiser les niveaux. Pour cela, nous avons mis en place un système de "*mémoire*", qui conserve les données obtenues à partir des nœuds de niveau supérieur. Lorsque plusieurs solutions sont possibles, nous conservons le niveau le plus élevé.

Entrées : Un sommet $vertex$
La mémoire $Memory$
Sorties : $Memory$ Complétée

$Sons \leftarrow GetSons(vertex)$
si $Sons = \emptyset$ **alors**
  $Memory[vertex] = 1$;
sinon
  pour chaque $i \in Sons$ **faire**
    si $Memory[i] = -1$ **alors**
      $RecursiveCovering(i, Memory)$
    fin
  fin
  pour chaque $i \in Sons$ **faire**
    $Memory[vertex] = max(Memory[vertex], Memory[i] + 1)$
  fin
fin

Algorithme 8: FreqRecursive

Discussion

Dans cette partie, nous avons présenté nos premiers résultats concernant l'extraction de règles graduelles de la forme *"Plus le mur est proche, plus le train doit freiner fort"*. Ces règles sont exploitées par de nombreux experts avec lesquels nous collaborons, notamment dans le domaine de la santé (INSERM, IRCM, Univ. Montpellier 3).

Dans un premier temps, nous nous sommes attardés sur la définition de ce qu'est un item graduel et un itemset graduel, dont découle la notion de règle graduelle. Nous avons montré, notamment à travers l'étude des approches exitantes, que le problème de l'extraction de telles règles est rendu difficile d'une part par les nombreuses définitions qu'il est possible d'associer aux concepts manipulés, et d'autre part par la difficulté de parcourir l'espace de recherche efficacement.

Pour faire face à ces difficultés, différentes approches sont considérées, selon qu'elles tentent de découvrir l'ensemble des règles (approche exhaustive) ou non (approche utilisant une heuristique).

Notons que les méthodes présentées ici ne permettent pas de prendre en compte la séquentialité, problème sur lequel nous sommes en train de travailler (voir publications récentes). Cependant, des approches ont également été proposées dans le cadre du post-doctorat de C. Fiot, par exemple pour extraire des motifs du type *"Plus le nombre de requêtes sur la page P1 est fort, plus le nombre de requêtes sur la page P2 sera faible quelques secondes plus tard"* ou encore des règles de la forme *"dans la plupart des cas, une augmentation des requêtes sur la page P1 pendant une courte période précède une augmentation des requêtes sur la page P2 après une très courte période de temps"* décrivant les relations temporelles qui interviennent entre les événements. Notre travail actuel est notamment lié à l'optimisation des algorithmes afin de permettre d'extraire des motifs séquentiels graduels à partir de bases de données denses contenant de nombreux attributs [17].

Étant parmi les premières équipes internationales à nous intéresser au problème d'algorithmes efficaces, il est encore difficile de dresser un bilan comparatif de nos approches, tant la difficulté est grande, d'une part pour définir ce que sont une règle et un motif graduels, mais surtout pour définir des algorithmes efficaces.

Il ne fait pourtant aucun doute que ces approches sont prometteuses, tant les premières règles et motifs extraits semblent satisfaire les experts avec lesquels nous collaborons. Il n'en reste pas moins que de très nombreuses perspectives sont associées à ces travaux (voir les perspectives détaillées à la fin de ce mémoire).

En particulier, nous souhaitons permettre l'extraction de règles graduelles à partir d'entrepôts de données, ou de flots de données. Les problèmes posés sont alors très grands en raison de la complexité des données qui s'ajoute à la complexité des traitements.

De plus, de nombreuses pistes de recherche seront exploitées dans le cadre de la thèse en co-tutelle avec la Tunisie de S. Ayouni (co-encadrée avec S. Ben Yahia), pour notamment étudier les formes de représentations condensées de tels ensembles de règles et motifs ainsi que les propriétés des sous-ensembles flous capables d'aider à définir des algorithmes à la fois souples et efficaces. Des collaborations avec l'institut Louis Pasteur de Tunis seront menées dans le cadre de cette thèse.

Dans le contexte de la gestion des imperfections, nous souhaitons également prendre en compte la *force de la variation* (une variation d'une unité n'ayant pas la même signification qu'une variation de cent unités).

Enfin, nous souhaitons, en accord avec les experts, définir des mesures de qualité spécifiques aux règles graduelles, ainsi que des systèmes de contraintes (par exemple par l'ajout de connaissances externes connues sous la forme d'ontologies) afin de gérer au mieux le grand nombre de règles extraites soit en le réduisant au vu des contraintes posées par l'expert, soit en les ordonnant selon les besoins de l'expert.

Sixième partie

Conclusion et Perspectives

Chapitre 12

Conclusion

Dans ce mémoire, nous avons rapporté les principaux travaux menés depuis 2003, dans un contexte local au laboratoire (encadrement d'étudiants en Master et thése), régional (collaborations scientifiques avec d'autres laboratoires et organismes de recherche montpelliérains), national (collaborations avec des organismes de recherche français), international (Malaisie, Pakistan, Indonésie, Tunisie), et industriel (sociétés incubées, grands groupes).

Ces travaux ont notamment été menés autour des thématiques liées aux entrepôts de données, à la fouille de données et à la théorie des sous-ensembles flous. Thémes récurrents dans le monde qui nous entoure, où plus que jamais les entreprises et organisations sont noyées sous de gros volumes de données dont elles ne savent que faire, ces domaines de recherche n'en sont pas moins de vrais challenges pour la recherche.

Ces travaux m'ont permis de confronter à chaque instant les idées de chacun aux enjeux de la recherche et de ses applications. Cette confrontation aux données réelles, amorcée lors de ma thèse avec la collaboration avec le Ministère de l'Éducation Nationale (données liées aux résultats du bacalauréat), loin de réduire le champ de recherche, a été une source de perpétuels enrichissements des études en cours et d'ouvertures vers de nouveaux problémes de recherche. La participation active d'élèves-ingénieurs de Polytech'Montpellier et d'étudiants en Licence et Master de la faculté des Sciences à ces travaux n'a fait qu'ajouter à cette dynamique de continuels échanges entre recherche, formation et entreprises.

12.1 Transferts technologiques

Nous listons ci-dessous les principaux projets et collaborations menés au cours de ces dernières années :

Responsabilités de projets de recherche et collaborations industrielles

- Responsable scientifique pour le LIRMM de l'ANR MIDAS (108Keuros)
- Reponsable du projet EXTRACAP (société SQLi). Transfert de technologie. Projet régional SPRINTT. "Classification de messages à l'aide des motifs séquentiels". 2004. (29Keuros)
- Responsable scientifique pour l'informatique dans le contrat équipe-conseil Satin - Polytech' (incubation LRI/polytech') (4Keuros)
- Responsable scientifique pour le LIRMM dans la collaboration de recherche New ID - LIRMM (incubation LRI/LIRMM) (8Keuros)
- Co-responsable scientifique pour le LIRMM et responsable scientifique pour POLYTECH dans la collaboration de recherche NOMEXCO - LIRMM - POLYTECH - PRAXILING (incubation LRI. 4Keuros LIRMM, 4Keuros Polytech')
- Co-responsable scientifique dans la collaboration de recherche EDF - LIRMM (50Keuros)
- Responsable scientifique pour le LIRMM du PEPS CNRS "LAMAL. Langage, Mémoire et Alzheimer : une approche des maladies neuro-dégénératives fondée sur la densité des idées. Fouille de données pour l'extraction de corrélations entre différents indices neuropsychologiques."
- Responsable scientifique pour le LIRMM et Polytech' dans la collaboration de recherche avec "We are Cloud" (incubation LRI/LIRMM/POLYTECH) (12Keuros)

Participation à des projets de recherche et collaborations industrielles

- projet STIC-ASIA Exploitation des Entrepôts de données-EXPEDO réunissant : Univ. Cergy-Pontoise, Univ. Orsay, Univ. Tours, HELP University College Malaisie, ITB Indonésie, STIKOM-Bali Indonésie, Pakistan Science Fundation (financement Ministère des Affaires Etrangères)
- Projet incubé KEOSIA (Fouille de données et santé)
- Projet incubé AIRTIST (Fouille de données et profils d'internautes téléchargeant de la musique)
- PEPS "GeneMining : Vers un processus de fouille de données adapté à l'analyse des bio-puces et basé sur les connaissances consensuelles du domaine".

Outre les collaborations listées ci-dessus, mes travaux ont été menés dans le cadre du co-encadrement d'étudiants en thèse, Master, ou mémoire d'ingénieur CNAM, que nous rappelons ci-dessous.

12.2 Encadrements d'étudiants

Thèses

- F. Del Razo Lopez : *Recherche de structures fréquentes dans les données semi-structurées* (débutée en 2003 - soutenue en juillet 2007, bourse Sfere Mexique. 40%)

- C. Fiot : *Prise en compte des données manquantes ou incomplètes lors de la recherche de motifs séquentiels dans de grandes bases de données* (débutée en 2004 - soutenue en septembre 2007, BDI CNRS. 40%)
- M. Plantevit : *Fouille de données pour les bases de données multidimensionnelles* (débutée en 2005 - soutenue en juillet 2008. allocataire-moniteur. 60%)
- H. Li : *Mesures de qualité et causalité pour les motifs séquentiels. Recherche de motifs exceptionnels* (2006-2009, bourse EMA. 70%)
- L. Di Jorio : *Fouille de Données et santé : découverte de règles graduelles* (2007-2010, BDI-CNRS. 50%)
- Y. Pitarch : *Bases de données multidimensionnelles et données en flots : stockage et fouille* (2008-2011, allocation sur ANR MIDAS. 60%)
- H. Saneifar : *Extraction d'information dans des masses de données complexes et évolutives* (2008-2011, Convention CIFRE société Satin-IP. 30%)
- S. Ayouni : *Extraction de règles graduelles floues : définition d'algorithmes efficaces et représentations concises* (2008-2011, Thèse en co-tutelle avec la Tunisie. 40%)

DEA et Master recherche

- C. Fiot : *Extraction de motifs séquentiels flous* (2004. 40%).
- M. Plantevit : *Recherche de motifs séquentiels au sein de bases de données multidimensionnelles* (2005. 50%).
- A. Rammal : *Fouille de données et préservation de la vie privée* (2006. 70%).
- H. Li : *Hypergraphes pour la recherche de motifs séquentiels* (2006. 50%).
- D. Jouve : *Comportement atypiques dans des données multidimensionnelles* (2007. 80%)
- Y. Pitarch : *Cubes de données et streams* (2008. 30%)
- H. Saneifar : *Clustering de données séquentielles : application à la détection d'intrusions* (2008. 50%)
- M. Sghaier : *Recherche de sous-structures de donnéees arborescentes fréquentes. Introduction d'une approche floue sur le niveau horizontal et proportion des nœuds* (2008. 60%)

Mémoires d'ingénieur CNAM encadrés au sein du LIRMM

- A. Beaud : *Implantation d'une plate-forme de fouille de données multidimensionnelles* (2005. 100%)
- E. Cazal : *Conception et mise en œuvre d'une plateforme d'analyse automatique de textes en Ancien Français* (2007. 50%)
- S. Sanchez : *Fouille de données arborescentes* (2006. 20%)

12.3 Vers de nouveaux défis

Initiés au moment où les communautés commencaient à se rapprocher, mes travaux ont profité des avantages de chacune des approches de communautés diverses. Sans nier les difficultés liées à de tels rapprochements, j'ai maintenant l'assurance qu'il est possible de concilier des approches jadis vues comme antagonistes. Il apparaît par exemple maintenant évident qu'il est possible de trouver des pistes pour lier passage à l'échelle et représentations et traitements des imperfections, comme le montrent les chapitres du livre co-édité avec MJ. Lesot. En ce sens, je suis ravie d'avoir participé à l'aventure de la mise en commun de résultats de communautés jusqu'alors trop éloignées.

Il n'en reste pas moins que de nombreuses brèches restent ouvertes, entre les communautés d'informaticiens-mathématiciens d'une part, et entre les informaticiens et experts d'autre part. Au-delà de l'effort de pédagogie, des différences existent, et doivent exister, entre les diverses approches. De notre côté, nous réaffirmons notre position, résumée ainsi : l'informaticien ne doit pas se substituer à l'expert. La représentation du monde qui nous entoure ne sera jamais parfaite, le monde ne l'étant pas lui-même, et encore moins les mesures que l'on peut en faire, il faut donc gérer ces imperfections tant au niveau de la représentation que de l'extraction.

Chapitre 13

Perspectives

Les perspectives associées à ce travail sont nombreuses. Certaines font d'ores et déjà partie de nos prospections. D'autres sont des perspectives à plus long terme.

13.1 Fouille de stream cubes

Les cubes de données construits à partir de flots de données sont de plus en plus nombreux. Des modèles de construction et de maintien de telles structures sont en cours de définition dans le cadre de l'ANR MIDAS. Il s'agit alors de déterminer les processus d'interrogation, en étudiant en particulier les formes d'imprécision qui devront être présentes dans les réponses apportées par les systèmes, toutes les données ne pouvant être stockées sur toute la période de vie de l'entrepôt. En liaison avec les experts psychologues, nous étudions donc de nouvelles formes de stockage d'historiques de données arrivant à très haute vitesse (flots) et les formes d'interrogation de cet historique particulier.

Nous étudions en ce sens différentes méthodes de fouille de données :
- d'une part pour construire et maintenir cet historique,
- d'autre part pour extraire de cet historique des connaissances pertinentes.

13.2 Gestion des incertitudes

Nous souhaitons développer les travaux associés à la représentation et la fouille de données imparfaites, avec un intérêt grandissant pour les données incertaines, par exemple pour la gestion de la fiabilité de flots issus de sources multiples. Cet aspect avait été abordé succintement lors de ma thèse. Cependant il reste un large champ d'exploration. La théorie des possibilités et la théorie de

Dempster Shaffer seront des cadres formels auxquels nous nous intéresserons.

13.3 Skylines flous

Les requêtes de type *skyline* correspondent à des requêtes renvoyant les objets qui répondent de manière optimale à au moins l'un des critères de l'interrogation, tout en restant meilleurs ou égaux à tous les autres objets sur les autres critères de l'interrogation [60]. Etudiées depuis longtemps en théorie de la décision (analyses multi-critères), ces requêtes gagnent maintenant le terrain de la définition d'algorithmes performants.

Nous nous intéresserons dans ce contexte à la prise en compte de l'imperfection, d'une part dans les données et les critères d'interrogation (e.g. proximité d'un hôtel par rapport au centre ville), et d'autre part dans les résultats retournés (appartenance graduelle des objets au résultat).

13.4 Entrepôts de données temps réel

Les entrepôts ont longtemps été étudiés comme des réceptacles d'information "hors-ligne" ne nécessitant pas de réel ancrage temps réel. Les mises à jour sont alors effectuées de manière régulière lorsque les systèmes opérationnels sous-jacents sont le moins actifs, et l'utilisateur accepte que le résultat de ses interrogations ne soit pas parfaitement juste, en fonction de ses mises à jour. Cependant, cette hypothèse est de moins en moins acceptée, et il s'agit maintenant de construire des entrepôts temps réel, au fait des derniers changements réalisés au niveau opérationnel. Or, si cette thématique est liée au traitement des données en flots, elle en est cependant quelque peu éloignée puisqu'il ne s'agit pas tant de gérer un flot très rapide que plutôt d'offir des outils de décision temps réel à l'utilisateur.

Dans ce contexte, la gestion d'entrepôts temps réel sur données en flots sera cependant un des prochains challenges sur nos thématiques. Les pistes seront notamment explorées en collaboration avec des industriels (IBM notamment), afin de travailler à de nouvelles architectures intégrant tous les aspects, depuis les matériels jusqu'aux outils intelligents. En particulier, les architectures massivement parallèles seront étudiées dans ce contexte. Notons qu'un projet est en cours de lancement sur le plan régional, et sera le support de cette recherche.

13.5 Fouille de cubes de données non-structurées et semi-structurées

Très récemment [43], une structure de cube de données textuelles reprenant les principales mesures liées à la recherche d'information, notamment les fréquences des termes dans les documents (TF) et les fréquences inverses (IDF) a été proposée. Si des travaux avaient été précédemment proposés, cette nouvelle approche permet d'envisager la construction de cubes de données associant les informations .

Dans nos perspectives, nous envisageons d'étendre ces travaux pour :
- définir de nouvelles caractéristiques pouvant être utilisées comme mesures de cubes,
- étendre la notion de hiérarchie pour prendre mieux en compte les caractéristiques de distances et relations entre termes : lien vers plusieurs termes plus généraux, prise en compte de la distance (e.g. wordnet),
- utiliser ces cubes de données comme support de la fouille de données.

Ces travaux seront menés en collaboration avec l'équipe-projet TAL (Traitement Algorithmique du Langage) du LIRMM, en particulier avec M. Roche, en lien étroit avec le Help University College avec lequel nous collaborons sur les aspects entrepôts de données.

13.6 Règles et motifs graduels

Les recherches menées sur cette thématique sont prometteuses. Étant l'une des premières équipes à nous intéresser à la définition d'algorithmes efficaces, et appliquant ces travaux dans différents domaines (e.g. biologie, psychologie cognitive), nous travaillons actuellement à de nombreuses pistes.

Règles graduelles séquentielles

Une attention particulière est actuellement portée pour l'extraction de motifs graduels à partir de données séquentielles. Notre but est d'extraire des règles exprimant des évolutions (gradualité) au cours du temps. Deux types de motifs sont considérés :
- motifs graduels intra-transations,
- motifs graduels inter-transactions.

Le premier type de motif graduel fait référence à des données dans lesquelles un ensemble de descripteurs seraient évalués plusieurs fois de manière séquentielle (à intervalles réguliers ou non). Par exemple, la base de données PAQUID comprend des indicateurs sur le niveau de performance cognitive de personnes au cours de plusieurs années. Nous pourrons ainsi comparer l'évolution et la coévalotion (croissante/décroissante) de différents descripteurs au cours du temps.

Patient	$Date_1$	$Date_2$	$Date_3$
P1	Test A : 5 - Test B : 22	Test A : 6 - Test B : 24	Test A : 8 - Test B : 30
P2	Test A : 2 - Test B : 12	Test A : 2 - Test B : 18	Test A : 4 - Test B : 20

TABLE 13.1 – Exemple : *Au cours du temps, plus la réussite au test A est importante, plus la réussite au test B est grande*

Client	$Date_1$	$Date_2$	$Date_3$
C1	DVDs (2)	Chips (1)	
C2	DVDs(4)	Carottes (8)	Chips (3)

TABLE 13.2 – Exemple : *Plus le nombre de DVDs achetés est important, plus le nombre de paquets de chips l'est quelque temps plus tard*

Le deuxième type de motif fait référence à des données exprimant également des comportements au cours du temps (ou présentant un quelconque ordre) mais comparant les valeurs des attributs d'un objet à l'autre.

Par exemple, le tableau 13.6 permet d'extraire la règle *Au cours du temps, plus la réussite au test A est importante, plus la réussite au test B est grande* tandis que le tableau 13.6 contient le motif graduel séquentiel inter-transaction *plus le nombre de DVDs achetés est important, plus le nombre de paquets de chips l'est quelque temps plus tard.*

Quel que soit le type de motif considéré, nous souhaitons définir des algorithmes efficaces permettant un passage à l'échelle et la prise en compte de l'ensemble des données et descripteurs présents dans les bases de données.

Règles graduelles et mesures de qualité

S'il existe de nombreuses mesures permettant d'évaluer la qualité d'une règles d'association ou d'autres motifs extraits par les méthodes de fouille de données [24], il n'en est pas de même pour les règles et motifs graduels. Nous proposons donc d'étudier de nouvelles mesures statistiques liées à la notion de gradualité. Liées aux mesures statistiques de corrélation, ces propositions devront néanmoins être facilement mises en œuvre, le but étant de pouvoir associer chaque règle/motif séquentiel(le) à de indicateurs de qualité en un temps raisonnable.

Règles graduelles et exceptions

Nous envisageons également d'extaire les objets étant le plus en contradiction avec les règles et motifs graduels présents au sein d'un ensemble de données et présentant en ce sens une caractéristique d'exception. Pour ce faire, nous nous appuierons d'une part sur les mesures de qualité actuellement étudiée, et définirons notamment une mesure de distance entre règles et motifs graduels, et

d'autre part sur la notion d'ensembles de conflits décrits dans ce mémoire. En effet, les objets présents dans les ensembles de conflits sont ceux qui s'opposent à l'ordonnancement des objets de la base. Deux problèmes seront alors étudiés : d'une part le fait qu'un objet, même s'il est présent dans l'ensemble de conflit à une étape donnée de notre processus, peut se révéler pas autant contradictoire que d'autres objets mais ne sera plus jamais étudié en raison de notre heuristique gloutonne, et d'autre part le fait que de très (trop) nombreux objets sont présents dans les ensembles de conflit, ce qui pourrait conduire à extraire un très grand nombre d'exceptions inexploitable pour l'utilisateur final.

Extraction de règles et motifs graduels à partir de flots de données

La fouille de données en flots est un domaine très actif dans lequel l'équipe TATOO a activement participé, notamment au travers la thèse de C. Raissi (encadrée par P. Poncelet) [54]. Cependant, si l'extraction de règles d'associations et de motifs séquentiels ont été étudiés, aucune proposition n'a été faite dans le cadre de motifs graduels. Or les motifs graduels sont très adaptés au contexte de flots de données pour décrire des tendances et des évolutions (accroissement, baisse). Nous proposons donc d'étudier comment des motifs graduels peuvent être extraits de manière efficace (temps/mémoire) à partir de flots de données.

Ce travail sera mené dans le cadre de la visite de Jordi Nin Guerrero (actuellement en post-doctorat à l'*Artificial Intelligence Research Institute* à Bellaterra, Espagne) au sein de l'équipe TATOO du LIRMM.

Extraction de règles et motifs graduels et flou

Une règle graduelle floue est une expression de la forme "plus/moins X est A, plus/moins Y est B", par exemple "plus l'âge est jeune, plus le Salaire est Bas". D'une manière approximative, la sémantique de ce type de règles floues est : "plus/moins le degré d'appartenance de la valeur de X dans A est grand, plus/moins le degré d'appartenance de la valeur de Y dans B est grand" [19, 6]. En d'autres termes, ces règles expriment la modification progressive du degré auquel l'entité Y satisfait la propriété graduelle B en fonction du degré auquel l'entité X satisfait la propriété graduelle A. Ainsi, une règle graduelle floue permet de rendre compte d'une influence positive ou négative continue d'un attribut sur un autre. Il existe plusieurs définitions de règles graduelles floues, selon que la gradualité concerne le degré d'appartenance à un sous-ensemble flou, ou le degré de certitude d'apparition d'un évènement etc. [18]. Nous proposons d'étudier davantage la sémantique des règles graduelles floues et leur formalisation dans le cadre de la fouille de données. Ceci nous mènera à étudier les différents types d'implications floues et leurs implémentations afin de parvenir à faire le choix de l'implication la plus adéquate pour un contexte d'extraction donné en fonction du sens des connaissances à représenter. Après avoir bien appréhendé le concept de règles graduelles floues nous allons nous intéresser à définir des

approches algorithmiques efficaces d'extraction de ce type de règles. Ceci en mettant en exergue les propriétés liées aux sous ensembles flous et aux partitions floues (i.e., α-coupures). En effet, le fait d'inclure ces propriétés dans les algorithmes d'extraction va minimiser le nombre de règles extraites sans pourtant perdre des informations (i.e., avec une telle coupure nous pouvons stopper l'exploration de " plus X est A " quand " plus X est A' " est fortement présent. Nous proposons également d'étudier la possibilité de réduire l'ensemble de toutes les règles graduelles floues sans perte de connaissance. Ceci en définissant une représentation concise (i.e., base générique) couvrant l'ensemble de toutes les règles graduelles floues, ainsi que les mécanismes d'inférence de l'ensemble de toutes les règles redondantes à partir de cette représentation concise.

Ces travaux sont et seront menés dans le cadre de la thèse de S. Ayouni réalisée en co-tutelle avec la Tunisie (S. Ben Yahia), et trouveront application auprès de différents partenaires, en particulier l'Institut Pasteur de Tunis.

Extraction de règles et motifs graduels et entrepôts de données

De nombreuses perspectives sont associées à l'extraction d'information sous la forme de règles graduelles à partir de données d'entrepôts. Nous souhaitons ici exploiter les caractéristiques des bases de données multidimensionnelles afin de proposer de nouvelles définition et méthodes.

De même que les travaux portant sur les entrepôts de données textuelles, ces travaux seront menés en collaboration avec le Help University College (Malaisie), dans le cadre de la thèse de L. Di Jorio.

En particulier, nous étudierons l'effet de l'impact de la notion de hiérarchie et d'agrégation sur la gradualité.

Une autre piste est de considérer les blocs de données présentés dans ce mémoire et de rechercher s'ils peuvent être ordonnés selon la valeur de mesure (valeur du bloc).

Nous avons également pour but de nous attarder sur le caractère lié à la présence de données symboliques (dimensions) décrivant des données numériques (mesures). Nous pensons alors proposer une méthode correspondant à une extension des travaux développés par J. Han dans son approche des motifs séquentiels multidimensionnels [49]. Notre idée est de coupler des données numériques (ou au moins munies d'un ordre total) et des données non ordinales. Ainsi, il sera possible de trouver des règles du type : `Chez les personnes habitant le sud de la France et appartenant à la catégorie socio-professionnelle des retraités,` *plus le nombre d'achats de a est important, plus le nombre d'achats de b est faible.*

Bibliographie

[1] R. Agrawal, T. Imielinski, and A. Swami. Mining association rules between sets of items in large databases. In *Proc. of ACM SIGMOD*, pages 207–216, 1993.

[2] R. Agrawal and R. Srikant. Mining sequential patterns. In *Proc. 1995 Int. Conf. Data Engineering (ICDE'95)*, pages 3–14, 1995.

[3] V. Barnett and T. Lewis. *Outliers in Statistical Data*. John Wiley & Sons, 1994.

[4] F. Berzal, J.C. Cubero, D. Sanchez, M.A. Vila, and J.M. Serrano. An alternative approach to discover gradual dependencies. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems (IJUFKS)*, 15(5) :559–570, 2007.

[5] Fernando Berzal, Juan-Carlos Cubero, and Nicolás Marín. Anomalous association rules. In *IEEE ICDM Workshop Alternative Techniques for Data Mining and Knowledge Discovery.*, 2004.

[6] B. Bouchon-Meunier, D. Dubois, LL. Godó, and H. Prade. Fuzzy sets and possibility theory in approximate and plausible reasoning. In *Handbooks of Fuzzy Sets*, pages 15–190. Kluwer Academic Publishers, 1999.

[7] L. Cabibbo and R. Torlone. A Logical Approach to Multidimensional Databases. In *Sixth International Conference on Extending Database Technology (EDBT'98)*, pages 183–197, Valencia, Spain, 1998. Lecture Notes in Computer Science 1377,Springer-Verlag.

[8] K.C. Chan and W.-H. Au. Mining fuzzy association rules. In *CIKM '97 : Proceedings of the sixth international conference on Information and knowledge management*, pages 209–215. ACM, 1997.

[9] S. Chaudhuri and U. Dayal. An overview of data warehousing and olap technology. *ACM-SIGMOD Records*, 26(1) :65–74, 1997.

[10] YW. Choong, A. Laurent, and D. Laurent. Summarizing data cubes using blocks. *Data Mining Patterns : New Methods and Applications*, 2007.

[11] Y.W. Choong, A. Laurent, and D. Laurent. Mining multiple-level fuzzy blocks from multidimensional data. *Fuzzy Sets and Systems*, 159(12), 2008.

[12] Y.W. Choong, D. Laurent, and A. Laurent. Pixelizing data cubes : a block-based approach. In *Visual Information Expert Workshop (VIEW'06)*, Lecture Notes in Computer Science, 2006.

[13] Y.W. Choong, D. Laurent, and P. Marcel. Computing appropriate representation for multidimensional data. *DKE Int. Journal*, 45 :181–203, 2003.

[14] Y.W. Choong, D. Laurent, and P. Marcel. Computing appropriate representations for multidimensional data. *Data Knowledge Eng.*, 45(2) :181–203, 2003.

[15] Y.W. Choong, P. Maussion, A. Laurent, and D. Laurent. Summarizing multidimensional databases using fuzzy rules. In *Proc. of the 10th Int. Conf. on Information Processing and Management of Uncertainty in Knowledge-Based Systems (IPMU'04)*, pages 99–106, 2004.

[16] E.F. Codd, S.B. Codd, and C.T. Salley. Providing olap (on-line analytical processing) to user-analysts : An it mandate. In *White Paper*, 1993.

[17] L. Di Jorio, A. Laurent, and M. Teisseire. Mining frequent gradual itemsets from large databases. In *Int. Conf. on Intelligent Data Analysis, IDA'09*, 2009.

[18] D. Dubois, E. Hülermeier, and H. Prade. A note on quality measures for fuzzy association rules. In *Int. Fuzzy Systems Association World Congress on Fuzzy Sets and Systems*, 2003.

[19] D. Dubois and H. Prade. Fuzzy rules in knowledge-based systems. modelling gradedness, uncertainty and preference. In R.R. Yager and L.A. Zadeh, editors, *An introduction to fuzzy logic applications in intelligent systems*, pages 45–68. Kluwer, 1992.

[20] Didier Dubois and Henri Prade. Gradual inference rules in approximate reasoning. *Information Sciences*, 61(1-2) :103–122, 1992.

[21] C. Fiot, A. Laurent, and M. Teisseire. From crispness to fuzziness : Three algorithms for soft sequential pattern mining. *IEEE Transactions on Fuzzy Systems*, 15(6) :1263–1277, 2007.

[22] S. Galichet, D. Dubois, and H. Prade. Imprecise specification of ill-known functions using gradual rules. *International Journal of Approximate Reasoning*, 35 :205–222, 2004.

[23] J. Gray, A. Bosworth, A. Layman, and H. Pirahesh. Data Cube : A Relational Aggregation Operator Generalizing Group-By, Cross-Tabs, and Sub-Totals. *Journal of Data Mining and Knowledge Discovery*, 1(1) :29–53, 1997.

[24] F. Guillet and J.H. Hamilton, editors. *Quality Measures in Data Mining*, volume 43 of *Studies in Computational Intelligence*. Springer Verlag, 2007.

[25] M. Gyssens and L.V.S. Lakshmanan. A Foundation for Multidimensional Databases. In *Proc. 23rd Int. Conf. on Very Large Data Bases*, pages 106–115, Athens, Greece, August 1997.

[26] J. Han. Olap mining : An integration of olap with data mining. In *In Proceedings of the 7th IFIP 2.6 Working Conference on Database Semantics (DS-7*, pages 1–9, 1997.

[27] J. Han and M. Kamber. *Data Mining - Concepts and Techniques*. Morgan Kaufmann, 2001.

[28] Jiawei Han. Olap mining : Integration of olap with data mining. In *DS-7*, pages 3–20, 1997.

[29] D. Hawkins. *Identification of Outliers*. Chapman and Hall, London, 1980.

[30] Y.-C. Hu, R.-S. Chen, G.-H. Tzeng, and J.-H. Shieh. A fuzzy data mining algorithm for finding sequential patterns. *Int. Journal of Uncertainty Fuzziness and Knowledge-Based Systems*, pages 173–193, 2003.

[31] E. Hüllermeier. Association rules for expressing gradual dependencies. In *PKDD '02 : Proceedings of the 6th European Conference on Principles of Data Mining and Knowledge Discovery*, pages 200–211. Springer-Verlag, 2002.

[32] W.H. Inmon. *Builing the Datawarehouse*. John Wiley and Sons, 2003.

[33] M. Jarke, M. Lenzerini, Y. Vassiliou, and P. Vassiliadis. *Fundamentals of Data Warehouses*. Springer-Verlag, 1998.

[34] L. Di Jorio, A. Laurent, and M. Teisseire. Extraction efficace de règles graduelles. In *Proc. Conf. EGC'2009*, 2009.

[35] J. Kacprzyk, R.R. Yager, and S. Zadrozny. A fuzzy logic based approach to linguistic summaries of databases. *Int. Journal of Applied Mathematics and Computer Science*, 10 :813–834, 2000.

[36] J. Kacprzyk and S. Zadrozny. Computing with words : Towards a New Generation of Linguistic Querying and Summarization in Databases. In P. Sinčák, J. Vaščák, V. Kvasnička, and R. Mesiar, editors, *Proc. of the Euro-International Symposium on Computational Intelligence (ISCI)*, volume 54, Kosice, Slovaquie, 2000. Springer-Verlag.

[37] C. Kim, J.-H. Lim, R. Ng, and K. Shim. SQUIRE : Sequential pattern mining with quantities. In *20th International Conference on Data Engineering (ICDE'04)*, page 827, 2004.

[38] R. Kimball. *The Datawarehouse Toolkit*. John Wiley & Sons, 1996.

[39] Edwin M. Knorr and Raymond T. Ng. A unified notion of outliers : Properties and computation. In *KDD*, pages 219–222, 1997.

[40] Edwin M. Knorr and Raymond T. Ng. Algorithms for mining distance-based outliers in large datasets. In *Proc. of Int. Conf. of Very Large Data Bases (VLDB'98)*, pages 392–403. Morgan Kaufmann, 1998.

[41] Chan Man Kuok, Ada Fu, and Man Hon Wong. Mining fuzzy association rules in databases. *SIGMOD Record*, 27 :41–46, 1998.

[42] L. Lakshmanan, J. Pei, and J. Han. Quotient cube : How to summarize the semantics of a data cube. In *Proc. of VLDB*, pages 778–789, 2002.

[43] C. X. Lin, B. Ding, J. Han, F. Zhu, and B. Zhao. Text cube : Computing ir measures for multidimensional text database analysis. In *Proc. 2008 Int. Conf. on Data Mining (ICDM'08)*, 2008.

[44] S. Lin and D.E. Brown. Criminal incident data association using the olap technology. In *ISI*, pages 13–26, 2003.

[45] P. Marcel. Modeling and querying multidimensional databases : An overview. *Networking and Information Systems Journal*, 2(5-6) :515–548, 1999.

[46] F. Masseglia, F. Cathala, and P. Poncelet. The PSP Approach for Mining Sequential Patterns. In *Proc. of PKDD*, volume 1510 of *LNCS*, pages 176–184, 1998.

[47] R. Ben Messaoud, S. Loudcher Rabaséda, O. Boussaid, and R. Missaoui. Enhanced mining of association rules from data cubes. In *DOLAP*, pages 11–18, 2006.

[48] J. Pei, J. Han, B. Mortazavi-Asl, J. Wang, H. Pinto, Q. Chen, U. Dayal, and M.-C. Hsu. Mining sequential patterns by pattern-growth : The prefixspan approach. *IEEE Transactions on Knowledge and Data Engineering*, 16(10), 2004.

[49] H. Pinto, J. Han, J. Pei, K. Wang, Q. Chen, and U. Dayal. Multi-dimensional sequential pattern mining. In *ACM CIKM*, pages 81–88, 2001.

[50] M. Plantevit. *Extraction de motifs séquentiels à partir de bases de données multidimensionnelles.* PhD thesis, Université Montpellier 2, 2008.

[51] M. Plantevit, A. Laurent, and M. Teisseire. Up and down : Mining multidimensional sequential patterns using hierarchies. In *International Conference on Data Warehousing and Knowledge Discovery (DaWaKÕ08)*, 2008.

[52] M. Plantevit, A. Laurent, and M. Teisseire. Olap-sequential mining : Summarizing trends from historical multidimensional data using closed multidimensional sequential patterns. *Annals of Information Systems, special issue in New Trends in Data Warehousing and Data Analysis*, 2009.

[53] Marc Plantevit, Yeow Wei Choong, Anne Laurent, Dominique Laurent, and Maguelonne Teisseire. M^2SP : Mining Sequential Patterns Among Several Dimensions. In Alípio Jorge, Luís Torgo, Pavel Brazdil, Rui Camacho, and João Gama, editors, *PKDD*, volume 3721 of *Lecture Notes in Computer Science*, pages 205–216. Springer, 2005.

[54] C. Raïssi. *Extraction de séquences fréquentes : des bases de données statiques aux flots de données.* PhD thesis, Université Montpellier 2, 2008.

[55] G. Raschia and N. Mouaddib. Evaluation de la qualité des partitions de concepts dans un processus de résumés de bases de données. In *Rencontres francophones sur la logique floue et ses applications*, pages 297–305, La Rochelle, France, 2000. Cépaduès Editions.

[56] R.Srikant and R. Agrawal. Mining quantitative association rules in large databases. In *ACM SIGMOD Conference Proceedings*, pages 1–12, 1996.

[57] H. Saneifar, S. Bringay, A. Laurent, and M. Teisseire. S2mp : Similarity measure for sequential patterns. In *The Australasian Data Mining Conference (AusDM08)*, 2008.

[58] S. Sarawagi, R. Agrawal, and N. Megiddo. Discovery-driven exploration of olap data cubes. In *EDBT*, pages 168–182, 1998.

[59] Einoshin Suzuki. Scheduled discovery of exception rules. In *DS '99 : Proceedings of the Second International Conference on Discovery Science*, pages 184–195. Springer-Verlag, 1999.

[60] Y. Tao, X. Xiao, and J. Pei. Efficient Skyline and Top-k Retrieval in Subspaces. *IEEE Trans. Knowl. Data Eng.*, 19(8) :1072–1088, 2007.

[61] P. Vassiliadis and T. Sellis. A survey of logical Models for OLAP Databases. *SIGMOD Record*, 28(4), 1999.

[62] L. Zadeh. Fuzzy Sets. *Information and Control*, 8 :338–353, 1965.

[63] Mohammed Javeed Zaki. Spade : An efficient algorithm for mining frequent sequences. *Machine Learning*, 42(1/2) :31–60, 2001.

Septième partie

Annexes

Chapitre 14

Liste des publications

Ouvrages et Actes édités

Ed2 S. Bringay, **A. Laurent**, M. Teisseire. *Actes des 5èmes Journées francophones sur les Entrepôts de Données et l'Analyse en ligne (EDA'09).* Numéro spécial de la Revue des nouvelles technologies de l'information (RNTI). Editions Cépaduès. Juin 2009.

Ed1 **A. Laurent**, MJ Lesot. *Scalable Fuzzy Algorithms for Data Management and Analysis : Methods and Design.* IGI Publishing. octobre 2009.

Chapitres de livre

Ch5 H. D. Li, **A. Laurent**, P. Poncelet. Mining Belief-Driven Unexpected Sequential Patterns and Implication Rules. *in Rare Association Rule Mining and Knowledge Discovery : Technologies for Infrequent and Critical Event Detection.* IGI Publishing, 2009.

Ch4 **A. Laurent**. Fuzzy Multidimensional Databases. Series on Computational Intelligence. Special volume dedicated to the memory of Dr. Ashley Morris. Springer Verlag, 2009.

Ch3 YW. Choong, **A. Laurent**, D. Laurent. Summarizing Data Cubes using Blocks. *In Data Mining Patterns : New Methods and Applications.* IDEA Group Inc. 2007

Ch2 **A. Laurent**, P. Poncelet, M. Teisseire. Fuzzy Data Mining for the Semantic Web : Building XML Mediator Schemas. *In Fuzzy Logic and the Semantic Web.* Chapitre 12. pages 249-264. Elsevier. E. Sanchez (ed). 2006.

Ch1 **A. Laurent**, C. Marsala, B. Bouchon-Meunier, Improvement of the Interpretability of Fuzzy Rule Based Systems : Quantifiers, Similarities andAggregators., in *Modelling with Words*, Springer-Verlag series 'Lecture Notes in Artificial Intelligence', LNAI 2873, pages 102-123, J. Lawry, J. Shanahan, A. Ralescu (eds), 2003.

Journaux internationaux avec comité de lecture

IJ18 M. Plantevit, Y.W. Choong, **A. Laurent**, D. Laurent, M. Teisseire. Mining Multidimensional and Multiple-Level Sequential Patterns. ACM Transactions on Knowledge Discovery from Data (ACM TKDD). 2010.

IJ17 H. Li, **A. Laurent**, P. Poncelet. Discovery of Unexpected Recurrence Behaviors in Sequence Databases. *International Journal of Computational Intelligence Research.* 2009.

IJ16 H. Li, **A. Laurent**, P. Poncelet. WebUser : Mining Unexpected Web Usage. *International Journal of Business Intelligence and Data Mining (IJBIDM).* 2009.

IJ15 H. Li, **A. Laurent**, P. Poncelet. Discovering Fuzzy Unexpected Sequences with Semantic Hierarchies. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems (IJUFKS).* 2009.

IJ14 H. Li, **A. Laurent**, P. Poncelet, M. Roche. *Int. Journal on Intelligent Data Analysis (IDA).* 14(1). 2010.

IJ13 C. Fiot, F. Masseglia, **A. Laurent**, M. Teisseire. Evolution Patterns and Gradual Trends. *International Journal of Intelligent Systems.* 2009.

IJ12 M. Plantevit, **A. Laurent**, M. Teisseire. Mining Convergent and Divergent Sequences in Multidimensional Data. *International Journal of Business Intelligence and Data Mining (IJBIDM).* 2009.

IJ11 F. Del Razo, **A. Laurent**, P. Poncelet, M. Teisseire. FTMnodes : Fuzzy Tree Mining Based on Partial Inclusion. *Fuzzy Sets and Systems*, Elsevier, 2009.

IJ10 F. Del Razo, **A. Laurent**, P. Poncelet, M. Teisseire. PIVOT : Equivalence Class-Based Optimized Generation of Candidates for Tree Mining. *Int. Journal on Intelligent Data Analysis (IDA)*, IOS Press, Volume 13(4), 2009.

IJ9 M. Plantevit, **A. Laurent**, M. Teisseire. OLAP-Sequential Mining : Summurizing Trends from Historical Multdimensional Data using Closed Multidimensional Sequential Patterns. *Annals of Information Systems.* special issue on new trends in data warehousing and data analysis. Springer, 2009.

IJ8 C. Fiot, **A. Laurent** and M. Teisseire. Fuzzy Sequential Pattern Mining In Incomplete Databases. *Mathware and Soft Computing.* 2008.

IJ7 C. Fiot, **A. Laurent** et M. Teisseire. Softening the Blow of Frequent Sequence Analysis : Soft Constraints and Temporal Accuracy. *International Journal of Web Engineering and Technology (IJWET)*, special issue on Web-based Knowledge Representation and Management. Vol. 5, No. 1, 2009

IJ6 Yeow Wei Choong, **A. Laurent**, Dominique Laurent. Mining Multiple-Level Fuzzy Blocks from Multidimensional Data. *Fuzzy Sets and Systems*, vol. 159, n¡ 12, June 2008.

IJ5 F. Del Razo Lopez, **A. Laurent**, P. Poncelet, M. Teisseire. Data structures for efficient tree mining : from crisp to soft embedding constraints. in *International Journal of Applied Mathematics and Computer Science.* 2008.

IJ4 C. Fiot, **A. Laurent** et M. Teisseire. From Crispness to Fuzziness : Three Algorithms for Soft Sequential Pattern Mining. in *International Journal IEEE Transactions on Fuzzy Systems.* 2007.

IJ3 S. Jaillet, **A. Laurent**, M. Teisseire. Sequential Patterns for Text Categorization. *International Journal of Intelligent Data Analysis (IDA)*, volume 10(3). 2006.

IJ2 **A. Laurent**, "Generating Fuzzy Summaries : a New Approach based on Fuzzy Multidimensional Databases", *Int. Journal on Intelligent Data Analysis (IDA)*, IOS Press, volume 7, numéro 2. 2003.

IJ1 **A. Laurent**, "Querying Fuzzy Multidimensional Databases : Unary Operators and their Properties", *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems (IJUFKS)*, special issue on Intelligent Information Systems, A. Vila, O. Pons and M. J. Martin-Bautista (eds), World Scientific Publishing. 2003.

Journaux nationaux avec comité de lecture

NJ4 H. Li, **A. Laurent**, P. Poncelet. Extraction de comportements inattendus dans le cadre du Web Usage Mining. in *Revue des Nouvelles Technologies de l'Information (RNTI)*, Cépaduès Editions. 2009.

NJ3 C. Serp, **A. Laurent**, M. Roche, M. Teisseire. La quête du Graal et la réalité numérique. in *Revue CORPUS, numéro sur "Constitution et exploitation des corpus d'ancien et de moyen français"*, numéro 7. 2008.

NJ2 F. Del Razo, **A. Laurent**, M. Teisseire . Une représentation des arborescences pour la recherche de sous-structures fréquentes . in *Revue des Nouvelles Technologies de l'Information*. Numéro spécial Extraction des connaissances : Etat et perspectives. E-5. pp. 299-308. 2006.

NJ1 **A. Laurent**, "FUB et FUB Miner : Deux systèmes pour la représentation, la manipulation et la fouille de données multidimensionnelles floues", revue I3, volume 3(1), Cepadues Editions, 2003.

Conférences et ateliers internationaux avec comité de lecture

IC35 **A. Laurent**, MJ Lesot, M. Rifqi. GRAANK : Exploiting Rank Correlations for Extracting Gradual Dependencies. In Proc. of the Eighth International Conference on Flexible Query Answering Systems (FQAS'09). 2009.

IC34 L. Di Jorio, **A. Laurent**, M. Teisseire. Mining Frequent Gradual Itemsets From Large Databases. Intelligent Data Analysis (IDA'09). 2009.

IC33 H. Saneifar, S. Bonniol, **A. Laurent**, P. Poncelet, M. Roche. Terminology Extraction from Log Files. DEXA'09. 2009.

IC32 S. Bringay, **A. Laurent**, B. Orsetti, P. Salle, M. Teisseire. Handling Fuzzy Gaps in Sequential Patterns : Application to Health. Fuzz'IEEE 2009.

IC31 H. Saneifar, S. Bringay, **A. Laurent**, M. Teisseire. S2MP : Similarity Measure for Sequential Patterns. The Australasian Data Mining Conference (AusDM08). 2008.

IC30 C. Low Kam, **A. Laurent**, M. Teisseire. Detection of Sequential Outliers using a Variable Length Markov Model. Seventh International Confe-

rence on Machine Learning and Applications (ICMLA 2008). IEEE. 2008.

IC29 L. Di Jorio, **A. Laurent**, M. Teisseire. Fast Extraction of Gradual Association Rules : A Heuristic Based Method. IEEE/ACM International Conference on Soft Computing as Transdisciplinary Science and Technology (CSTST). 2008.

IC28 D.H. Li, **A. Laurent**, P. Poncelet. Recognizing Unexpected Recurrence Behaviors with Fuzzy Measures in Sequence Databases. IEEE/ACM International Conference on Soft Computing as Transdisciplinary Science and Technology (CSTST). 2008.

IC27 L. Di Jorio, S. Bringay, C. Fiot, **A. Laurent**, M. Teisseire. Sequential Patterns for Maintaining Ontologies over Time. *ODBASE'08*. LNCS. 2008.

IC26 M. Plantevit, **A. Laurent**, M. Teisseire. Up and Down : Mining Multidimensional Sequential Patterns Using Hierarchies. *DaWaK'08*. 2008.

IC25 D.H. Li, **A. Laurent**, M. Roche and P. Poncelet. Extraction of Opposite Sentiments in Classified Free Format Text Reviews. *DEXA'08*. 2008.

IC24 D.H. Li, **A. Laurent** and P. Poncelet. Discovering fuzzy unexpected sequences with beliefs. *In Proc. IPMU 2008*.

IC23 D.H. Li, **A. Laurent** and P. Poncelet. Mining Unexpected Web Usage Behaviors. *In The 8th Industrial Conference on Data Mining,* Leipzig, Germany, 2008. Best Paper Award.

IC22 C. Fiot, F. Masseglia, **A. Laurent**, M., Teisseire. Ted and Eva : Expressing Temporal Tendencies Among Quantitative Variables Using Fuzzy Sequential Patterns, *Fuzz'IEEE*, 2008.

IC21 C. Fiot, F. Masseglia, **A. Laurent**, M. Teisseire. Gradual trends in fuzzy sequential patterns. *In Proc. IPMU'08*, 2008.

IC20 C. Fiot, G.A. P. Saptawati, **A. Laurent**, M. Teisseire. Learning Bayesian Network Structure from Incomplete Data without any Assumption. *In Proc. DASFAA'08*. LNCS. 2008.

IC19 M. Plantevit, S. Goutier, F. Guisnel, **A. Laurent** and M. Teisseire. Mining Unexpected Multidimensional Rules. *ACM DOLAP'07*, Lisbon, Portugal, 2007.

IC18 D. (Haoyuan) Li, **A. Laurent**, M. Teisseire. On transversal hypergraph enumeration in mining sequential patterns. *In Proc. IDEAS*, pages 303Ð307, Banff, Canada, September 2007.

IC17 C. Fiot, **A. Laurent** and M. Teisseire. Extended Time Constraints for Sequence Mining. in *Proc. of the 14th International Symposium on Temporal Representation and Reasoning (TIME 2007)*. 2007.

IC16 C. Fiot, **A. Laurent** and M. Teisseire. Approximate Sequential Patterns for Incomplete Sequence Database Mining. In *Proc. of the IEEE International Conference on Fuzzy Systems* (FUZZ-IEEE 2007). 2007.

IC15 F. Del Razo Lopez, **A. Laurent**, P. Poncelet and M. Teisseire. Fuzzy Tree Mining : Go Soft on your Nodes. International Conference IFSA'07 (Theory and Applications of Fuzzy Logic and Soft Computing). 2007.

IC14 C. Fiot, **A. Laurent**, M. Teisseire et B. Laurent - Why Fuzzy Sequential Patterns can Help Data Summarization : an Application to the INPI Trademark Database. 15th IEEE International Conference on Fuzzy Systems (FuzzIEEE'06), juillet 2006.

IC13 M. Plantevit, **A. Laurent** and M. Teisseire. HYPE : Mining Hierarchical Sequentiel Patterns slides In *Proc. of the ACM DOLAP'06 Workshop* , Arlington, VA(USA). 2006.

IC12 C. Fiot, **A. Laurent**, M. Teisseire. Web Access Log Mining With Soft Sequential Patterns. *In Proc. of the 7th Internat. FLINS Conference on Applied Artificial Intelligence*. 2006.

IC11 Y.W. Choong, D. Laurent and **A. Laurent**. Pixelizing Data Cubes : a Block-Based Approach. *In Proc. of the Visual Information Expert Workshop (VIEW'06)*. 2006. LNCS 4370. 2007.

IC10 S. Sanchez, **A. Laurent**, P. Poncelet, M. Teisseire. FuzBT : a Binary Approach for Fuzzy Tree Mining. *in Proc. 11th International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems (IPMU'06)*. 2006.

IC9 Y.W. Choong, **A. Laurent**, D. Laurent. Building Fuzzy Blocks from Data Cubes. *in Proc. 11th International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems (IPMU'06)*. 2006.

IC8 F. Del Razo Lopez, **A. Laurent**, P. Poncelet and M. Teisseire. "RSF - A New Tree Mining Approach with an Efficient Data Structure". Proceedings of the joint Conference : 4th Conference of the European Society for Fuzzy Logic and Technology (EUSFLAT 2005) and Eleventh Rencontres Francophones sur la Logique Floue et ses Applications (LFA 2005), special session on Text Mining and Web Mining, pp. 1088-1093. Barcelona, Spain, September, 2005.

IC7 M. Plantevit, Y.W. Choong, **A. Laurent**, D. Laurent, M. Teisseire. M2SP : Mining Sequential Patterns Among Several Dimensions. PKDD'05 : Principles and Practice of Knowledge Discovery in Databases, LNAI 3721, pages 205-216. 2005.

IC6 S. Jaillet, **A. Laurent**, M. Teisseire, J. Chauché, "Order and Mess in Text Categorization : Why Using Sequential Patterns to Classify", Third Workshop on Mining Temporal and Sequential Data, ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining (KDD-2004), 2004.

IC5 Y.W. Choong, P. Maussion, **A. Laurent**, D. Laurent. Summarizing Multidimensional Databases Using Fuzzy Rules in Proc. 10th International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems (IPMU'04). pp 99-106. 2004.

IC4 F. Denis, R. Gilleron, **A. Laurent**, M. Tommasi, "Text Classification and Co-Training from Positive and Unlabeled Examples", in Proc. of the *ICML 2003 Workshop The Continuum from Labeled to Unlabeled Data in Machine Learning and Data Mining*, pp 80-87, Washington, DC, 2003.

IC4 **A. Laurent**, B. Bouchon-Meunier, A. Doucet, "Flexible Unary Multidimensional Queries and their Combinations", *Proc. 9th International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems (IPMU'02)*, 1-5 July, Annecy, France, 2002.

IC3 **A. Laurent**, B. Bouchon-Meunier, A. Doucet, "Towards Fuzzy-OLAP Mining", in *Proc. Workshop PKDD "Database Support for KDD"*, Freiburg, sept. 2001.

IC2 **A. Laurent**, "Generating Fuzzy Summaries from Fuzzy Multidimensional Databases", *Proc. Fourth International Symposium on Intelligent Data Analysis*, Lisbonne, 13-15 sept., 2001, LNCS 2189, pp. 24-33, Springer-Verlag, 2001.

IC1 **A. Laurent**, B. Bouchon-Meunier, A. Doucet, S. Gançarski and C. Marsala, "Fuzzy Data Mining from Multidimensional Databases", *Proc.*

Int. Symposium on Computational Intelligence (ISCI), Kosice, Slovakia, Springer-Verlag, Studies CI series Vol. 54, pp. 278-283, J. Kacprzyk (Ed.), 2000.

Conférences nationales avec comité de lecture

NC26 B. Bouchon-Meunier, **A. Laurent**, et MJ Lesot. Extraction de règles graduelles floues renforcées. *Actes de LFA'09*, 2009.

NC25 P. Salle, S. Bringay **A. Laurent**, et M. Teisseire. Motifs séquentiels et écarts flous. *Actes de LFA'09*, 2009.

NC24 L. Di Jorio, **A. Laurent**, et M. Teisseire. Extraction efficace de règles graduelles. *Actes de EGC'09*, 2009.

NC23 C. Low Kam, **A. Laurent**, et M. Teisseire. Détection de séquences atypiques basée sur un modèle de Markov d'ordre variable. *Actes de EGC'09*, 2009.

NC22 Y. Pitarch, **A. Laurent**, Marc Plantevit et P. Poncelet. Fenêtres sur cubes. In Actes des 24èmes Journées Bases de Données Avancées Octobre 2008.

NC21 C. Serp, E. Cazal, **A. Laurent**, M. Roche. Tervotiq : un système de vote pour l'extraction de la terminologie d'un corpus en français médiéval. *Actes des 9ièmes Journées internationales d'Analyse statistique des Données Textuelles (JADT)*. 2008.

NC20 H.D. Li, **A. Laurent**, et P. Poncelet. Découverte de motifs séquentiels et de règles inattendus. In Actes des 8ièmes Journées Francophones Extraction et Gestion des Connaissances (EGC 2008), pages 535Ð540, Sophia Antipolis, France, January 2008.

NC19 M. Plantevit, **A. Laurent** and M. Teisseire Fouille de données multidimensionnelles : différentes stratégies pour prendre en compte la mesure *Actes Entrepôts de Données et Analyse en Ligne (EDA08)* , 2008.

NC18 M. Plantevit, **A. Laurent**, M. Teisseire Extraction de Motifs Séquentiels Multidimensionnels Sans Gestion d'Ensemble Candidats. *Actes de EGC'08*, 2008.

NC17 C. Fiot, F. Masseglia, **A. Laurent** and M. Teisseire. Séquences, tendances et connaissances. *in Proc. Congrès Informatique des organisations et systèmes d'information et de décision (InforSID'08)*. 2008.

NC16 M. Plantevit, **A. Laurent** and M. Teisseire. Motifs Séquentiels Multidimensionnels Convergents et Divergents slides *In Proc. of the Extraction et Gestion des Connaissances Conference (EGC07),* 2007.

NC15 C. Fiot, **A. Laurent** et M. Teisseire. SPoID : Extraction de motifs séquentiels pour les bases de données incomplètes. *7èmes journées d'Extraction et Gestion des Connaissances (EGC'07).* 2007.

NC14 M. Plantevit, **A. Laurent** and M. Teisseire. HYPE : Prise en compte des hiérarchies lors de l'extraction de motifs séquentiels multidimensionnels slides In Proc. of the EDA'06 : Entrepôts de Données et Analyse en ligne Conference (EDA'06), Versailles (France), June 2006.

NC13 F. Del Razo Lopez, **A. Laurent**, P. Poncelet, M. Teisseire. Recherche de sous-structures fréquentes pour l'intégration de schémas XML. pages 487-498. EGC 2006.

NC12 C. Fiot, **A. Laurent**, M. Teisseire. Des motifs séquentiels généralisés aux contraintes de temps étendues. pages 603-614. EGC 2006.

NC11 M. Plantevit, Y.W. Choong, **A. Laurent**, D. Laurent, M. Teisseire. Motifs séquentiels multidimensionnels étoilés. BDA 2005.

NC10 F. Del Razo Lopez, **A. Laurent**, M. Teisseire.Représentation efficace des arborescences pour la recherche des sous -structures fréquentes.Actes de l'atelier Fouille de données complexes,Conférence Extraction et Gestion des Connaissances (EGC'2005), Janvier 2005, pp 113-120.

NC9 C. Fiot, **A. Laurent**, M. Teisseire.Motifs séquentiels flous : un peu, beaucoup, passionément. in Actes de la Conférence Extraction et Gestion des Connaissances (EGC'2005), Revue des Nouvelles Technologies de l'Information, Cepadues, Janvier 2005, pp 507-518.

NC8 C. Fiot, G. Dray, **A. Laurent**, M. Teisseire, "À la recherche des motifs séquentiels flous", *In Proc. Journées sur la Logique floue et ses applications (LFA'2004)*. 2004.

NC7 S. Jaillet, M. Teisseire, **A. Laurent**, J. Chauché. "Ordre et désordre dans la catégorisation de textes". BDA 2004. Montpellier. 2004.

NC6 Y.W. Choong, **A. Laurent**, D. Laurent, P. Maussion, "Résumés de cubes de données multidimensionnelles à l'aide de règles floues", Actes de la conférence EGC, in Revue des Nouvelles Technologies de l'Information, volume I, Cepaduès Editions, pp 95-106, Janvier 2004.

NC5 **A. Laurent**, B. Bouchon-Meunier, C. Tijus, "Étude cognitive des proportions approximatives d'objets", *Journées du réseau sciences cognitives*, Collège de France, poster, octobre 2002.

NC4 **A. Laurent**, B. Bouchon-Meunier, "Détection des cellules anormalement vides dans les bases de données multidimensionnelles", *Revue Extraction des connaissances et apprentissage (ECA)*, Volume 1, Numéro 4, numéro spécial : Actes des journées francophones d'Extraction et Gestion des Connaissances (EGC 2002), 2002.

NC3 **A. Laurent**, "De l'OLAP Mining au F-OLAP Mining, Représentation de données imparfaites dans les bases de données multidimensionnelles pour la fouille de données", *Revue Extraction des connaissances et apprentissage (ECA)*, Volume 1 - n°1-2/2001, Hermès, numéro spécial : actes des Journées francophones d'Extraction et Gestion des Connaissances (EGC 2001), 2001.

NC2 **A. Laurent**, "Bases de données multidimensionnelles floues", *Actes des 17èmes Journées Bases de Données Avancées*, 29 octobre - 2 novembre 2001, Agadir (Maroc), Cépaduès Editions, pp. 107-117, 2001.

NC1 **A. Laurent**, S. Gançarski, C. Marsala, "Coopération entre un système d'extraction de connaissances floues et un système de gestion de bases de données multidimensionnelles", *Actes des Rencontres Francophones sur la Logique Floue et ses Applications*, La Rochelle, 18-20 oct. 2000, Cepaduès editions, pp 325-332, 2000.

Rapports collectifs

R1 Y.W. Choong, **A. Laurent**, D. Laurent, P. Marcel, F. Ravat, O. Teste, G. Zurfluh, *Entrepôts de données et OLAP : un aperçu orienté recherche*,

rapport du groupe de travail GaFo OLAP, Action Spécifique GaFoDonnées, novembre 2002.

Thèse

T1 **A. Laurent**, *Bases de données multidimensionnelles floues et leur utilisation pour la fouille de données*, thèse de doctorat de l'Université Paris 6, 20 septembre 2002. Rapport LIP6 2002/22.