Extraction de règles d'association selon le couple support-MGK : Graphes implicatifs et Applications en didactique des mathématiques

Parfait Bemarisika

Universite d'Antananarivo

Ecole Doctorale: Problématique de l'Education et Didactique des Disciplines
Equipe d'Accueil: Education et Didactique des Mathématiques et de l'Informatique

THESE

présentée à l'Université d'Antsiranana

pour l'obtention du grade de
Docteur de l'Universite d'Antananarivo
Spécialité: Didactique des Mathématiques et de l'Informatique

par

BEMARISIKA Parfait

Extraction de règles d'association selon le couple support-M_{GK}: Graphes implicatifs et Applications en didactique des mathématiques

soutenue publiquement le 20 avril 2016, devant le jury composé de:

Judith RAZAFIMBELO	Professeur, Université d'Antananarivo	Présidente
Dominique TOURNÈS	Professeur, Université de La Réunion	Rapporteur interne
Jean-Claude RÉGNIER	Professeur, Université de Lyon 2	Rapporteur externe
Victor HARISON	Professeur, Université d'Antananarivo	Rapporteur externe
Jean Emile RAKOTOSON	Professeur, Université de Fianarantsoa	Examinateur
Daniel Rajaonasy FENO	Maître de Conférences, Université de Toamasina	Examinateur
Jean DIATTA	Professeur, Université de La Réunion	Co-Directeur
André TOTOHASINA	Professeur, Université d'Antsiranana	Directeur

Remerciements

Cette thèse a été préparée conjointement au sein du Laboratoire de Mathématiques et d'Informatique-LMI de l'ENSET, Université d'Antsiranana, Madagascar et du Laboratoire d'Informatique et de Mathématiques-LIM, Université de La Réunion, France. Elle a été financée par une allocation de recherche de l'Agence Universitaire de la Francophonie-AUF. Je remercie l'AUF pour ses soutiens financiers pendant mes trois années de thèse.

Je tiens tout d'abord à exprimer mes vifs remerciements à mes directeurs de thèse, Messieurs André TOTOHASINA, Professeur à l'Université d'Antsiranana, Madagascar, Responsable scientifique de l'équipe d'accueil EDMI, et Jean DIATTA, Professeur à l'Université de La Réunion, Directeur du LIM, qui m'ont permis d'effectuer cette thèse sous leur direction. J'ai particulièrement apprécié leur enthousiasme, leurs suggestions et leur soutien moral et scientifique du début à la fin de cette thèse.

Je suis très reconnaissant à mes rapporteurs, Monsieur Dominique TOURNÈS, Professeur à l'Université de La Réunion, Directeur de l'IREM, Monsieur Jean-Claude RÉGNIER, Professeur à l'Université de Lyon 2, et Monsieur Victor HARISON, Professeur à l'Université d'Antananarivo, Directeur Général de l'INSCAE. Je les remercie pour l'attention avec laquelle ils ont lu et évalué ce mémoire ainsi que pour les remarques et critiques constructives qu'ils m'ont adressées.

Je voudrais remercier sincèrement Madame Judith RAZAFIMBELO, Professeur à l'Université d'Antananarivo, Madagascar, Directrice de l'École doctorale PE2Di, pour l'honneur qu'elle me fait d'avoir accepté de présider le jury de cette thèse.

Je tiens également à remercier Monsieur Jean Emile RAKOTOSON, Professeur à l'Université de Fianarantsoa, Madagascar, et Monsieur Daniel Rajaonasy FENO, Maître de Conférences à l'Université de Toamasina, Madagascar, Doyen de la Faculté de Droit, d'Economie et de Gestion, qui ont accepté d'être membres du jury en tant qu'examinateurs.

Je souhaite adresser mes chaleureux remerciements à toute l'équipe du LIM et du LMI, avec laquelle j'ai collaboré, en particulier mon collègue Harrimann RAMANANTSOA, pour ses discussions passionnées sur certains aspects algorithmiques.

Mes remerciements vont aussi à tous mes amis, doctorants ou étudiants Malagasy à l'Université de La Réunion, en particulier Elisa, Rivo et Christelle, pour l'accueil et l'ambiance véritablement chaleureux durant tous mes séjours.

Je ne saurais trouver les justes mots pour remercier ma famille qui m'a toujours soutenu quelle que soit ma situation durant mes longues années d'études, et sans qui ce travail n'aurait pas pu avoir lieu.

Je remercie mille fois ma femme Lucie et nos filles Marina et Juliana, pour leur soutien précieux moral et affectif.

Résumé

Dans ce mémoire, nous nous intéressons au problème de l'extraction des règles d'association positives et négatives valides et ses applications en didactique des mathématiques. L'extraction de telles règles est décomposée en deux étapes qui sont l'extraction des motifs fréquents et la génération des règles d'association pertinentes à partir de ceux-ci. La complexité de chacune de ces deux étapes est très souvent exponentielle: le nombre d'itemsets fréquents potentiels est exponentiel, et le nombre des règles d'association générées peut être excessivement élevé, dû en partie aux mesures de qualité utilisées. En effet, l'immense majorité des approches existantes se limitent uniquement à l'utilisation du couple classique support-confiance, alors que ce couple n'est pas efficace en présence des données fortement corrélées et pour des seuils de support faibles. La découverte des règles d'association pertinentes est donc d'autant plus délicate que de nombreuses règles s'avèrent inintéressantes ou redondantes.

Pour y faire face, nous proposons deux nouveaux modèles permettant respectivement l'extraction optimisée des motifs fréquents, et l'optimisation de la génération des règles d'association potentiellement pertinentes en utilisant le nouveau couple support-M_{GK}. Dans le cadre des graphes implicatifs, la plupart des travaux existants se limitent à l'utilisation unique de l'approche de Gras, reposant sur la mesure *intensité d'implication*, basée sur une approximation gaussienne. Toutefois, l'évaluation de cette approche est assez critiquable dans le sens où ladite mesure de qualité utilisée a tendance à ne plus être discriminante en présence de données denses, à cause de cette approximation encourue, ce qui n'est donc pas à l'abri de perte d'information. De plus, seules les règles d'association positives sont considérées, les règles d'association négatives qui peuvent se révéler une source d'information pertinente pour l'utilisateur sont ignorées, ce qui pose donc un réel problème pour qualifier les résultats obtenus. Pour pallier ces défauts, nous proposons une nouvelle approche de construction de ces graphes implicatifs à l'aide de l'autre mesure plus sélective, M_{GK}, en intégrant à la fois les règles d'association positives et les règles d'association négatives. Nous y avons défini un nouvel algorithme afin d'automatiser la construction. Egalement, nous avons élaboré un nouvel outil, `CHIC-`M_{GK}, permettant entre autres de servir d'appui à la recherche en didactique des mathématiques. Les expérimentations menées sur quelques bases de données de référence, et sur un problème réel de didactique de la statistique à Madagascar montrent la faisabilité notable de nos modèles.

Mots-clés: fouille de données, règles d'association positives et négatives, support-M_{GK}, `CHIC-`M_{GK}, graphes implicatifs, didactique des mathématiques.

Abstract

In this work, we investigate the problem of mining positive and negative association rules and its application in mathematics education. The extraction of this knowledge type is divided into two phases that are mining frequent patterns and generation of association rules from the set of frequent patterns. Very often, the cost of extracting frequent patterns in large and dense contexts is exponential, and the number of association rules generated could be excessively high, most of being not interesting.

We propose a new model for mining frequent patterns, and a new model for generating the interesting association rules, using the couple support-M_{GK}. In the elaboration of the implicative graphs, most of existing methods use one quality measure, implication intensity, based on a gaussian approximation, which is not safe of lossing information. To resolve this defect, we propose a new method in which we use another quality measure, M_{GK}. We propose a new algorithm of implicative graphs. The experiments conducted in data reference, show that our models reduce the costs of inputs/outputs and memory space. The application of these approach in mathematics education has highlighted the practical value of our approach. We propose a new tool `CHIC-`M_{GK} to serve as support for the educational searching activity. The effectiveness of this tool has been shown a real problem of educational statistic involving difficulties of our students in L1 level through proposed exercise solution.

Keywords: data mining, positive and negative association rules, support-M_{GK}, `CHIC-`M_{GK}, implicative graphs, mathematics éducation.

Abréviations et sigles

ECD	Extraction de Connaissances à partir de Données
KDD	Knowledge Discovery from Databases
ASI	Analyse Statistique Implicative
ION	Implicative Orientée Normalisée
CPIR	Conditional Probability Increment Ratio
AVL	Algorithme de Vraisemblance du Lien
EOMF	Extraction Optimisée des Motifs Fréquents
GenPNR	Generation of Positive and Negative association Rules
RAPN	Règles d'Association Positives et Négatives
M_{GK}	Mesure de Guillaume Khenchaff
CHIC	Classification Hiérarchique Implicative et Cohésitive
CHIC-M_{GK}	CHIC selon la mesure M_{GK}
MCV	Maladie cardiovasculaire
SPD	Savoir Pédagogique Disciplinaire
STPD	Savoir Technopédagogique Disciplinaire
UNESCO	United Nations Educational, Scientific and Culturel Organization
AUF	Agence Universitaire de la Francophonie
LMI	Laboratoire de Mathématiques et d'Informatique
LIM	Laboratoire d'Informatique et de Mathématiques
ENSET	Ecole Normale Supérieure pour l'Enseignement Technique
INSCAE	Institut National des Sciences Comptables et de l'Administration d'Entreprises
IREM	Institut de Recherche sur l'Enseignement de Mathématiques
PE2Di	Problématiques de l'Éducation et Didactique des Disciplines
EDMI	Éducation et Didactique de Mathématiques et de l'Informatique
DREN	Direction Régionale de l'Education Nationale
CISCO	Circonsription scolaire
BAC	Baccalauréat de l'enseignement secondaire
BEPC	Brevet d'Etude du Premier Cycle

Notations utilisées

Relations d'ordre	
$<$	ordre strict
$\leq$	ordre large
Quantificateurs	
$\forall$	quantificateur universel
$\exists$	quantificateur existentiel
Ensemblistes divers	
$\mathcal{I}$	ensemble d'attributs
$\mathcal{T}$	ensemble de transactions
$\mathcal{FM}$	famille de Moore
$\mathcal{B}d^+$, $\mathcal{B}d^-$	bordure positive, bordure négative
$\mathcal{GM}$	générateur minimal
$\mathcal{CGM}$	ensemble des candidats générateurs minimaux
$\mathcal{M}$	matrice d'adjacence
$S_{\mathcal{M}}$	ensemble de sommets de la matrice $\mathcal{M}$
$\mathcal{E}_{PNR}$	ensemble des règles positives et négatives valides
$\mathcal{E}_{SRC}$	ensemble des sommets sources
Ensemblistes classiques	
$\cup$	union classique
$\bigcup$	union généralisée
$\cap$	intersection classique
$\bigcap$	intersection généralisée
$\subset$	inclusion stricte
$\subseteq$	inclusion large
$\in$	appartenance
$\setminus$	soustraction
$\lvert\mathcal{I}\rvert$	cardinalité de l'ensemble $\mathcal{I}$
$2^{\lvert\mathcal{I}\rvert}$	ensemble des parties de $\mathcal{I}$
Logiques des prédicats.	
$\vee$	disjonction
$\wedge$	conjonction
$\neg$	négation
$\Rightarrow$ (ou $\rightarrow$)	implication
$\Leftrightarrow$	équivalence

Table des matières

Liste des tableaux

Table des figures

Chapitre 1

Introduction générale

1.1 Contexte et problématique

Les progrès des technologies de l'information offrent, de la dernière décennie, de nombreux moyens pour collecter et stocker une quantité de données extrêmement importante et véhiculent une quantité d'informations prodigieuses dans plusieurs secteurs d'activité, tels que Commerce, Biologie, Médecine, Télécommunication, Contrôle de qualité, didactique disciplinaire, etc. Cependant, l'exploitation optimale de ces masses de données reste encore difficile. C'est pourquoi, au carrefour de thèmes de recherche variés, des différents travaux (Guillet et Hamilton, 2007 [GH07a, GH07b] ; Massaglia et al., 2007 [MPTM07, PTM07] ; Baqueiro et al., 2009 [BWMC09] ; Toussain, 2011 [Tou11] ; Han et Kamber, 2012 [HKP12] ; Bertrand et Diatta, 2013 [BD13] ; Serrano, 2014 [Ser14] ; Ruiz, 2014 [Rui14]) se sont intéressés à l'extraction de ce flot d'information, en intégrant des techniques de fouille des données. Dans ce mémoire, nous nous intéressons principalement à l'optimisation de l'extraction des règles d'association (Agrawal et al., 1993 [AIS93]) plus pertinentes, l'une des techniques les plus populaires de la fouille de données, et montrons leur utilité dans le cadre d'applications réelles, telles que la didactique des mathématiques. Une telle démarche s'inscrit dans une double problématique : (*i*) définir un modèle adéquat pour la fouille des motifs fréquents, et (*ii*) un modèle efficace pour l'extraction de ces règles réellement intéressantes.

Une règle d'association est une quasi-implication entre deux motifs de la forme $X \to Y$, où X et Y sont des motifs disjoints ($X \cap Y = \emptyset$), appelés respectivement la *prémisse* et la *conlusion* de la règle. Cela peut se traduire par « si X, alors Y ». Notons que X et Y peuvent être composés de plusieurs attributs, mais un attribut ne peut pas figurer simultanément dans les deux parties de la règle. Un exemple souvent cité de ce sujet concerne l'*analyse du panier de la ménagère* permettant d'analyser les tickets de caisse des clients afin de comprendre leurs habitudes de consommation, d'agencer les rayons du magasin, d'organiser les promotions, de gérer les stocks, dans le naturel but d'améliorer le profit.

Liée à l'analyse des tableaux croisés 2×2, l'étude portant sur un problème des règles d'association est déjà ancienne. Comme le soulignent Hajek et Rauch [HR99], l'une des premières méthodes de ce sujet est la méthode GUHA [HHC66], où apparaissent déjà les notions de `support` qui est la probabilité qu'une transaction contienne les événements composant la règle d'association, et de `confiance`, qui n'est autre la probabilité conditionnelle d'observer la conclusion sachant qu'on a observé la prémisse. L'algorithme `Apriori` (Agrawal et Srikant, 1994 [AS94]) basé sur le couple support-confiance est le premier modèle efficace qui traite le

problème de l'extraction des règles d'association. C'est un algorithme par niveaux qui s'appuie sur la propriété d'anti-monotonie du support. Il fonctionne sous deux étapes qui sont l'extraction des motifs fréquents, et la génération des règles d'association valides à partir de ceux-ci : un motif est dit fréquent si son support est au moins égal à un support minimum *minsupp* fixé ; une règle est valide lorsque sa confiance dépasse ou égale à une confiance minimale *minconf* fixée. Cet algorithme marque le véritable commencement de la recherche dans le domaine de la fouille des règles d'association, et est aujourd'hui classé dans les dix algorithmes clés en fouille des données (Wu et al. [WKQ+07], Motoda and Ohara [MO09]). Il engendre néanmoins un très grand nombre des règles dont la plupart sont inintéressantes ou redondantes qui ralentissent simplement le temps de réponse pour l'extraction.

Pour y faire face, diverses approches ont été proposées dont nous en citons quelques-unes. L'algorithme `Apriori-TID` proposé par Agrawal lui-même (Agrawal et Srikant [AS94]) cherche à garder le contexte en mémoire pour diminuer l'espace de stockage. Malgré son apport incontestable, l'algorithme parcourt encore plusieurs fois possible la base de données. L'algorithme `Partition` (Savasere et al. [SON95]) partitionne la base entière en sous-bases d'intersection vide pour tenir en mémoire. Il n'effectue que deux passes, et reste facilement parallélisable, mais considère plus de motifs qui se révèlent globalement peu fréquents. L'algorithme `Eclat` (Zaki et al. [ZPOL97]) dédié à la recherche de motifs ensemblistes fréquents parcourt l'espace de recherche en profondeur. L'originalité de son approche est de calculer le support d'un motif en faisant l'intersection des ensembles des transactions contenant ses spécialisations. En revanche, une telle méthode ne permet pas de bénéficier pleinement des capacités de la condition d'élagage. L'algorithme `FP-growth` (Han et al. [HPYM00]) exploite quant à lui une structure de données particulière appelée FP-tree (Frequent-Pattern tree). L'idée est de construire un arbre résumant la base de données et de le parcourir en profondeur afin de générer tous les motifs fréquents. Bien qu'il soit efficace, l'algorithme est assez complexe en terme d'exécution. L'une des propositions assez développée de la dernière décennie est le `résumé d'ensembles` (Mielikäinen et Mannila [MM03], Afrati et al. [AGM04], Yan et al. [YCHX05], Carlos et al. [OEC06], Ndiaye et al. [NDG+10]). Ces approches essayent de limiter l'espace de recherche mais conduisent à une phase d'élagage très complexe, et génèrent un grand nombre de motifs difficilement gérables nécessistant des étapes de traitement très coûteuses. L'étude basée sur les `motifs minimaux` est également très développée (Calders et al. [CRB04], Li et al. [LLW+06], Liu et al. [LLW08], Szathmary et al. [SVNG09]). Ce sont des méthodes en largeur utilisant l'approche support-confiance, elles ont donc des limites similaires à celles d'Apriori. Plus récemment, l'algorithme `DeFMe` (Soulet et Rioult [SR14]) offre une nouvelle méthode pour extraire efficacement les motifs fréquents. C'est un algorithme en profondeur qui étend le concept de fermeture développé dans (Han et al. [HPYM00]). Malgré son apport notable, cet algorithme requiert un nombre polynomial d'opérations qui ralentit la vitesse de compilation.

Même si ces différentes approches ont permis d'améliorer efficacement les performances de l'algorithme historique Apriori, les chercheurs du domaine de l'extraction des règles d'association font généralement face au même obstacle, dû en partie aux mesures de qualité utilisées. En effet, l'immense majorité de ces approches se limitent uniquement à l'utilisation du couple classique support-confiance, alors que ce couple n'est pas efficace en présence des données fortement corrélées et/ou pour des seuils de support faibles qui peuvent être intéressante pour l'utilisateur. Le nombre des règles d'association obtenu est excessivement élevé. La découverte des règles potentiellement pertinentes est donc d'autant plus délicate que de

nombreuses règles s'avèrent inintéressantes ou redondantes. En fait, ce couple ne permet toujours de discerner les règles d'association porteuses de sens, ce qui ne suffit donc pas pour garantir la qualité des résultats.

1.2 Contributions

Dans ce travail, nous avons tenté de répondre aux problématiques exposées ci-dessus. Nos travaux ont donné lieu à plusieurs contributions. Différents modèles mathématiques ont été proposés, en particulier des modèles basés sur le couple support-M_{GK}. Pour chacun de nos modèles, nous nous attachons à développer les algorithmes permettant leur mise en œuvre.

Nous avons tout d'abord réalisé un état de l'art des différents axes de recherche abordés par le sujet. Suite à cela, la première contribution répond au problème majeur levé de cet état de l'art, à savoir la découverte des motifs fréquents dans un contexte transactionnel. De nouveau, le temps de réponse pour l'extraction n'étant pas meilleur lorsque les données sont corrélées ou denses. Pour pallier ce problème, nous proposons une nouvelle approche d'extraction optimisée les motifs fréquents, fondée sur une nouvelle structure de données, notée `MatriceSupport`, et des nouvelles propriétés sur le `générateur minimal` afin de compter efficacement les supports des motifs candidats. Nous proposons également un nouvel algorithme, appelé `Eomf` qui étend l'algorithme historique Apriori (Agrawal et Srikant, 1994 [AS94]), en optimisant les coûts de calcul de l'étape d'extraction. L'algorithme proposé est validé par des expérimentations menées sur quatre jeux de données de référence de la littérature, comparé aux algorithmes sémantiquement proches, tels que Apriori et Pascal. Cette partie a abouti à une publication (Bemarisika et Totohasina, 2016 [BT16a]).

La seconde contribution se focalise sur l'optimisation de la génération des règles d'association réellement pertinentes. A notre connaissance, la plupart des études associées se sont intéressées à l'extraction des règles d'association positives, très peu d'approches traitent les règles d'association négatives. Comme nous l'avons déjà mentionné, un nombre important des règles n'étant pas toujours intéressantes, dû en partie aux mesures de qualité utilisées. En effet, la plupart des travaux existants utilisent le couple classique support-confiance, alors que celui-ci engendre très vite un nombre prohibitif de règles dont plusieurs sont inintéressantes. Pour y faire face, nous proposons une nouvelle méthode de génération des règles d'association positives et négatives potentiellement pertinentes en utilisant le nouveau couple support-M_{GK}, dans laquelle nous introduisons des nouvelles propriétés d'élagage de l'espace de recherche. Nous proposons également un nouvel algorithme afin d'automatiser la génération. Nous comparons notre modèle avec d'autres modèles représentatifs de la littérature, issu de quelques bases de données de référence. Cette partie a abouti à deux publications (Bemarisika et Totohasina, 2014 [BT14c, BT14b]).

La troisième contribution concerne l'élaboration des graphes implicatifs des règles d'association valides. Celle-ci a été introduite pour la première fois dans (Gras, 1979 [Gra79]) et raffiné dans (Gras et al., 1996 [GAB+96]) qui repose sur la mesure de qualité *intensité d'implication* basée sur une approximation gaussienne. L'utilisation de tels graphes (Gras [GSK05], Serge et al. [SPJ+05], Ritschard et al. [RMM09], J.-C. Régnier et Gras [Rég09], Lerman et Pascal [LP09], Matthias et al. [MR10], Lucia et Dusan [LD12]) prend aujourd'hui un essor intense dans la communauté de l'ASI-analyse statistique implicative. Jusqu'à

présent, le graphe implicatif de Gras [GAB+96] est à notre connaissance le seul modèle couramment utilisé au sein de cette communauté de l'ASI. Ce modèle est cependant critiquable : ladite mesure utilisée a tendance à ne plus être discriminante, à cause de cette approximation encourue, en présence des données denses, ce qui n'est donc pas à l'abri de perte d'information. De plus, seules les règles d'association positives sont considérées, les règles d'association négatives qui peuvent se révéler une source d'information pertinente sont ignorées, ce qui pose donc un réel problème pour qualifier les résultats obtenus. Afin de compenser ces limites, nous proposons un nouveau modèle d'élaboration de ces mêmes graphes implicatifs en utilisant une autre mesure de qualité plus sélective, M_{GK}. Nous y définissons un nouvel algorithme afin d'automatiser la construction. Bien sûr, le nouveau modèle proposé intègre à la fois les règles d'association positives et négatives. Cette partie a abouti à une publication (Bemarisika et Totohasina, 2014 [BT14a]).

L'implémentation d'un nouvel outil d'élaboration des graphes implicatifs et applications en didactique des mathématiques constituent notre quatrième contribution. Un rapide survol état de l'art montre que l'élaboration d'un tel outil dans le cadre de la didactique des mathématiques demeure encore un défi majeur. Nous proposons en ce sens un nouvel outil, CHIC-M_{GK} (séquence du logiciel CHIC de Gras [GAB+96], version Couturier, R. 2008 [Cou08]), de représentation en graphe les chaînes implicatives entre les règles d'association valides. L'outil CHIC-M_{GK} implémente deux vues graphiques supplémentaires : la hiérarchie de similarité et la hiérarchie cohésitive. A cet effet, le graphe implicatif repose sur notre algorithme 25 M_{GK}-ImplicativeGraph. La hiérachie de similarité quant à elle repose sur l'algorithme de vraisemblance du lien-AVL (Lerman [Ler81]), tandis que la hiérarchie cohésitive repose à son tour sur l'algorithme de classification hiérarchique orientée (Gras et Kuntz [GK05]), mais avec notre propre programmation. Le verrou principal que nous allons lever ici est tout d'abord d'implémenter le nouvel outil CHIC-M_{GK}, puis de proposer un nouveau modèle d'identification du problème de l'enseignement-apprentissage de la statistique à Madagascar. Par suite, nous effectuons, à l'aide de l'outil CHIC-M_{GK}, des expérimentations menées sur un problème réel de didactique de la statistique, faisant intervenir les difficultés de nos étudiants en L1 dans la solution d'exercice proposé. Aussi, la thèse concrétise un travail novateur sur la formation des enseignants de mathématiques. Cette partie a abouti à deux communications (Bemarisika et al., 2012 [BRTR12] ; Ramanantsoa et al., 2012 [RBATR12]).

1.3 Organisation du mémoire

Ce mémoire est organisé en deux parties reflétant l'ensemble des travaux accomplis de cette thèse. La première partie portant sur état de l'art est divisée en deux chapitres et présente un panorama des travaux de la littérature connexes à notre problématique. La seconde partie vouée aux contributions de la thèse, composée de quatre chapitres, expose l'ensemble de nos résultats et apports réalisés.

Le chapitre 2 propose un tour d'horizon, centré sur notre problématique, du domaine de la fouille de données et de la modélisation des connaissances. Dans ce cas, une description de l'ECD est faite, l'accent est mis sur le cadre théorique de règles d'association. Nous clôturons ce chapitre en abordant un aperçu de quelques domaines d'applications.

Le chapitre 3 réalise un état de l'art autour de la problématique principale de nos travaux,

à savoir l'extraction de règles d'association à sémantique d'implication logique statistique. Nous y dressons un panorama des travaux existants en fouille de règles d'association. Nous nous intéressons plus particulièrement aux techniques d'extraction des motifs fréquents et celles de génération des règles d'association pertinentes à partir de ceux-ci. Nous avons étudié trois groupes d'algorithmes sur les motifs fréquents et deux algorithmes sur la génération des règles pertinentes. Ce chapitre clôt cette partie de l'état de l'art. Quelques remarques y sont tirées et guident les nouvelles modélisations proposées.

Dans la seconde partie, le chapitre 4 détaille le premier apport de nos travaux de thèse. Il s'intéresse comme nous l'avons dit à l'extraction optimisée des motifs fréquents dans une base de données. Le chapitre a été élaboré en fonction des conclusions de notre état de l'art et de façon adaptée à notre problématique. Nous y détaillons, tout d'abord, notre méthode de découverte optimisée des motifs fréquents, et présentons l'algorithme qui en découle. Le chapitre fournit aussi des expérimentations menées sur quelques bases de données de référence en vue de valider les performances de cette méthode proposée. Nous concluons ce chapitre sur un bilan des forces et faiblesses de notre approche.

Le chapitre 5 constitue le cœur de nos travaux de recherches et présente notre seconde contribution sur l'optimisation de la génération des règles d'association positives et négatives valides [1] en utilisant le nouveau couple support-M_{GK}. A cela, nous commençons par présenter tout d'abord les motivations liées à ce chapitre en dégageant les limites du couple classique support-confiance. Ensuite, nous décrivons en détail la modélisation de l'approche proposée. Nous présentons de manière synthétique le nouvel algorithme proposé, GenPNR-Generation of Positive and Negative association Rules. Celui-ci se poursuit à des expérimentations menées sur quelques bases de données de référence, et comparé aux algorithmes sémantiquement proches voire représentatifs de la littérature, afin de le valider. Le chapitre est clôturé par une conclusion décrivant l'intérêt et les limites de notre approche.

Le chapitre 6 s'intéresse à l'élaboration de graphes implicatifs et expose notre troisième contribution proposée dans ce mémoire. Tout d'abord, nous commençons par présenter les notions de base, préliminaires à ce chapitre. Puis, nous présentons de manière synthétique notre approche pour l'élaboration de tels graphes implicatifs. Nous y avons décrit un nouveau modèle du parcours les chemins implicatifs, et défini un nouvel algorithme qui en résulte. L'évaluation de la performance de cette approche est menée sur quelques bases de données de référence de la littérature. En conclusion, nous présentons une synthèse de l'approche proposée et en dégageons quelques perspectives.

Le chapitre 7 s'articule sur la pertinence de l'implantation d'un nouvel outil `CHIC-`M_{GK} et ses applications en didactique des mathémtiques, plus particulièrement aux problèmes de l'enseignement-apprentissage de la statistique à Madagascar. Nous nous y intéressons principalement au paradigme des graphes implicatifs permettant, entre autres, la représentation en graphe les chaînes implicatives entre les règles d'association potentiellement pertinentes. Nous ajoutons comme supplémentaire l'implantation de la hiérarchie de similarité (Lerman, 1981 [Ler81]) et la hiérarchie cohésitive (Gras et Kuntz [GK05]).

Enfin, la conclusion générale présente une synthèse des contributions apportées ainsi que les pistes définissant des perspectives possibles pour de futurs travaux.

1. Le terme "valide" est ici utilisé dans le cadre de fouille de données, qui ne doit donc pas être confondu avec le fait que l'implication (en tant que formule en logique classique) soit valide ou non.

Première partie

Etat de l'art autour de l'extraction de règles d'association

Introduction de la première partie

Cette partie a pour objectif de réaliser un tour d'horizon de l'existant dans les principaux domaines abordés par cette thèse. Le premier chapitre de cet état de l'art (chapitre 2) est centré sur les concepts et approches actuels en Extraction des Connaissances à partir des Données (ECD). L'accent est mis sur la théorie de l'extraction des règles d'ssociation. Le chapitre suivant (chapitre 3) explore quelques modèles algorithmiques de l'extraction de telles règles d'association, à savoir l'extraction de l'ensemble des motifs fréquents et la génération de la famille des règles d'association pertinentes à partir de cet ensemble des motifs fréquents. Pour conclure chacun de ces deux chapitres, nous dressons un bilan des forces et faiblesses des approches explorées, et nous y introduirons chacune de nos contributions en réponse à cet état de l'art.

Chapitre 2

ECD et Fondements mathématiques de l'extraction des règles d'association

2.1 Introduction

Ce chapitre est consacré à un état de l'art sur la théorie de l'extraction des règles d'association valides. Il est structuré comme suit. Après la présentation en bref des étapes du processus d'Extraction des Connaissances à partir de Données-ECD (section 2.2), nous exposons les fondements mathématiques issus de l'Extraction des Règles d'Association-ERA (section 2.3). La section 2.4 donne la conclusion de ce chapitre.

2.2 Processus de l'ECD

Au début des années 90, Piatetsky-Shapiro [PSF91, FPSM92] introduit le terme de "Knowledge Discovery from Databases", abrégé par la suite en KDD, dont l'équivalent français est Extraction de Connaissances à partir de Données (ECD). Celui-ci est une discipline qui recoupe les domaines des bases de données, des statistiques, de l'intelligence artificielle et de l'interface homme/machine. Son objectif est de découvrir automatiquement des informations généralisables en connaissances nouvelles sous le contrôle des experts des données. Cela nécessite la conception et la mise au point de méthodes pour extraire les informations essentielles et cachées qui seront interprétées par les experts afin de les transformer, si possible, en connaissances. Il y a plusieurs définitions pour l'ECD, parmi lesquelles les suivantes sont les plus utilisées par la communauté scientifique.

Définition 1. *L'ECD désigne le processus non trivial d'extraction des informations implicites, de structures inconnues, valides et potentiellement utiles ou exploitables dans des bases de données [PS91, FPSS96b].*

Définition 2. *L'ECD est définie comme un processus interactif et itératif d'analyse d'un grand ensemble de données brutes afin d'en extraire des connaissances exploitables par l'utilisateur [KNZ01]).*

Définition 3. *L'ECD désigne le processus global de découverte de connaissances qui permet de passer de données brutes à des connaissances [Tou11].*

Fayyad [FPSS96a] décrit le processus d'extraction de connaissances à partir de bases de données comme un processus itératif composé de plusieurs étapes. Ce processus suscite un fort intérêt industriel, notamment pour son champ d'application très large, pour son coût de mise en œuvre relativement faible, et surtout pour l'aide qu'il peut apporter à la prise de décision. Cependant, il est rarement possible d'appliquer directement la technique de fouille de données sur les données brutes. Il faut tout d'abord sélectionner le sous-ensemble des données qui peuvent être effectivement intéressantes. Vient ensuite l'étape de prétraitement visant quant à elle à corriger les données manquantes ou erronées. Puis, il faut transformer les données pour qu'elles soient utilisables par l'algorithme de choix. Celui-ci génère un certain nombre de motifs qu'il faut interpréter pour enfin obtenir des nouvelles connaissances. Ce processus peut être découpé en cinq grandes étapes illustrées par la figure 2.1, qui seront expliquées dans les cinq sous-section qui suivent.

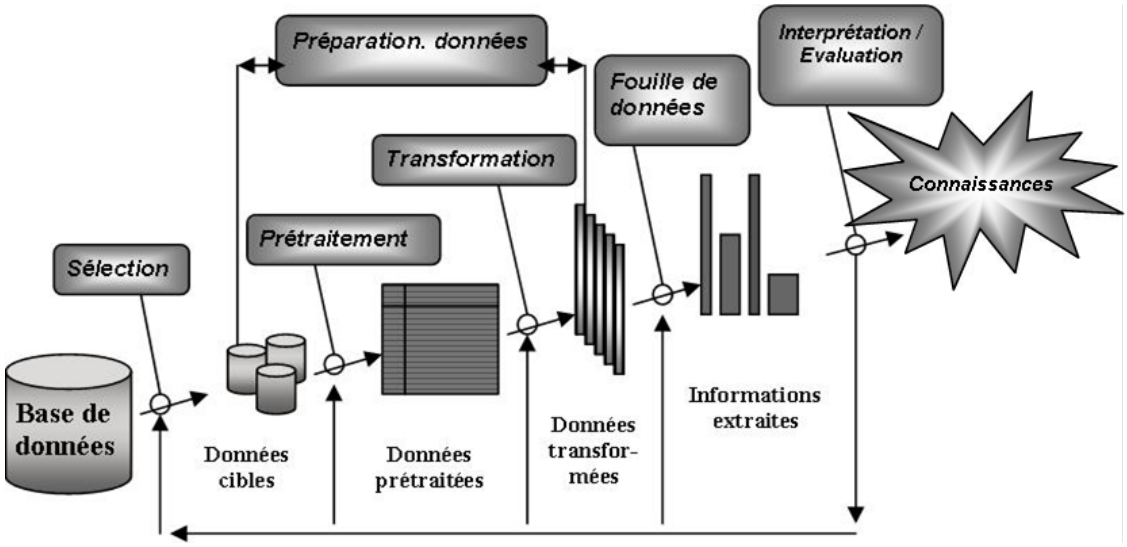

Figure 2.1 – Etapes du processus de l'ECD

2.2.1 Sélection de données

La sélection de données, première étape du processus de l'ECD, permet d'élaborer un jeu d'essai adapté au cadre de l'étude, c'est-à-dire les bases de données qui serviront à l'extraction sont sélectionnées. En effet, il est possible que les données proviennent de diverses sources et soient enregistrées sous différents formats. Cette phase d'acquisition comprend diverses tâches d'intégration et de nettoyage : repérer lors de la sélection les inconsistances, les données trop bruitées, les nettoyer avant de les stocker dans les bases de données ciblées.

2.2.2 Prétraitement de données

La seconde étape est une étape de prétraitement en vue de fabriquer les jeux de données adéquats à l'étape de la fouille. Il s'agira dans ce cas de sélectionner les items appropriés au processus décisionnel en cours, normaliser les données, les agréger, réduire le nombre de

dimensions etc. Cela aura pour conséquence de réduire la taille de la base de données, et donc d'augmenter les chances de succès de l'algorithme de fouille.

2.2.3 Transformation des données

La transformation des variables, troisième étape du processus, présente les données sous la forme exigée par l'algorithme d'extraction de connaissances. Plusieurs algorithmes de data mining sont contraignants sur la forme des données qu'ils acceptent. Cette étape consiste à préparer les données brutes et à les convertir en données appropriées afin de les représenter dans un codage adéquat pour l'application de méthodes de fouille de données.

2.2.4 Fouille de données

La fouille de données est l'étape centrale du processus d'extraction de connaissance. Elle consiste à découvrir de nouveaux modèles au sein de grandes quantités de données. Les différentes tâches de fouille de données peuvent être classées en trois catégories.

Classification supervisée Le modèle de classification est construit en choisissant un échantillon significatif d'objets de la base de données dans lequel chaque objet va être associé à un nom de classe. Le processus de classification se décompose en deux étapes. La première consiste à construire un modèle modélisant l'ensemble d'apprentissage "instances → classes". La seconde correspond à l'utilisation du modèle ainsi construit, c'est-à-dire tester la prédiction du modèle en ciblant le taux d'erreur, puis prédire des nouvelles instances construites. Ce modèle peut être appliqué dans de nombreux domaines, entre autres, en statistiques [AGI+92, HCC92, MST94, EP96, FPSSU96], en réseaux de neurones [Lip87, LSL95].

Classification non supervisée La classification non supervisée (ou segmentation), en anglais "Clustering" consiste à trouver, dans l'ensemble hétérogène d'objets de la base de données, des groupes relativement homogènes appelés "clusters". Il s'agit de grouper les données ayant un haut degré de similitudes au sein de classes. Contrairement à la classification supervisée, les classes y sont construites en fonction d'un ensemble d'observation, mais ne sont pas connues initialement [HK01]. La segmentation se décompose en deux étapes. Il faut tout d'abord découvrir les classes appropriées, puis classifier les données selon les classes découvertes. Ce modèle a couramment été étudié en statistiques [EP96, MST94], en apprentisage numérique [MS83, Fis87], et en analyse de données spatiales [BKSS90, EKX95].

Recherche de règles d'association Introduit par Agrawal et al. [AIS93], la recherche des règles d'association est l'un des sérieux problèmes du KDD. Le principe est de trouver des règles dans les données de types « si `Condition`, alors `Résultats` », notées `Conditions`→ `Résultats`. Cette technique permet la découverte de règles intelligibles et exploitables dans un ensemble de données volumineux, règles exprimant des associations entre items ou attributs dans une base de données. Cetta approche fut développée pour l'analyse de base de données de transactions de ventes, chacune constituée d'une liste d'articles achetés, afin d'identifier les groupes d'articles achetés le plus fréquemment ensemble. Une règle d'association permettra alors de définir une stratégie commerciale ou de marketing dans le but de

promouvoir les ventes, en plaçant par exemple *l'ordinateur* et *l'imprimante* dans le même rayonnage du magasin. Cette problématique constitue la thématique centrale de ce mémoire. Elle sera présentée plus en détails dans le chapitre 3.

2.2.5 Interprétation et évaluation des résultats

L'interprétation et l'évaluation des informations extraites, dernière étape du processus, a pour objectif de produire des connaissances sur le domaine d'études en s'assurant que les conclusions émises correspondent à des phénomènes réels. Cette phase est l'ultime étape de l'ECD. Elle est constituée de l'évaluation, qui mesure l'intérêt des motifs extraits, et de la présentation des résultats à l'utilisateur grâce aux différentes techniques de visualisation. Il s'agit d'une étape de la suppression des connaissances inutiles ou redondantes, et la transformation des connaissances intéressantes en connaissances compréhensibles par l'utilisateur. L'interprétation conduit à la validation ou à la réfutation qui pourrait remettre en cause tous les processus ou une partie du processus de l'ECD [LT04]. Diverses méthodes de validation sont alors envisageables. Les modèles issus d'une classification pourront être vérifiés en premier lieu par un expert, puis la validation sera complétée par des tests statistiques sur des bases de cas existant. Pour des techniques d'apprentissage non supervisées, la détermination de la pertinence des modèles obtenus est essentiellement une affaire d'expertise. Pour ce qui est de la classification supervisée, une validation croisée est récommandable à partir de trois ensembles de données, c'est-à-dire un ensemble d'apprentisage, un ensemble de tests et un ensemble de validations.

2.3 Fondements mathématiques de l'ERA

Nous présentons dans cette section les fondements mathématiques issus de l'Extraction des Règles d'Association-ERA. Nous commençons par décrire une synthèse sur le contexte d'extraction et terminons par un aperçu de quelques domaines d'application.

2.3.1 Contexte d'extraction des règles d'association

Rappelons brièvement tout d'abord les notions de treillis, de famile de Moore, de correspondance de Galois et de fermeture avant de voir le contexte de fouille de données.

Structures de treillis On trouve dans la littérature deux définitions d'un treillis : algébrique et ordinale. Celles-ci introduisent toutes deux la notion de borne supérieure (ou *supremum*) et de borne inférieure (ou *infimum*) : alors qu'il s'agit d'opérateurs binaires dans la définition algébrique, ce sont des éléments particuliers dans la définition ordinale.

Définition 4. (**Définition algébrique**) *[Ded00] Un treillis est un triplet* $\mathcal{L} = (E, \vee, \wedge)$*, où* $\vee$ *et* $\wedge$ *sont deux opérateurs binaires de l'ensemble* E *qui vérifient les propriétés suivantes :*

- *Associativité : pour tous* $x, y, z \in E$, $x \vee y) \vee z = x \vee (y \vee z)$ *et* $(x \wedge y) \wedge z = x \wedge (y \wedge z)$
- *Commutativité : pour tous* $x, y \in E$, $x \vee y = y \vee x$ *et* $x \wedge y = y \wedge x$
- *Idempotence : pour tout* $x \in E$, $x \vee x = x = x \wedge x$
- *Absorption : pour tous* $x, y, z \in E$, $x \vee (x \wedge y) = x = x \wedge (x \vee z)$

Définition 5. (**Définition ordinale**) *Un treillis est une paire $\mathcal{L} = (E, \leq)$, où $\leq$ est une relation d'ordre sur l'ensemble E qui vérifie les propriétés suivantes :*
- *réflexivité : pour tout $x \in E$, on a $x\mathcal{R}x$*
- *antisymétrie : pour tous $x, y \in E$, $x\mathcal{R}y$ et $y\mathcal{R}x$ impliquent $x = y$*
- *transitivité : pour tous $x, y, z \in E$, $x\mathcal{R}y$ et $y\mathcal{R}z$ impliquent $x\mathcal{R}z$*

Toute paire d'éléments $\{x, y\}$ de E admet à la fois une borne inférieure et une borne supérieure. La *borne inférieure* de x et y, notée $x \wedge y$, est l'unique élément maximal (i.e. plus grand élément) de l'ensemble des prédécesseurs (ou minorants) de x et y (ensemble des éléments $z \in E$ tels que $z \leq x$ et $z \leq y$), tandis que la *borne supérieure* de x et y, notée $x \vee y$, est l'unique élément minimal (i.e. plus petit élément) de l'ensemble des successeurs (ou majorants) de x et y (ensemble des éléments $z \in E$ tels que $z \geq x$ et $z \geq y$).

Lorsque seule l'existence de la borne inférieure est vérifiée, on parle d'*inf-demi-treillis*. Un *sup-demi-treillis* est défini dans le cas dual où seule l'existence d'une borne supérieure est vérifiée. L'ensemble des nœuds d'un arbre muni de la relation "est ancêtre de" forme ainsi un sup-demi-treillis où aucune borne inférieure d'éléments incomparables n'existe, alors qu'il forme un inf-demi-treillis lorsqu'on considère la relation "est descendant de".

Définition 6. *Un treillis est dit complet s'il est à la fois inf-demi-treillis complet et sup-demi-treillis complet.*

Thérème 1. *Un sup-demi-treillis (resp. inf-demi-treillis) ayant un plus petit élément (resp. plus grand) est un treillis [Mon03].*

A toute relation d'ordre $\leq$, on associe sa relation d'ordre strict : notée $<$, c'est une relation asymétrique ($(x, y) \in \mathcal{R}$ implique $(y, x) \notin \mathcal{R}$) transitive et irréflexive définie par $x < y$ si $x \leq y$ et $x \neq y$. Il est d'usage d'identifier relation binaire et graphe orienté simple (i.e. un graphe sans arcs multiples) où chaque élément est représenté par un sommet du graphe, et où la relation entre deux éléments x et y est représentée par un arc du graphe entre le sommet x et le sommet y. Le *diagramme de Hasse* est une représentation d'un ordre proche de la représentation habituelle d'un graphe : les nœuds sont positionnés de telle sorte que x sera au-dessous de y lorsque $x \leq y$. Il permet entre autres de ne pas surcharger le dessin pour faciliter une meilleure lisibilité.

Famille de Moore Ce paragraphe présente quelques notions sur les familles de Moore. Des études menées sur les familles de Moore appelée aussi Systèmes de Fermeture permettent d'obtenir qu'elles sont criptomorphes à d'autres notions telles que Opérateurs de Fermeture, Systèmes implificatif, etc. [Dom02].

Définition 7. *Soit E un ensemble. Une famille de Moore $\mathcal{FM}$ sur E est une partie de l'ensemble $\mathcal{P}(E)$ vérifiant les propriétés suivantes (Feno, 2007 [Fen07]) :*

$$(i) : E \in \mathcal{FM}; \qquad (ii) : \mathcal{FM}' \subseteq \mathcal{FM} \Rightarrow \mathcal{FM}' \in \mathcal{FM}$$

Si $\mathcal{FM}$ est un ensemble fini, donc E l'est aussi, l'assertion (ii) peut être remplacée par l'assertion (iii) $F_1, F_2 \in \mathcal{FM}' \subseteq F_1 \cap F_2 \in \mathcal{FM}$. Les éléments de $\mathcal{FM}$ sont appelés aussi les *fermés* de $\mathcal{FM}$. Dans la suite, nous considérons que les familles de Moore finies.

Exemple 1. *Soit* $E = \{A, B, C, D, E\}$. *On a* $\mathcal{FM} = \{\emptyset, A, B, C, D, DE, BCD, ABCDE\}$, *une famille de Moore sur* E. *Ici, les ensembles finis sont notés comme des mots. Par exemple "AE" désigne la partie de l'ensemble* $\{A, E\}$.

Exemple 2. *Soit* E *un ensemble.* $A, B \in E$, *alors* $\mathcal{FM}_{A,B} = \{X \subseteq E \quad ou \quad B \subseteq X\}$, $\mathcal{FM}_{A,B}$, *de sous ensemble de* E, *est une famille de Moore. En particulier, pour* $A = \emptyset$, *on a :* $\mathcal{FM}_{\emptyset,B} = \{X \subseteq E : B \subseteq X\}$ *; pour* $B = \{i\}$, *on a :* $\mathcal{FM}_{A,i} = \{X \subseteq E : A \nsubseteq X où\ i \in X\}$.

Remarque 1. *Notons que* $\mathcal{FM}_{A,B} = \mathcal{FM}_{A,B\setminus A}$. *En effet, si* $X \in \mathcal{FM}_{A,B}$ *alors il est clair que* $X \in \mathcal{FM}_{A,B\setminus A}$. *Réciproquement, si* $X \notin \mathcal{FM}_{A,B\setminus A}$, *i.e.* $A \subseteq B$ *et* $B \nsubseteq X$, *alors* $A \subseteq X$ *et* $B\setminus A \nsubseteq X$, *i.e.* $X \notin \mathcal{FM}_{A,B\setminus A}$. *Par ailleurs, d'après la définition de* $\mathcal{FM}_{A,B}$, *si* F *est un fermé qui contien* A *alors* F *contient aussi* B, *en d'autres termes* A *implique* B. *Ainsi, l'ensemble* $\mathcal{FM}_{A,B}$ *sera dit famille de Moore implicative.*

Rappelons que l'ensemble $(\mathcal{P}(E), \cap, \cup)$ est un treillis. Comme $\mathcal{FM}$ est un sous ensemble de $\mathcal{P}(E)$ stable par intersection, donc $\mathcal{FM}$ est un inf-demi-treillis. Par ailleurs, $\mathcal{FM}$ contient un plus grand élément qui est l'ensemble E, donc la famille de Moore $\mathcal{FM}$ est un treillis, par application du Théorème 1. Ainsi, nous avons le résultat suivant.

Thérème 2. *Soit* $\mathcal{FM}$ *une famille de Moore sur* E. *L'ensemble ordonné* $(\mathcal{FM}, \subseteq)$ *est un treillis, avec les infimums et supremuns définis respectivement :*

$$\forall X, Y \in \mathcal{FM}, X \wedge Y = X \cap Y \text{ et } X \vee Y = \cap\{F \in \mathcal{FM} : X \cup Y \subseteq F\}$$

Correspondances de Galois Ce paragraphe présente les notions de correspondances de Galois associées à une relation binaire, $\mathcal{R}$. Deux fonctions f et g sont ensuite définies :

$$\begin{array}{rcll} \mathcal{P}(E) & \to & \mathcal{P}(F) & \\ X & \mapsto & f(X) = & \{y \in F, \text{ pour tout } x \in X, x\mathcal{R}y\} \\ \mathcal{P}(F) & \to & \mathcal{P}(E) & \\ Y & \mapsto & g(Y) = & \{x \in E, \text{ pour tout } y \in Y, x\mathcal{R}y\} \end{array}$$

Définition 8. *Soient* $(E, \leq), (F, \leq)$ *deux ensembles ordonnées et* $f : E \to F$ *et* $g : F \to E$ *deux applications. Le couple d'applications* (f, g) *sera dit correspondance de Galois entre* E *et* F *si, pour tous* $x, x' \in E, y, y' \in F$, *les trois propriétés suivantes sont vérifiées :*

- $x \leq x'$ *implique* $f(x) \geq f(x')$ *(antitonie) ;*
- $y \leq y'$ *implique* $g(y) \geq g(y')$ *(antitonie) ;*
- $x \leq gof(x)$ *et* $y \leq fog(y)$ *(extensivité).*

Les applications composées $\gamma = fog$ et $\gamma' = gof$ sont des opérateurs de fermeture dans les ensembles ordonnés respectifs $(F, \leq)$ et $(E, \leq)$. De façon intuitive, un treillis est un ensemble ordonné dont tout couple d'éléments admet un infimum et un supremum. On parle alors de treillis de Galois lorsqu'on a affaire à deux treillis auxquels est associé une correspondance de Galois. Le treillis de Galois peut être vu comme un regroupement conceptuel et hiérarchique d'objets, et interprété comme une représentation de toutes les implications entre les attributs (Stumme et al., 2002 [STB+02]).

Proposition 1. *Soient* $(E, \leq), (F, \leq)$ *deux ensembles ordonnés, et que* $f : E \to F$ *et* $g : F \to E$ *deux applications. Le couple* (f, g) *est une correspondance de Galois si et seulement si pour tout* $(x, y) \in E \times F$, *on a* $x \leq g(y) \Leftrightarrow y \leq f(x)$.

Contexte formel Ce paragraphe expose quelques notions sur les contextes formels proprement dits. Un contexte formel définit un ensemble de transactions (ou objets) qui possède des items (ou attributs). Il est représenté sous forme d'un tableau dont les transactions sont en lignes et les attributs en colonnes. Dans ce mémoire, nous utilisons parfois des contextes formels dont la valeur des attributs est booléenne.

Définition 9. *Un contexte transactionnel est un triplet $\mathcal{B} = (\mathcal{I}, \mathcal{R}, \mathcal{T})$ décrivant un ensemble fini $\mathcal{I}$ d'attributs (ou items), un ensemble fini $\mathcal{T}$ de transactions (ou objets), et une relation binaire $\mathcal{R}$ (i.e. $\mathcal{R} \subseteq \mathcal{I} \times \mathcal{T}$) entre $\mathcal{I}$ et $\mathcal{T}$. Un couple $(i, t) \in \mathcal{R}$ dénote le fait que la transaction $t \in \mathcal{T}$ contient l'item $i \in \mathcal{I}$.*

On suppose que les données $\mathcal{B}$ à explorer sont binaires, i.e. qu'on peut décrire chaque transaction au moyen d'un ensemble fini d'items $\mathcal{I} = \{i_1, \ldots, i_m\}$, également appelés attributs. Chaque transaction t sera donc un sous-ensemble de $\mathcal{I}$. De plus, on associe à chaque transaction un identifiant `TID` (Transaction IDentifier) : $\mathcal{T} = \{i_1, \ldots, i_n\}$, c'est-à-dire $\forall (i, t) \in \mathcal{I} \times \mathcal{T}$ tel que $t[i] = 1$ si l'objet i est présent dans t que l'on note par $i\mathcal{R}t$, et $t[i] = 0$ sinon que l'on note par $\neg i\mathcal{R}t$. Dans un contexte d'extraction, les transactions sont généralement dénotées par des nombres et les items par des lettres. Le tableau 2.1 ci-dessous représente un exemple de contexte binaire ayant 5 items $\{A, B, C, D, E\}$ et 6 transactions $\{1, 2, 3, 4, 5, 6\}$.

TID	Attributs
1	ACD
2	BCE
3	ABCE
4	BE
5	ABCE
6	BCE

Tableau 2.1 – Base de données $\mathcal{B}$, $\mathcal{T} = \{1, 2, 3, 4, 5, 6\}$ et $X = \{A, B, C, D, E\}$

Le tableau 2.2 ci-dessous représente également cette même base, mais en mode binaire.

TID	A	B	C	D	E
1	1	0	1	1	0
2	0	1	1	0	1
3	1	1	1	0	1
4	0	1	0	0	1
5	1	1	1	0	1
6	0	1	1	0	1

Tableau 2.2 – Représentation en mode binaire d'une base de données $\mathcal{B}$

Définition 10. *Une base transactionnelle $\mathcal{B}$ est un ensemble de couples formés d'un identificateur de transaction `TID` et de la transaction t proprement dite, telle que :*

$$\mathcal{B} = \{(t, X_t) | t \in \mathcal{T}, X_t \subseteq \mathcal{I}\}$$

Exemple 3. *Le tableau 2.2 représente une base transactionnelle de 6 transactions sur 5 items :* $\mathcal{B} = \{(1, ACD), (2, BCE), (3, ABCE), (4, BE), (5, ABCE), (6, BCE)\}$

Définition 11. *Le terme item est la traduction anglaise d'articles ou d'attributs appartenant à un ensemble fini d'éléments distincts* $\mathcal{I} = \{i_1, \ldots, i_m\}$. *Un élément de* $\mathcal{I}$ *est appelé un item ou attribut.*

Exemple 4. *Dans le tableau 2.2, l'ensemble fini d'éléments* $\mathcal{I} = \{A, B, C, D, E\}$ *contient 5 items* A, B, C, D, E.

Définition 12. *Un motif négatif de* X *est la négation logique de ce motif notée* $\overline{X}$, *désignant l'absence d'au moins un item composant* X, *i.e. la disjonction des absences des items de* X :

$$\overline{X} = \{t \in \mathcal{T} | \exists i \in X : (i, t) \notin \mathcal{R}\}$$

Exemple 5. *Considérons la base de transactions illustrée par le tableau 2.2. Nous avons* $AB = \{3, 5\}$, *alors* $\overline{AB} = \{1, 2, 4, 6\}$.

Définition 13. *Un itemset (ou motif) est un sous-ensemble de* $\mathcal{I}$, *tel que :* $X \subseteq \mathcal{I}$, *où* $\mathcal{I}$ *un ensemble fini d'items. La conjonction* $\{ABC\}$ *indique un itemset composé des trois items.*

Définition 14. *Un sous-ensemble* $X = \{i_1, \ldots, i_k\} \subseteq \mathcal{B}$ *est appelé* k*-itemset, i.e. un itemset avec* k *items, où* k *est la taille (ou longueur) de l'itemset. En particulier, la conjonction* $\{ABC\}$ *est un 3-itemset, donc de longueur 3.*

Définition 15. *Soit le contexte transactionnel* $\mathcal{B} = (\mathcal{I}, \mathcal{R}, \mathcal{T})$. *Les deux fonctions* f *et* g *forment les opérateurs de la connexion de Galois entre les ensembles* $\mathcal{P}(\mathcal{T})$ *et* $\mathcal{P}(\mathcal{I})$, *tels que :*

$$f : \mathcal{P}(\mathcal{T}) \to \mathcal{P}(\mathcal{I}), f(\mathcal{T}) = \{i \in \mathcal{I} | \forall t \in \mathcal{T}, i\mathcal{R}t\}$$
$$g : \mathcal{P}(\mathcal{I}) \to \mathcal{P}(\mathcal{T}), g(\mathcal{I}) = \{t \in \mathcal{T} | \forall i \in \mathcal{I}, i\mathcal{R}t\}$$

où $f(\mathcal{T})$ *dénote l'ensemble de tous les items communs à un groupe de transactions* $\mathcal{T}$ *(intension), et* $g(\mathcal{I})$ *l'ensemble de toutes les transactions partageant les même items de* $\mathcal{I}$ *(extension). Le couple d'applications* (f, g) *définit une correspondance de Galois entre* $\mathcal{I}$ *et* $\mathcal{T}$.

Par exemple, dans la base de données du tableau 2.2, nous avons $f(\{4, 6\}) = \{B, E\}$ et $g(\{A, C\}) = \{1, 3, 5\}$. Ce qui signifie que l'ensemble de transactions $\{4, 6\}$ possède en commun l'ensemble d'attributs $\{B, E\}$. De la même manière, l'ensemble d'attributs $\{A, C\}$ possède en commun l'ensemble de transactions $\{1, 3, 5\}$.

Définition 16. *Les deux applications* f *et* g *vont servir à calculer la fermeture de* X *et* Y *représentant respectivement un sous-ensemble de transactions et un sous-ensemble d'attributs. On dit qu'un ensemble est fermé s'il est égal à sa fermeture [GW99].*

Ainsi, les applications $\gamma = fog$ et $\gamma' = gof$ sont appelées les opérateurs de fermeture de la connexion de Galois [GW99]. Par exemple, dans le contexte $\mathcal{B}$ du tableau 2.2, on a si $X = \{3, 5\}$, alors $f(X) = \{ABCE\}$ et donc $\gamma' = \{3, 5\}$. Si $X = \{1, 2, 3\}$, alors $f(X) = \{C\}$ et donc $\gamma' = \{1, 2, 3, 5, 6\}$. Si $Y = \{AC\}$, alors $g(Y) = \{1, 3, 5\}$ et donc $\gamma = \{AC\}$. Dans ces exemples, les ensembles $\{3, 5\}$ et $\{AC\}$ sont fermés.

A un motif donné, l'opérateur γ, tout comme γ', est caractérisé par le fait qu'il est :

- Isotonie : $\forall X_1, X_2 \subseteq \mathcal{I}$, on a $X_1 \subseteq X_2 \Rightarrow \gamma(X_1) \subseteq \gamma(X_2)$;
- Extensivité : $\forall X \subseteq \mathcal{I}$, on a $X \subseteq \gamma(X)$;
- Idempotence : $\forall X \subseteq \mathcal{I}$, on a $\gamma(\gamma(X)) = \gamma(X)$.

Dans la pratique, le calcul d'une fermeture est très coûteux puisqu'il nécessite de calculer les deux correspondances f et g. Le calcul d'une seule correspondance nécessite à lui seul de parcourir tout le contexte (pouvant contenir un gros volume de données).

Définition 17. *X est un motif fermé si X est le plus grand motif de sa classe d'équivalence. Un motif X est fermé si $X = \gamma(X)$ (Pasquier et al., 1999 [PBTL99c]). Le motif X est un ensemble maximal d'items communs à un ensemble d'objets.*

Exemple 6. *Dans la base de données $\mathcal{B}$ du tableau 2.2, les motifs $\{AB\}$, $\{ABC\}$ et $\{ABCE\}$ sont dans la même classe d'équivalence. Donc, $\{ABCE\}$ est le motif fermé.*

Définition 18. *Générateur minimal (Bastide et al., 2000 [BTP+ 00]) : Un motif X_1 est un générateur minimal d'un motif fermé X, si et seulement si,*

$$\gamma(X_1) = X \text{ et } \nexists X_1' \subset X_1 \text{ tel que } \gamma(X_1') = X$$

Exemple 7. *Dans la base $\mathcal{B}$ du tableau 2.2, le motif $\{AB\}$ est un générateur minimal de $\{ABCE\}$ étant donné que $\gamma(\{AB\}) = \{ABCE\}$ et aucun de ses sous-ensembles propres a le terme $\{ABCE\}$ comme fermeture.*

Ainsi, un motif fermé fréquent apparaît dans le même ensemble d'objets et par conséquent il a le même support que celui de ses générateurs. Il représente donc un ensemble maximal partageant les mêmes items, tandis que ses générateurs minimaux représentent les plus petits éléments décrivant l'ensemble d'objets. De ce fait, nous pouvons dire qu'un motif fermé englobe l'expression la plus spécifique qui décrit les objets qui lui sont associés alors que le générateur contient une des expressions les plus générales (Latiri, 2013 [Lat13]).

Définition 19. *On appelle support d'un motif X, noté $supp(X)$, le rapport entre le nombre de transactions t contenant X ($X \subseteq t$) et le nombre total de transactions dans la base de données $\mathcal{B}$.*

$$supp(X) = P(X') = \frac{|\{t \in \mathcal{T} | X \subseteq t\}|}{|\mathcal{B}|}$$

où $|A|$ désigne la cardinalité d'un ensemble A, tandis que P signifie la probabilité discrète uniforme sur l'espace probabilisable $(\mathcal{T}, \mathcal{P}(\mathcal{T}))$, et X' est l'ensemble de toutes les entités communes à tous les ensembles de X, i.e. $X' = \{t \in \mathcal{T} | \forall i \in X, i\mathcal{R}t\}$, c'est le dual (ou l'intension) d'un motif X de $\mathcal{I}$.

Du tableau 2.2, on a : $supp(A) = \frac{|\{1,3,5\}|}{6} = 1/2$ et $supp(BC) = \frac{|\{2,3,5,6\}|}{6} = 2/3$.

Dans un souci de simplification d'écriture et sans nuire à la compréhension du lecteur, notons par $P(X) = P(X')$, $supp(X \cup Y) = supp(X' \cup Y')$ et $P(Y|X) = P(Y'|X')$.

Propriété 1. $supp(X \cup Y) \leq min(supp(X), supp(Y)) \leq max(supp(X), supp(Y)) \leq supp(X \cap Y)$

Démonstration. Plus un itemset est grand, plus le nombre de transactions le contenant est faible, d'où une diminution de son support. L'union de deux itemsets est l'intersection des transactions les contenant. Cette vérification est évidente, en utilisant le tableau 2.2, on a : $supp(A \cup BC) = \frac{|\{2,5\}|}{6} = 1/3 \leq min(supp(A) = 1/2, supp(BC) = 2/3) = 1/2$. □

Définition 20. *Le cadre mathématique est classique et formalise celui défini par Agrawal [AIS93]. Un itemset X est dit fréquent dans une base de données $\mathcal{B}$ si son support est plus grand qu'un seuil de support minimal minsupp donné :*

$$supp(X) \geq minsupp$$

Un seuil minimal de support *minsupp* est un nombre fixé par l'utilisateur, compris entre 0 et 1, à partir duquel un ensemble d'items est dit fréquent.

Exemple 8. *Dans le tableau 2.2, si l'on fixe le minsupp = 3/6, on obtient que $\{A\}$ et $\{BC\}$ sont fréquents, mais non pas que $\{ABC\}$. En revanche, en choisissant un seuil à 2/6, ces trois motifs sont fréquents.*

Définition 21. *Un motif X est dit fréquent maximal s'il est fréquent et que tous ses sur-ensembles sont non fréquents. Formellement, l'ensemble $\mathcal{F}_m$ des motifs fréquents maximaux d'un contexte $\mathcal{B}$ est défini par :*

$$\mathcal{F}_m = \{X \in \mathcal{I}; \forall Y \supset X, Y \notin \mathcal{I}\}$$

Définition 22. *Un motif fermé est un ensemble maximal de motifs communs à un ensemble d'objets. Un motif $X \subseteq \mathcal{I}$ est dit $\gamma -$ fermé (ou tout simplement fermé) si $\gamma(X) = X$. Il est dit $\gamma -$ fréquent fermé s'il est à la fois $\gamma -$ fréquent et fermé. Formellement, l'ensemble des motifs fréquents fermés d'un contexte $\mathcal{B}$ est défini par :*

$$\mathcal{F}_f = \{X \in \mathcal{I} : \gamma(X) = X \text{ et } supp(X) \geq minsupp\}$$

Pour un motif $X \subseteq \mathcal{I}$, l'image $\gamma(X)$ sera appelée la fermeture de X, qui correspond au plus petit fermé contenant X. L'opérateur de fermeture γ induit une relation d'équivalence sur l'ensemble des motifs fermés fréquents. Les éléments distincts d'une classe d'équivalence donnée apparaissent ainsi dans les mêmes objets et partagent par conséquent la même fermeture et donc le même support. L'unique élément maximal, par rapport à l'inclusion ensembliste, d'une classe d'équivalence est le motif fermé, tandis que les éléments minimaux représentent les générateurs minimaux (Latiri, 2013 [Lat13]). A cet effet, la localisation d'un motif fermé ou d'un générateur minimal nécessite un voisinage restreint, à savoir ses sur-ensembles immédiats et ses sous-ensembles immédiats, respectivement. Il suffit alors de comparer son support avec ceux des motifs du voisinage associé. Par ailleurs, tout motif est nécessairement compris entre un générateur minimal et le motif fermé associé.

Définition 23. *Etant donné un minsupp. La bordure positive $\mathcal{B}d^+$ est l'ensemble des plus grands itemsets fréquents (au sens de l'inclusion) dont tous les sur-ensembles sont fréquents, et est définie comme suit.*

$$\mathcal{B}d^+ = \{X \in \mathcal{I} | supp(X) \geq minsupp, \forall X_1 \supseteq X, supp(X_1) \geq minsupp\}$$

Ce concept joue un rôle très important dans le problème d'extraction des itemsets fréquents maximaux.

Définition 24. *Etant donné un seuil minimum de support minsupp. La bordure négative $\mathcal{B}d^-$ est l'ensemble des plus petits itemsets qui ne sont pas fréquents dont tous les sous-ensembles sont fréquents, et est définie comme suit.*

$$\mathcal{B}d^- = \{X \in \mathcal{I} | supp(X) < minsupp, \forall X_1 \subseteq X, supp(X_1) \geq minsupp\}$$

Définition 25. *Une règle d'association est une quasi-implication logique de la forme $X \rightarrow Y$, où X et Y sont des motifs disjoints ($X, Y \subseteq \mathcal{I}$ et $X \cap Y = \emptyset$) appelés respectivement la premisse et le conséquent de la règle.*

Une règle d'association peut être quantifiée par un ensemble de mesures. Les deux mesures les plus utilisées dans la littérature pour cela sont le *support* et la *confiance*.

Définition 26. *Le support d'une règle d'association, noté $supp(X \rightarrow Y)$, est la proportion de transactions dans la base contenant l'itemset $X \cup Y$, ou encore la probabilité d'observer les événements X et Y dans la base $\mathcal{B}$. Formellement, on a :*

$$supp(X \rightarrow Y) = supp(X \cup Y) = \frac{|\{t \in \mathcal{T} | X \subseteq t, Y \subseteq t\}|}{|\mathcal{B}|}$$

Le support représente en général la portée ou la force de la règle.

Exemple 9. *De la règle $\{AB\} \rightarrow \{C\}$, le support de l'ensemble $\{ABC\}$ étant égal à 2 et le nombre total de transactions étant égal à 6, donc le support de cette règle est $2/6$.*

Définition 27. *La confiance d'une règle associative $X \rightarrow Y$, notée $conf(X \rightarrow Y)$, est le rapport entre le nombre de transactions contenant le motif $X \cup Y$ et celui contenant X, qui est définie par :*

$$conf(X \rightarrow Y) = P(Y|X) = \frac{supp(X \cup Y)}{supp(X)}$$

La confiance peut être assimilée à la probabilité conditionnelle $P(Y|X)$, c'est-à-dire la probabilité d'avoir Y sachant X. Elle est parfois utilisée comme un indicateur de précision de l'implication entre les motifs X et Y (Agrawal et al., 1993 [AIS93]).

Exemple 10. *Dans le tableau 2.2, la confiance de la règle $AB \rightarrow C$ est obtenue en divisant le support de l'itemset $\{AB \cup C\}$ par le support de l'itemset $\{AB\}$: $conf(AB \rightarrow C) = 2/2 = 1$.*

Définition 28. *Etant fixé un $minsupp$ et une $minconf$, une règle d'association $X \rightarrow Y$ est valide au sens du couple support-confiance si : $supp(X \rightarrow Y) \geq minsupp$ et $conf(X \rightarrow Y) \geq minconf$.*

Au sens du support-confiance, une telle règle d'association est dite valide si sa valeur de confiance, i.e., $conf(X \rightarrow Y)$, est supérieure ou égale à un seuil prédéfini noté par $minconf$. Ce seuil de confiance minimal est utilisé pour exclure les règles dites non valides. Par ailleurs, le seuil de support minimal $minsupp$ est utilisé pour écarter les règles d'association qui ne sont pas suffisamment fréquentes, i.e., les règles ayant un support inférieur à $minsupp$.

Exemple 11. *Dans le tableau 2.2, si nous fixons les seuils de minsupp et de $minconf$, respectivement, à 0.3 et 0.8, la règle $AB \rightarrow C$, est valide au sens support-confiance, car $supp(AB \rightarrow C) = 2/6 = 0.33 > 0.3\%$ et $conf(AB \rightarrow C) = 1 > 0.8$*

Dans la littérature, deux types de règles d'association sont définis à savoir : les règles exactes et les règles approximatives.

Définition 29. *Une règle d'association $X \rightarrow Y$ est dite support-confiance exacte si son indice de confiance est égal à 1 (Zaki, 2004 [Zak04]).*

Définition 30. *Une règle d'association $X \rightarrow Y$ est dite support-confiance approximative si son indice de confiance n'est pas égal à 1 (Zaki, 2004 [Zak04]).*

Ces deux types d'association s'identifient par deux propriétés différentes [Zak04] :

Propriété 2. *Une règle d'association exacte est de la forme $X_1 \rightarrow X2 \backslash X_1$ et représente une implication entre deux motifs fréquents X_1 et X_2 tel que $X_1 \subseteq X_2$ et X_1 et X_2 ont des fermetures identiques, i.e., $\gamma(X_1) = \gamma(X_2)$.*

Propriété 3. *Une règle d'association approximative est de la forme $X_1 \rightarrow X2 \backslash X_1$ et représente une implication entre deux itemsets fréquents X_1 et X_2, tel que $X_1 \subseteq X_2$ et la fermeture de X_1 est un sous-ensemble strict de la fermeture de X_2, i.e., $\gamma(X_1) \subset \gamma(X_2)$.*

Exemple 12. *Si l'on prend notre exemple précédent, la règle $conf(AB \rightarrow ABCE \backslash AB)$ est dite exacte avec une confiance égale à 1, puisque $\gamma(AB) = \gamma(ABCE) = \{ABCE\}$. Tandis que $conf(A \rightarrow B)$ est approximative puisque sa confiance $\frac{2}{3}$ est différente de 1.*

La découverte des règles d'association à partir des motifs fermés fréquents se fait de la façon suivante. Pour chaque motif fréquent X, on génère les divers sous-ensembles non vides de X. Ensuite, pour chaque sous-ensemble X_1 de X, une règle de la forme $X_1 \rightarrow X \backslash X_1$ est retenue si le rapport $supp(X)/supp(X_1)$ est au moins égal à $minconf$. Dans le cadre des motifs fermés fréquents, certains travaux ([LPBT99, VMHG03]) se sont intéressés à la génération des ensembles non redondants.

2.3.2 Quelques domaines d'applications

Les règles d'association [AIS93] sont traditionnellement utilisées pour « l'analyse du panier de la ménagère » dans le secteur de la distribution. De nos jours, leurs domaines d'application sont multiples dont les principaux sont les suivants.

- `Planification commerciale` [AIS93, FPSSU96, BHB+09, PBG11]. Les règles d'association permettent aux sociétés de vente par correspondance de déterminer quels articles il est préférable de placer sur la même page d'un catalogue afin d'identifier quels articles en promotion pouront inciter les clients à effectuer d'autres achats. Dans le cas de transactions de ventes dans lesquelles le client est identifié, les règles d'association permettent de définir les catalogues personnalisés.

- `Finances et Réseaux de télécommunications` [SC09, SN10]. Les règles d'association sont utilisées avec succès au filtrage des alarmes non informatives à l'identification des causes d'anomalies, à la détection et la prédiction d'incidents dans les processus de télémaintenance [HKM+96, KMT97].

- `Recherche médicale et Biologique` [MYGS91, OO98, AS09, SHFM09, VTL10]. La plupart des organismes médicaux (hôpitaux, laboratoires d'analyse, cabinet médicaux,...) stockent systématiquement les informations relatives à leurs patients dans des bases de données. Dans ce cadre, l'extraction de règles d'association permet d'apporter par exemple une aide au diagnostic en identifiant les symptômes, l'identification de population à risque vis à vis de certaines maladies. Les règles d'association sont utilisées dans le cadre de la prédiction de résultats médicaux. Un grand nombre de phénomènes biologiques sont naturellement modélisés par une approche des règles d'association.

- `Analyse de données spatiales`. Les bases de données spatiales sont largement utilisées dans les systèmes d'information géographiques, en cartographie, en astronomie et pour les études environnementales. Elles stockent des informations relatives aux objets occupant un espace. Par exemple, dans le système GeoMiner, les règles d'association ont été utilisées, notamment, pour l'aide à la prédiction d'événements naturels et à l'aménagement du territoire, à la prévision météorologique, aux études biologiques, et à la recherche démographique et géographique.

- `Multimédia et internet` [Fau07, BWMC09, Rui14]. Des quantités croissantes de données de divers types (images, vidéo, etc.) sont stockées dans des bases de données dont le nombre ne cessent d'augmenter. L'extraction des règles d'association a donné lieu à de nombreuses études, principalement dans le cadre de l'analyse d'images. Les applications concernent, entre autres, la reconnaissance militaire, le filtrage des données parasites, la prévision météorologique, l'imagerie médicale, l'aide dans les enquêtes criminelles. De même, un grand nombre de ressources sont accessibles par les réseaux internets. La taille et le nombre croissants des sites Internets entrainent d'importants besoins d'outils pour la réorganisation de ces sites en fonction des cheminements des usagers : l'aide à la navigation dans les systèmes de gestion d'informations. L'extraction de règles d'association à partir des historiques des accès par les usagers aux ressources des sites Internets ont été utilisées dans ce cadre pour l'aide à la conception et l'organisation des sites.

- `Analyse de données statistiques` [WWR+09, IPB+09, Idi13, Ser14]. L'analyse de données statistiques constitue un défi important pour le KDD. L'intérêt tient au nombre d'applications pouvant bénéficier de l'analyse des données statistiques qu'elles utilisent. Les organismes financiers, les administrations (résultats de recensements, de sondages,...) sont parmi les exemples utilisant les données statistiques. L'analyse de ces données constitue une part importante de l'activité de ces organismes dont les règles d'association peuvent constituer des indicateurs utiles dans ce cadre.

- `didactique disciplinaire` Plus précisément, en didactique des mathématiques, l'approche des règles d'association est couramment utilisée dans la communauté de l'ASI-analyse statistique implicative (Serge et al. [SPJ+05], Ritschard et al. [RMM09], J.-C. Régnier et Gras [Rég09], Lerman et Pascal [LP09], Matthias et al. [MR10], Lucia et Dusan [LD12]). Elle permet de représenter l'enchaînement d'idées des élèves, voire identifier leurs difficultés lors d'une résolution d'un exercice donné, afin de donner des prescriptions pédagogiques.

2.4 Conclusion partielle

Dans ce chapitre, nous avons introduit les différents concepts nécessaires à la compréhension de la suite de ce manuscrit, qui se positionne dans le cadre de l'extraction des règles d'association dans un contexte transactionnel. A travers ce premier état de l'art, nous avons pu nous familiariser dans un premier temps le processus de l'extraction de connaissances à partir de données (ECD). Nous avons, par la suite, présenté les différentes théories issues de l'extraction des règles d'association. L'ensemble de ces observations seront prises en compte lors de l'élaboration de nos approches présentées dans la partie II de ce manuscrit. Nous allons continuer cet état de l'art dans le chapitre 3 en présentant quelques approches algorithmiques d'extraction des règles d'association valides.

Chapitre 3

Extraction de règles d'association

3.1 Introduction

Ce chapitre dresse un état de l'art sur les approches en extraction des règles d'association (Agrawal et al., 1993 [AIS93]). Le problème de l'extraction de telles règles est abordé pour la première fois par l'algorithme Apriori (Agrawal et Srikant, 1994 [AS94]). Cependant, cette approche présente 3 problèmes majeurs : (*i*) le coût de l'extraction des motifs fréquents, notamment pour les contextes denses, est exponentiel ; (*ii*) le nombre de règles d'association générées peut être excessivement élevé ; (*iii*) les règles d'association produites sont en grande partie redondantes ou inintéressantes. Ce constat a poussé la communauté à s'intéresser à la recherche des nouvelles méthodes plus efficaces. Ainsi, nous décrivons dans ce chapitre un aperçu sur les principaux algorithmes d'extraction des motifs fréquents, et de génération des règles d'association, en nous réservant une grande place à l'algorithme Apriori [AS94] avec son principe et ses limites. Ce chapitre est structuré comme suit. La section 3.2 présente les principales approches sur l'extraction des motifs fréquents. Nous nous présentons les algorithmes d'extraction d'itemsets fréquents (sous-section 3.2.1), les algorithmes d'extraction d'itemsets fréquents maximaux (sous-section 3.2.2) et les algorithmes d'extraction d'itemsets fermés fréquents (sous-section 3.2.3). La section 3.3 traite l'algorithme de génération des règles d'association. La section 3.4 est dédiée à la conclusion partielle.

3.2 Extraction d'itemsets fréquents

L'extraction des motifs fréquents est la première étape de l'extraction des règles d'association. Elle consiste à extraire le contexte de l'ensemble d'attributs binaires $\mathcal{I}$. Le problème de recherche des motifs fréquents associés aux données $(\mathcal{I}, \mathcal{R}, \mathcal{T})$ consiste à déterminer le sous-ensemble $X_k \subset X$ des motifs fréquents ainsi que le support de chaque motif fréquent. Les algorithmes de recherche de ces motifs doivent parcourir la totalité de la base de données chaque fois qu'ils ont déterminé le support de motifs candidats. Dans la plupart de cas, l'espace de recherche est exponentiel, de l'ordre de $2^{|\mathcal{I}|}$ itemsets candidats. Afin de limiter cet espace de recherche, les algorithmes reposent sur les propriétés 4 et 5 d'anti-monotonie ci-après. L'intérêt de ces propriétés repose particulièrement par le fait qu'elles permettent d'affirmer que *si un motif de taille n n'est pas fréquent, alors aucun de ses sur-motifs ne le seront non plus*. Cela permet de ne pas tester ou même générer les sur-motifs d'un motif non fréquent.

3.2.1 Algorithmes d'extraction d'itemsets fréquents

Dans la littérature, il y a un large éventail d'algorithmes permettant l'extraction des motifs fréquents dans une base de données transactionnelles. Nous en présentons un panorama loin d'être exhaustif, en identifiant l'avantage et l'inconvénient de chacun d'eux.

Algorithme APRIORI Nous présentons ici l'algorithme historique de découverte des motifs fréquents, Apriori (Agrawal et Srikant, 1994 [AS94]). Apriori est le premier algorithme qui résout le problème de la découverte des motifs fréquents. Il procède en deux phases : (i) recherche des motifs fréquents, ceux dont le support est plus grand que le support minimum *minsupp* choisi par l'utilisateur ; (ii) pour chaque itemset fréquent X, on conserve les règles de type $X \backslash Y \rightarrow Y$, avec $Y \supset X$, dont ladite "Confiance" dépasse le seuil $minconf$. Apriori est un algorithme itératif de recherche d'itemsets fréquents par niveaux. Son idée générale est de générer, à chaque itération k, un ensemble d'itemsets potentiels. Un balayage est réalisé pour élaguer les itemsets non fréquents : les k-itemsets fréquents obtenus sont réutilisés lors de l'itération $(k+1)$. A chaque itération k, l'algorithme effectue un passage dans la base de transactions pour calculer le support de chaque k-itemset. Il utilise une représentation horizontale de la base de données dont les lignes (resp. colonnes) sont représentées par les transactions (resp. motifs). Afin de limiter le nombre de candidats à générer et de réduire dynamiquement l'espace de recherche, l'algorithme exploite les propriétés 4 et 5 d'anti-monotonicité ci-après.

Propriété 4. *Tout sous-ensemble d'un itemset fréquent est fréquent.*

Cette propriété permet de limiter le nombre de candidats de taille k générés lors de la k^{e} itération en réalisant une jointure conditionnelle des itemsets fréquents de taille $(k-1)$ découverts lors de l'itération précédente.

Propriété 5. *Tout sur-ensemble d'un itemset non fréquent est aussi non fréquent.*

Cette propriété permet de supprimer un candidat de taille k lorsqu'au moins un de ses sous-ensembles de taille $(k-1)$ ne fait pas partie des itemsets fréquents découverts lors de l'itération précédente.

L'algorithme Apriori reçoit un contexte d'extraction $\mathcal{B}$ et un seuil de support minimum *minsupp* comme paramètres. Il fournit en sortie l'ensemble L_k des motifs fréquents. Son pseudo-code est présenté dans l'algorithme 1 ci-après. Dans ce cas, notons C_k l'ensemble des k-itemsets candidats.

Durant la première itération, tous les 1-itemsets sont considérés et un balayage du contexte $\mathcal{B}$ est réalisé afin de déterminer l'ensemble F_1 des 1-motifs fréquents (ligne 1). Chaque itération k suivante (lignes 2 à 10) se subdivise en deux phases. Durant la première phase (ligne 3), l'ensemble C_k des k-itemsets candidats est construit en joignant les $(k-1)$-itemsets fréquents dans l'ensemble L_{k-1}. Cette phase est réalisée par la procédure Apriori-Gen. Durant la deuxième phase (lignes 4 à 10), un balayage du contexte est réalisé afin de déterminer le support de chacun des k-itemsets candidats dans C_k et les k-itemsets fréquents sont insérés dans l'ensemble L_k. Lors de ce balayage, pour chaque transaction t du contexte $\mathcal{B}$, l'ensemble C_t des k-itemsets candidats est déterminé par la procédure Subset (ligne 5) et le support de chacun des itemsets est incrémenté (ligne 7). Ces itérations cessent lorsqu'aucun nouveau candidat n'est généré ($L_{k-1} = \emptyset$).

Algorithm 1 Algorithme APRIORI

Require: Contexte $\mathcal{B}$; seuil minimum de support *minsupp*.
Ensure: Ensemble L_k des k-itemsets fréquents.

$L_1 \leftarrow$ {1-itemsets fréquents}
for $(k \leftarrow 2; L_{k-1} \neq \emptyset; k++)$ **do**
  $C_k \leftarrow$ Apriori-Gen(L_{k-1})
  for all (instance $t \in \mathcal{B}$) **do**
    $C_t \leftarrow Subset(C_k, t)$
    for all (candidat $c \in C_t$) **do**
      $c.support++$
    end for
    $L_k \leftarrow \{c \in C_k | c.support \geq minsupp\}$
  end for
  Retourner $\bigcup_k L_k$
end for

Algorithm 2 Procédure Apriori-Gen

Require: Ensemble L_{k-1} de (k-1)itemsets fréquents
Ensure: Ensemble C_k de k-itemsets candidats

insert to C_k
select $p.item_1, \ldots, p.item_{k-1}, q.item_{k-1}$
from $L_k p, L_k q$
where $p.item_1 = q.item_1, \ldots, p.item_{k-2} = p.item_{k-2}, p.item_{k-1} < q.item_{k-1}$
for all (itemsets $c \in C_k$) **do**
  for all ($(k-1)$ subsets s of C) **do**
    if $(s \notin L_{k-1})$ **then**
      delete c from C_k
    end if
  end for
end for
Retourner C_k

Apriori-Gen La procédure Apriori-Gen prend comme paramètre l'ensemble L_{k-1} des motifs fréquents de $(k-1)$-attributs, et retourne l'ensemble C_k des k-motifs candidats. Le pseudo-code de cette procédure est présenté dans l'algorithme 2 ci-dessus. La génération d'un candidat s'effectue en deux étapes : la première étape est une étape de `jointure` entre deux motifs de taille $(k-1)$ qui ont $(k-2)$ attributs en commun. Par exemple, la jointure des motifs ABC et ABD donne $ABCD$, par contre, la jointure de ABC et CDE ne donne rien, car il n'y a pas $(k-2)$ attributs en commun. La seconde étape est une étape d'`élagage`. Pour chaque k-motifs M_k générés par l'étape de jointure, il faut tester si tous les $(k-1)$-motifs $M_{k-1} \subseteq M_k$ sont fréquents, i.e. s'ils sont tous présents dans L_{k-1}. Par exemple, prenons $L_3 = \{ABC, ABD, ACD, ACE, BCD\}$, après l'étape de jointure $C_4 = \{ABCD, ACDE\}$. L'étape d'élagage supprime le motif $ACDE$, car son sous-motif ADE n'est pas présent dans L_3. A la fin de l'exécution de `Apriori-gen(`L_3`)`, on a $C_4 = \{ABCD\}$. La fonction `Subset`

prend pour paramètre l'ensemble des candidats C_k et une transaction t et elle retourne le sous-ensemble de motifs $c \in C_k$ présents dans t. Cela permet de calculer la fréquence exacte de chaque candidat (lignes 8 et 9), et ensuite, seul les motifs fréquents sont conservés (ligne 11) pour le prochain pas de la boucle principale.

Fonction Subset La fonction Subset calcule le sous-ensemble $C_t \subseteq C_k$ qui correspond à des sous-ensembles présents dans les transactions contenues dans $\mathcal{B}$. Le pseudo-code de cette fonction est présenté dans l'algorithme 3 ci-dessous.

Algorithm 3 Fonction Subset

Require: Ensemble L_{k-1} de $(k-1)$-itemsets fréquents ; Un candidat c.
Ensure: Un booléen

```
1: for all (sous-ensemble s de c, de taille k − 1) do
2:     if (s ∉ L_{k−1}) then
3:         Retourner TRUE
4:     end if
5:     Retourner FALSE
6: end for
```

La fonction s'assure, après avoir généré un candidat de taille k à partir de $(k-1)$-itemsets fréquents, que le nouveau candidat ne contient pas un sous-ensemble peu fréquent auquel cas le candidat lui-même serait peu fréquent selon la propriété d'antimonotonicité. Une fois l'ensemble C_k des candidats de taille k est calculé, la base de transactions est parcourue transaction par transaction afin de déterminer le support de chaque candidat. La fonction sous-ensemble $(C_k; t)$ recherche parmi les candidats de C_k ceux qui sont contenus dans la transaction t. Si c'est le cas, alors le support de ces candidats est augmenté (ligne 7 de l'algorithme 1). La recherche est optimisée en utilisant un arbre de hachage pour stocker les itemsets candidats. Parmi les candidats, seuls les candidats fréquents, i.e. de support suffisant, sont gardés dans l'ensemble L_k (ligne 9 de l'algorithme 1). L'ensemble L_k de tous les itemsets fréquents est ensuite mis à jour. C'est cet ensemble qui est retourné (ligne 11 de l'algorithme 1) à la fin de l'étape d'extraction des itemsets fréquents.

Exemple 13. *L'exemple d'exécution de l'algorithme Apriori sur la base des données $\mathcal{B}$, pour un minsupp $= 2/6$, est représenté sur la figure 3.1 ci-dessous.*

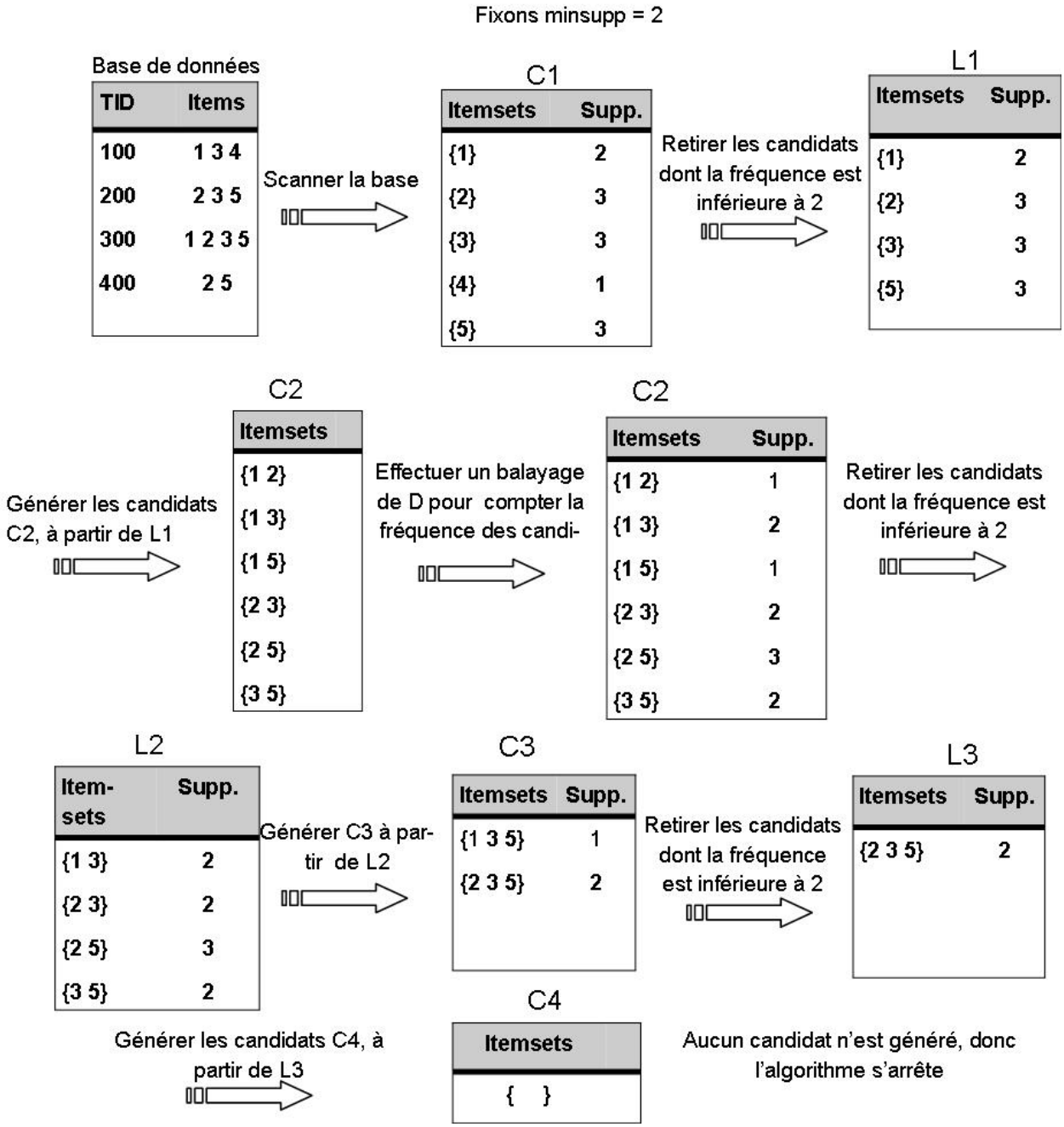

Figure 3.1 – Exemple d'exécution d'APRIORI à l'extraction des itemsets fréquents

A la première itération, chaque item de $\mathcal{I}$ est un 1-itemset de C_1. Un premier parcours de $\mathcal{B}$ permet de trouver le support de chaque 1-itemset. Ainsi, l'itemset $\{4\}$ est retiré en L_1, car son support minimal est inférieur à 2. Afin de découvrir les 2-itemsets fréquents, Apriori effectue dans la seconde itération une jointure de $L_1 \otimes L_1$ pour trouver l'ensemble C_2 des candidats de taille 2. Seuls les 2-candidats n'ayant pas de sous-ensembles peu fréquents sont

gardés. Un second parcours de $\mathcal{B}$ est alors effectué pour déterminer le support de chacun des 2-itemsets candidats, seuls les 2-itemsets fréquents sont gardés dans L_2. Ainsi l'itemset $\{12\}$ et $\{15\}$ n'ayant pas de support suffisant sont supprimés. Les 3-itemsets sont obtenus en combinant les itemsets de L_2 deux à deux, i.e. par jointure $L_2 \otimes L_2$. Seuls les 2-itemsets ayant le même préfixe de taille 1 sont générés. Par exemple les 2-itemsets $\{23\}$ et $\{25\}$ forment le candidat $\{235\}$. On s'assure également que les candidats générés n'ont pas de sous-ensembles peu fréquents. Un troisième parcours de $\mathcal{B}$ est alors effectué pour déterminer les 3-itemsets fréquents. De nouveau, on effectue la jointure de $L_3 \otimes L_3$ pour trouver l'ensemble C_4 des candidats de taille 4, qui est dans ce cas vide, car on n'a plus qu'un seul élément de taille 3. L'algorithme s'arrête après avoir trouvé tous les itemsets fréquents.

L'algorithme Apriori (Agrawal et Srikant, 1994 [AS94]) est relativement efficace, toutefois ses performences diminuent terriblement en présence de données denses et des supports relativement faibles. Plusieurs approches ont été proposées pour compenser ces limites. Nous citons, entre autres, APRIORI-TID [AS94], PARTITION [SON95], ECLAT [ZPOL97], Max-CLIQUE et MaxECLAT [ZPOL97], Max-MINER [Bay98], CLOSE [PBTL98, PBTL99c], A-CLOSE [PBTL99a] et FP-Growth [HPYM00].

Algorithme APRIORI-TID L'algorithme Apriori-TID (Agrawal et Srikant, 1994 [AS94]) est une variante de l'algorithme Apriori. Il permet de diminuer la taille du contexte afin de le stocker en mémoire. L'algorithme cherche à garder le contexte en mémoire afin de limiter les accès répétitifs. Son pseudo-code est présenté dans l'algorithme 4. A cet effet, notons par $\mathcal{C}_k$ l'ensemble des k-motifs candidats, par L_k l'ensemble des k-motifs fréquents.

Algorithm 4 Algorithme APRIORI-TID

Require: Contexte $\mathcal{B}$; seuil minimal de support *minsupp*.
Ensure: Ensemble L_k des k-itemsets fréquents.

$L_1 \leftarrow$ {1-itemsets fréquents} ;
$\overline{C}_1 \leftarrow \mathcal{B}$;
for $(k \leftarrow 2; L_{k-1} \neq \emptyset; k++)$ **do**
  $C_k \leftarrow$ Apriori-Gen(L_{k-1}) ;
  $\overline{C_k} \leftarrow \emptyset$;
  for all (objet o tel que $o.TID \in \overline{C}_{k-1}$) **do**
    $C_o \leftarrow \{c \in C_k | (c - c_k) \in o.listcandidat \wedge (c - c_{k-1}) \in o.listcandidat\}$;
    for all (candidat $c \in C_o$) **do**
      $c.support++$;
      if $(C_o \neq \emptyset)$ **then**
        $\overline{C}_k \leftarrow \overline{C}_k \bigcup \{(o.TID, C_k)\}$;
      end if
    end for
    $L_k \leftarrow \{c \in C_k | c.support \geq minsupp\}$;
  end for
  Retourner $\bigcup_k L_k$;
end for

Ici, la fonction Apriori-Gen est utilisée pour générer les itemsets candidats pour l'ité-

ration suivante dans l'ensemble d'enregistrement $\overline{C}_k$. Chaque élément de $\overline{C}_k$ est un couple $(TID, \{c_k\})$, où $\{c_k\}$ est la liste des k-itemsets candidats contenus dans l'objet dont l'identifiant est TID. Le support des itemsets de $\overline{C}_k$ est égal au nombre d'apparitions c_k dans $\overline{C}_k$. L'ensemble $\overline{C}_1$ correspond au contexte après transformation de chaque item i en itemsets $\{i\}$ (ligne 2). Pour chaque k^{e} itérations, l'ensemble $\overline{C}_k$ est généré en utilisant la procédure Apriori-Gen (ligne 4) et l'ensemble $\overline{C}_k$ est construit en utilisant les ensembles $\overline{C}_{k-1}$ et $\overline{C}_k$ (lignes 6 à 13). Chaque élément de $\overline{C}_k$ correspond à un objet o de $\overline{C}_{k-1}$ et contient son TID et la liste des k-itemsets candidats. L'ensemble L_k est construit en déterminant pour chaque candidat de C_k son nombre d'apparition $\overline{C}_k$ et en insérant dans L_k les candidats fréquents (ligne 14).

L'exemple 3.2 ci-dessous présente son application à la base $\mathcal{B}$, pour un *minsupp* de 2/6.

Algorithme PARTITION L'algorithme Partition (Savasere et al., 1995 [SON95]) ne réalise que deux balayages du contexte afin d'extraire les motifs fréquents. Durant le premier balayage, la base est divisée en n partitions disjointes. Pour chaque partition, l'ensemble des itemsets fréquents (fréquents locaux) est extrait. Les ensembles d'itemsets fréquents pour chaque partition sont ensuite fusionnés pour obtenir un ensemble d'itemsets candidats (itemsets fréquents globaux). L'ensemble ainsi obtenu un sur-ensemble des itemsets fréquents. Durant le second balayage, les supports pour ces candidats sont calculés sur toute la base afin d'identifier les motifs fréquents. La taille de la partition est choisie de telle façon que chaque partition tienne en mémoire. Le pseudo-code de cet algorithme est présenté dans l'algorithme 5 ci-dessous. Par la suite, notons par n le nombre de partition du contexte, par Pr la r^{e} partition du contexte, par C_k^g l'ensemble des k-motifs globaux, par C^g l'ensemble des candidats globaux, par F^r l'ensemble des motifs fréquents dans Pr, par F^g l'ensemble des motifs fréquents globaux.

Algorithm 5 Algorithme PARTITION

Require: Contexte $\mathcal{B}$; seuil *minsupp* ; nombre de partition n ;
Ensure: Ensemble F^g des itemsets fréquents.

1: **Partitionner**$(\mathcal{B}, n)$
2: **for** $(r \leftarrow 1; r \leq n; r++)$ **do**
3: Lire partition P_r
4: $F^r \leftarrow$ Partition-Gen$(P_r, minsupp)$
5: **end for**
6: **for** $(r \leftarrow 1; F_k^r \neq \emptyset; k++)$ **do**
7: $C_k^g \leftarrow \bigcup_{r=1}^{r=n} F_k^r$
8: **end for**
9: **for** $(r \leftarrow 1; r \leq n; r++)$ **do**
10: Lire partition P_r
11: $F^r \leftarrow$ Partition-Count(C^g, P_r)
12: **end for**
13: $F^g \leftarrow \{c \in C^g | c.suppoort \geq minsupp\}$
14: **Retourner** F^g

L'algorithme procède en quatre phases. Durant la première phase, le contexte est divisé

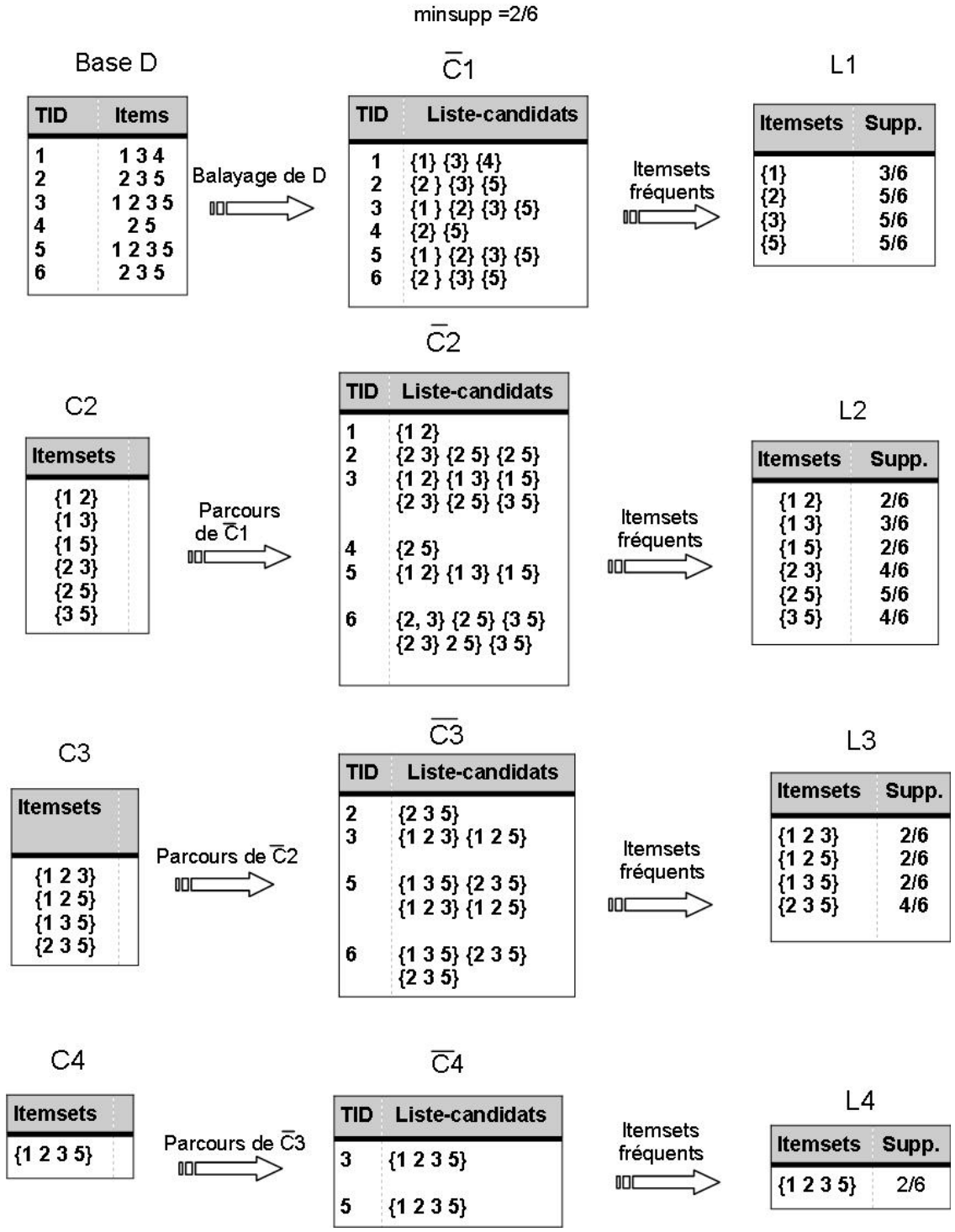

Figure 3.2 – Exemple d'exécution d'APRIORI-TID sur la base de données $\mathcal{B}$

en n partitions disjointes (ligne 1). Durant la seconde phase (lignes 2 à 5), chaque partition

P^r pour $1 \leq r \leq n$ est lu et la procédure Partition-Gen est appliquée afin d'en extraire l'ensemble F^r des itemsets fréquents locaux. Les ensembles $F^r (1 \leq r \leq n)$ ainsi obtenus sont ensuite fusionnés lors de la troisième phase (lignes 6 à 8) afin d'obtenir l'ensemble C^g des candidats globaux. Chaque ensemble C_k^g des candidats globaux de taille k est construit par l'union des ensembles F_k^r contenant les k-itemsets fréquents dans la partition P_r (ligne 7). Durant la quatième phase (lignes 9 à 13), les n partitions sont lues successivement et les supports des itemsets candidats de C^g sur l'ensemble de supports globaux sont déterminés (lignes 9 à 12). L'ensemble F^g des itemsets fréquents globaux dont le support global est supérieur ou égal à *minsupp* (ligne 13).

Partition-Gen La procédure `Partition-Gen` reçoit une partition Pr et un seuil *minsupp* comme paramètres. Elle retourne l'ensemble F^r des itemsets fréquents locaux. Le pseudo-code de cette procédure est présenté dans l'algorithme 6 ci-dessous.

Algorithm 6 Procédure Partition-Gen

Require: Partition P_r du contexte $\mathcal{B}$; seuil *minsupp* ;
Ensure: Ensemble F^r des itemsets fréquents locaux.
1: $F_1^{Pr} \leftarrow$ {1-itemsets fréquents dans P_r avec liste de TID}
2: **for** $(k \leftarrow 2; F_k^{Pr} \neq \emptyset; k++)$ **do**
3: **for all** (itemsets $p \in F_{k-1}^{Pr}$) **do**
4: **for all** (itemsets $p \in F_{k-1}^{Pr}$) **do**
5: **if** $(p[1] = q[1] \wedge \ldots \wedge p[i-2] = q[i-2] \wedge p[i-1] < q[i-1])$ **then**
6: $c \leftarrow p[1] \cup p[2] \cup \ldots \cup p[i-1] \cup q[i-1]$
7: **if** ($\forall s$ sous-ensemble de $|c|-1$ nous avons $s \in F_{k-1}^{Pr}$) **then**
8: $c.tidset \leftarrow p.tidset \cap q.tidset$
9: **if** $(|c.tidset|/|Pr| \geq minsupp)$ **then**
10: $F_k^{Pr} \leftarrow F_k^{Pr} \cup \{c, c.support\}$
11: **end if**
12: **end if**
13: **end if**
14: **end for**
15: **end for**
16: **end for**
17: **Retourner** $\bigcup F_k^{Pr}$

Après la lecture de la partition Pr, l'ensemble F_1^{Pr} des 1-itemsets fréquents est construit (ligne 1). Chaque élément de F_1^{Pr} est un couple $(f_1, \{TID\})$ dans lequel f_1 est un 1-itemset et $\{TID\}$ est la liste des identifiants des objets de la partition contenant f_1. La première phase de la procédure (lignes 3 à 6) est conforme à la première phase d'Apriori-Gen. Deux $(k-1)$-itemsets fréquents p et q de F_{k-1}^{Pr} sont joints si les $(k-2)$ premiers items qui les composent sont identiques afin de créer un k-itemsets candidats c. Durant la deuxième phase (lignes 7 à 12), chaque candidat c créé pendant la première phase est considéré. Si tous les sous-ensembles de c de taille $k-1$ sont présents dans F_{k-1}^{Pr} (ligne 7), le champ *tidset* de c est initialisé avec l'intersection des champs *tidset* des itemsets p et q (ligne 8). Cette intersection fournit la liste des TID des objets contenant c. Dans ce cas, les supports des itemsets sont

calculés par intersection des $Tids$ et non par comptage comme dans Apriori. Par exemple, considérons les itemsets i_a et i_b ayant respectivement *tidset* {1,2,5} et {1,4,5}, on en déduit le support de l'itemset (i_a, i_b) par $\{1,2,5\} \cap \{1,4,5\} = \{1,5\}$ représentant les transactions contenant i_a et i_b.

Partition-Count La procédure Partition-Count reçoit l'ensemble C^g des candidats globaux et une partition P_r de la base $\mathcal{B}$ comme paramètres. Elle met à jour le support global de chaque candidat c de C^g. Son pseudo-code est présenté dans l'algorithme 7 ci-après.

Algorithm 7 Procédure Partition-Count

Require: Partition C^g de candidats globaux ; partition P_r de la base $\mathcal{B}$;
Ensure: Champs du support de C^g mis à jour
1: **for all** (1-itemset $c_1 \in C_1^g$) **do**
2: $c_1.tidset \leftarrow \{t.TID | t \in P_r$ et c_1 contenu dans t$\}$
3: $c_1.support \leftarrow c_1.support + |c_1.tidset|$
4: **end for**
5: **for** $(k \leftarrow 2; C_k^g \neq \emptyset; k++)$ **do**
6: **for all** (k-itemset $c_1 \in C_k^g$) **do**
7: $templist \leftarrow c_1.tidset \cap c_2.tidset \cap \ldots \cap c_k.tidset$
8: $c.support \leftarrow c.support + |templist|$
9: **end for**
10: **end for**

La procédure commence par construire pour chaque 1-itemset c_1 de C^g la liste des objets de la partition contenant c_1 et met à jour le support de c_1 en fonction de la taille de cette liste (lignes 1 à 4). Pour chaque itemset $c \in C^g$ de taille supérieure à un, la procédure détermine la liste *templist* des TID des objets contenant c par intersection des *tidset* de tous les 1-itemsets inclus dans c (ligne 7). Le support global de l'itemset c est ensuite mis à jour en augmentant sa valeur de la taille de liste *templist* (ligne 8).

Algorithme ECLAT Eclat a été introduit par Zaki dans (Zaki et al., 1997 [ZPOL97]). A l'inverse de Apriori, Eclat n'utilise pas explicitement la propriété d'anti-monotonie du support. Il s'appuie essentiellement sur la notion de base conditionnelle et sur une représentation verticale des données, en ce sens que les lignes représentent les attributs et que l'on trouve dans chaque ligne l'ensemble des TID des transactions contenant l'attribut en question. Pour un attribut donné A, nous appellerons cet ensemble de transactions la couverture de A. Plus généralement, l'algorithme Eclat propose de fixer un seuil minimum de support *minsupp*, permettant un élagage non pas des attributs de couverture nulle, mais des attributs dont la couverture aurait une cardinalité inférieure au seuil. La recherche en profondeur d'abord commence par les 1-itemsets. Contrairement à Apriori, Eclat ne connaît pas tous les itemsets fréquents à un niveau donné avant de considérer les candidats du niveau suivant, ce qui diminue l'efficacité, car la propriété d'antimonotonicité n'est plus utilisable pour élaguer l'espace de recherche. Ceci reste acceptable pour les petites bases de données, mais l'élagage reste pauvre et altère les performances quand il s'agit de traiter des bases de données de taille importante. Le pseudo-code de cet algorithme est présenté dans l'algorithme 8 ci-dessous.

Algorithm 8 Algorithme ECLAT

Require: un ensemble de motifs $\mathcal{I}$; une base $\mathcal{B}$; un *minsupp*
Ensure: $\mathcal{F}[\mathcal{I}]$, un ensemble des itemsets fréquents.

```
1: F[I] ← {}
2: for all i ∈ I occurring in B do
3:    F[I] ← F[I] ∪ {I ∪ {i}}
4:    B^i ← {}
5:    for all j ∈ I occurring in B such that j > i do
6:       C ← cover({i}) ∩ cover({j})
7:       if |C| ≥ minsupp then
8:          B^i ← B^i ∪ {j, C}
9:       end if
10:   end for
11:   compute F[I ∪ {i}](B^i, minsupp)
12:   F[I] ← F[I] ∪ F[I ∪ {i}]
13: end for
```

Eclat utilise le format vertical de la base de données ; celui-ci a l'avantage de rendre le calcul du support plus simple puisqu'il s'agit d'effctuer dans ce cas des intersections des tidsets. De plus, ceci réduit automatiquement la taille de la base de données puisque seules les transactions concernant un itemset sont utilisées pour l'intersection. Eclat effectue une recherche des itemsets fréquents en profondeur d'abord et se base sur le concept de classes d'équivalence. Par exemple, ABC et ABD appartiennent à la même classe d'équivalence. Deux k-itemsets appartiennent à une même classe d'équivalence s'ils ont en communun préfixe de taille $(k-1)$. Chaque classe peut être traitée séparément en mémoire, ce qui permet de décomposer le treillis en sous-treillis où chaque sous-treillis représente une classe d'équivalence.

Algorithme FP-Growth Afin de remédier au problème de fouille des motifs fréquents dans une grande base de données transactionnelle dont souffrent les algorithmes variantes d'Apriori, Han et al. [HPYM00] ont proposé l'algorithme FP-Growth (Frequent-Pattern Growth). L'algorithme utilise une structure de données compacte appelée *Frequent-Pattern tree* représentant les transactions des motifs fréquents. FP-Growth construit les itemsets fréquents sans génération des candidats, i.e. il ne génère pas des candidats, mais ne fait que les tester pour essayer d'en trouver des nouveaux, afin de réduire les coûts des entrées/sorties. De ce fait, l'algorithme repose sur le paradigme de *Diviser pour Régner*, et n'accède que deux passes à la base de transactions. Le premier parcours de la base permet d'extraire la liste des 1-itemsets fréquents. Cette liste est ensuite triée par ordre décroissant des supports. Le deuxième parcours consiste à trier le contenu de chacune des transactions. L'algorithme présente un important gain par le fait qu'il suffit de suivre les liens inter-nœuds pour connaître les associations fréquentes, mais nécessite un espace mémoire important en présence de données éparses et de valeurs des supports relativement faibles, en raison en partie des structures auxiliaires utilisées. A cet effet, l'algorithme est contraint d'effectuer un nombre important d'opérations pour concaténer les fragments d'itemsets, sans parvenir à trouver les motifs fréquents. Son pseudo-code est présenté dans l'algorithme 9 ci-dessous.

Algorithm 9 Algorithme FP-Growth($\mathcal{B}$, *minsupp*)

Require: Set of items $\mathcal{I}$ Database $\mathcal{B}$; minimum support *minsupp*.
Ensure: F-List $\mathcal{F}$

```
Define and clear F-list F ;
for all transaction t_i ∈ B do
    for all Items a_i ∈ t_i do
        F[a_i] + + ;
    end for
end for
Sort F ;
Define and clear the root of FP-Tree : r ;
for all transaction t_i ∈ B do
    Make t_i ordered according to F ;
    Call ConstructTree(t_i, r) ;
end for
for all transaction a_i ∈ I do
    Call Growth(r, a_i, minsupp) ;
end for
```

L'Algorithme 10 ci-dessous représente la procédure Growth($r, a, minsupp$).

Algorithm 10 Procédure Growth($r, a, minsupp$)

```
if r contains a single path Z then
    for all combination (denoted as γ) of the node in Z do
        Generate pattern β = γ ∪ a ;
        if β.support > minsupp then
            Call Output(β) ;
        end if
    end for
else
    for all ν ∈ r do
        Generate pattern β = ν ∪ a ;
        if β.support > minsupp then
            Call Output(β) ;
        end if
        Construct conditional pattern base ;
        Construct conditional FP-Tree Tree_β ;
        if Tree_β ≠ ∅ then
            Call Growth(Tree_β, β, minsupp) ;
        end if
    end for
end if
```

L'algorithme commence par définir un ordre sur les attributs en parcourant une première

fois la base de données pour calculer les supports de chaque attribut. Puis, il classe les attributs fréquents de chaque transaction (i.e. deux motifs ayant le même support sont donnés dans un ordre arbitraire), et construit une liste appelée table de liens auxquels sont associés leur support et un pointeur. Ce pointeur est initialement vide, mais pointera ensuite vers la première occurence de l'attribut dans l'arbre. Une fois FP-tree construit, l'extraction des itemsets fréquents se fait de la manière suivante : l'algorithme commence par l'extraction des itemsets suffixes de taille 1, auxquels il construit des bases conditionnelles. Ce sont des chemins préfixes dans FP-tree partageant les 1-itemsets à un suffixe considéré. L'extraction se fera récursivement sur cette nouvelle sous-structure. Les itemsets fréquents sont obtenus par concaténation du suffixe de la base conditionnelle aux itemsets fréquents du sous-arbre conditionnel. L'exemple (inspiré de [BL09]) ci-dessous illustre le principe de FP-Growth, mené à une base de données du tableau 3.1 contenant des items descriptifs $\mathcal{I}$ et ceux de classe C. On choisit ici un seuil de support *minsupp* de 1.

TID	Attributs
t_1	I_1,I_2,I_5,C_1
t_2	I_2,I_4,C_2
t_3	I_1,I_2,I_3,C_1
t_4	I_1,I_2,I_4,C_2

Tableau 3.1 – Exemple d'une base de données

Le tableau 3.2 ci-dessous présente les items associés à leurs supports.

Items	Support
I_1	3
I_2	4
I_3	1
I_4	2
I_5	1
C_1	2
C_2	2

Tableau 3.2 – Items associés à leur support

Dans un premier temps, l'algorithme parcourt la base de données et lit tous les motifs en calculant leur support et les compare avec le seuil de support préfixé. FP-Growth enregistre les items fréquents dans une liste L telle que : $L = \{(I_2 : 4), (I_1 : 3), (I_4 : 2), (C_1 : 2), (C_2 : 2), (I_3 : 1), (I_5 : 1)\}$.

Dans un second temps, un FP-Tree est construit par la création d'une *root* assignée nulle, où chaque transaction est décrite dans l'ordre des items donné par la liste L. Par exemple, la première transaction $t_1 : I_1, I_2, I_5, C_1$ sera sauvegardé en $t_1 : I_2, I_1, C_1, I_5$. Cette transaction va constituer la première branche de FP-Tree avec 4 nœuds : $(I_2 : 1), (I_1 : 1), (C_1 : 1), (I_5 : 1)$. La seconde transaction t_2 de l'ordre I_2, I_4, C_2 construit une deuxième branche, où le nœud I_2 est lié à la *root*, le nœud I_4 lié à I_2, et le nœud C_2 lié à I_4. Cette branche partage un préfixe commun I_2 avec transaction t_1. Ainsi, lors de construction du nœud $(I_4 : 1)$, doit-on incrémenter de 1 le compteur du nœud $(I_2 : 2)$.

En dernier temps, FP-Tree est fouillé par la création des sub-fragments conditionnels de base. En fait, pour trouver ces fragments, on extrait pour chaque fragments de longueur 1 (*suffix pattern*) l'ensemble des itemsets existants dans le chemin FP-Tree *conditionnal pattern base*. L'itemset fréquent est obtenu par la concaténation du suffixe avec les fragments fréquents extraits des FP-Tree conditionnals (cf. tableau 3.3).

Items	Motifs conditionnels	FP-Tree conditionnels	Motifs fréquents
I_5	$I_1, I_2, C_1, I_2, I_1, I_3$	$(I_2 : 2, I_1 : 1)$	$I_2I_5 : 2, I_1I_5 : 2, I_2I_5I_1 : 2$
I_4	$I_2 : 1, I_2I_1 : 1$	$(I_2 : 2)$	$I_2I_4 : 2$

Tableau 3.3 – Fouille du FP-Tree

3.2.2 Algorithmes d'extraction d'itemsets fréquents maximaux

Plusieurs algorithmes ont été conçus pour déterminer l'ensemble des itemsets fermés fréquents maximaux. Nous en citons entre autres MaxClique et MaxEclat [ZPOL97], et Max-Miner [Bay98] dont l'objectif est de réduire l'espace de recherche, de diminuer le nombre de balayage du jeu de données. L'ensemble des itemsets fréquents maximaux forme une bordure positive de l'ensemble des itemsets fréquents. L'extraction des itemsets fréquents maximaux est par la suite réalisée par une exploration itérative du treillis, en avançant d'un niveau du bas vers le haut, et d'un ou plusieurs niveaux du haut vers le bas lors de chaque itération. À partir des itemsets fréquents maximaux, tous les itemsets fréquents sont dérivés et leurs supports sont déterminés en réalisant un balayage du contexte.

MaxClique et MaxEclat MaxClique et MaxEclat ont été proposés par Zaki (Zaki et al., 1997 [ZPOL97]. Dans ce cas, le calcul des supports se fait par intersection des listes TID et des items inclus dans l'itemset. L'algorithme MaxEclat est basé sur la classe d'équivalences des itemsets. C'est-à-dire, deux k-itemsets font partie de la même classe d'équivalence s'ils partagent les mêmes $(k-1)$ premiers items. A partir de la seconde itération, les itemsets fréquents déterminés sont ensuite utilisés afin de créer un ensemble d'itemsets maximaux candidats. Pour k-itération, tous les k-itemsets fréquents de F_k possédant les mêms $(k-1)$ premiers items sont combinés pour créer un itemset maximal candidat. L'ensemble ainsi généré contient chaque itemset maximal fréquent de taille supérieur à k, ou bien un de ses sur-ensembles [ZPOL97]. Les supports des itemsets maximaux candidats sont calculés lors de l'itération suivante.

L'algorithme MaxClique utilise la même démarche que MaxEclat, mais basé sur les cliques maximales des hypergraphes uniformes. A tout ensemble F_k de k-itemsets fréquents peut être associé un hypergraphe uniforme dont les sommets sont les items des k-itemsets et les arcs relient les itemsets contenus dans les même k-itemsets. Rappelons qu'une clique maximale d'un hypergraphe est un ensemble maximale de sommets de l'hypergraphe tous reliés entre eux. Lors d'une itération k consécutives à la seconde itération, les k-itemsets fréquents de F_k appartenant à la même clique de l'hypergraphe associé à F_k sont combinés afin de générer un itemset maximal candidat. Ce qui implique qu'un itemset maximal candidat est crée, si tous ses sous-ensembles de taille k sont des k-itemsets fréquents. Le nombre d'itemsets maximaux candidats générés par MaxClique est inférieur à celui de ceux générés par MaxEclat.

Toutefois, cette plus grande précision des itemsets maximaux candidats entraîne un coût supplémentaire dû au nombre d'opérations requises pour les générer.

Algorithme Max-Miner Max-Miner a été proposé par Bayardo [Bay98]. Le pseudo-code est présenté dans l'algorithme 11 ci-dessous. Dans ce cas, notons par C l'ensemble des groupes candidats, par F_m l'ensemble des itemsets fréquents maximaux.

Algorithm 11 Algorithme Max-Miner

Require: Contexte $\mathcal{B}$; seuil minimal de support $minsupp$.
Ensure: Ensemble F_m des itemsets fréquents maximaux
1: $C \leftarrow \emptyset$;
2: $F_m \leftarrow$ Gen-Inital-Groups$(\mathcal{B}, C, minsupp)$;
3: **while** $C \neq \emptyset$ **do**
4: **lire** Contexte $\mathcal{B}$
5: $Support - Count(\mathcal{B}, C)$
6: **for all** candidat $c \in C | h(t) \cup t(c)$ est fréquent **do**
7: $F_m \leftarrow F_m \cup \{h(c) \cup t(c)\}$
8: **end for**
9: $C_{new} \leftarrow \emptyset$
10: **for all** candidat $c \in C | h(t) \cup t(c)$ est infréquent **do**
11: $F_m \leftarrow F_m \cup$ Gen-Sub-Nodes$(c, C_{new}, minsupp)$
12: **end for**
13: $C_{new} \leftarrow \emptyset$
14: **supprimer** de F_m les itemsets f tel que $\exists f' \in F_m, f \subseteq f$
15: **supprimer** de C les groupes c tel que $\exists f' \in F_m$ avec $h(c) \cup t(c) \subseteq f'$
16: **end while**
17: **Retourner** F_m

Durant la première phase (lignes 1 et 2), les ensemble C et F_m sont respectivement initialisés avec l'ensemble vide et le 1-itemset possédant le plus grand support dans le contexte. L'ensemble F_m est réalisé par la procédure Gen-Inital-Groups. Durant chacune des itérations suivantes (lignes 3 à 16), un balayage du contexte est réalisé (ligne 4), le support du groupe candidat est calculé (ligne 5). Les itemsets fréquents maximaux sont découverts sont insérés dans F_m. Les candidats de l'itération suivante sont crées (lignes 9 à 15) comme suit. Un ensemble C_{new} est crée et initialisé par l'ensemble vide. Pour chaque groupe candidat $c \in C$, les candidats de l'itération suivante dérivée de c sont insérés dans C_{new} en utilisant la procédure Gen-Sub-Nodes (ligne 11). Cette procédure est également utilisée pour insérer dans F_m l'itemset fréquent qui est maximal vis-à-vis du groupe candidat c. Cet itemset peut ne pas être un itemset fréquent maximal final. Lorsque tous les groupes candidats ont été considérés, l'ensemble C_{new} devient le nouvel ensemble candidat pour l'itération suivante (ligne 13). Les itemsets qui ne sont pas maximaux dans F_m sont suprimés de F_m (ligne 14) et les groupes candidats de C qui ont un sur-ensemble dans F_m sont supprimés de C (ligne 15). L'algorithme cesse si aucun groupe de candidats ne peut être créé et retourne l'ensemble F_m contenant tous les itemsets fréquents maximaux du contexte (ligne 17).

Procédure Gen-Inital-Groups La procédure Gen-Inital-Groups initialise l'ensemble C des itemsets candidats avec les 1-itemsets fréquents et leur liste d'items extensions et retourne les 1-itemsets possédant le plus grand support. Les itemsets servent à initialiser l'ensemble F_m. Le pseudo-code de la procédure est présenté dans l'algorithme 12 ci-après, dont l'objectif est de faire apparaître les items les plus fréquents dans le contexte. En ordonnant les items par ordre croissant de leurs supports, les items les plus fréquents seront les derniers dans l'ordre et apparaîtront donc dans le plus grand nombre de groupes candidats.

Algorithm 12 Procédure Gen-Inital-Groups

Require: Contexte $\mathcal{B}$; Ensemble C de groupe candidat ; Seuil minimal de support *minsupp*.
Ensure: Ensemble C initialisé ; 1-itemset possédant le plus grand support.
1: $F_{m1} \leftarrow$ {1-itemsets fréquents dans $\mathcal{B}$} ;
2: **ordonner** les itemsets contenus dans F_{m1} par supports croissants ;
3: **for all** item $i \in F_{m1}$ autre que le plus grand item dans l'ordre **do**
4: **créer** un groupe candidat c
5: $h(c) \leftarrow \{i\}$
6: $t(c) \leftarrow \{i' \in \mathcal{I} | i < i'$ dans l'ordre sur les items$\}$
7: $C \leftarrow C \cup \{c\}$
8: **end for**
9: **Retourner** l'itemset $f \in F_{m1}$ contenant le plus grand item dans l'ordre

La procédure commence par créer un ensemble F_{m1} contenant tous les 1-itemsets fréquents dans $\mathcal{B}$ (ligne 1). Les items contenus dans F_{m1} sont ensuite ordonnés par ordre croissant de leurs supports (ligne 2). Tous ces items (à l'exception du plus grand dans l'ordre défini) sont considérés successivement en créant un groupe candidat pour chacun de ces items (lignes 3 à 8). Cette procédure retourne les 1-itemsets fréquents de F_{m1} contenant le plus grand item dans l'ordre (ligne 9).

Procédure Gen-Sub-Nodes Le pseudo-code de cette procédure est présenté dans l'algortithme 13.

L'algorithme met à jour l'ensemble C de groupes candidats. Il retourne l'itemset fréquent sur-ensemble de c de taille $|c|+1$. La première partie de la procédure (lignes 1 à 3) supprime de la liste d'items extension de c les listes i pour lesquels $h(c) \cup \{i\}$ est infréquent. Les items i restant dans $t(c)$ sont ensuite ordonnés par ordre croissant des supports de $h(c) \cup \{i\}$ (ligne 4). Un nouveau groupe candidat c' est créé dans C, avec itemset candidat $h(c') = h(c) \cup \{i\}$ (lignes 5 à 10). Si aucun des itemsets $h(c) \cup \{i\}$ n'est pas fréquent alors la procédure retourne $h(c)$ (ligne 11). Sinon, elle retourne l'itemset $h(c) \cup \{i\}$ (ligne 12).

3.2.3 Algorithmes d'extraction d'itemsets fermés fréquents

Les notions d'itemset fermé et de fermeture d'un itemset sont issues de la théorie des treillis (Marc et al., 1970 [MM70]). L'utilisation de cette théorie dans l'extraction d'itemsets fréquents est nombreuse. Nous pouvons citer, entre autres, l'algorithme Bordat [Bor86], l'algorithme Carpineto [CR93], l'algorithme Ganter [Bur98]. Ces algorithmes sont relativement efficaces, ils souffrent parfois du temps de calcul trop élevé. Plusieurs travaux ont été

Algorithm 13 Procédure Gen-Sub-Nodes

Require: Groupe candidat c ; Ensemble C de groupe candidat ; Seuil minimal de support *minsupp*.
Ensure: Ensemble C mis à jour ; itemset fréquent $h(c) \cup \{i\}$ possédant le plus grand support pour $i \in t(c)$ ou $h(c)$.

1: **for all** item $i \in t(c)$ **do**
2: **if** $h(c) \cup \{i\}$ est fréquent **then**
3: $t(c) \leftarrow t(c)\ \{i\}$
4: **end if**
5: **end for**
6: **ordonner** les itemsets i de $t(c)$ par supports de $h(c) \cup \{i\}$ croissants
7: **for all** item $i \in t(c)$ autre que le plus grand itemset dans $t(c)$ **do**
8: **créer** un groupe candidat c'
9: $h(c') \leftarrow h(c) \cup \{i\}$
10: $t(c') \leftarrow \{i' \in t(c) | i < i' \text{ dans } t(c)\}$
11: $C \leftarrow C \cup \{c'\}$
12: **end for**
13: **Si** $t(c) = \emptyset$ **alors retourner** $h(c)$
14: **sinon retourner** $h(c) \cup \{i\}$ avec i plus grand item dans $t(c)$

proposés pour y faire face. Parmi les plus classiques sont les travaux de Pasquier (Close [PBTL98, PBTL99c] et A-Close [PBTL99a]).

Algorithme CLOSE CLOSE [PBTL99b] est un algorithme d'extraction par niveau. Son pseudo-code est présenté dans l'algorithme 14 ci-après. Dans ce cas, notons par FC_k l'ensemble de k-groupes candidats, par F_k celui de k-groupes fréquents générateurs.

Algorithm 14 Algorithme CLOSE

Require: Contexte $\mathcal{B}$; Seuil minimal de support *minsupp*.
Ensure: F : Ensemble des motifs fermés fréquents

1: $FC_1.générateurs \leftarrow \{1 - itemsets\}$
2: **for** $(k = 1\ ;\ FC_k.\text{générateurs} \neq \emptyset; k++)$ **do**
3: Gen-Closure(FC_k)
4: **for all** $c \in FC_k$ **do**
5: **if** $c.support \geq minsupp$ **then**
6: $F_k \leftarrow F_k \cup \{c\}$
7: **end if**
8: **end for**
9: $FC_{k+1} \leftarrow$ Gen-Generator(F_k)
10: **end for**
11: **Retourner** $F_k \bigcup_k F_k$

L'ensemble des 1-générateurs est initialisé des 1-itemsets du contexte (ligne 1). Chacune des k-itérations suivantes (lignes 2 à 10) est décomposée en 3 phases. La première, qui est

réalisée par la procédure Gen-Closure, consiste à déterminer les fermetures des k.générateurs de FC_k et calculer les supports de chacun des itemsets fermés féquents ainsi obtenus (ligne 3). Durant la seconde phase, les itemsets fermés fréquents sont insérés dans F_k (lignes 4 à 8). Durant la troisième phase, les $(k-1)$.générateurs de FC_{k+1} sont crées en appliquant de la procédure Gen-Generator aux k.générateurs de F_k (ligne 9). Ces itérations sont répétées jusqu'à ce que l'ensemble FC_{k+1} soit vide.

Procédure Gen-Closure La procédure Gen-Closure reçoit un ensemble FC_k des k-groupes candidats comme paramètre. Le pseudo-code de cette procédure est présenté dans l'algorithme 15 ci-après.

Algorithm 15 Procédure Gen-Closure

Require: FC_k ; Contexte $\mathcal{B}$;
Ensure: FC_k.fermés ; FC_k.supports
1: **for all** $o \in \mathcal{B}$ **do**
2: $G_o \leftarrow$ Sous-ensemble(FC_k.générateurs, $f(o)$)
3: **for all** p.générateur$\in G_o$ **do**
4: **if** p.fermé $= \emptyset$ **then**
5: p.fermé$\leftarrow f(o)$
6: **else**
7: p.fermé $\leftarrow$ p.fermé $\cap f(\{o\})$
8: **end if**
9: $p.support++$
10: **end for**
11: **end for**
12: **Retourner** $\bigcup\{p \in FC_k | \text{p.fermé} \neq \emptyset\}$

Pour chaque objet o, l'ensemble G_o est créé (ligne 2). Chaque générateur $p \in G_o$ et la fermeture associée ainsi que leur support sont mis à jour (lignes 3 à 10). Si l'objet p.fermé est vide, on affecte le p.fermé à l'intersection de l'ancien p.fermé et $f\{o\}$ (lignes 6 à 8). Ensuite, le support p.support du motif p.fermé est incrémenté (ligne 9). Lorsque tous les objets du contexte ont été considérés, la procédure retourne l'ensemble FC_k des mis à jour pour chaque générateur, sa fermeture et leur support.

Gen-Generator La procédure Gen-Generator reçoit F_k en argument, et retourne FC_{k+1}. Son pseudo-code est présenté dans l'algorithme 16 sous-dessous.

La procédure Gen-Generator génère tout d'abord les $(k+1)$-générateurs candidats en appliquant la même phase de jonture de l'Apriori (Agrawal et Srikant, 1994 [AS94]) (ligne 1). Les $(k+1)$-générateurs candidats dont on sait qu'ils sont soit fréquents, soit non maximaux sont ensuite supprimés (lignes 2 à 8). Enfin, on supprime parmi les générateurs ceux dont la fermeture est déjà calculée (lignes 9 à 17).

Algorithme A-Close A-Close [PBTL99a] génère itérativement les k-générateurs de k-groupes fréquents de F_k. Son pseudo-code est présenté dans l'algorithme 17 ci-dessous.

Algorithm 16 Procédure Gen-Generator

Require: FC_k ;
Ensure: FC_{k+1}
1: $FC_{k+1} \leftarrow$ Apriori-Gen(FC_k.générateurs)
2: **for all** p.générateur $\in FC_{k+1}$.générateurs **do**
3: **for all** $s \subseteq$ p.générateur($s : k - motifs$) **do**
4: **if** $s \notin FC_k$.générateur **then**
5: $FC_{k+1} \leftarrow FC_{k+1}$ $\{p\}$
6: **end if**
7: **end for**
8: **end for**
9: **for all** p.générateur $\in FC_{k+1}$ **do**
10: $S_p \leftarrow$ Sous-ensemble(F_k.générateurs, p.générateurs
11: **for all** $s \in S_p$ **do**
12: **if** (p.générateur $\subseteq$ s.fermé) **then**
13: $FC_{k+1} \leftarrow FC_{k+1}\{p\}$
14: **end if**
15: **end for**
16: **end for**
17: **Retourner** FC_{k+1}

Algorithm 17 Algorithme A-Close

Require: Contexte $\mathcal{B}$; seuil minimal de support *minsupp*
Ensure: Ensemble F_k des k-groupes fréquents
1: F_1.générateurs $\leftarrow \{1 - itemsets\}$
2: Support-Count($\mathcal{B}, F_1$.générateurs)
3: **for all** générateur g.générateur$\in F_1$ **do**
4: **Si** (g.support $<$ minsupp) **alors supprimer** g de F_k
5: **end for**
6: **for** ($k \leftarrow 1; F_k$.générateurs $\neq \emptyset; k++$) **do**
7: $F_{k+1} \leftarrow$ AC-Generator(F_k)
8: **end for**
9: AC-Closure($\bigcup F_k$)
10: **Retourner** $\bigcup_k F_k$

Durant la première itération (ligne 1), l'ensemble des 1-générateurs de F_k est initialisé avec la liste des 1-itemsets du contexte. La procédure Support-Count est ensuite appliquée afin de déterminer les supports de ces 1-itemsets générateurs en réalisant un balayage du contexte (ligne 2). Les 1-générateurs infréquents sont supprimés de F_k (lignes 3 à 5). Durant chaque itération k suivantes (lignes 6 à 8), les $(k+1)$-générateurs de l'ensemble F_{k+1} sont crées en utilisant les k-générateurs. La procédure AC-Generator est appliquée à l'ensemble F_k (ligne 7). Ces itérations s'arrêtent lorsque aucun nouvel itemset générateur ne peut être crée. Tous les générateurs des ensembles F_k ont alors été créés et la procédure AC-Closure

est appliquée à l'ensemble de ces générateurs afin de déterminer leurs fermetures (ligne 9). L'algorithme retourne à la collection des ensembles F_k (ligne 10).

Procédure AC-Generator La procédure AC-Generator reçoit en entrée un ensemble F_k de k-groupes fréquents contenant les k-générateurs fréquents, et donne en sortie l'ensemble FC_{k+1}. Le pseudo-code de cette procédure est représenté dans l'algorithme 18 ci-après.

Algorithm 18 Procédure AC-Generator

Require: FC_k ;
Ensure: FC_{k+1}
1: **insert to** F_{k+1}.générateur
2: **select** $p[1], p[2], \ldots, p[k], q[k]$
3: **from** F_k.générateur p, F_k.générateur q
4: **where** $p[1] = q[1], p[2] = q[2], \ldots, p[k-1] = q[k-1], p[k] < q[k]$
5: **for all** générateur g.générateur $\in FC_{k+1}$ **do**
6: **for all** k-subset $s \in g.générateur$ **do**
7: **if** ($s \notin F_k$.générateur) **then** delete g from FC_{k+1}
8: **end for**
9: **end for**
10: Support-Count($\mathcal{B}, F_1$.générateur)
11: **for all** générateur g.générateur$\in FC_{k+1}$ **do**
12: **if** (g.support $<$ minsupp) **then** delete g from FC_{k+1}
13: **else**
14: **for all** k-subset s.générateur$\in F_k$ **do**
15: **if** (s.support=g.support) **then** delete g from FC_{k+1}
16: **end for**
17: **end for**
18: **Retourner** FC_{k+1}

La première phase de cette procédure (lignes 1 à 4) applique la phase de jointure d'Apriori-Gen aux k-générateurs afin d'initialiser les $(k+1)$-générateurs potentiels de FC_{k+1}. La seconde phase (lignes 5 à 9) vérifie la présence de tous les k-générateurs dans F_k. Durant la troisième phase, un balayage du contexte est réalisé (ligne 10) afin de déterminer le support de chaque $(k+1)$-générateurs potentiels restant dans FC_{k+1}. Tous les $(k+1)$-générateurs sont examinés (lignes 11 à 18). Si $(k+1)$-générateurs g est infréquent, il est supprimé (ligne 12). Sinon, s'il existe un générateur $s \in F_k$ qui est un sous-ensemble de g, possédant le même support que g, alors on supprime g dans FC_{k+1} (ligne 15).

Procédure AC-Closure La procédure AC-Closure reçoit un ensemble F_k des groupes fréquents contenant tous les générateurs fréquents en argument. Elle détermine la fermeture de chaque générateur dans le champs fermé du groupe fréquent à un balayage du contexte. Le pseudo-code de cette procédure est présenté dans l'algorithme 19 ci-après.

AC-Closure traite chaque objet du contexte successivement (lignes 1 à 9) et créée, pour chaque objet o, un ensemble G_o (ligne 2). Ensuite, pour chaque générateur $g.générateur$ dans

Algorithm 19 Procédure AC-Closure

Require: Ensemble F_k des k-groupes des k-générateurs fréquents ; Contexte $\mathcal{B}$;
Ensure: Champs fermé des F_k mis à jour
1: **for all** instance $o \in \mathcal{B}$ **do**
2: $G_o \leftarrow$ Subset(F_k.générateurs, $\phi(\{o\})$)
3: **for all** générateur g.générateur $\in G_o$ **do**
4: **if** (g.fermé $= \emptyset$) **then**
5: g.fermé $\leftarrow \phi(\{o\})$
6: **else**
7: g.fermé $\leftarrow$ g.fermé $\bigcap \phi(\{o\})$
8: **end if**
9: **end for**
10: **end for**
11: **Retourner** $\bigcup\{g \in F_k\}$

G_o, la fermeture *g.fermé* est mis à jour (lignes 3 à 10). Lorsque tous les objets du contexte ont été considérés, la procédure retourne l'ensemble F_k, dans lequel les champs fermés qui sont les fermetures de générateurs fréquents sont mis à jour.

3.3 Génération des règles d'association

La génération de règles d'association est la seconde étape essentielle du processus d'extraction des règles d'association. Un algorithme efficace à ce sujet a été proposé par Agrawal dans (Agrawal et al., 1994 [AS94]). Son principe général est le suivant. Pour chaque itemset fréquent $I_1 \in \mathcal{F}$ de taille supérieure ou égale à 2, tous les sous-ensembles I_2 de I_1 sont déterminés et la valeur du support $supp(I_1)/supp(I_2)$ est calculée. Si ce rapport est supérieur ou égal au seuil de confiance *minconf* fixé par l'utilisateur, la règle d'association $I_1 \rightarrow I_2 \backslash I_1$ est générée. L'algorithme est basé sur la propriété suivante.

Propriété 6. *Etant donné un itemset I, le support d'un sous-ensemble I' est supérieur ou égal au support de I.*

Etant donnés trois itemsets I_1, I_2, I_3 tels que $I_1 \supset I_2 \supset I_3$, il est possible de déduire de cette propriété que $supp(I_3) \geq supp(I_2) \geq supp(I_1)$ [Pas00]. En conséquence, la confiance de la règle $r : I_2 \rightarrow I_1 \backslash I_2$ est supérieure ou égale à la confiance de la règle $r' : I_3 \rightarrow I_1 \backslash I_3$. Si la règle r n'est pas valide, alors la règle r' ne le sera pas non plus. Cela signifie que si la règle d'association $AC \rightarrow DE$ n'est pas valide, par conséquent les règles $A \rightarrow CDE$ et $C \rightarrow ADE$ ne seront pas valides non plus, et il n'est pas nécessaire de calculer leurs confiances. Cette propriété permet de diminuer le nombre de règles d'association testées par l'algorithme. Réciproquement, la confiance de la règle $r'' : I_1 \backslash I_2 \rightarrow I_2$ est supérieure ou égale à la confiance de la règle $r''' : I_1 \backslash I_3 \rightarrow I_3$. Si la règle r''' est valide alors la règle r'' le sera également. Cela signifie que si la règle d'association $A \rightarrow BC$ est valide, alors les règles $AB \rightarrow C$ et $AC \rightarrow B$ le seront également.

3.3.1 Algorithme de génération des règles d'association

Soit $\mathcal{F}$ un ensemble d'itemsets fréquents dans lequel chaque élément de cet ensemble possède deux champs qui sont l'itemset en lui-même et son support. H_m représente les m-itemsets qui sont les conséquences de règles valides générées à partir de l'itemset I_k. Le pseudo-code est représenté dans l'algorithme 20.

Algorithm 20 Algorithme de génération des règles d'association

Require: $\mathcal{F}$ ensemble des itemsets fréquents ; seuil de confiance *minconf*
Ensure: R ensemble des règles d'association valides
1: **for all** k-itemsets fréquents $I_k \in \mathcal{F}$ tel que $k \geq 2$ **do**
2: $H_1 \leftarrow$ 1-itemset sous ensembles de I_k
3: **for all** $h_1 \in H_1$ **do**
4: confiance(r) $\leftarrow$ support(I_k)/support($I_k \backslash h_1$)
5: **if** (confiance(r)$\geq$ minconf) **then**
6: $R \leftarrow R \cup \{r : I_k \backslash h_k \rightarrow h_1\}$
7: **else**
8: $H_1 \leftarrow H_1 \backslash \{h_1\}$
9: **end if**
10: **end for**
11: Gen-Rules(I_k, H_1)
12: **end for**
13: **Retourner** R

L'algorithme considère successivement chaque itemset fréquent de $\mathcal{F}$ de taille supérieur à un (lignes 1 à 12). Pour chacun des itemsets I_k, l'ensemble H_1 des 1-itemsets qui sont des sous-ensembles de I_k est généré (ligne 2). Et pour chacun de ces itemsets h_1, la règle $I_k \backslash h_1 \rightarrow h_1$ est générée si sa confiance est supérieure ou égale à *minconf* (lignes 4 à 6). Sinon, si cette règle n'est pas valide, alors le 1-itemset h_1 est supprimé de H_1 (ligne 8). Lorsque tous les 1-itemsets de H_1 ont été testés, h_1 contient la liste des 1-itemsets qui sont les conséquences des règles valides générées à partir de I_k. Les règles valides générées à partir de I_k sont les règles dont l'union de l'antécédent et de la conséquence donne l'itemset I_k. La procédure Gen-Rules est alors appelée (ligne 11) afin d'insérer dans R les règles valides générées à partir de I_k dont la conséquence contient plus de un item. L'algorithme termie lorsque tous les k-itemsets fréquents pour $k \geq 2$ ont été considérés. L'ensemble R renvoyé par l'algorithme (ligne 13) contient alors toutes les règles d'association valides générées à partir de l'ensemble F.

Procédure Gen-Rules La procédure Gen-Rules met à jour l'esemble R des règles d'associations en y insérant les règles valides générées à partir de I_k dont la conséquence est un $(m+1)$-itemsets. Cette procédure est récursive et réalise en fin d'exécution un appel afin de générer, à partir de I_k, les règles valides dont la conséquence est un $(m+2)$-itemsets. Ces appels se répètent récursivement jusqu'à ce que les règles, dont la conséquence est un $(|I_k|+1)$-itemsets, aient été insérées dans R. La démarche de cette procédure est représentée dans l'algorithme 21 ci-après.

Algorithm 21 Algorithme Gen-Rules : Insertion de règles d'association dans R

Require: k-itemsets fréquents ; Ensemble H_m de m-itemsets ; Seuil *minconf*
Ensure: Ensemble R de règles d'associations valides

```
1:  if k > m + 1 then
2:      H_{m+1} ← Apriori-Gen(H_m)
3:      for all pour chaque h_{m+1} ∈ H_{m+1} do
4:          confiance(r) ← support(I_k)/support(I_k\h_{m+1})
5:          if confiance(r) ≥ minconf then
6:              R ← R ∪ {r : I_k\h_{m+1} → h_{m+1}}
7:          else
8:              Supprimer h_{m+1} de H_{m+1}
9:          end if
10:     end for
11:     Gen-Rules(I_k, H_{m+1})
12: end if
```

Le premier test de l'algorithme (ligne 1) correspond au test d'arrêt des appels récursifs de la procédure. Ensuite, l'ensemble H_{m+1} des $(m+1)$-itemsets qui peuvent être des conséquences de règles valides générées à partir de I_k est créé. Cette création est réalisée en appliquant la procédure Apriori-Gen à l'ensemble H_m des m-itemsets qui sont les conséquences de règles valides générées à partir de I_k (ligne 2). Chaque règle dont la conséquence est un $(m+1)$-itemsets de H_{m+1} est alors testée (lignes 3 à 10). Si la règle testée est valide, elle est insérée dans R (ligne 6). Sinon, les $(m+1)$-itemsets qui en est la conséquence est supprimée de H_{m+1} (ligne 8). Cette suppression correspond à la diminution du nombre de règles testées basé sur la propriété 6 ci-dessus. En effet, si la règle d'association $AC \leftarrow DE$ n'est pas valide, DE est supprimé de H_2. Lors de l'appel récursif suivant, les itemsets CDE et ADE ne seront pas crées par Apriori-Gen dans H_3 car DE est un sous-ensemble de CDE et de ADE. Les règles $A \leftarrow CDE$ et $C \leftarrow ADE$ ne seront donc pas testées. L'appel récursif Gen-Rules est réalisé en fin de procédure (ligne 11) avec comme paramètre l'itemset I_k et l'ensemble H_{m+1}.

3.4 Conclusion partielle

La réalisation de cet état de l'art a mis en exergue une suite logique à nos travaux sur l'extraction de règles d'association dans un contexte binaire, à savoir la recherche des motifs fréquents, et la génération de règles d'association au moyen d'une mesure de qualité plus pertinente, M_{GK} (Totohasina et Feno, 2008 [TF08, Tot08]), par rapport à ladite mesure *confiance* d'Agrawal (Agrawal et al., 1993 [AIS93]). Cet état de l'art fait apparaître un panorama de l'existant autour de l'extraction de règles d'association, nous y avons recencé un nombre important d'approches. Il nous a permis de comparer différentes approches adaptées avec notre problématique. Il ne s'agit pas ici d'une description exhaustive de l'état de l'art, étant donné qu'il existe de nombreux travaux dans la littérature, mais les enseignements tirés de cette étude ont été exploités lors de l'élaboration de notre approche. Ce chapitre clôt notre partie état de l'art sur les différents domaines de recherche abordés par ce mémoire.

Deuxième partie

Contributions de la thèse

Introduction

Cette seconde partie présente nos apports en réponse à notre problématique. Les apports proposés sont situés par rapport aux familles de méthodes présentées dans l'état de l'art. En effet, nous proposons, dans le chapitre 4, notre premier apport, *Extraction optimisée des motifs fréquents*, une nouvelle approche conçue en accord avec les observations faites sur les approches au problème des motifs fréquents. Cette proposition décrit dans un premier temps une nouvelle approche d'extraction optimiseé des motifs fréquents, fondée sur une nouvelle structure de données, appelée `MatriceSupport`, et une nouvelle technique de comptage des supports utilisant le concept des générateurs minimaux. Elle introduit dans un second temps un nouvel algorithme, dénommé EOMF, qui étend l'algorithme de référence Apriori (Agrawal et Srikant, 1994 [AS94]). Notre second apport sera exposé dans le chapitre 5, *Optimisation de la génération des règles d'association positives et négatives valides*. Nous y commençons par présenter quelques motivations et les limites du couple classique "support-confiance". Puis, le cœur du chapitre sera dédié à un nouveau modèle d'extraction des règles d'association positives et négatives potentiellement pertinentes à l'aide du nouveau couple support-M_{GK}. Le modèle proposé est fondé sur deux nouvelles techniques : la première consiste à réduire l'ensemble des règles, tandis que la seconde reflète la minimisation du parcours de l'espace de recherche. Il est validé à des expérimentations menées sur quelques jeux de données de référence de la littérature. Le chapitre 6 expose notre troisième apport, *Graphes implicatifs selon la mesure M_{GK}*. Nous y présentons tout d'abord les notions de base sur la théorie des graphes classiques. Puis, nous détaillons notre modèle d'élaboration de graphes implicatifs, dans lequel nous définissons un nouvel algorithme afin d'automatiser l'élaboration. Ce modèle quant à lui est validé à des expérimentations menées sur quelques bases de données de référence. Enfin, le dernier chapitre (chapitre 7), *Outil `CHIC`-M_{GK}, Applications en didactique des mathématiques*, sera centré sur l'implémentation de notre prototype `CHIC`-M_{GK}, un outil d'aide à l'analyse de données permettant la représentation en graphe les chaînes implicatives entre les règles d'association valides. Cette dernière proposition comprend, d'une part, la description informatique de ce nouvel outil, et d'autre part des applications en didactique des mathématiques, particulièrement problème de l'enseignement-apprentissage de la statistique à Madagascar.

Chapitre 4

Extraction optimisée des motifs fréquents

4.1 Introduction et Motivations

L'extraction des motifs fréquents dans une base de données transactionnelles est un problème classique ayant, dans la communauté de fouille de données, de multiples applications telles la recherche des règles d'association et la classification [MT96]. Elle est une étape primordiale dans le processus d'extraction des règles d'association (Agrawal et al., 1993 [AIS93]) valides. L'algorithme historique Apriori (Agrawal et Srikant, 1994 [AS94]) est le premier algorithme qui traite cette question. Bien qu'il soit efficace, cet algorithme souffre de la vitesse de compilation en présence de données denses et volumineuses : plusieurs balayages de la base de données, conduisant à une insuffisance d'espace mémoire, sont activés.

Un certain nombre de travaux ont été par la suite consacrés à la résolution de ce problème et ont donné naissance aux diverses méthodes plus ou moins efficaces dont nous en citons quelques-unes. L'algorithme Apriori-TID (Agrawal et Srikant [AS94]) cherche à garder le contexte en mémoire, mais parcourt encore plusieurs fois possible la base de données. L'algorithme Partition (Savasere et al. [SON95]) partitionne la base entière en sous-bases d'intersection vide pour tenir en mémoire. Il n'effectue que deux passes, et reste facilement parallélisable, mais considère plus de motifs qui se révèlent globalement peu fréquents. L'algorithme Eclat (Zaki et al. [ZPOL97]) dédié à la recherche de motifs ensemblistes fréquents parcourt l'espace de recherche en profondeur. L'originalité de son approche est de calculer le support d'un motif en faisant l'intersection des ensembles des transactions contenant ses spécialisations. En revanche, une telle méthode ne permet pas de bénéficier pleinement des capacités de la condition d'élagage. Pour y faire face, Pasquier propose deux algorithmes (Close [PBTL98, PBTL99c] et A-Close [PBTL99a]) basés sur le mécanisme de fermeture. Ces algorithmes sont relativement efficaces, mais leur consommation en espace mémoire est élevée dûe à un calcul redondant des fermetures : un motif fermé fréquent peut admettre plusieurs générateurs minimaux et sera calculé plusieurs fois, surtout dans le contexte dense. Plusieurs heuristiques algorithmiques (PF-Growth par Han et al., 2000 [HPYM00] ; Pascal par Bastide et al., 2002 [BTP+02] ; CHARM par Zaki et Hsiao, 2002 [ZH02] ; *Valid Group-Growth Algorithm* par Wang et al., 2006 [WLH06]) ont par la suite été proposées. Bien qu'ils soient efficaces, ces modèles sont assez complexes en terme d'exécution, ce qui désavantage donc la qualité des résultats.

L'une des propositions assez développée est le résumé d'ensembles des motifs (Mielikäinen et Mannila [MM03], Afrati et al. [AGM04], Yan et al. [YCHX05], Chandola et al. [CK07], Jin

et al. [JAAXR08]). Ces approches essayent de limiter l'espace de recherche mais conduisent à une phase d'élagage très complexe, et génèrent un grand nombre de motifs difficilement gérables nécessistant des étapes de traitement très coûteuses. Leur validation pose en général des problèmes pratiques et théoriques ardus qui soulèvent de véritables défis scientifiques.

La prise en compte de bruits a fait aujourd'hui l'objet d'un nombre important de travaux de recherche en fouille des motifs (Seppänen et Mannila [SM04], Besson et al. [BRB04], Liu et al. [JPS+06], Cheng et al. [CYH08]). Ces travaux ont repris le principe de l'algorithme Apriori [AS94], ils sont donc limités à l'utilisation de la contrainte anti-monotonie pour élaguer l'espace mémoire. Afin d'y remédier, Poernomo et Gopalkrishnan [PG09] définissent une nouvelle contrainte de bruit proportionnelle au support des motifs. Cette méthode est efficace, mais extrait un grand nombre de résultats redondants. Pour surmonter cette limite, des heuristiques (Deodhar et al. [DGG+09], Hanczar et Nadif [HN11], Mouhoubi et al. [KLR11, KLR12]) ont été proposées, ce sont des méthodes biclustering. Bien qu'elles soient efficaces, ces approches identifient séquentiellement quelques motifs coclusters qui détériorent la qualité des résultats.

La fouille de motifs séquentiels est également étudiée dans divers travaux (Srikant et Agrawal [SA96], Zaki [Zak01], Nanni et Rigotti [NR07], Li et al. [LLP08], Charnois et al. [CPRC09], Nicolas et al. [NCCC12], Makhlouf et al. [MDT14], Quiniou et al. [QCC14]). Bien qu'obtenant généralement de bons résultats, ces approches ont leurs limites : les résultats obtenus ne sont pas directement exploités en raison de leur grand nombre, ce qui nécessite un poste intermédiaire.

L'étude basée sur les motifs minimaux est également très développée (Calders et al. [CRB04], Li et al. [LLW+06], Liu et al. [LLW08], Szathmary et al. [SVNG09]). Ce sont des méthodes en largeur utilisant l'approche support-confiance, elles ont donc des limites similaires à celles d'Apriori. Plus récemment, l'algorithme DEFME (Soulet et Rioult [SR14]) offre une nouvelle méthode pour extraire efficacement les motifs fréquents. C'est un algorithme en profondeur qui étend le concept de fermeture développé dans (Han et al. [HPYM00]). Malgré son notable apport, DEFME requiert un nombre polynomial d'opérations qui ralentit la vitesse de compilation.

Observant d'après ce rapide survol état de l'art que l'immense majorité de ces travaux se sont confrontés à un problème du coût de calculs dû en partie à la structure de données utilisées. En effet, la plupart d'entre eux reposent sur une structure de type booléen dans Apriori (Agrawal et Srikant [AS94]) nécessitant un grand nombre d'accès à la base de données, ce qui complique la tâche d'exécution lorsque la base de données est volumineuse ou dense. Pour y faire face, nous proposons une nouvelle approche d'extraction optimisée des motifs fréquents, fondée sur une nouvelle structure de données, notée `MatriceSupport`, et à des concepts de générateur minimal afin d'éviter les accès répétitifs et coûteux à la base de données. Nous proposons également un nouvel algorithme, appelé EOMF (Bemarisika et Totohasina, 2016 [BT16a]), qui s'inscrit dans la lignée des travaux cités ci-dessus, en l'occurrence l'algorithme historique Apriori (Agrawal et Srikant, 1994 [AS94]). Il s'en distingue cependant sur deux points principaux : (*i*) comptage de support et (*ii*) stratégie d'élagage.

Le chapitre est organisé de la manière suivante. La section 4.2 décrit le cadre théorique de notre modèle d'optimisation de la rechercche des motifs fréquents. L'algorithme proposé est présenté dans la section 4.3. L'évaluation expérimentale est synthétisée dans la section 4.4. En conclusion (section 4.5), nous présentons le bilan et les perspectives de notre approche.

4.2 Extraction optimisée des motifs fréquents

Il est bien connu que l'étape la plus complexe et la plus consommatrice en temps d'exécution est celle de l'extraction des motifs fréquents. Nous décrivons ici la théorie globale de notre modèle, qui étend comme nous l'avons dit les travaux d'Agrawal (Agrawal et al., 1993 [AIS93] ; Agrawal et Srikant, 1994 [AS94]). Nous définissons dans un premier temps le cadre théorique de l'optimisation proposée (sous-section 4.2.1). Nous présentons dans un second temps la nouvelle structure de données `MatriceSupport` (sous-section 4.2.2).

4.2.1 Cadre théorique de comptage des supports

Dans un souci de simplification d'écriture et sans nuire à la compréhension du lecteur, nous noterons par la suite l'itemset {ABC} par ABC, et par $supp$ le support réel d'Apriori (Agrawal et Srikant, 1994 [AS94]).

Définition 31. *Deux motifs X_1 et X_2 sont équivalents ($X_1 \approx X_2$) s'ils ont même ensemble support.*

Proposition 2. *La relation « $\approx$ » est une relation d'équivalence sur $\mathcal{I}$. La classe d'équivalence du motif X est notée $[X]$.*

Définition 32. *Classe d'équivalence (Bastide et al., 2000 [BTP+00]) : l'opérateur de fermeture γ induit une relation d'équivalence sur l'ensemble des parties de $\mathcal{I}$, i.e. l'ensemble des parties est subdivisé en des sous-ensembles, appelé aussi classe d'équivalence. Dans chaque classe, tous les éléments possèdent la même fermeture : soit $X \subseteq \mathcal{I}$, la classe d'équivalence de X est définie par $[X] = \{X_1 \subseteq \mathcal{I} | \gamma(X_1) = \gamma(X)\}$. Les éléments de $[X]$ ont ainsi la même valeur de support.*

La définition d'une classe d'équivalence nous amène à celle d'un générateur.

Définition 33. *Un motif X est générateur s'il est minimal dans sa classe d'équivalence :*

$$\forall X' \in [X], X' \subseteq X \Rightarrow supp(X') = supp(X)$$

Exemple 14. *Dans la base $\mathcal{B}$ du tableau 2.2, les motifs $\{AB\}$, $\{ABC\}$ et $\{ABCE\}$ sont dans la même classe d'équivalence. Donc, $\{AB\}$ est le motif générateur.*

Définition 34. *Générateur minimal (Bastide et al., 2000 [BTP+00], Hamrouni et al., 2011 [HYN11]) : soit un motif X fermé et $[X]$ sa classe d'équivalence. L'ensemble $\mathcal{GM}_X$ des générateurs minimaux de X est défini comme suit :*

$$\mathcal{GM}_X = \{A \in [X] | \nexists A' \subset A \text{ tel que } A' \in [X]\}$$

Les notions de motif générateur développées ci-après sont utilisées pour rendre notre algorithme plus efficace.

Proposition 3. *Pour tous $X_1, X_2 \in \mathcal{I}$, on a : $supp(X_1 \cup X_2) = supp(X_1) \cap supp(X_2)$.*

Proposition 4. *Quels que soient les ensembles d'attributs A, B, C de $\mathcal{I}$, si $A \approx B$ alors $A \cup C \approx B \cup C$.*

Démonstration. Si $A \approx B$ alors $supp(A) \cap supp(C) = supp(B) \cap supp(C)$. La proposition 3 implique que $supp(A \cup C) = supp(B \cup C)$, et donc que $A \cup C = B \cup C$. □

Proposition 5. *Les motifs contenus dans un motif générateur sont tous générateurs.*

Démonstration. Soient deux motifs X_1 et X_2 vérifiant $X_1 \subset X_2$. Il existe un ensemble d'attributs Y_1 non vide et disjoint de X_1 tel que $X_2 = X_1 \cup Y_1$. Si X_1 est supposé être non générateur, alors X_1 admet un sous-ensemble propre Y_2 qui lui est équivalent : $Y_2 \subset X_1$ et $Y_2 \approx X_1$. La proposition 4 entraîne que $Y_2 \cup Y_1 \approx X_1 \cup Y_1$. De plus, $X_1 \cap Y_1 = \emptyset$ donc $Y_2 \cup Y_1 \subset X_1 \cup Y_1$. X_2 étant équivalent à un sous-ensemble propore $Y_2 \cup Y_1$, il est donc non générateur. La contraposée donne le résultat. □

Proposition 6. *Le support d'un motif X non générateur est égal au minimum de support des motifs inclus strictement dans celui-ci : $supp(X) = min\{supp(X')|X' \subset X\}$*

Démonstration. Soit l'ensemble $\mathcal{I}'$ des motifs inclus dans X. Soit X_1 un motif de $\mathcal{I}'$ qui minimise le support dans cet ensemble. Du fait de la croissance du support, on a $X_1 \subseteq X$ entraîne $supp(X) \leq supp(X_1)$. Par ailleurs, X est non générateur : il existe donc un motif $X' \in \mathcal{I}'$ tel que $supp(X') = supp(X)$. Or, $supp(X_1)$ minimal, donc $supp(X_1) \leq supp(X')$. Finalement, $supp(X) = supp(X_1) = min\{supp(X')|X' \subset X\}$. □

Le support de ce type de modèle est parfois appelé support estimé, noté *supp-estimé*.

Proposition 7. *Un motif X générateur minimal a un support strictement inférieur à celui de ses sous-ensembles : $supp(X) < min\{supp(X')|X' \subset X\}$*

Démonstration. Si X est générateur, alors $\forall X' \in \mathcal{I}, X' \subset X \Rightarrow supp(X) \subset supp(X')$, donc $supp(X) < supp(X')$. Le passage au minimum sur l'ensemble fini des minorants X' donne $supp(X) < min\{supp(X')|X' \subset X\}$. Et, si $supp(X) < min\{supp(X')|X' \subset X\}$, alors $supp(X) \neq min\{supp(X')|X' \subset X\}$. La contraposée de la proposition 6 implique que X est générateur. □

Définition 35. *Un motif X non fréquent minimal est un motif non fréquent dont tous les sous-motifs sont fréquents : $\forall X' \subset X, supp(X') \geq minsupp$.*

Proposition 8. *Un motif non fréquent minimal est un motif générateur.*

Démonstration. Tout motif inclus strictement dans un motif non fréquent minimal X est fréquent et ne peut donc pas être équivalent à X. Le motif X est donc minimal dans sa classe d'équivalence et donc générateur. □

L'économie du coût de calculs est particulièrement fonction de la *stratégie d'élagage*. Nous avons adopté les techniques d'élagage (cf. propositions 9 et 10) ci-dessous, qui sont conformes de celles d'Apriori (Agrawal et Srikant, 1994 [AS94]).

Proposition 9. *Tout sous-ensemble d'un itemset estimé fréquent est fréquent. Et tout sur-ensemble d'un itemset estimé non féquent est aussi non fréquent.*

Démonstration. Puisque $X' \subset X$, on a $supp(X) \leq supp(X') \Leftrightarrow supp(X) \leq min\{supp(X')|X' \subset X\}$. Comme X est fréquent, alors $minsupp \leq supp(X) \leq min\{supp(X')|X' \subset X\}$, donc le sous-ensemble X' est aussi fréquent. Et si X est un sur-ensemble de X', on a $supp(X) \leq supp(X')$ équivaut à $supp(X) \leq min\{supp(X')|X' \subset X\}$. Puisque X' est estimé non fréquent ($min\{supp(X')|X' \subset X\} < minsupp$), alors le sur-ensemble X est aussi non fréquent. □

Proposition 10. *Tout sous-ensemble d'un motif générateur est générateur. Et tout sur-ensemble d'un motif non générateur est aussi non générateur.*

La preuve est évidente en utilisant celle que nous l'avons vu dans proposition 5.

4.2.2 Structure de données utilisée

Nous avons introduit une nouvelle structure de données, appelée `MatriceSupport`, qui est une projection de la base $\mathcal{B}$ par rapport à ses attributs. La nouvelle structure a été privilégiée dans plusieurs travaux, tels que (Zaki et Hsiao [ZH02], Szathmary [Sza06]). L'idée générale est d'acquérir les données au fur et à mesure de la structure et de les stocker. La spécificité de cette structure réside du fait qu'elle permet de calculer les supports des 1-motifs et des 2-motifs en une seule passe à la base de données, étant donnée que la famille de ces motifs constitue la majorité des candidats. En pratique, comme la projection est symétrique, ladite matrice doit être triangulaire afin d'éviter un problème de redondance. A cet effet, seule une demie matrice (supérieure) doit être stockée. A chaque attribut correspond donc une cellule de celle-ci dans laquelle on associe la fréquence, notée m_{ij}, qui représente le nombre de fois que l'item m_j apparaît avec l'item m_i, où i (resp. j) dénote la $i^{\text{è}}$ ligne (resp. $j^{\text{è}}$ colonne) de cette base projetée. Une fois la structure constituée, les supports des motifs candidats sont extraits de celle-ci, c'est-à-dire chacun de ses attributs projetés est ensuite utilisé pour trouver leur support. Pour ce faire, la diagonale de la matrice triangulaire est utilisée pour récupérer les supports des 1-itemsets, la partie supérieure pour les 2-itemsets. Le tableau 4.1 ci-dessous illustre le formalisme de la structure `MatriceSupport` sur la base $\mathcal{B}$.

Base de données $\mathcal{B}$	
TID	Attributs
1	ACD
2	BCE
3	ABCE
4	BE
5	ABCE
6	BCE

MatriceSupport					
i/j	A	B	C	D	E
A	3	2	3	1	2
B	-	5	4	0	5
C	-	-	5	1	4
D	-	-	-	1	0
E	-	-	-	-	5

Tableau 4.1 – Formalisme de la structure `MatriceSupport` sur la base $\mathcal{B}$

Dans ce cas, le support d'un motif X peut se calculer de la propriété 11 ci-après.

Proposition 11. *Pour tout motif X de longueur 1 ou 2, son support est défini par :*

$$supp(X) = \frac{MatriceSupport[i,j]}{|\mathcal{B}|} = \frac{m_{ij}}{|\mathcal{B}|}$$

Par exemple, le support des motifs A, B, AB et BC est obtenu par :

$$\begin{cases} supp(A) &= MatriceSupport[1,1]/6 = m_{11}/6 = 3/6 \\ supp(B) &= MatriceSupport[2,2]/6 = m_{22}/6 = 5/6 \\ supp(AB) &= MatriceSupport[1,2]/6 = m_{12}/6 = 2/6 \\ supp(BC) &= MatriceSupport[2,3]/6 = m_{23}/6 = 4/6 \end{cases}$$

Un parcours entier de la base est effectué pour construire la structure proposée. Le calcul des supports des 1-itemsets et des 2-itemsets s'effectue également en un simple parcours. Ce qui va réduire considérablement les coûts couteux de calcul des supports.

4.3 Algorithme EOMF

Cette section présente l'algorithme EOMF, sa stratégie d'élagage, son éxecution issue d'une base de données, et son évaluation théorique.

4.3.1 Stratégie d'élagage adoptée

La méthode adoptée afin d'extraire de façon optimisée les motifs fréquents repose sur l'algorithme EOMF. A partir des supports de l'ensemble des motifs générateurs minimaux fréquents, nous pouvons, en vertu des propositions 6, 10 et 11 ci-dessus, calculer les supports de tous les motifs fréquents sans accès à la base de données $\mathcal{B}$, ni le calcul des fermetures : on ne calcule pas les supports des candidats dès qu'on connaît qu'ils ne sont pas générateurs. Pour les motifs générateurs, EOMF fonctionne comme Apriori, il passe au contexte $\mathcal{B}$ pour déterminer les supports. En général, ces motifs sont dans la pratique moins nombreux que les non générateurs, et sont généralement à des niveaux inférieurs. Cela veut dire qu'à un certain niveau, tous les candidats peuvent être des candidats non générateurs. A cet effet, l'algorithme n'a plus à accéder à la base de données $\mathcal{B}$.

En outre, nous avons utilisé d'autres stratégies d'élagage : (i) Recherche des k-itemsets fréquents ($k < 3$) en utilisant la nouvelle structure `MatriceSupport` ; (ii) Recherche des k-itemsets fréquents ($k \geq 3$) à partir de l'itération précédente. Dans le premier point, nous déterminons tout d'abord les supports des motifs de taille 1 et ceux de taille 2 via la structure `MatriceSupport`. Une fois le calcul terminé, vient ensuite la recherche des motifs fréquents. En effet, les 2-motifs sont déduits des 1-motifs fréquents, en tenant compte les caratéristiques des générateurs minimaux, seulement au niveau d'élagage. Lors du second point, nous calculons pour chaque itération le support des k-itemsets candidats, puis on élague ceux qui sont infréquents. L'algorithme procède ainsi en deux sous-étapes. La première consiste à générer les k-motifs, i.e. à l'itération k, les candidats sont engendrés à partir des motifs fréquents de l'étape $(k-1)$. La génération de tels candidats s'effectue par auto-jointure des $(k-1)$-itemsets générés de la précédente itération. La seconde sous-étape est une étape d'élagage. Dans ce cas, pour chaque motif généré de l'étape de jointure, il faut tester si tous les motifs sous-ensembles sont fréquents. De plus, il faut tester comme nous l'avons mentionné ci-dessus si les motifs sont générateurs ou non : si le motif est générateur, l'algorithme passe à la base, sinon on n'a pas besoin l'accès à la base de données en utilisant la proposition 6.

4.3.2 Présentation de l'algorithme EOMF

L'algorithme 22 EOMF prend en entrée un contexte $\mathcal{B}$, un *minsupp* support minimum, et donne en sortie la liste $\mathcal{F}_k$ des motifs fréquents. D'un point vue technique, EOMF parcourt en largeur l'espace de recherche. Sa principale originalité réside au fait qu'il permet générer un ensemble des motifs fréquents en une seule passe à la base $\mathcal{B}$. Notons par $X.supp$ le support de X, C_k l'ensemble des k-itemsets candidats, $\mathcal{CGM}_k$ l'ensemble des générateurs minimaux.

Algorithm 22 Algorithme EOMF

Require: Une base de données $\mathcal{B}$, une liste d'items $\mathcal{I}$, un seuil *minsupp*.
Ensure: Un ensemble $\mathcal{F}_k$ des k-itemsets fréquents.

1: $MatriceSupport \leftarrow ConstruireBase(\mathcal{B}, \mathcal{I})$; //Construire la base de données $\mathcal{B}$
2: $\mathcal{F}_1 \leftarrow \{c_1 \in MatriceSupport | c_1.supp \geq minsupp\}$; //Générer les 1-itemsets
3: $\mathcal{F}_2 \leftarrow \{c_2 \in MatriceSupport | c_2.supp \geq minsupp\}$; //Générer les 2-itemsets
4: **for** $(k = 3; \mathcal{F}_{k-1} \neq \emptyset; k++)$ **do**
5: $C_k \leftarrow$ EOMF-Gen$(\mathcal{F}_{k-1})$; //Générer les motifs candidats
6: **for all** (candidat $c \in C_k$) **do**
7: **if** $(c \in \mathcal{CGM}_k)$ **then**
8: **for all** (transaction $t \in \mathcal{B}$) **do**
9: $C_t \leftarrow$ subset(C_k, t) ; //Sélectionner les candidats C_k présents dans t
10: **for all** (candidat $c \in C_t$) **do**
11: $c.supp++$;
12: **end for**
13: **end for**
14: **else**
15: $c.supp \leftarrow min\{c'.supp | c' \subset c\}$;
16: **end if**
17: $\mathcal{F}_k \leftarrow \mathcal{F}_k \cup \{c\}$; //Générer les fréquents
18: **end for**
19: **end for**
20: **return** $\bigcup_k \mathcal{F}_k$;

Détaillons maintenant chacune des lignes de l'algorithme 22. La base $\mathcal{B}$ est construite par la fonction `ConstruireBase` (ligne 1) pour avoir la structure `MatriceSupport`. Une fois $\mathcal{B}$ construite, les ensembles respectifs $\mathcal{F}_1$ et $\mathcal{F}_2$ des 1-itemsets et des 2-itemsets fréquents sont ensuite générés en une seule passe à la base de données (lignes 2 et 3), au lieu de deux passes pour Pascal (Bastide et al., 2002 [BTP+02]) et son ancêtre Apriori (Agrawal et Srikant, 1994 [AS94]). La boucle principale (lignes 4 à 19) est quasiment similaire à celle d'Apriori. En effet, EOMF-Gen (ligne 5) est appelé pour générer les motifs candidats. Dans ce cas, pour chaque élément c de C_k, la procédure EOMF-Gen parcourt ensuite deux étapes récursives suivantes. Si c est un candidat générateur (lignes 7 à 13), alors un accès au contexte d'extraction permettra de calculer les supports réels des candidats retenus dans $\mathcal{CGM}_k$ (lignes 10 à 12), i.e. pour toute instance t du contexte $\mathcal{B}$, l'ensemble C_t des k-iemsets candidats qui sont contenus dans t est déterminé (ligne 8) et le support de chacun de ces itemsets est incrémenté (ligne 11). Sinon, c est non générateur (lignes 14 à 16), son support

est alors le support estimé de c (cf. proposition 6) qui est égal au minimum des supports de ses sous-ensembles de taille k (ligne 15). Dans ce cas, aucun accès à la base de données n'est effectué pour calculer le support des motifs candidats. Le candidat ainsi retenu est ajouté dans l'ensemble $\mathcal{F}_k$ des motifs fréquents (ligne 17). Après l'exécution de ces deux phases, l'algorithme retourne l'ensemble des motifs fréquents de taille k (ligne 20), et il s'arrête lorsqu'il n'y a plus des motifs qui puissent être générés.

La procédure EOMF-Gen (cf. algorithme 23) fonctionne pratiquement comme celle de Apriori-Gen. A cet effet, EOMF-Gen prend en entrée $\mathcal{F}_{k-1}$, ensemble des (k-1)-motifs fréquents et retourne l'ensemble C_k de k-motifs candidats.

Algorithm 23 Procédure EOMF-Gen

Require: Ensemble $\mathcal{F}_{k-1}$ de $(k-1)$-itemsets fréquents
Ensure: Ensemble C_k de k-itemsets candidats

```
1: for all itemset p ∈ F_{k-1} do
2:   for all itemset q ∈ F_{k-1} do
3:     if (p[1] = q[1], ..., p[k-2] = q[k-2], p[k-1] < q[k-1]) then
4:       c ← p ∪ q(k-1) ; //Etape de jointure : Générer les candidats
5:       for all (itemset candidat c ∈ C_k) do
6:         for all ((k-1)-sous-ensemble s de c) do
7:           if (s ∉ F_{k-1}) then
8:             Delete c from C_k ;
9:           else
10:            c.supp ← min{c.supp, s.supp|s ⊂ c} ;
11:            if (s ∉ CGM_k) then
12:              candidat=faux ;
13:            end if
14:          end if
15:        end for
16:        if (c ∉ CGM_k) then
17:          c.supp ← min{c'.supp|c' ⊂ c} ;
18:        end if
19:      end for
20:    end if
21:  end for
22: end for
23: return ⋃_k C_k ;
```

La procédure EOMF-Gen procède comme nous l'avons déjà dit en deux étapes. La première étape consiste à générer les motifs candidats. A l'itération k, les candidats sont engendrés à partir des motifs fréquents de l'étape $(k-1)$. La génération des k-itemsets candidats est donc effectuée par auto-jointure des $(k-1)$-itemsets générés dans l'itération précédente. Par exemple, la jointure des motifs ABC et ABD donne $ABCD$, par contre, la jointure de ABC et CDE ne donne rien, car il n'y a pas $(k-2)$ attributs en commun. La seconde étape consiste à son tour à élaguer les motifs non fréquents : pour chaque motif généré par l'étape de jointure, il faut tester à l'itération k si tous les $(k-1)$-motifs sous-ensembles sont

fréquents, i.e. vérifier s'ils sont présents dans $\mathcal{F}_{k-1}$. En plus de la jointure et de l'élagage, nous testons pour chaque candidat c de C_k s'il est générateur ou non (lignes 5 à 21). A cela, l'algorithme procure en deux sous-phases. La première sous-phase consiste, pour chaque sous-ensemble s de c, à tester que si s soit fréquent ou non (lignes 7 à 14). Si s est non fréquent ($s \notin \mathcal{F}_{k-1}$), alors c est éliminé (lignes 7 à 9), sinon le support de ce même candidat est égal au minimum de support de s et de son support lui-même (ligne 10). En même temps, nous testons si s est générateur ou non (lignes 11 à 13). Lors de la deuxième sous-phase, nous testons pour chaque élément c de C_k s'il est générateur ou non (lignes 16 à 18). Cette procédure EOMF-Gen retourne l'ensemble C_k des motifs candidats de taille k (ligne 23).

4.3.3 Exemple d'exécution d'EOMF sur $\mathcal{B}$, à un $minsupp = 2/6$

Nous notons par Gen. le générateur minimal, par supp. le support d'un motif.

Matricesupport					
i/j	A	B	C	D	E
A	3	2	3	1	2
B	-	5	4	0	5
C	-	-	5	1	4
D	-	-	-	1	0
E	-	-	-	-	5

1er élagage

Générer $\mathcal{F}_1$			Générer $\mathcal{F}_2$		
1-motif	Gen.	supp.	2-motif	Gen.	supp.
A	oui	3/6	AB	oui	2/6
B	oui	5/6	AC	non	3/6
C	oui	5/6	AE	oui	2/6
D	oui	1/6	BC	oui	4/6
E	oui	5/6	BE	non	5/6
			CE	oui	4/6

Générer C3		
3-motif	Gen.	supp.
ABC	non	min(2/6,3/6,4/6)=2/6
ABE	non	min(2/6,2/6,5/6)=2/6
ACE	non	min(3/6,2/6,4/6)=2/6
BCE	non	min(4/6,5/6,4/6)=4/6

2e élagage

Générer $\mathcal{F}_3$		
3-motif	Gen.	supp.
ABC	non	2/6
ABE	non	2/6
ACE	non	2/6
BCE	non	4/6

Générer C4		
4-motif	Gen.	supp.
ABCE	non	min(2/6,2/6,4/6)=2/6

3e élagage

Générer $\mathcal{F}_4$		
4-motif	Gen.	supp.
ABCE	non	2/6

Tableau 4.2 – Exemple d'exécution de l'algorithme EOMF du contexte $\mathcal{B}$, $minsupp = 2/6$

La première passe de l'algorithme EOMF donne, en un seul parcours, les $\mathcal{F}_1$ et $\mathcal{F}_2$. On voit que les 1-motifs sont des générateurs et sont tous fréquents sauf D, donc élagué, ce qui donne $\mathcal{F}_1 = \{A, B, C, E\}$. Les 2-motifs sont aussi tous fréquents $\mathcal{F}_2 = \{AB, AC, AE, BC, BE, CE\}$, où AC et BE sont non générateurs (car $supp(AC) = supp(A)$ et $supp(BE) = supp(B) = supp(E)$). Nous avons ensuite l'ensemble $C_3 = \{ABC, ABE, ACE, BCE\}$ des candidats qui sont des sur-ensembles de AC et de BE, donc non générateurs, aucun parcours n'est alors

effectué. Et on voit qu'ils sont tous fréquents, donc $\mathcal{F}_3 = \{ABC, ABE, ACE, BCE\}$. Enfin, nous avons le seul candidat $ABCE$ qui est un sur-ensemble de $\mathcal{F}_3$, donc non générateur et aucun parcours n'est effectué. Puis, on voit qu'il est fréquent, ce qui donne $\mathcal{F}_4 = \{ABCE\}$. Nous avons finalement l'ensemble $\mathcal{F} = \mathcal{F}_1 \cup \mathcal{F}_2 \cup \mathcal{F}_3 \cup \mathcal{F}_4$ des motifs fréquents.

Bref, Apriori l'effectue en 4 parcours de 4 passes au contexte $\mathcal{B}$, Pascal l'effectue en 3 parcours de 2 passes, notre algorithme Eomf l'effectue en 3 parcours d'une seule passe. La différence importante réside donc au fait qu'Eomf permet de générer l'ensemble des motifs fréquents en une seule passe, tandis que Pascal l'effectue au moins en 2 passes, et Apriori l'effectue en plusieurs fois possible.

4.3.4 Complexité de l'algorithme Eomf

Avant d'entrer dans une phase d'évaluation empirique d'un algorithme, nous pouvons chercher à évaluer de manière théorique la quantité d'appels aux opérations les plus coûteuses en temps. L'estimation du temps d'exécution se fait généralement en trois optiques : *au pire des cas*, *au mieux*, et *à la moyenne*. La première consiste à évaluer la durée maximale afin de contrôler le déroulement d'un programme donné. C'est le cas le plus utilisé dans la communauté scientifique. La seconde consiste à estimer le temps d'exécution minimum, dans le cas le plus favorable. En pratique, cette complexité n'est pas très utile. La dernière consiste à évaluer la durée moyenne d'exécution, fondée sur des modèles probabilistes complexes. Elle se revèle souvent très difficile à mettre en œvure et sort du cadre de ce travail. Nous nous intéressons, tout au long de ce mémoire, à la première méthode, car nous voulons borner le temps d'exécution.

Proposition 12. *La complexité en temps de l'algorithme 22 est, dans le pire cas, $\mathcal{O}(m2^n)$, où m est le nombre de transactions, et n le nombre d'attributs de la base $\mathcal{B}$.*

Démonstration. Au niveau k, le coût du calcul de support des itemsets est, dans le pire cas, en $\mathcal{O}(m|C_k|)$, où C_k est l'ensemble des k-motifs candidats. Le coût du test des candidats fréquents est, au pire des cas, en $\mathcal{O}(|C_k|) \approx \mathcal{O}(2^n)$, et le coût de la génération des candidats du niveau $(k+1)$ est $\mathcal{O}(k|C_k|)$. Si le contexte est suffisamment grand, c'est le coût du calcul de support qui domine. Ce qui donne la complexité totale de l'algorithme en $\mathcal{O}(m2^n)$. Cette quantité montre que le temps d'exécution de notre algorithme croît linéairement avec le nombre de transactions, et exponentiellement avec le nombre d'attributs. En pratique, grâce à l'optimisation introduite (cf. proposition 10), cette complexité sera beaucoup plus faible, ce qui rendra l'algorithme utilisable même si avec un contexte relativement dense. □

4.4 Evaluation expérimentale

Nous évaluons la performance d'Eomf [BT16a] comparée à celle d'Apriori [AS94] et de Pascal [BTP+02]. Les raisons sont les suivantes. Les trois algorithmes effectuent une recherche par niveau, donc sémantiquement proches, mais de processus d'élagage différent. A cet effet, Apriori utilise les contraintes d'antimonotonicité (cf. proprétés 4 et 5). Pascal introduit, en plus des contraintes d'Apriori, une nouvelle technique de comptage des supports, appelée comptage par inférence, basée sur le concept des motifs clés. Dans Eomf, en plus des contraintes d'Apriori, nous avons utilisé ladite structure `MatriceSupport`. De plus, nous exploitons aussi le concept des générateurs minimaux.

Protocole expérimental Les algorithmes sont implémentés en C++ et R. Nous avons utilisé l'implémentation de Christian B. [Bor03] pour Apriori, et librairie `rtools` pour compiler Pascal. Les expérimentations ont été réalisées sur un PC de 4 Go de RAM tournant sous Windows, menées sur quatre jeux de données qui sont présentées dans le tableau 4.3.

Base	Nombre de transactions	Nombre d'items	Taille moy. des objets
T20I6D100K	100 000	1 000	20
T25I10D10K	10 000	1 000	25
C20D10K	10 000	386	20
MUSHROOMS	8 416	128	23

Tableau 4.3 – Caractéristiques des données d'expérimentations.

Les jeux de données T20I6D100K [1] et T25I10D10K [2] sont des données synthétiques. Ils contiennent respectivement 100 000 objets d'une taille moyenne de 20 items, et 10 000 objets d'une taille moyenne de 25 items. Le jeu de données C20D10K [3] est un échantillon du fichier PUMS90KS (Public Use Microdata Samples) contenant des données du recensement du Kansas en 1990. Il contient 10 000 objets correspondant aux 10 000 premières personnes recensées, chaque objet contient 20 attributs (20 items par objets et 386 items au total). MUSHROOMS [4] décrit les caractéristiques de champignons, et contient 8 416 objets d'une taille moyenne de 23 items (23 items par objets et 128 items au total).

Temps de réponse Dans ce cas, nous faisons varier le *minsupp* (colonne *minsupp*).

Base	minsupp(%)	CPU(s)			Fréquents
		Apriori	Pascal	EOMF	
T20I6D100K	2.00	9	8	3	378
	1.00	35	33	12	1 534
	0.75	50	47	20	4 710
	0.50	75	70	30	26 836
	0.25	100	97	40	155 163
T25I10D10K	2.00	8	4	2	2 543
	1.00	50	30	8	3 300
	0.75	85	50	15	17 583
	0.50	120	85	35	331 280
	0.25	150	120	50	2 270 573
C20D10K	20.0	30	11	8	20 239
	15.0	75	25	15	36 359
	10.0	120	40	30	89 883
	7.5	210	120	40	153 163
	5.0	340	175	50	352 611
MUSHROOMS	20.0	25	20	12	53 337
	15.0	150	40	20	99 079
	10.0	350	75	45	600 817
	7.5	750	100	80	936 247
	5.0	1000	140	95	4 140 453

Tableau 4.4 – Résultats expérimentaux sur 4 bases de données.

1. http ://www.almaden.ibm.com/cs/quest/syndata.html
2. http ://www.almaden.ibm.com/cs/quest/syndata.html
3. ftp ://ftp2.cc.ukans.edu/pub/ippbr/census/pums/pums90ks.zip
4. ftp ://ftp.ics.uci.edu/pub/machine-learning-databases/mushroom/agaricus-lepiota.data

Pour chacune des bases, nous avons restitué le temps CPU exprimé en secondes (colonne CPU(s)) dans lesquels les sous-colonnes Apriori , Pascal, EOMF représentent respectivement les temps d'exécution d'Apriori, de Pascal et d'EOMF. La dernière colonne (colonne Fréquents) représente le nombre des motifs fréquents extraits pour chacun de trois algorithmes. La figure 4.1 compare le temps d'exécution de notre algorithme à celui offert par Apriori et Pascal, selon le protocole de l'expérience ci-dessus.

A la lumière de ces résultats, constatons que le temps de calcul pour chacun des algorithmes est très sensible à la valeur de support. Il diminue rapidement lorsque le support augmente, ceci est dû au fait que le nombre de motifs est réduit.

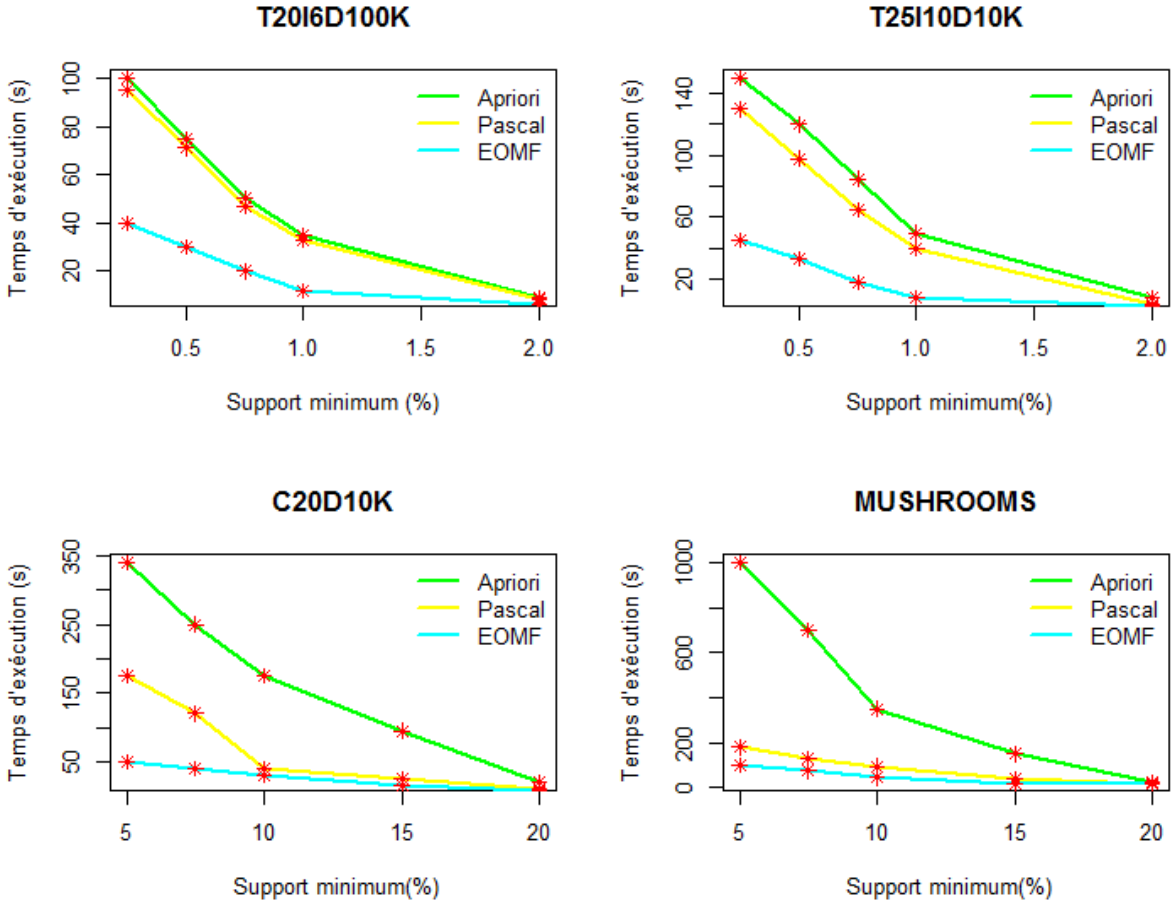

Figure 4.1 – Temps d'exécution en fonction de *minsupp* pour les quatre bases

Par ailleurs, le comportement des algorithmes varie fortement selon les caractéristiques du jeux de données utilisé. A cet effet, les données faiblement corrélées, telles que T20I6D100K et T25I10D10K, constituent des cas faciles pour l'extraction, car elles présentent peu de motifs fréquents. Dans ce cas, même si avec un *minsupp* assez faible, le temps d'exécution pour chacun des trois algorithmes reste encore raisonnable. Pour le jeu de données T20I6D100K, Apriori et Pascal se comportent de manière quasiment similaire avec un léger avantage pour Pascal. Notre algorithme EOMF, quant à lui, donne de temps largement inférieur pour

toutes les valeurs de *minsupp*. Pour la base T25I10D10K, Apriori donne de temps relativement supérieur dû au nombre de passes sur le contexte de données. Le temps d'exécution de notre algorithme EOMF, sur cette base de données, est toujours inférieur à celui de Pascal, qui lui-même est inférieur à celui d'Apriori.

Les données corrélées et denses (C20D10K et MUSHROOMS) constituent des cas plus difficiles du fait de l'importante proportion des motifs, contrairement à ce qui se passe pour les données faiblement corrélées. Dans ce cas, le temps de réponses obtenu (cf. figure 4.1) varie fortement selon l'algorithme utilisé. A cet effet, Apriori donne de temps largement supérieur dû au grand nombre d'accès à la base de données. Tandis que notre algorithme EOMF donne de temps très inférieur, donc plus efficace. L'algorithme Pascal, pour le jeu de données C20D10K, donne l'allure typiquement particulière, fournit de temps d'exécution relativement suppérieur pour le *minsupp* allant de 5% à 7.5%, et rejoint l'EOMF sur le seuil de 10% à 20%. Pour la base MUSHROOMS, Pascal et EOMF se comportent quasiment similaire pour toutes les valeurs de support, et cela est toujours au profit de notre algorithme EOMF.

Ces différentes performances peuvent être expliquées par le fait que ces deux jeux de données sont fortement corrélées, ceci complique la tâche d'Apriori au niveau du calcul des supports dû au nombre de passes au contexte. EOMF évite ce considérable côut grâce aux différents points d'optimisations introduites. Il en est de même pour Pascal grâce aux concepts de comptage par inférence. Néanmoins, ses performances tendent parfois à diminuer lorsque le seuil de support *minsupp* est faible, à cause du nombre de passes sur la base pour l'extraction des 1-motifs et 2-motifs, qui constituent généralement un volume important du candidat.

4.5 Conclusion partielle et perspectives

Dans ce travail, nous avons proposé une nouvelle approche permettant l'extraction optimisée des motifs fréquents dans une base de données transactionnelles. Nous y avons défini une nouvelle optimisation de comptage des supports et un nouvel algorithme AOMF qui en résulte. Nous avons conduit des expérimentations menées sur quelques bases de données de réference. Nos résultats comparés à ceux obtenus par Apriori (Agrawal et Srikant [AS94]) et Pascal (Bastide et al. [BTP+02]) semblent compétitifs pour chacune de ces quatre bases d'expérimentation : les temps d'exécution de notre algorithme EOMF surpassent clairement ceux d'Apriori et de Pascal. Il ressort que notre modèle est capitale pour la robustesse de l'algorithme d'extraction des motifs fréquents. Au delà de la vitèsse de compilation, ces expérimentations ont permis de juger positivement de la validité de notre approche, au niveau de l'extraction des motifs fréquents.

En guise des perspectives, nous souhaitons étendre notre approche en penchant les concepts de bordures négatives et de générateurs minimaux. La combinaison de ces deux concepts pourrait encore améliorer la robustesse de notre algorithme EOMF. Dans notre approche, nous avons limité à l'étude des motifs positifs, nous n'avons pas pu étudier les motifs négatifs de type $\overline{X}$, $\overline{X}Y$, $X\overline{Y}$, $\overline{X}\ \overline{Y}$, alors que des nouvelles connaissances importantes peuvent être cachées dans ce genre, ce qui serait une piste pour des travaux futurs. Une fois le modèle établi, son extension au problème d'extraction des règles d'association pertinentes vient ensuite naturelle, qui nous donne aussi une autre piste intéressante.

Chapitre 5

Optimisation de la génération des règles d'association positives et négatives valides

5.1 Introduction et motivations

La génération des règles d'association est une seconde phase de la découverte des règles d'association (Agrawal et al. [AIS93]) pertinentes. Comme nous le savons, ce sujet a été résolu par l'algorithme Apriori (Agrawal et Srikant [AS94]), et depuis, il trouve un développement notable. A notre connaissance, la plupart des travaux proposés se concentrent sur l'extraction des règles d'association *classiques (ou positives)*, assez peu d'approches s'intéressent sur les règles négatives. Cependant, dans certaines situations, il peut être intéressant de prendre en compte l'absence de certains motifs, en particulier en médecine, et en didactique des disciplines. Afin d'extraire ce genre, comme le souligne Guillaume, l'approche naïve consiste à adjoindre à notre base de données, la même base mais avec la négation des motifs. Très souvent, la complexité de l'approche existante est exponentielle du fait d'un important nombre de motifs, celle-ci est dramatique en termes de coûts de calculs, sans parler du nombre prohibitif de règles redondantes et non intéressantes extraites, ce qui empêche le décideur de pouvoir exploiter les résultats.

Afin de compenser ces défauts, différentes approches ont été proposées dont nous en citerons quelques-unes. Dans (Brin et al., 1997 [BMS97]), une technique d'extraction des règles de corrélation contenant des attributs négatifs, à l'aide de la statistique de Khi-carré χ^2 (Pearson, K., 1900 [Pea00]), est élaborée. Savasere et al. [SON98] proposent une nouvelle méthode d'extraction des règles d'association négatives à partir des motifs fréquents [1]. C'est une méthode utilisant le couple support-confiance d'Apriori ([AS94]). Boulicaut et al. [BBJ00] proposent une approche par extraction contrainte des motifs généralisés afin d'extraire les règles de type $X \wedge Y \rightarrow \overline{Z}$ ou $\overline{X} \wedge Y \rightarrow Z$. Teng et al. [THC02] proposent une méthode permettant de générer les règles négatives de type $X \rightarrow \overline{Y}$. Wu et al. [WZZ04]) proposent une méthode d'extraction des règles d'association positives et négatives, basée sur le couple support-CPIR (Conditional Probability Increment Ratio). Antonie et Zaïane [AZ04] étudient les règles du type $X \rightarrow \overline{Y}$, $\overline{X} \rightarrow Y$, $\overline{X} \rightarrow \overline{Y}$ via le couple support-confiance. Cornelis et al. [CYZC06] proposent une nouvelle approche d'extraction des règles d'associa-

1. Rappelons qu'un motif X est dit fréquent si son support $supp(X)$ est au moins égal à un seuil minimum $minsupp$ fixé par l'utilisateur : $supp(X) \geq minsupp$.

tion positives et négatives valides, qui se limite sur l'approche support-confiance. Bouker et al. [BGYS07] proposent une nouvelle méthode d'extraction des règles généralisées contenant des règles positives et négatives, qui utilise également l'approche classique support-confiance.

La tendance actuelle de ce propos se focalise à des heuristiques algorithmiques. Ramasubbareddy et al. [RGR11] proposent un nouvel algorithme qui s'appuie aussi sur le couple support-confiance. Cet algorithme est ensuite utilisé afin de construire le modèle d'un classifieur. Malgré son apport incontestable, l'algorithme se trouve confronté à un grand coût de balayage. Peng et al. [PCW12] proposent un nouvel algorithme efficace dans le cadre de l'analyse décisionnelle. L'algorithme comprend cependant une phase d'élagage complexe, ce qui rend son usage très difficile. Plus récemment, l'algorithme RAPN (Guillaume et Papon, 20013 [GP13]) propose un nouveau modèle d'extraction optimisée des règles d'association positives et négatives, basé sur le couple support-confiance et une autre mesure complémentaire, M_{GK} modifiée [Gui10], qui perd le caractère favorablement implicatif.

Nous constatons, d'après ce rapide survol état de l'art, que l'optimisation de l'extraction des règles d'association positives et négatives valides demeure encore un défi majeur que l'on doit ainsi profiter des techniques récentes plus efficaces. L'ajout de règles négatives va augmenter exponentiellement le nombre de règles à extraire, soit quatre fois plus de l'ensemble : $4(3^m - 2^{m+1} + 1)$, où m est la taille des motifs. L'autre constat essentiel, à part des performances locales sur des données éparses [2], est que la majorité de ces approches citées se limitent à l'utilisation du couple classique support-confiance, alors que celui-ci ne suffit pas pour garantrir la qualité des résultats extraits, et a été remis en cause dans de nombreux travaux (Brin et al. [BMS97], Lallich et Teytaud [LT04], Guillet et Hamilton [GH07a], Feno et Totohasina [Fen07, Tot08], Y. Toussaint [Tou11], Ruiz [Rui14], Sourour et al. [SLS15]).

Nous proposons ainsi une nouvelle approche d'extraction des règles d'association positives et négatives réellement pertinentes en utilisant le nouveau couple support-M_{GK}. Afin d'automatiser l'extraction, nous proposons également un nouvel algorithme GenPNR-Generation of Positive and Negative association Rules, prolongeant nos travaux (Bemarisika et Totohasina, 2014 [BT14b, BT14c]). Notre démarche consiste à générer, à partir de la famille $\mathcal{F}$ des motifs fréquents, l'ensemble de ces règles positives et négatives potentiellement intéressantes en partitionnant celui-ci en deux sous-classes, *classe attractive* [3] et *classe répulsive* [4]. Ce partitionnement proprement dit est effectué selon la dépendance du couple (X, Y), soit en fonction de la classe de $X \to Y$. En effet, lorsque X favorise Y (i.e. $P(Y|X) > P(Y)$), cas où $X \to Y$ est dans la classe d'attraction, nous évaluons les règles $X \to Y$, $Y \to X$, $\overline{Y} \to \overline{X}$ et $\overline{X} \to \overline{Y}$. Lorsque X défavorise Y (i.e. $P(Y|X) < P(Y)$), cas où $X \to Y$ est dans la classe de répulsion, nous évaluons les règles $X \to \overline{Y}$, $\overline{Y} \to X$, $\overline{X} \to Y$ et $Y \to \overline{X}$.

Le reste de ce chapitre est organisé comme suit. Nous présentons dans la section 5.2 quelques motivations. La section 5.3 modélise les optimisations du parcours des règles réellement pertinentes. L'algorithme proposé est décrit dans la section 5.4. L'évaluation expérimentale est synthétisée dans la section 5.5. Nous présentons dans la section 5.6 le bilan.

2. Les coûts de calculs dans ce type de contextes pèsent lourd sur les performances de ces algorithmes.

3. C'est aussi une zone d'attraction entre deux motifs X et Y de la règle, où X favorise Y (Y favorise X, $\overline{Y}$ favorise $\overline{X}$ et $\overline{X}$ favorise $\overline{Y}$) et les règles $X \to Y$, $Y \to X$, $\overline{Y} \to \overline{X}$ et $\overline{X} \to \overline{Y}$ pourront être intéressantes.

4. C'est aussi une zone de répulsion entre deux motifs disjoints X et Y de la règle, dans laquelle X défavorise Y mais (X favorise $\overline{Y}$, $\overline{Y}$ favorise X, $\overline{X}$ favorise Y et Y favorise $\overline{X}$) et la règle $X \to Y$ ne sera donc pas intéressante, mais les règles $X \to \overline{Y}$, $\overline{Y} \to X$, $\overline{X} \to Y$ et $Y \to \overline{X}$ seront potentiellement pertinentes.

5.2 Définitions et limites de support-confiance

Nous présentons ici quelques définions de base et l'intérêt des règles négatives (Bemarisika et Totohasina, 2014 [BT14c]) en dégageant les limites de l'approche support-confiance.

Définition 36. *Une règle d'association est dite négative, si l'un au moins de deux motifs de la règle est négatif, qui peut être de type :* $X \to \overline{Y}, \overline{X} \to Y, \overline{X} \to \overline{Y}$.

En fait, en plus de générer les règles positives $X \to Y$, nous générons également les règles négatives du type $X \to \overline{Y}, \overline{X} \to Y$ et $\overline{X} \to \overline{Y}$ qui représentent les 3/4 de l'ensemble.

La découverte de ce type de connaissances peut se révéler très intéressante, particulièrement, en médecine et en didactique de mathématiques. En médecine, considérons un exemple simulé d'une base de données médicales (Szathmary et al., 2006 [SSP+06]) pour l'identification des causes de maladies cardiovasculaires (MCV). Une règle d'association fréquente telle que {niveau élevé de cholestérol}→ {MCV} permet de faire émerger l'hypothèse que les individus ayant un fort taux de cholestérol ont un risque élevé de MCV, il s'agit d'une règle de type $X \to Y$, donc des règles classiques (ou positives). Une règle $\{\overline{\text{niveau élevé de cholestérol}}\} \to \{\overline{\text{MCV}}\}$ peut valider l'hypothèse que les individus qui n'ont pas un fort taux de cholestérol n'ont pas aussi un risque élevé de MCV, c'est une règle de type $\overline{X} \to \overline{Y}$. A l'opposé, si notre base de données contient un grand nombre de végétariens, une règle d'association {végétariens}→$\{\overline{\text{MCV}}\}$ peut valider l'hypothèse que les végétariens ont une forte chance de ne pas contracter un MCV, il s'agit d'une règle de type $X \to \overline{Y}$. Une règle $\{\overline{\text{végétariens}}\} \to \{\text{MCV}\}$ permet de faire émerger l'hypothèse que les individus non végétariens ont un fort risque de contracter un MVC, il s'agit d'une règle de type $\overline{X} \to Y$.

En didactique des mathématiques, prenons un exemple portant sur l'enseignement et l'apprentissage du calcul des probabilités (CALPRO) avec pré-requis théorie des ensembles et séries entières (THENS∧SER). Une règle d'association $\{\overline{\text{THENS} \wedge \text{SER}}\} \to \{\text{trop d'erreur CALPRO}\}$ permet de faire émerger l'hypothèse que l'étudiant n'ayant pas des notions suffisantes en théorie des ensembles et séries entières a commis trop d'erreurs en calcul des probabilités. L'enseignant doit donc trouver une méthode efficace pour contourner cette situation. Il suffit, dans notre approche, de prendre la négation (rappelons que $\overline{\overline{X}} = X$), on a : $\{\text{THENS} \wedge \text{SER}\} \to \{\overline{\text{trop d'erreur CALPRO}}\}$, ce qui nous dit que l'étudiant ayant un bon niveau en théorie des ensembles et des séries entières ayant commis moins d'erreur en calcul des probabilités. Autrement dit, l'étudiant ayant un bon niveau en théorie des ensembles et en séries entières a bien maîtrisé le calcul des probabilités, ce qui correspond à la règle positive : $\{\text{THENS} \wedge \text{SER}\} \to \{\text{CALPRO}\}$. Déduite de cette dernière, une règle négative à gauche et à droite est donnée par $\{\overline{\text{THENS} \wedge \text{SER}}\} \to \{\overline{\text{CALPRO}}\}$; ce qui permet de faire émerger l'hypothèse que l'étudiant n'ayant pas des notions suffisantes en théorie des ensembles et en séries entières est fort probable faible en calcul des probabilités.

Extraire ce genre de modèle à l'aide des bonnes mesures de qualité est un défi majeur. La mesure *confiance*, malgré sa simplicité, est interprétable : le couple support-confiance sélectionne facilement les règles non intéressantes, et ne prend pas en compte les exemples $P(Y)$ et les contre-exemples $P(X \cap \overline{Y})$. A cela, nous avons opté pour la mesure M_{GK} qui, outre sa qualité implicative conduisant à la solution du problème d'exemples/contre-exemples, optimise le paradigme de dépendance. Les exemples ci-après, inspirés de (Lallich et Teytaud, 2004[LT04] ; Brin et al., 1997[BMS97]), permettent d'illustrer ce propos.

Exemple 15. *Supposons que l'on souhaite analyser la relation qui existe entre des personnes achetant les produits A et B. Le tableau 5.1 (Lallich et Teytaud [LT04]), montre la répartition des achats sur un groupe de 100 personnes.*

	A	$\overline{A}$	$\sum$ lignes
B	72	18	90
$\overline{B}$	8	2	10
$\sum$ colonnes	80	20	100

Tableau 5.1 – Limite de l'approche Confiance

D'après les statistiques de ce tableau 5.1, on s'ensuit que le support de la règle $A \rightarrow B$ est 0.72 et sa cconfiance de 0.9. Ces valeurs raisonnablement élevées nous invitent à considérer que les personnes qui achètent le produit A achètent également le produit B. Or, nous remarquons que la confiance de cette règle est égale à la probabilité de la partie conclusion, indépendamment de la partie hypothèse, c'est-à-dire $\text{conf}(A \rightarrow B) = p(B|A) = p(B)$. Donc, la règle $A \rightarrow B$ qui semblait intéressante selon la confiance est trompeuse ; elle est en fait non pertinente, car les motifs A et B sont indépendants.

Exemple 16. *Le scénario est présenté dans le tableau 5.2 (Brin et al. [BMS97]).*

	café	¬café	$\sum$ lignes
thé	20	5	25
¬thé	70	5	75
$\sum$ colonnes	90	10	100

Tableau 5.2 – Inconvénient de l'approche Confiance

A ce sujet, nous considérons la règle `thé` $\rightarrow$ `café`. Ce qui donne le support $supp(thé \wedge café) = 20/100 = 0.2$, et la confiance $p(café|thé) = p(thé \wedge café)/p(thé) = 20/25 = 0.8$. Ces valeurs raisonnablement élevées nous invitent à penser que les clients qui achètent le thé achètent également le café, i.e. thé *favorise* café. Pourtant, la part des clients achetant café, indépendamment du fait qu'ils achètent thé, est de p(café)=90/100 = 0.9, alors que la règle nous dit que la proportion des clients consommant thé et café est inférieure, puisqu'elle est égale à 0.8, donc une dépendance négative entre thé et café, i.e. thé *défavorise* café. Ainsi, la règle `thé`$\rightarrow$`café` est en fait non pertinente. Afin d'y remédier, nous utilisons la mesure M_{GK} (Guillaume, 2000 [Gui00]), qui est définie de la manière suivante.

Définition 37. *Soient X et Y deux motifs, la mesure M_{GK} est définie par :*

$$M_{GK}(X \rightarrow Y) = \begin{cases} \frac{P(Y|X)-P(Y)}{1-P(Y)}, & si\ X\ favorise\ Y,\ (P(Y) \neq 1) \\ \frac{P(Y|X)-P(Y)}{P(Y)}, & si\ X\ défavorise\ Y,\ (P(Y) \neq 0) \end{cases}$$

X favorise Y signifie $P(Y|X) > P(Y)$ et X défavorise Y signifie $P(Y|X) \leq P(Y)$. Indépendamment de Guillaume [Gui00], Totohasina [Tot03] et Wu [WZZ04] ont introduit

cette même mesure, sous les noms respectifs d'ION-Implicative Orientée Normalisée (où apparaissent les termes `favorise` et `défavorise`), et de CPIR-Conditional Probability Increment Ratio. La mesure M_{GK} varie dans l'intervalle $[-1, +1]$: plus la mesure s'éloigne de 0, plus X, Y sont fortement dépendants négativement ou positivement, i.e. plus elle est proche de 1 (resp. -1) plus le couple (X, Y) caractérise spécifiquement la dépendace positive (resp. négative). M_{GK} mesure la distance de la "confiance" par rapport à l'indépendance. Si

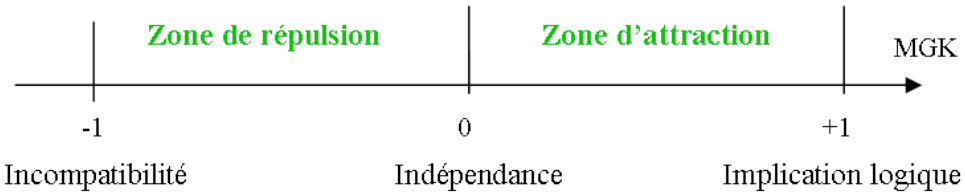

Figure 5.1 – Situations de référence pour M_{GK}

$M_{GK}(X \to Y) = -1$, alors X et Y sont incompatibles, ce qui correspond à la répulsion limite entre X et Y. Si $M_{GK}(X \to Y) = 0$, alors $X \to Y$ n'est pas intéressante, ce qui caractérise l'indépendance stochastique entre X et Y. Si $M_{GK}(X \to Y) = 1$, alors X *favorise* Y, ce qui dénote l'attraction forte entre X et Y.

Avec le même exemple du tableau 5.2 ci-dessus, on a $M_{GK}(thé \to café) = -0.2$, c'est une dépendance négative entre thé et café, donc thé défavorise café, ce qui confirme la non pertinence de cette règle thé→café.

Propriété 7. *Si* $X_1 \subseteq \ldots \subseteq X_i \subseteq X_{i+1} \subseteq X_p$ *:* $M_{GK}(X_1 \to X_p) = \prod_{i=1}^{p-1} M_{GK}(X_i \to X_{i+1})$

La mesure M_{GK} est donc multiplicative sur une chaîne du treillis de motifs. La démonstration de cette propriété 7 est disponible dans (Totohasina et Feno, 2008 [TF08, Tot08]).

Remarque 2. Remarque importante sur la pratique avec support-M_{GK}. *De façon simplifiée, pour tous* X, Y *deux motifs de* $(\mathcal{I}, \mathcal{R}, \mathcal{T})$*, si* X *favorise* Y*, nous considérons alors les règles* $X \to Y$*,* $Y \to X$*,* $\overline{Y} \to \overline{X}$ *et* $\overline{X} \to \overline{Y}$*, sinon les règles* $X \to \overline{Y}$*,* $\overline{Y} \to X$*,* $\overline{X} \to Y$ *et* $Y \to \overline{X}$*. Notons ainsi que la composante défavorisante de cette mesure* M_{GK} *n'est pas active dans l'extraction des règles d'association support-*M_{GK} *valides. Elle indique juste s'il convient d'étudier des règles d'association négatives ou pas.*

Notre préférence en ladite mesure M_{GK} originelle est justifiée par ses propriétés mathématiques, notamment le caractère implicatif de sa seule composante favorisante alors que, comme cela a été mentionné ci-dessus, sa composante défavorisante joue juste un rôle de détecteur des règles négatives. A signaler que la mesure M_G (Guillaume et Papon, 2010 [Gui10]), M_{GK} modifiée, perd une telle qualité (cf. Ramanantsoa et Totohasina, 2015 [RT15]).

Définition 38. *Une règle* $X \to Y$ *est potentiellement intéressante, si* $P(Y|X) > P(Y)$*, qui correspond à* $M_{GK}(X \to Y) > 0$*. Elle est dite non intéressante, si* $P(Y|X) \leq P(Y)$*, ce qui équivaut à* $M_{GK}(X \to Y) \leq 0$*.*

Une règle d'association intéressante n'implique pas qu'elle soit valide. En effet, sa validité est arbitraire selon un seuil fixé par l'utilisateur. Afin de la valider, nous introduisons une contrainte, appelée valeur critique, qui sera définie de la manière suivante.

Etant données une base de données $\mathcal{B}$ de n transactions et une règle d'association $X \to Y$, dans laquelle n_X et n_Y sont des fréquences respectives des motifs X et Y. Pour ce faire, nous considérons un tableau de contingence par le croisement des motifs X et Y, s'appuyant sur la statistique de Khi-carré (χ^2) de Pearson à un degré de liberté, tel que :

$$M_{GK}(X \to Y) = \begin{cases} \sqrt{\frac{1}{n}\frac{n-n_X}{n_X}\frac{n_Y}{n-n_Y}\chi^2}, & \text{si } X \text{ favorise } Y \\ -\sqrt{\frac{1}{n}\frac{n-n_X}{n_X}\frac{n-n_Y}{n_Y}\chi^2}, & \text{si } X \text{ défavorise } Y \end{cases}$$

A partir de la table de χ^2, cette relation permet de nous donner les abaques des valeurs critiques d'une règle d'association M_{GK}-valide. Au risque d'erreur α pris dans l'intervalle réel $[0, 1]$, ladite valeur critique, cas où X favorise Y, notée par soucis de simplicité ξ_α, pourrait également être définie par :

$$\xi_\alpha = \sqrt{\frac{1}{n}\frac{n-n_X}{n_X}\frac{n_Y}{n-n_Y}\chi^2_\alpha} = \sqrt{\frac{n-n_X}{n_X}\frac{n_Y}{n-n_Y}}\phi_\alpha(X \to Y),$$

un seuil de prise en compte de la qualité implicative de la règle $X \to Y$, où ϕ_α est la fonction de Laplace-Gauss à un α fixé. Par exemple, au seuil de signification 95% (i.e. au risque d'erreur $\alpha = 5\%$), la valeur critique associée de $\chi^2_{0.05}$ étant 3.84, ce qui donne M_{GK} critique

$$\xi_{0.05} = \sqrt{\frac{1}{n}\frac{n-n_X}{n_X}\frac{n_Y}{n-n_Y}3.84} = \sqrt{\frac{n-n_X}{n_X}\frac{n_Y}{n-n_Y}}\frac{1.96}{n}, \text{ car } \chi^2_\alpha = n\phi^2_\alpha(X \to Y)$$

Dans notre approche, une règle d'association $X \to Y$ est donc dite *valide* si $X \cup Y$ est fréquent au sens de la mesure *support*, et $M_{GK}(X \to Y) \geq \xi_\alpha$. Ladite règle est appellée *règle exacte* si son M_{GK} est 1, et *approximative*, dans le cas contraire.

5.3 Optimisation du parcours des règles pertinentes

Une des solutions pour réduire le très grand nombre des règles extraites consiste à les mesurer pour ne garder que des plus pertinentes. Ainsi, nous présentons dans cette section notre modèle d'optimisation au parcours d'extraction des règles d'association potentiellement intéressantes, en utilisant le nouveau couple support-M_{GK}. En effet, nous allons opter les deux points suivants : (i) Description des propriétés d'élagage de l'espace de recherche ; (ii) Parcours optimisé de cet espace de recherche. Dans un premier temps, nous montrons comment réduire considérablement, à l'aide des nouvelles propriétés mathématiques introduites, le nombre des règles d'association à extraire en éliminant celles qui ne sont pas pertinentes. Dans un second temps, nous présentons notre stratégie de parcours de l'espace de recherche grâce également à des nouvelles propriétés énoncées. La tâche de notre modèle est donc de dégager d'une part un modèle mathématique d'élagage proposé visant à réduire efficacement les coûts de l'extraction des règles réellement intéressantes, ainsi que l'espace mémoire pour ces règles valides. D'autre part, d'introduire un nouveau modèle de parcours de recheche des règles en partitionnant l'ensemble en deux sous-classes, *classe d'attraction* et *celle de répulsion* que nous allons expliciter dans les deux paragraphes 5.3.1 et 5.3.2 ci-après. Afin d'éviter le paradigme de classes minoritaires, les deux classes seront élaguées sous la même contrainte ξ_α à un risque α fixé.

5.3.1 Propriétés du parcours d'élagage

Dans le but de ne pas générer toutes les règles candidates, nous définissons les propriétés d'élagage de l'espace de recherche. L'algorithme 24 peut être rendu plus efficace en considérant les propriétés ci-après.

Proposition 13. *Si la règle $X \to Y$ est potentiellement intéressante, alors les règles $Y \to X$, $\overline{Y} \to \overline{X}$, $\overline{X} \to \overline{Y}$ le seront également, sinon les règles $X \to \overline{Y}$, $\overline{Y} \to X$, $\overline{X} \to Y$, $Y \to \overline{X}$ pourront être intéressantes (cf. Totohasina, 2008 [Tot08]).*

Le corollaire 1 ci-dessous partitionne l'ensemble des règles en deux sous-classes, classe d'attraction et celle de répulsion, et permet d'éviter de plus le problème des redondances.

Corollaire 1. *Si X favorise Y (i.e. $P(Y|X) > P(Y)$), seul l'ensemble des règles $X \to Y$, $Y \to X$, $\overline{X} \to \overline{Y}$ et $\overline{Y} \to \overline{X}$ de la classe attractive est à évaluer ; sinon nous allons étudier $X \to \overline{Y}$, $\overline{X} \to Y$, $\overline{Y} \to X$ et $Y \to \overline{X}$ qui sont des règles dans la classe répulsive.*

Il est possible, en vertu des propositions ci-dessous, de générer sur chacune des classes toutes les règles potentiellement intéressantes. En effet, les trois premières propositions (prop. 14, 15 et 16) permettent de générer les règles de la classe attractive.

Proposition 14. *$\forall X, Y \in \mathcal{F}$ t.q. $P(Y|X) > P(Y)$ et $P(X) \geq P(Y)$ (resp. $P(X) \leq P(Y)$), on a $M_{GK}(X \to Y) \leq M_{GK}(Y \to X)$ (resp. $M_{GK}(X \to Y) \geq M_{GK}(Y \to X)$).*

Démonstration. Puisque $P(Y|X) > P(Y)$, on a $M_{GK}(Y \to X) = \frac{P(\overline{Y})}{P(Y)} \frac{P(X)}{P(\overline{X})} M_{GK}(X \to Y)$. Or, par hypothèse $P(X) \geq P(Y) \Leftrightarrow P(\overline{X}) \leq P(\overline{Y})$, donc $P(X)P(\overline{Y}) \geq P(\overline{X})P(Y)$, entraîne $M_{GK}(Y \to X) > M_{GK}(X \to Y)$. Ce qui montre que si la règle $X \to Y$ est M_{GK}-intéressante, alors la règle $Y \to X$ le sera également. Et, si $P(X) \leq P(Y)$ équivaut à $P(\overline{X}) \geq P(\overline{Y})$, entraîne $P(X)P(\overline{Y}) \leq P(\overline{X})P(Y)$. Finalement $M_{GK}(X \to Y) \geq M_{GK}(Y \to X)$, ce qui montre que si $X \to Y$ est non pertinente, alors $Y \to X$ ne le sera pas non plus. □

Proposition 15. *Etant donnés deux motifs fréquents X et Y de $\mathcal{F}$, si $P(Y|X) > P(Y)$ et $P(X) \geq P(Y)$, alors $M_{GK}(X \to Y) = M_{GK}(\overline{Y} \to \overline{X})$.*

Démonstration. Puisque $P(Y|X) > P(Y)$, on a : $M_{GK}(\overline{Y} \to \overline{X}) = \frac{P(\overline{X}|\overline{Y}) - P(\overline{X})}{1 - P(\overline{X})} = \frac{-P(X|\overline{Y}) + P(X)}{P(X)}$ $= \frac{-P(X \cap \overline{Y}) + P(X)P(\overline{Y})}{P(X)P(\overline{Y})} = \frac{-P(X)[1 - P(Y|X)] + P(X)[1 - P(Y)]}{P(X)[1 - P(Y)]} = \frac{P(Y|X) - P(Y)}{1 - P(Y)} = M_{GK}(X \to Y)$, d'où $M_{GK}(\overline{Y} \to \overline{X}) = M_{GK}(X \to Y)$. Donc, si $X \to Y$ est pertinente, alors $\overline{Y} \to \overline{X}$ le sera. □

Proposition 16. *$\forall X, Y \in \mathcal{F}$ t.q. $P(Y|X) > P(Y)$ et $P(X) \geq P(Y)$ (resp. $P(X) \leq P(Y)$), on a $M_{GK}(\overline{X} \to \overline{Y}) \geq M_{GK}(X \to Y)$ (resp. $M_{GK}(\overline{X} \to \overline{Y}) \leq M_{GK}(X \to Y)$).*

Démonstration. Puisque $P(Y|X) > P(Y)$ et $P(X) \geq P(Y)$, nous avons : $M_{GK}(\overline{X} \to \overline{Y}) = \frac{P(\overline{Y}|\overline{X}) - P(\overline{Y})}{1 - P(\overline{Y})} = \frac{-P(Y|\overline{X}) + P(Y)}{P(Y)} = \frac{-P(\overline{X} \cap Y) + P(\overline{X})P(Y)}{P(\overline{X})P(Y)} = \frac{P(X|Y) - P(X)}{1 - P(X)} = M_{GK}(Y \to X)$, ce qui donne $M_{GK}(\overline{X} \to \overline{Y}) = \frac{P(X)}{P(\overline{X})} \frac{P(\overline{Y})}{P(Y)} M_{GK}(X \to Y)$. Par hypothèse, $P(X) \geq P(Y)$ équivaut à $P(\overline{X}) \leq P(\overline{Y})$, entraîne $P(X)P(\overline{Y}) > P(\overline{X})P(Y)$, ce qui donne, pour tous X et Y de $\mathcal{F}$, $M_{GK}(\overline{X} \to \overline{Y}) \geq M_{GK}(X \to Y)$. Ce qui montre que si la règle $X \to Y$ est intéressante, alors la règle $\overline{X} \to \overline{Y}$ le sera également. Et, si $P(X) \leq P(Y)$ ou bien $P(\overline{X}) \geq P(\overline{Y})$, alors $P(X)P(Y) \leq P(\overline{X})P(\overline{Y})$, d'où $M_{GK}(\overline{X} \to \overline{Y}) \leq M_{GK}(X \to Y)$, $\forall X, Y \in \mathcal{F}$. Ce qui montre que si la règle $X \to Y$ est non intéressante, alors la règle $\overline{X} \to \overline{Y}$ ne le sera pas non plus. □

Les propositions 17 et 18 suivantes caractérisent les règles de la classe répulsive.

Proposition 17. *$\forall X, Y \in \mathcal{F}$ t.q. X défavorise Y (i.e. X favorise $\overline{Y}$, $\overline{X}$ favorise Y et $\overline{Y}$ favorise X), on a $M_{GK}(X \to \overline{Y}) = M_{GK}(Y \to \overline{X})$ et $M_{GK}(\overline{X} \to Y) = M_{GK}(\overline{Y} \to X)$.*

Démonstration. Puisque $P(\overline{Y}|X) > P(\overline{Y})$, nous avons : $M_{GK}(X \to \overline{Y}) = \frac{P(\overline{Y}|X)-P(\overline{Y})}{1-P(\overline{Y})} = \frac{1-P(Y|X)-1+P(Y)}{P(Y)} = \frac{-P(X\cap Y)+P(X)P(Y)}{P(X)P(Y)} = \frac{-P(X|Y)+P(X)}{P(X)} = \frac{1-P(X|Y)-1+P(X)}{P(X)} = \frac{P(\overline{X}|Y)-P(\overline{X})}{1-P(\overline{X})}$, d'où $M_{GK}(X \to \overline{Y}) = M_{GK}(Y \to \overline{X})$. Ce qui montre que si $X \to \overline{Y}$ est intéressante, alors $Y \to \overline{X}$ le sera également. Et, si $X \to \overline{Y}$ n'est pas intéressante, alors $Y \to \overline{X}$ ne le sera pas non plus. Comme $P(\overline{Y}|X) \geq P(\overline{Y})$, nous avons $M_{GK}(X \to \overline{Y}) = \frac{P(\overline{Y}|X)-P(\overline{Y})}{1-P(\overline{Y})} = \frac{P(\overline{X}\cap Y)-P(\overline{X})P(Y)}{P(X)P(Y)} = \frac{P(\overline{X})[P(Y|\overline{X})-P(Y)]}{P(X)P(Y)} = \frac{P(\overline{X})P(\overline{Y})}{P(X)P(Y)}\frac{P(Y|\overline{X})-P(Y)}{P(\overline{Y})}$, donc $M_{GK}(X \to \overline{Y}) = \frac{P(\overline{X})P(\overline{Y})}{P(X)P(Y)}M_{GK}(\overline{X} \to Y)$ (4). De plus, $M_{GK}(X \to \overline{Y}) = \frac{P(X\cap\overline{Y})-P(X)P(\overline{Y})}{P(X)P(Y)} = \frac{P(\overline{Y})[P(X|\overline{Y})-P(X)]}{P(X)P(Y)} = \frac{P(\overline{X}P(\overline{Y})}{P(X)P(Y)}\frac{P(X|\overline{Y})-P(X)}{P(\overline{X})}$, donc $M_{GK}(X \to \overline{Y}) = \frac{P(\overline{X})P(\overline{Y})}{P(X)P(Y)}M_{GK}(\overline{Y} \to X)$ (4′). En identifiant (4) et (4′), nous avons $M_{GK}(\overline{X} \to Y) = M_{GK}(\overline{Y} \to X)$, $\forall X, Y \in \mathcal{F}$. Donc, si $\overline{X} \to Y$ est intéressante, alors $\overline{Y} \to X$ le sera. Et, si $\overline{X} \to Y$ n'est pas intéressante, alors $\overline{Y} \to X$ ne le sera pas non plus. □

Proposition 18. *Etant donnés deux motifs X et Y fréquents (i.e. $X, Y \in \mathcal{F}$), tels que $P(Y|X) \leq P(Y)$ et $P(X) \geq P(Y)$, on a $M_{GK}(X \to \overline{Y}) \leq M_{GK}(\overline{X} \to Y)$.*

Démonstration. On est dans le cas où X défavorise Y ($P(Y|X) \leq P(Y)$), mais X favorise $\overline{Y}$ ($P(\overline{Y}|X) > P(\overline{Y})$) : le nombre de contre-exemples est supérieur au nombre d'exemples. Donc, la quantité $M_{GK}(X \to \overline{Y})$ doit être supérieure au point d'équilibre : $\frac{1}{2} \leq M_{GK}(X \to \overline{Y})$. Il faut donc que $P(X) \geq \frac{1}{2} \geq P(Y)$, équivaut à $P(\overline{X}) \leq \frac{1}{2} \leq P(\overline{Y})$, entraîne $P(\overline{X})P(\overline{Y}) \leq P(X)P(Y)$. D'où, d'après (4) de la propriété 17, $M_{GK}(X \to \overline{Y}) \leq M_{GK}(\overline{X} \to Y)$. Ce qui montre si $X \to \overline{Y}$ est intéressante, alors $\overline{X} \to Y$ le sera également. □

La propriété 19 ci-dessous permet d'éviter le problème des redondances, et complémente les propriétés ainsi évoquées.

Proposition 19. *Etant donnés trois motifs X, Y, Z de $\mathcal{F}$ tels que $X \subseteq Y \subseteq Z$, on a : $M_{GK}(X \to Z\backslash X) \leq M_{GK}(Y \to Z\backslash Y)$ et $M_{GK}(Z\backslash Y \to Y) \geq M_{GK}(Z\backslash X \to X)$.*

Démonstration. Puisque $X \subseteq Y \subseteq Z$, du fait de la croissance de support, on a : $supp(Z) \leq supp(Y) \leq supp(X)$, entraine $P(Z|X) \leq P(Z|Y) \Leftrightarrow P(Z|X) - P(Z) \leq P(Z|Y) - P(Z)$. Par association, pour $P(Z) \neq 1$, on a : $\frac{P(Z|X)-P(Z)}{1-P(Z)} \leq \frac{P(Z|Y)-P(Z)}{1-P(Z)}$, d'où $M_{GK}(X \to Z\backslash X) \leq M_{GK}(Y \to Z\backslash Y)$. Et puisque $supp(X) \geq supp(Y) \geq supp(Z)$ et les trois motifs X, Y et Z se favorisent mutuellement deux à deux, on a : $1 \geq M_{GK}(Y\backslash X \to X) \geq 0$, équivaut à $M_{GK}(Z\backslash Y \to Y) \geq M_{GK}(Z\backslash Y \to Y)M_{GK}(Y\backslash X \to X) \geq 0$. Or, M_{GK} est multiplicative (ou transitive), finalement, on a $M_{GK}(Z\backslash Y \to Y) \geq M_{GK}(Z\backslash X \to X)$. □

Comme nous le savons, l'économie du coût de l'extraction est fonction de la stratégie d'élagage. Nous allons décrire dans la sous-section 5.3.2 ci-après notre modèle d'optimisation du parcours de l'espace de recherche.

5.3.2 Parcours de l'espace de recherche

Nous partitionnons tout d'abord l'ensemble en deux sous-classes, *classe attractive* et *classe répulsive*, ce qui partitionne également notre espace de recherche. Rappelons que si X favorise Y, les règles $X \to Y$, $Y \to X$, $\overline{Y} \to \overline{X}$ et $\overline{X} \to \overline{Y}$ seront étudiées dans la classe attractive ou zone d'attraction. Si X défavorise Y, les règles $X \to \overline{Y}$, $\overline{Y} \to X$, $\overline{X} \to Y$ et $Y \to \overline{X}$ quant à elles seront étudiées dans la classe répulsive ou zone de répulsion. Autrement dit, lorsque la règle $X \to Y$ est dans la classe attractive, les règles $X \to Y$, $Y \to X$, $\overline{Y} \to \overline{X}$ et $\overline{X} \to \overline{Y}$ pourront être potentiellement pertinentes, tandis que les quatre autres règles $X \to \overline{Y}$, $\overline{Y} \to X$, $\overline{X} \to Y$ et $Y \to \overline{X}$ ne seront pas non plus intéressantes. Celles-ci pourront être intéressantes lorsque $X \to Y$ est dans la classe répulsive, mais cette fois-ci, les quatre autres $X \to Y$, $Y \to X$, $\overline{Y} \to \overline{X}$ et $\overline{X} \to \overline{Y}$ ne seront pas non plus pertinentes. Ce qui va permettre d'élaguer les règles qui ne sont pas même classe que la règle $X \to Y$.

Dans la *zone attractive*, sous l'hypothèse $P(X) \geq P(Y)$, nous avons prouvé d'après la propriété 14 que $M_{GK}(Y \to X) > M_{GK}(X \to Y)$, ce qui implique que si la règle $X \to Y$ est valide selon M_{GK}, alors la règle $Y \to X$ le sera également. Ensuite, d'après la propriété 15, on a $M_{GK}(\overline{Y} \to \overline{X}) = M_{GK}(X \to Y)$, s'agissant que $\overline{Y} \to \overline{X}$ et $X \to Y$ sont équivalentes. Par conséquent, si la règle $X \to Y$ est valide, alors la règle $\overline{Y} \to \overline{X}$ le sera également. Enfin, nous avons montré d'après la propriété 16 que $M_{GK}(\overline{X} \to \overline{Y}) > M_{GK}(X \to Y)$, cela signifie que si la règle $X \to Y$ est valide, alors la règle $\overline{X} \to \overline{Y}$ le sera également. En fait, une seule règle, $X \to Y$, va permettre de déduire l'intérêt de trois autres $Y \to X$, $\overline{Y} \to \overline{X}$ et $\overline{X} \to \overline{Y}$. Ce qui donne une notable optimisation de l'espace de recherche, soit 3/4 de réduction.

Dans la *zone répulsive*, sous l'hypothèse $P(\overline{X}) \leq P(\overline{Y})$, nous avons montré, d'après la propriété 17, que $M_{GK}(X \to \overline{Y}) = M_{GK}(Y \to \overline{X})$, signifiant si la règle $X \to \overline{Y}$ est valide, alors la règle $Y \to \overline{X}$ le sera également. Nous avons aussi montré que $M_{GK}(\overline{X} \to Y) = M_{GK}(\overline{Y} \to X)$, qui signifie si la règle $\overline{X} \to Y$ est valide, alors la règle $\overline{Y} \to X$ le sera également. D'autre part, nous avons montré que $M_{GK}(X \to \overline{Y}) \leq M_{GK}(\overline{X} \to Y)$, ce implique si la règle $X \to \overline{Y}$ est valide, alors $\overline{X} \to Y$ le sera également. En conclusion, nous retrouvons aussi le même constat que précédemment : une seule règle, $X \to \overline{Y}$, va permettre d'inférer les trois autres règles $X \to \overline{Y}$, $\overline{Y} \to X$, $\overline{X} \to Y$ et $Y \to \overline{X}$. Ce qui donne également des 3/4 de réduction de l'espace de recherche, seuls les quarts de la classe vont étudier.

De plus, pour tous $X, Y, Z \in \mathcal{F}$ tels que $X \subseteq Y \subseteq Z$, on a d'après la propriété 19 $M_{GK}(X \to Z \backslash X) \leq M_{GK}(Y \to Z \backslash Y)$. Cela signifie que si la règle $Y \to Z \backslash Y$ n'est pas valide, alors la règle $X \to Z \backslash X$ ne le sera pas non plus. Par exemple, si la règle $AC \to DE$ n'est pas valide, alors les règles $A \to CDE$ et $C \to ADE$ ne seront pas valides non plus et il n'est pas néccesaire de poursuivre leur évaluation selon la mesure M_{GK}. Réciproquement, puisque $M_{GK}(Z \backslash Y \to Y) \geq M_{GK}(Z \backslash X \to X)$ entraîne si la règle $Z \backslash X \to X$ est valide, alors la règle $Z \backslash Y \to Y$ le sara également. Par exemple, si $AB \to C$ et $AC \to B$ sont valides, alors $A \to BC$ le sera également. Ce qui permet d'élaguer sans perte d'information les règles d'association redondantes, afin de minimiser aussi l'espace de recherche.

Ces stratégies montrent un comportement efficace de notre modèle. Celui-ci va générer les règles potentiellement pertinentes à partir de la classe contenant la règle $X \to Y$. Ce qui nous assure un double gain : amélioration de la qualité des règles et réduction de l'espace de recherche. A cet effet, nous n'étudions que des quarts de la classe, soit la moitié de l'ensemble.

5.4 Présentation de l'algorithme GenPNR

L'algorithme GenPNR-*Generation of Positive and Negative association Rules* prend en argument un ensemble $\mathcal{F}$ des motifs fréquents, un seuil *minsupp* de support minimum et un risque d'erreur α, et retourne l'ensemble $\mathcal{E}_{PNR}$ des règles d'association positives et négatives valides. D'un point de vue technique, l'algorithme GenPNR partitionne l'ensemble en deux sous-classes, classe d'attraction et celle de répulsion, et parallélise l'extraction. Puis, pour générer les règles potentiellement fréquentes, nous utilisons pour chacune des classes les mêmes contraintes, telles que le support minimum *minsupp* et un risque α fixé. L'orginalité de l'algorithme réside d'une part dans l'utilisation du nouveau couple support-M_{GK}, et d'autre part du fait qu'il ne génénère que la moitié de l'ensemble. Formellement défini par :

$$\mathcal{E}_{PNR} = \{X \to Y, X \to \overline{Y} : X, Y \in \mathcal{F} | (supp(X \to Y) \geq minsupp \wedge M_{GK}(X \to Y) \geq \xi_\alpha) \\ \text{et } (supp(X \to \overline{Y}) \geq minsupp \wedge M_{GK}(X \to \overline{Y}) \geq \xi_\alpha)\}$$

Il s'agit en fait, pour tous X et Y de $\mathcal{F}$, de générer à partir des deux seules règles $X \to Y$ et $X \to \overline{Y}$ l'ensemble des règles d'association positives et négatives satifaisant les contraintes *minsupp* et ξ_α. Son pseudo-code est reporté dans l'algorithme 24 ci-après.

Algorithm 24 GenPNR

Require: Un ensemble $\mathcal{F}$ de motifs fréquents, un *minsupp* et un risque d'erreur α.
Ensure: Un ensemble $\mathcal{E}_{PNR}$ de règles positives et négatives valides.

1: $\mathcal{E}_{PNR} = \emptyset$;
2: **for all** $(X \in \mathcal{F})$ **do**
3: **for all** $(Y \in \mathcal{F})$ **do**
4: **if** $(P(Y|X) > P(Y) \ \wedge \ P(X) \geq P(Y))$ **then**
5: calculate $supp(X \to Y)$; $M_{GK}(X \to Y)$; $\xi_\alpha = \sqrt{\frac{1}{n}\frac{n-n_X}{n_X}\frac{n_Y}{n-n_Y}\chi^2_\alpha}$;
6: **if** $(supp(X \to Y) \geq minsupp \ \wedge \ M_{GK}(X \to Y) \geq \xi_\alpha)$ **then**
7: $\mathcal{E}_{PNR} \leftarrow \mathcal{E}_{PNR} \cup \{(X \to Y \backslash X), (Y \to X \backslash Y), (\overline{Y} \to \overline{X} \backslash \overline{Y}), (\overline{X} \to \overline{Y} \backslash \overline{X})\}$;
8: **end if**
9: **else**
10: calculate $supp(X \to \overline{Y})$; $M_{GK}(X \to \overline{Y})$; $\xi_\alpha = \sqrt{\frac{1}{n}\frac{n-n_X}{n_X}\frac{n_{\overline{Y}}}{n-n_{\overline{Y}}}\chi^2_\alpha}$;
11: **if** $(supp(X \to \overline{Y}) \geq minsupp \ \wedge \ M_{GK}(X \to \overline{Y}) \geq \xi_\alpha)$ **then**
12: $\mathcal{E}_{PNR} \leftarrow \mathcal{E}_{PNR} \cup \{(X \to \overline{Y} \backslash X), (Y \to \overline{X} \backslash Y), (\overline{Y} \to \overline{X} \backslash \overline{Y}), (\overline{X} \to Y \backslash \overline{X})\}$;
13: **end if**
14: **end if**
15: **end for**
16: **end for**
17: **return** $\mathcal{E}_{PNR}$;

Détaillons maintenant chacune des lignes de l'algorithme GenPNR. L'algorithme commence par initialiser l'ensemble $\mathcal{E}_{PNR}$ (ligne 1). La boucle principale (lignes 2 à 16) qui constitue la partie optimisation de cet algorithme 24 prend fin lorsqu'il n'y a plus de règles à générer. En effet, l'algorithme procède en deux sous-étapes récursives. La première (lignes

4 à 9) consiste à générer les règles de type $X \to Y$, $Y \to X$, $\overline{X} \to \overline{Y}$ et $\overline{Y} \to \overline{X}$, qui sont des règles de la classe attractive, dans laquelle X favorise Y. Dans ce cas, l'algorithme n'étudie que de la règle $X \to Y$, les autres règles ($Y \to X$, $\overline{X} \to \overline{Y}$, $\overline{Y} \to \overline{X}$) peuvent être dérivées de celle-ci en utilisant les propriétés 14, 15 et 16. Si les contraintes ($P(Y|X) > P(Y)$ et $P(X) \geq P(Y)$) sont respectées (ligne 4), alors l'algorithme calcule successivement les quantités $supp(X \to Y)$, $M_{GK}(X \to Y)$ et la valeur critique ξ_α (ligne 5). Si $supp(X \to Y)$ (resp. $M_{GK}(X \to Y)$) est meilleur que $minsupp$ (resp. ξ_α) (ligne 6), l'algorithme met à jour l'ensemble $\mathcal{E}_{PNR}$ en y ajoutant les règles valides $X \to Y \backslash X$, $Y \to X \backslash Y$, $\overline{Y} \to \overline{X} \backslash \overline{Y}$ et $\overline{X} \to \overline{Y} \backslash \overline{X}$ (ligne 7). Sinon ($P(\overline{Y}|X) > P(\overline{Y})$ et $P(\overline{X}) \leq \frac{1}{2} \leq P(\overline{Y})$), l'algorithme génère, à sa seconde phase (lignes 9 à 14), les règles de type $X \to \overline{Y}$, $\overline{X} \to Y$, $Y \to \overline{X}$, $\overline{Y} \to X$, ce sont des règles de la classe répulsive, pour laquelle X défavorise Y, mais X favorise $\overline{Y}$, $\overline{X}$ favorise Y, et $\overline{Y}$ favorise X. Il y parcourt seulement la règle $X \to \overline{Y}$, les autres règles ($\overline{X} \to Y$, $Y \to \overline{X}$, $\overline{Y} \to X$) quant à elles peuvent être dérivées de celle-ci, grâce aux propriétés 17 et 18. A cet effet, l'algorithme calcule les quantités $supp(X \to \overline{Y})$ et $M_{GK}(X \to \overline{Y})$, et la valeur critique ξ_α (ligne 10). Si $supp(X \to \overline{Y})$ (resp. $M_{GK}(X \to \overline{Y})$) dépasse ou égale au $minsupp$ (resp. ξ_α) (ligne 11), l'algorithme met à jour l'ensemble $\mathcal{E}_{PNR}$ en y ajoutant les règles $X \to \overline{Y} \backslash X$, $Y \to \overline{X} \backslash Y$, $\overline{Y} \to X \backslash \overline{Y}$ et $\overline{X} \to Y \backslash \overline{X}$ (ligne 12). Il retourne l'ensemble $\mathcal{E}_{PNR}$ (ligne 17), et s'arrête lorsqu'il n'y a plus des règles qui puissent être générées.

5.4.1 Complexité de `GenPNR`

Ladite complexité est définie théoriquement dans la proposition 20 ci-après.

Proposition 20. *Soit $\mathcal{F}$ l'ensemble des motifs fréquents. La complexité en temps de l'algorithme 24 est, au pire des cas, $\mathcal{O}(\frac{|\mathcal{F}|^2}{4}(2^{\frac{|\mathcal{F}|}{2}} - \frac{|\mathcal{F}|}{2}))$.*

Démonstration. Soit $\mathcal{E}_{PNR}$ l'ensemble des règles positives et négatives valides. La boucle **for** (ligne 2) est itérée au maximum $|\mathcal{F}|$ fois, car nous stockons une règle valide pour chaque occurrence d'un motif, donc au maximum il y a autant d'occurrences que de règles dans la moitié l'ensemble $\mathcal{F}$. Cela s'effectue en $\mathcal{O}(|\mathcal{F}|/2)$ dans le pire des cas. La seconde boucle **for** (ligne 3) s'effectue également en $\mathcal{O}(|\mathcal{F}|/2)$, au pire des cas. Puis, la complexité du bloc d'instructions 4-9 générant les règles de la classe attractive étant égale à celle de répulsive aux instructions 9-14, nous ne comptabiliserons que l'une seulement. Donc, la complexité des instructions 4-14 est $\mathcal{O}(\sum_{i=2}^{|\mathcal{F}|/2} C^i_{|\mathcal{F}|/2}) = \mathcal{O}(\sum_{i=0}^{|\mathcal{F}|/2} C^i_{|\mathcal{F}|/2} - 1 - \frac{|\mathcal{F}|}{2}) \approx \mathcal{O}(2^{\frac{|\mathcal{F}|}{2}} - \frac{|\mathcal{F}|}{2})$, dans le pire des cas. Finalement, nous avons $\mathcal{O}(\frac{|\mathcal{F}|^2}{4}(2^{\frac{|\mathcal{F}|}{2}} - \frac{|\mathcal{F}|}{2}))$ □

Bien que, GenPNR donne plus d'optimisations que les algorithmes de Wu [WZZ04] et RAPN [GP13], sa complexité reste encore linéaire et exponentielle en fonction de $|\mathcal{F}|$.

5.5 Evaluation expérimentale

Nous présentons la performance de l'algorithme GenPNR, comparée à celle des algorithmes sémantiquement plus proche, tels que l'algorithme de Wu et al., 2004 [WZZ04] et RAPN (Guillaume et Papon, 2013 [GP13]). L'objectif est de démontrer la faisabilité en nous intéressant non seulement au temps d'exécution mais aussi à la qualité des règles extraites.

Protocole expérimental Les trois algorithmes ont été implémentés en C++ et R. Les expérimentations ont été réalisées sur un PC de 4 Go de RAM tournant sous système Windows. Pour avoir une meilleure idée du comportement de GenPNR, Wu et RAPN face aux paradigmes de l'extraction des meilleures règles, nous les expérimentons sur 4 bases de données disponibles sur l'UCI machine learning repository [5]. Les caractéristiques de ces jeux de données sont présentées dans le tableau 5.3 ci-après. La base `Adult` est un questionnaire

Base	Nombre de transactions	Nombre d'items	Items par transaction
`Adult`	48842	115	67
`German`	1000	71	69
`Income`	6876	50	40
`Iris`	150	15	15

Tableau 5.3 – Caractéristiques des bases d'expérimentations.

décrivant le personnel du bureau de recensement des Etat-Unis. `German` décrit les caractéristiques des clients ayant contracté un prêt bancaire. `Income` est une donnée de marketing décrivant la caractéristique socio-professionnelle des ouvriers. `Iris` décrit les typologies des plantes. Nous expérimentons les algorithmes avec les mêmes contraintes. Faute de place, notons par ms le support minimum d'Apriori, par η_α le niveau de signification à un risque d'erreur α. La colonne étiquetée "++" correspond à la règle $X \to Y$, la colonne "-+" à $\overline{X} \to Y$, la colonne "+-" à $X \to \overline{Y}$, et la colonne "- -" à $\overline{X} \to \overline{Y}$. En faisant varier ms et η_α, nous avons les résultats sur les tableaux 5.4 et 5.5, et sur les figures 5.2, 5.3 et 5.4.

			Algorithmes											
			Wu et al.2004				**RAPN**				**GenPNR**			
Base	ms	η_α	+ +	- +	+ -	- -	+ +	- +	+ -	- -	+ +	- +	+ -	- -
`Adult`	1%	60%	97956	625	1215	785	87800	542	615	421	27500	422	510	352
	2%	70%	55925	453	852	556	53950	323	503	344	25536	354	385	225
	3%	80%	38750	345	412	310	22033	156	254	145	18523	124	154	124
	4%	90%	12553	154	333	175	9020	75	95	56	8150	56	95	75
	5%	95%	5450	75	95	35	4523	15	25	13	1233	25	25	15
`German`	1%	60%	51478	456	1148	555	41235	401	565	412	26456	340	380	245
	2%	70%	39683	352	744	456	38555	234	425	384	18800	220	203	156
	3%	80%	9835	144	545	321	18500	75	325	232	12157	95	85	55
	4%	90%	2833	95	105	85	2533	45	97	65	942	15	15	12
	5%	95%	4209	45	44	35	4113	21	51	28	56	5	5	4
`Income`	1%	60%	2800	227	527	254	2130	350	385	286	1552	95	103	84
	2%	70%	2200	127	327	213	2054	214	330	185	1433	55	63	65
	3%	80%	1325	87	252	121	1212	65	156	45	923	35	30	17
	4%	90%	1056	25	95	54	956	30	75	25	523	11	30	2
	5%	95%	553	15	38	30	321	20	35	11	256	11	30	2
`Iris`	1%	60%	2437	159	196	160	1954	10	60	24	1500	150	165	124
	2%	70%	2000	159	145	120	1323	10	55	20	1122	75	85	65
	3%	80%	1200	159	59	45	1056	25	30	-	965	14	22	18
	4%	90%	950	157	35	45	756	-	-	-	561	10	10	3
	5%	95%	500	157	35	3	470	-	-	-	374	3	2	3

Tableau 5.4 – Nombre de règles d'association positives et négatives extraites

5. URL http ://www.ics.uci.edu/MLRepository.html

Qualité de règles extraites Nous constatons, pour chacun des algorithmes, qu'une baisse de support augmente exponentiellement le nombre des règles. Pour les jeux de données `Adult` et `German`, l'algorithme de Wu et l'algorithme RAPN donnent respectivement des nombres supérieurs à ceux de GenPNR. A cet effet, les règles positives valides pour Wu varient plus de 4 200 à plus de 97 900, plus de 4 100 à plus de 87 000 pour RAPN, qui sont relativement élevés, ce qui rend difficile la découverte de relations intéressantes. GenPNR quant à lui donne des nombres très raisonnables variant de 56 à 27 500, facilement utilisables pour l'utilisateur. Comme le synthétisent le tableau 5.4 et les graphiques des figures 5.2 et 5.3 sur les bases `Income` et `Iris`, les trois algorithmes donnent des nombres très raisonnables et cela au profit de l'algorithme GenPNR.

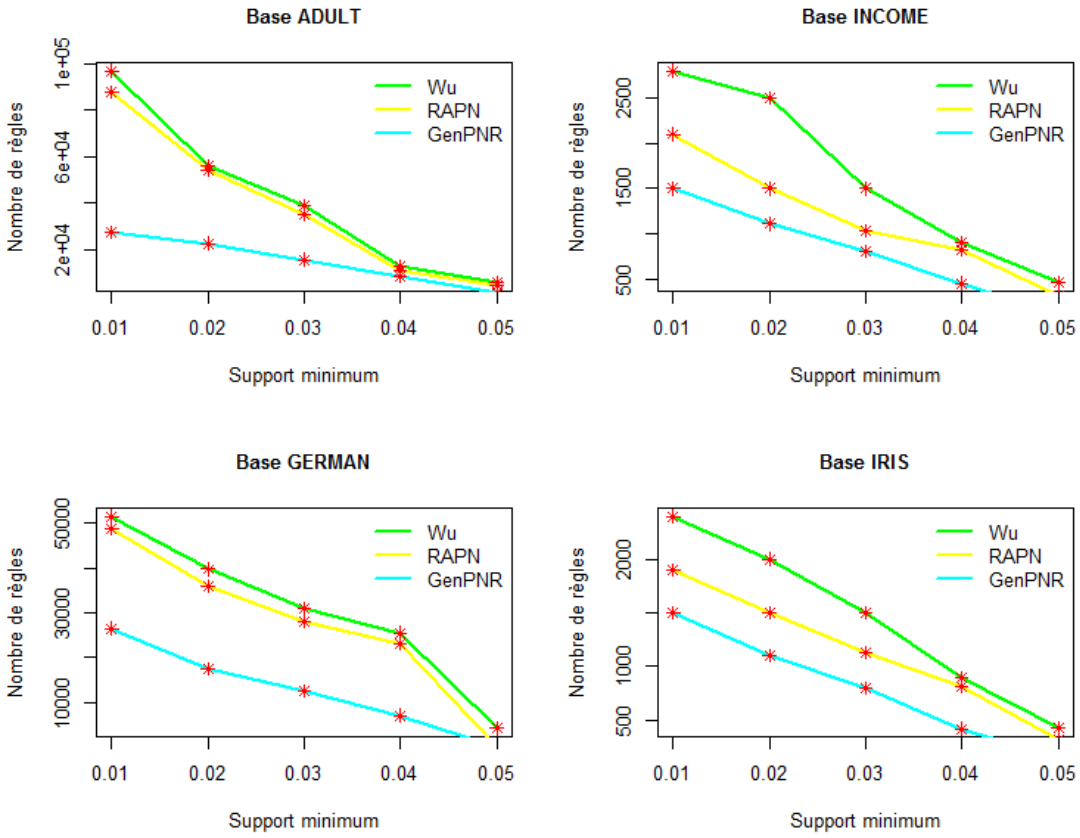

Figure 5.2 – Règles positives en fonction de support minimum *ms*

En fait, les graphiques de la figure 5.3 ci-après reportent l'ensemble des règles négatives pour les trois algorithmes sur les mêmes contraintes que la figure 5.2. Constatons sur l'ensemble de quatre bases utilisées, il apparaît, aux contextes fortement corrélés (pire des cas), que *plus de tiers* (cf. tableau 5.5) des règles générées par Wu et RAPN ne sont pas pertinents, ce qui alourdit inutilement la procédure. Pour les bases `Adult, German` et `Income`, les trois algorithmes donnent de nombre raisonnable. En effet, l'algorithme de Wu donne de nombre variant de 83 à 2625, RAPN fournit de nombre variant de 53 à 1578, et GenPNR

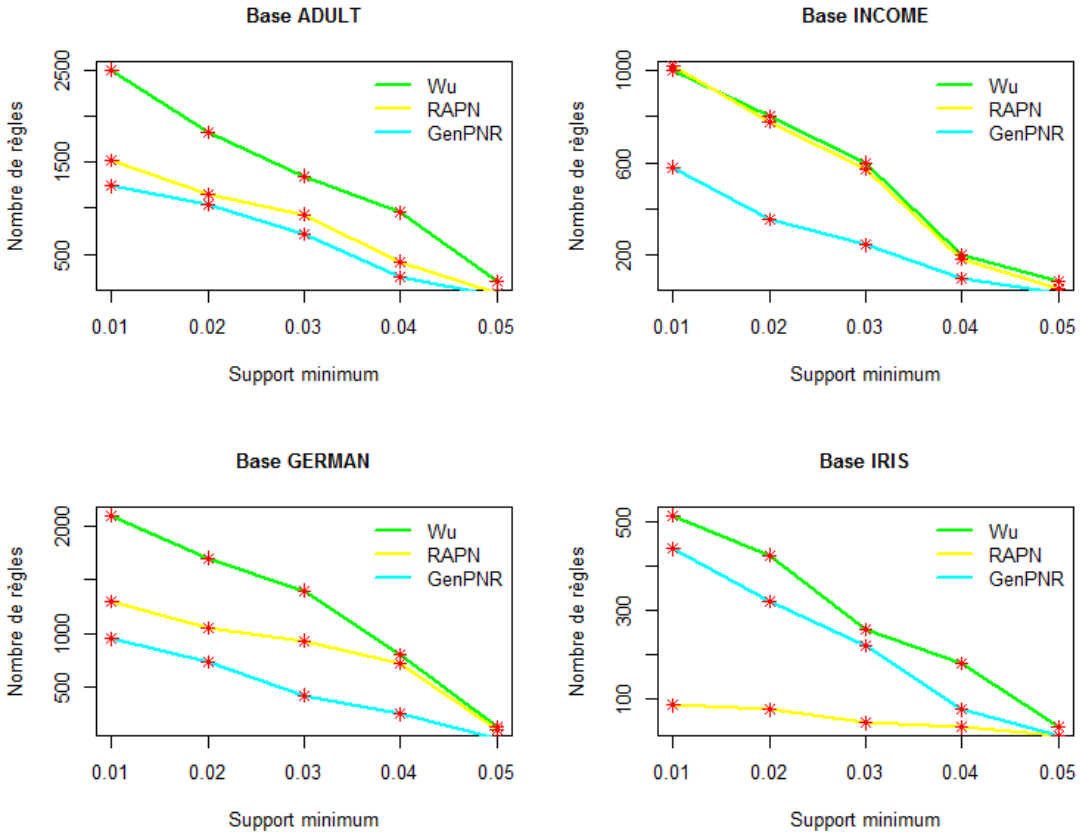

Figure 5.3 – Règles négatives en fonction de support minimum ms

de 14 à 1284. Pour la base `Iris`, RAPN offre une réduction très importante (de 0 à 94), mais avec perte d'information : trop de valeurs manquantes pour quelques catégories des règles (cf. tableau 5.4). L'algorithme de Wu et GenPNR donnent de nombres supérieurs, mais acceptables qui ne présentent aucune perte d'information. Nous retrouvons les mêmes observations que précédemment : la qualité de GenPNR reste encore stable même si on prend des seuils relativement faibles.

A titre indicatif, si on ne considère que de contexte très dense (comme par exemple `Adult`), on voit pour un ms égale à 1% que le nombre des règles de Wu (égal à 100 581) et celui de RAPN (égal à 89 378) sont respectivement presque à 3.50 et 3.10 fois à celui de notre algorithme GenPNR (égal à 28 784). Donc, GenPNR bénéficie clairement d'un nombre des règles d'association plus réduit que ceux de Wu et RAPN, donc plus sélectif.

Temps d'exécution Le temps que nous avons comptabilisé ici est le temps de réponse des trois algorithmes pour chacune des quatre bases de données utilisées. Nous y avons souhaité connaître le comportement de notre algorithme GenPNR par rapport aux algorithmes de Wu et RAPN. Les différents temps d'exécutions par rapport au seuil de support minimum sont reportés numériquement sur le tableau 5.5 et graphiquement sur la figure 5.4. Nous avons

restitué, pour chaque algorithme, le temps d'exécution exprimé en secondes (colonne cpu), le nombre total des règles positives par P, des négatives par N, et l'ensemble par $\mathcal{E}_{PNR}$.

Base	ms	η_α	Algorithmes											
			Wu et al.2004				**RAPN**				**GenPNR**			
			P	N	$\mathcal{E}_{PNR}$	cpu	P	N	$\mathcal{E}_{PNR}$	cpu	P	N	$\mathcal{E}_{PNR}$	cpu
`Adult`	1%	60%	97956	2625	100581	300	87800	1578	89378	280	27500	1284	28784	40
	2%	70%	55925	1861	57786	200	53950	1170	55120	185	25536	964	26500	30
	3%	80%	38750	1067	39817	150	22033	555	22588	150	18523	402	18925	25
	4%	90%	12553	662	13215	75	9020	226	9246	65	8150	226	8376	15
	5%	95%	5450	205	5655	12	4523	53	4576	11	1233	65	1298	4
`German`	1%	60%	51478	2159	53637	200	41235	1378	42613	195	26456	965	27421	30
	2%	70%	39683	1552	41235	175	38555	1043	39598	165	18800	579	19379	20
	3%	80%	9835	1010	10845	125	18500	632	19132	120	15157	235	15392	14
	4%	90%	2833	285	3118	25	2533	207	2740	22	942	42	984	10
	5%	95%	4209	124	4333	10	4113	100	4213	9	56	14	70	3
`Income`	1%	60%	2800	1008	3808	50	2130	1021	3151	48	1552	282	1834	15
	2%	70%	2200	667	2867	40	2054	720	2783	37	1433	183	1616	11
	3%	80%	1325	460	1785	30	1212	266	1478	28	923	82	1005	8
	4%	90%	1056	174	1230	15	956	130	1086	6	523	41	576	5
	5%	95%	553	83	636	5	321	66	387	3	256	41	299	2
`Iris`	1%	60%	2437	515	2952	40	1954	94	2048	37	1500	439	1939	15
	2%	70%	2000	424	2424	30	1323	85	1407	37	1122	225	1347	10
	3%	80%	1200	263	1463	15	1056	55	1111	17	965	54	1019	7
	4%	90%	950	237	1187	9	756	-	756	5	561	23	584	3
	5%	95%	500	195	695	3	470	-	470	3	374	8	382	2

Tableau 5.5 – Temps d'extraction en fonction des seuils minima ms et η_α

Nous avons la même observation que précédente, le temps d'exécution augmente exponentiellement lorsque le support diminue. On constate également que les bases de données de taille conséquente comportant un grand nombre d'attributs (`Adult` et `German`) sont très gourmandes en temps de calculs. Dans ce cas, l'algorithme de Wu et l'algorithme RAPN donnent des temps de réponse très supérieurs. Notre algorithme GenPNR quant à lui offre de meilleur temps d'exécution quel que soit le seuil *minsupp* de support minimum.

Ces différentes performances peuvent être expliquées par le fait que ces deux bases sont fortement corrélées, ce qui complique les tâches de Wu et RAPN au niveau de génération des règles potentiellement pertinentes. De plus, les deux algorithmes en question utilisent le couple support-confiance qui produit facilement des règles non intéressantes, ce qui alourdit inutilement le temps d'extractions. Notre approche GenPNR, afin d'éviter ce conséquent coût, propose une solution disons optimale, en utilisant un nouveau couple plus sélectif support-M_{GK}. GenPNR a démontré la grande concision de ce nouveau couple : plusieurs propriétés d'optimisations ont été introduites. A cet effet, notre algorithme GenPNR n'étudie que la moitié de l'ensemble, ce qui diminue notablement le coût d'extractions, et gagne 7 fois de plus la meilleure vitesse d'exécution par rapport aux algorithmes de Wu et RAPN.

Pour les jeux de données faiblement corrélés (`Income` et `Iris`), les trois algorithmes donnent des temps d'exécution très acceptables. On voit, pour des seuils de support de 0.01 à 0.03, que l'algorithme de Wu et RAPN fournissent des temps de réponse similaires, légèrement au gain de RAPN. Ce dernier rejoint notre algorithme GenPNR sur le seuil qui va de 0.04 à 0.05. Au vu des résultats de la figure 5.4, GenPNR semble le meilleur pour toutes les valeurs de support, ce qui nous assure la faisabilité notable de notre approche.

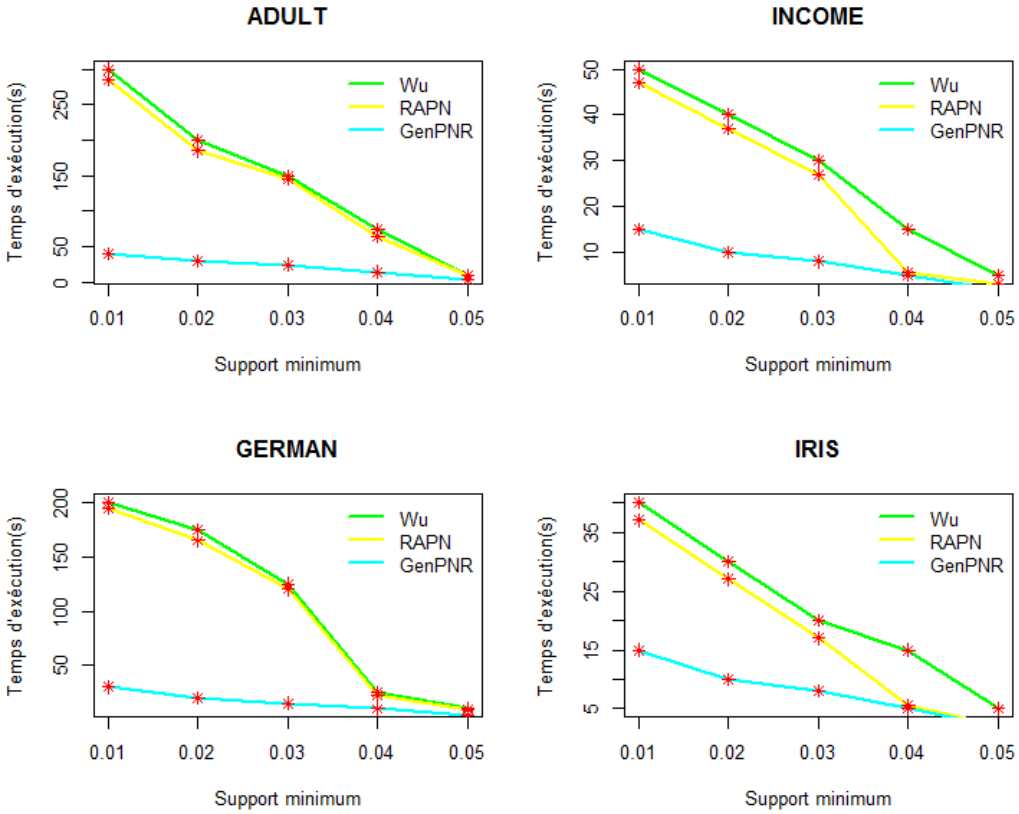

Figure 5.4 – Temps d'exécution en fonction de support minimum ms

Même si sa complexité est linéaire et parfois exponentielle en fonction de la taille de base, GenPNR obtient encore de meilleures performances que les approches Wu et RAPN.

5.6 Conclusion partielle et perspectives

Nous avons proposé une nouvelle approche d'extraction des règles d'association positives et négatives selon le nouveau couple support-M_{GK}. Nous y avons défini un nouvel algorithme GenPNR afin d'automatiser le modèle. L'algorithme a été testé sur quatre bases de données de références, et comparé à des algorithmes sémantiquement proches tels que Wu et RAPN. Les résultats ont montré que notre modèle réduit d'une manière notable les conséquents coûts que subissent Wu et RAPN, et reste meilleur au niveau du temps d'exécution et de la qualité des résultats. Enfin, le fait d'utiliser la mesure plus discriminante M_{GK}, nous évitons déjà des redondances de règles et obtenons un ensemble concis.

Notons toutefois la principale limitation de GenPNR : nous n'avons pas pu optimiser succinctement les problèmes des règles redondantes et ceux des règles négatives ayant dans la prémisse ou dans la conclusion une négation d'une conjonction des motifs à la fois négatifs et positifs, ce qui serait une piste pour des travaux futurs.

Chapitre 6

Graphes implicatifs selon la mesure M_{GK}

6.1 Introduction et motivations

Ce chapitre s'inscrit dans la continuité du chapitre précédent et poursuit la fouille des règles d'association valides, mais dans les paradigmes des graphes implicatifs afin de mettre en évidence les liens implicatifs de l'ensemble des règles retenues.

La notion de graphes est née en 1736 par la communication d'Euler où il proposait une solution au célèbre problème des 7 ponts de Königsberg. A partir de 1950, ce concept a connu un développement intense (Khun 1955 [MAQ03], Ford-Fulkerson 1956, Roy-Warshall 1959 [CLRS01], Dijkstra 1959 [MAQ03]). Depuis, de nombreux problèmes peuvent être étudiés et résolus sous l'angle de l'analyse de graphes. Jusqu'à présent, la littérature s'est beaucoup intéressée à l'étude des graphes attribués et dynamiques, très peu s'intéressent au sujet des graphes implicatifs. Alors qu'il peut être intéressant de prendre en compte la représentation des chaînes de règles d'association valides à l'aide des graphes implicatifs, en particulier en didactique de discipline pour une fin taxonomique, voire une structure organisatrice. Très souvent, la complexité de construction des graphes (classiques ou implicatifs) est exponentielle en raison essentiellement du parcours des chemins.

Des procédés ont été proposés dont nous en présenterons un rapide état de l'art qui est loin d'être exhaustif. Sur les graphes attribués, Moser et al. [MCRE09] proposent une approche permettant l'extraction des motifs cohésitifs des graphes, basée sur des vecteurs de propriétés. Miyoshi et al. [MOO09] étendent les motifs cohésitifs à des propriétés quantitatives. Dans (Berlingerio et al. 2009 [BBBG09]), des nouvelles propriétés ont été proposées en vue d'extraire les règles à partir des motifs locaux. Mougel et al. [MPR+10] proposent une nouvelle technique visant à extraire des ensembles de cliques homogènes. Silva et al. [SMZ10, SJZ12] proposent une méthode permettant l'extraction des paires de sous-graphes denses. Mougel et al. [MRG12] recherchent des collections de k-cliques percolées homogènes. Prado et al. [PPRB13] proposent une méthode pour trouver des régularités sur les descripteurs des attributs dans des graphes attribués. Sanhes et al. [SFP+13] étudient les motifs condensés dans un unique graphe orienté acyclique attribué.

La fouille de graphes dynamiques a aussi été très étudiée. Jin et al. [JMA07] étudient les graphes dynamiques permettant d'extraire les groupes de nœuds interconnectés. Lahiri et Berger-Wolf [LBW10] proposent une approche basée sur les sous-graphes similaires apparaissant périodiquement. Prado et al. [PJFD13] définissent un algorithme dédié à la

fouille de graphes planaires. La tendance actuelle de la fouille de graphes vise à combiner les graphes attribués et dynamiques. Boden et al. [BGS12] proposent une approche d'extraction des clusters combinant les graphes attribués et dynamiques. Récemment, Desmier et al. [DPB14] proposent un algorithme d'extension de (Jin et al. [JMA07], Boden et al. [BGS12]) afin d'extraire les motifs de co-évolution multiniveaux.

Bien qu'ils soient efficaces, ces différents travaux se trouvent confrontés à des limites notables : ils ne tiennent pas compte de la représentation en graphe des liaisons implicatives entre les règles valides. Pour cela, Gras [Gra79, GAB+96] a introduit une nouvelle approche d'élaboration des graphes implicatifs qui repose sur la mesure *intensité d'implication* basée sur une approximation gaussienne. C'est une approche couramment utilisée (Gras [GSK05], Serge et al. [SPJ+05], Ritschard et al. [RMM09], J.-C. Régnier et Gras [Rég09], Lerman et Pascal [LP09], Matthias et al. [MR10], Lucia et Dusan [LD12]) dans la communauté de l'ASI-analyse statistique implicative. Comme cela a été dit au début, l'évaluation de ce modèle est toutefois critiquable dans le sens où ladite mesure utilisée a tendance à ne plus être discriminante, à cause de l'approximation encourue, en présence des données denses, ce qui n'est donc pas à l'abri de perte d'information. De plus, il ne considère que des règles d'association positives, les règles d'association négatives qui peuvent se révéler une source d'information pertinente pour l'utilisateur sont ignorées, ce qui pose donc un réel problème pour qualifier les résultats obtenus.

Afin de compenser ces limites, nous proposons un nouveau modèle de construction de ces mêmes graphes implicatifs en utilisant une autre mesure de qualité plus sélective, M_{GK} (Guillaume [Gui00], Totohasina [Tot03], Wu [WZZ04]). Nous y définissons un nouvel algorithme afin d'automatiser la construction. Le modèle proposé intègre à la fois les règles d'association positives et les règles négatives. Pour le valider, nous effectuons des expérimentations menées sur quelques bases des données de référence.

Le reste de ce chapitre est organisé comme suit. La section 6.2 regroupe les définitions de base et les notations qui semblent nécessaires à la compréhension de ce chapitre. La section 6.3 détaille notre approche d'élaboration de l'ensemble des graphes implicatifs. La section 6.4 expose le nouvel algorithme qui construit cet ensemble des graphes. L'évaluation expérimentale est synthétisée dans la section 6.5. En conclusion (section 6.6), nous présentons un bilan de notre apport et des perspectives de recherche.

6.2 Définitions de base et notations

Cette section rappelle quelques définitions de base particulièrement sur les graphes orientés. Un graphe peut être défini de manière `intuitive` (définition 39) ou `mathématique` (définition 40).

Définition 39. *Un graphe est un schéma constitué par un ensemble de points et par un ensemble de flèches reliant chacune deux ceux-ci. Les points sont appelés les sommets du graphe, les flèches les arêtes (ou arcs) du graphe.*

Définition 40. *Un graphe $G = (S, A)$ est le couple constitué par un ensemble de sommets S, et par une famille d'arêtes A, tels que : $S \times S = \{(u, v) | u \in S, v \in S\}$.*

Les définitions suivantes sont énoncées surtout dans le cadre de graphes orientés.

Définition 41. *On appelle graphe non orienté $G = (S, A)$ la donnée de S dont les éléments sont appelés les sommets et d'une partie de A symétrique $((u, v) \in A \Leftrightarrow (v, u) \in A)$, les éléments sont appelés arêtes (ou arcs).*

Définition 42. *Un graphe $G = (S, A)$ est dit orienté si A est un ensemble de paires ordonnés, $A \subset S^2$. Un arc est constitué de deux sommets ayant des rôles distincts, une origine et une extremité terminale. On appelle ordre d'un graphe, noté $n = |S|$, le nombre de sommets du graphe. La taille d'un graphe, notée m, correspond au nombre d'arêtes : $m = |A|$.*

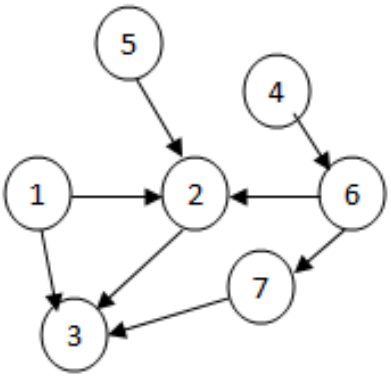

Figure 6.1 – Un exemple de graphe orienté d'ordre 7

La figure 6.1 est un exemple de graphe orienté d'ordre 7, comportant 7 sommets tels que $S = \{1, 2, 3, 4, 5, 6, 7\}$ et 8 arêtes $A = \{(1, 2), (1, 3), (2, 3), (5, 2), (4, 6), (6, 2), (6, 7), (7, 3)\}$. Le nombre d'arêtes *incidentes* à un sommet est le nombre d'arêtes sortant ou entrant, appelé degré qu'on le note $d^\circ(.)$. De la figure 6.1, 3 arêtes sont incidentes au sommet 6, soit $d^\circ(6) = 3$.

Définition 43. *Etant donné un graphe orienté $G = (S, A)$, $A \subset S^2$. Si $(u, v) \in A$, alors v est le successeur, et u prédécesseur. Deux sommets reliés par une arête sont dits adjacents.*

Dans notre exemple de la figure 6.1, le sommet 2 admet les sommets 1, 5 et 6 comme prédécesseurs, mais il admet 3 comme successeur, le voisinage du sommet 2 est donc $\{1, 3, 5, 6\}$. Dans ce cas, les sommets (1 et 2) ou (2 et 3) sont adjacents ou voisins.

Définition 44. *Etant donné un graphe orienté $G = (S, A)$, $A \subset S^2$. L'ensemble de successeurs de u est $\Gamma_{out}(u) = \{v \in S | (u, v) \in A\}$; et $\Gamma_{in}(u) = \{v \in S | (v, u) \in A\}$ celui de prédécesseurs. Ce qui donne l'ensemble des voisins $\Gamma(u) = \Gamma_{in}(u) \cup \Gamma_{out}(u)$.*

Toujours de la figure 6.1, l'ensemble de successeurs du sommet 1 est $\Gamma_{out}(1) = \{2, 3\}$, mais il n'a pas de prédécesseur, donc $\Gamma_{in}(1) = \{\}$. Finalement, l'ensemble de voisins du sommet 1 est donc $\Gamma(1) = \{2, 3\}$.

Définition 45. *Pour chaque sommet $u \in S$, on définit alors $d^\circ_{out}(u)$, le degré sortant de u comme étant le nombre d'arcs partant de u, qui correspond au cardinal du voisinage sortant $|\Gamma_{out}(u)|$; et $d^\circ_{in}(u)$, le degré entrant de u, le nombre d'arcs arrivant sur u, correspondant au nombre du voisinage entrant $|\Gamma_{in}(u)|$.*

De l'exemple de la figure 6.1, aucun sommet qui entre dans 1, son degré est donc nul ($d^\circ_{in}(1) = 0$), tandis qu'il y a deux arcs entrants, donc $d^\circ_{out}(1) = 2$. Il y a trois arcs entrant dans 3, donc $d^\circ_{in}(3) = 3$, mais aucun arc sortant, donc $d^\circ_{out}(3) = 0$.

Définition 46. *Etant donné un graphe orienté $G(S, A)$, $A \subset S^2$. On appelle degré de u, noté $d^\circ(u)$, la somme de son degré entrant et de son degré sortant : $d^\circ(u) = d^\circ_{in}(u) + d^\circ_{out}(u)$.*

Toujours de la figure 6.1, le degré du sommet 1 n'est autre que son degré sortant, car son degré entrant est nul, donc $d^\circ(1) = d^\circ_{out}(1) = 2$. Un sommet de degré entrant nul mais de degré sortant non nul est appelé *source*, tandis qu'un sommet de degré entrant non nul et de degré sortant nul est appelé *puits*. De ce même exemple, les sommets 1, 4 et 5 sont des sources, tandis que 3 est puits.

Définition 47. *Etant donnés un graphe $G = (S, A)$, et deux sommets distincts u et v de S. S'il existe une suite d'arêtes (ou d'arcs correctement orientés) permettant d'atteindre v à partir de u, alors on dit qu'il existe un chemin de u vers v.*

Définition 48. *Un chemin de cardinalite k, noté k-chemin, d'un graphe orienté $G = (S, A)$ est une séquence de sommets $u_1, \ldots, u_{k+1}$, non nécessairement distincts, telle que $(u_i, u_{i+1}) \in A$ pour tout $i \in \{1, \ldots, k\}$. Un chemin $[u_1, \ldots, u_k]$ est dit élémentaire, si $\forall i$, $1 \leq i \leq k$: $u_i \neq u_{i+1}$, i.e. ne contient pas deux fois le même sommet. La longueur d'un chemin est définie par le nombre d'arcs composant le chemin qui est égal au nombre de sommets moins un ($|S| - 1$). Un chemin constitué d'un seul sommet est de longueur nulle.*

Le chemin $[4, 6, 7, 3]$ de la figure 6.1 est un chemin élémentaire, car chacun de ces sommets du parcours est visité une seule fois.

Définition 49. *Un circuit de cardinalite k, noté k-circuit, d'un graphe orienté $G = (S, A)$ est une séquence de sommets $u_1, \ldots, u_k, u_1$, non nécessairement distincts, telle que $(u_i, u_{i+1}) \in A$ pour tout $i \in \{1, \ldots, k-1\}$ et $(u_k, u_1) \in A$.*

Définition 50. *Les chemins pondérés sont des chemins avec un poids sur chaque arc, représentant le nombre d'occurrences différentes de cet arc parmi toutes les occurrences dans le chemin.*

Par exemple, $\omega = u_1 \to u_2 \to u_3$ est un chemin pondéré tel que : $u_1 \overset{0.75}{\to} u_2 \overset{0.95}{\to} u_3$. Dans ce cas, les arêtes $u_1 \to u_2$ et $u_2 \to u_3$ ont leur poids respectif 0.75 et 0.95.

Définition 51. *On dit qu'un graphe $G = (S, A)$ est pondéré si à chaque élément a de A est associé à une valeur réelle positive $poids(a) \in \mathbb{R}^*_+$, qui s'appelle poids de a. Le poids d'un sous-ensemble X d'éléments de A, est la somme des poids des éléments de X, définie par $poids(X) = \sum_{a \in X} poids(a)$.*

Proposition 21. *Tout sous-chemin d'un chemin de valeur minimale d'un graphe pondéré est un chemin de valeur minimale.*

Démonstration. Soit ω_0 un chemin allant de u_l à u_t et soit ω' un sous-chemin de ω_0 allant de u_m à u_q, alors $\omega_0 = \omega_1 \omega' \omega_2$, où $\omega_1 = [u_l, u_m]$ et $\omega_2 = [u_q, u_t]$. On suppose ω_0 de valeur minimale. Si ω' n'était pas de valeur minimale, il existerait un chemin ω'' de valeur strictement plus faible allant de u_m à u_q et le chemin $\omega_1 \omega' \omega_2$ serait de valeur strictement inférieure à celle de ω_0, ce qui contredit le fait que ω_0 est de valeur minimale. □

6.3 Recherche des chemins implicatifs

La taille conséquente de sommets et d'arcs à représenter pèse lourd sur les coûts de constructions, ce qui rend difficile le parcours du chemin. Dans le pire des cas, un calcul combinatoire donne, pour un graphe dense de n sommets, un nombre près de $n!$ chemins possibles. Prenons par exemple un nombre très raisonnable 20 sommets, le nombre de chemins possibles avoisine de $2.432902e + 18$.

A titre de rappel, seule l'approche de Gras [Gra79, GAB+96] qui traite le problème de graphes implicatifs, dans la communauté de graphes. Nous présentons ici une démarche différente que celle de Gras [GAB+96]. L'originalité principale de notre approche vient surtout du rôle joué par l'ensemble des règles d'association considérées. En effet, notre approche basée sur la mesure M_{GK} intègre à la fois les règles positives et les règles négatives (type absent dans l'approche classique de Gras). Alors que celles-ci comme nous l'avons mentionné peuvent se révéler une source d'information importante. L'approche proposée souligne à la fois les points communs, les règles positives comme le fait l'approche classique de Gras, mais aussi met en évidence leurs traits distinctifs, les règles d'association négatives.

Pour ce faire, nous commençons par présenter tout d'abord nos résultats théoriques justifiant l'approche proposée (sous-section 6.3.1). Puis, nous décrivons une nouvelle méthode de parcours des chemins implicatifs (sous-section 6.3.2).

6.3.1 Résultats théoriques de l'approche

Nous définissons ici de manière formelle nos concepts théoriques de notre approche. Nous définissons tout d'abord la notion de graphes implicatifs. En effet, la prémisse et le conséquent de la règle correspondent respectivement aux sommets de départ (prédécesseur) et d'arrivé (successeur), et les règles sont matérialisées par des arêtes (ou arcs). Donc, on parle parfois d'arêtes orientées plutôt que de règles, et de sommets (ou nœuds) plutôt que de motifs.

Définition 52. *Soient un ensemble de sommets S et celui d'arêtes $\mathcal{A}$. Un graphe implicatif $\mathcal{G}(S, \mathcal{A})$, dont les sommets et les arêtes correspondent respectivement aux motifs fréquents et aux règles valides, est un graphe orienté, sans cycle, et pondéré, permettant la représentation des chaînes implicatives entre ces règles d'association valides.*

La matrice de données $\mathcal{M}$, contenant l'ensemble de ces règles d'association valides, peut être vue à l'aide d'un graphe implicatif. La définition 53 définit un formalisme de cette matrice.

Définition 53. *Soit un graphe implicatif $\mathcal{G} = (S, \mathcal{A})$ d'ordre $n = |S|$. Sa matrice d'adjacence $\mathcal{M}$ est une matrice carrée de taille $n \times n$, qui est définie de la manière suivante :*

$$\mathcal{M}[u][v] = \begin{cases} 1, & si\ M_{GK}(u \rightarrow v) \geq \xi_\alpha \\ 0, & sinon \end{cases}$$

Où M_{GK} désigne le poids de l'arête $u \rightarrow v$, et ξ_α la valeur critique à un α donné.

La matrice de données prend la valeur 1 si le poids (ou coût) de l'arête est meilleur que la valeur critique, et 0 sinon. Autrement dit, la cellule de ladite matrice vaut 1 si les sommets

correspondants sont adjacents, et 0 sinon. Ce qui va constituer une abaque de construction de l'ensemble des graphes implicatifs. Afin d'élaguer les chemins implicatifs, nous définissons dans la définition 54 la notion de distance minimale pour les sommets du graphe. Nous y introduisons une nouvelle fonction coût, M_{GK}.

Définition 54. *Etant donnés u et v de S, la distance minimale de l'arête $u \to v$ à l'itération k est la quantité, notée d^k_{uv}, vérifiant l'équation de récurrence suivante :*

$$\begin{cases} d^0_{uv} & = M_{GK}(u \to v) \\ d^k_{uv} & = min\{d^{k-1}_{uv}, d^{k-1}_{p(u)u} + M^k_{GK}(u \to v)\}, \ \forall k \geq 1 \end{cases}$$

où $p(u)$ dénote le prédécesseur du sommet u.

De façon simplifiée, nous notons d^k_{uv} par d_{uv}. Un arc dont la distance est minimale sera choisi un chemin optimal pour notre approche. Cette distance est par construction fonction du coût M_{GK}. Dans la théorie classique, le coût peut être réel, mais comme nous ne nous intéressons qu'au problème stochastique, nous n'utiliserons que de coûts réels positifs.

Les propriétés 22, 23 et 24 ci-dessous caractérisent le graphe de notre approche.

Proposition 22. *Un graphe implicatif $\mathcal{G} = (S, \mathcal{A})$ est nécessairement sans circuit.*

Démonstration. **Par l'absurde**. Soit M_{GK} une mesure transitive et non réflexive sur un ensemble fini S. Et si $\omega = [u_1, \ldots, u_k]$ est un circuit élémentaire, on obtient alors :

$$M_{GK}(u_1 \to u_2) M_{GK}(u_2 \to u_3) \ldots M_{GK}(u_{k-1} \to u_k) = M_{GK}(u_1 \to u_k)$$

Puisque $u_1 = u_k$, nous avons $M_{GK}(u_1 \to u_1)$, contradiction car M_{GK} est non réflexive. □

Proposition 23. *Dans un graphe implicatif $\mathcal{G} = (S, \mathcal{A})$, les 2 propriétés suivantes sont équivalentes : (i) $\mathcal{G}$ est sans circuit ; (ii) il n'y a pas d'arcs de retour.*

Démonstration. (i) implique (ii) est trivial. Montrons (ii) implique (i). Considérons un chemin $\omega^* = [u_1, \ldots, u_k, u_1]$ qui est un circuit du graphe $\mathcal{G}$, et supposons en outre que u_1 soit le premier sommet de ω^*, visité. Nécessairement, le parcours du chemin va visiter les sommets de ω^* dans l'ordre $u_1, \ldots, u_k$. Mais l'arc $u_k u_1$ est un arc de retour du parcours, d'où la contradiction. □

Proposition 24. *Un graphe implicatif $\mathcal{G}$ possède au moins une source et un puits.*

Démonstration. Considérons un chemin ω de $\mathcal{G}$ qui soit maximal au sens suivant ω tel que $\omega = [u_1, \ldots, u_k]$ et il n'existe pas de sommet r de $\mathcal{G}$ tel que $[r, u_1, \ldots, u_k]$ ou $[u_1, \ldots, u_k, r]$ soient de chemins de $\mathcal{G}$. Un tel chemin existe puisque $\mathcal{G}$ est sans circuit. Cela signifie que u_1 est une source et u_k est un puits. □

6.3.2 Parcours du chemin implicatif

Nous présentons ici la stratégie globale de notre approche, en utilisant la théorie que nous avons introduite précédemment. Nous générons tout d'abord l'ensemble des sommets sources $\mathcal{E}_{SRC}$. Les ensembles de sommets S et d'arêtes orientées $\mathcal{A}$, construits incrémentalement, seront ensuite générés à partir de cet ensemble $\mathcal{E}_{SRC}$. Ils sont élagués par rapport à la distance minimale afin d'obtenir l'ensemble des graphes implicatifs $\mathcal{G} = (S, \mathcal{A})$. Cette démarche s'arrête lorsque tous les sommets de l'ensemble $S_{\mathcal{M}}$ sont présents dans S.

Génération d'un ensemble des sommets sources Cette première étape consiste à identifier, à partir d'une matrice des données, l'ensemble $\mathcal{E}_{SRC}$ des sommets sources, sommets ascendants de tous les autres sommets du graphe. Rappelons que, pour un ensemble des sommets S de taille n, et un ensemble d'arêtes $\mathcal{A}$, la représentation par matrice d'adjacence consiste en une matrice booléenne $\mathcal{M}$ de taille $n \times n$ telle que $\mathcal{M}[u][v] = 1$ si $(u, v) \in \mathcal{A}$ et $\mathcal{M}[u][v] = 0$ sinon. Dans cette étape, il s'agit de rechercher tous les sommets ayant un degré entrant nul et un degré sortant non nul, c'est-à-dire des sommets n'ayant pas des prédécesseurs mais admettent des successeurs, que l'on définit formellement :

$$\forall u \in S,\ \exists v \text{ tel que } v \neq u,\ \mathcal{M}[v][u] = 0 \text{ et } \mathcal{M}[u][v] = 1$$

Les sommets u et v sont respectivement appelés la source et le successeur naturel. L'exemple ci-après, en considérant une matrice d'adjacence $\mathcal{M}$ du tableau 6.1 ci-dessous, illustre cette étape. On y suppose qu'on a toujours un sommet n'ayant pas d'arête de lui-même ($\mathcal{M}[u][u] = 0,\ \forall u$), donc des graphes sans circuit.

	1	2	3	4	5	6	7
1	0	1	1	0	0	0	0
2	0	0	1	0	0	0	0
3	0	0	0	0	0	0	0
4	0	0	0	0	0	1	0
5	0	1	0	0	0	0	0
6	0	1	0	0	0	0	1
7	0	0	1	0	0	0	0

Tableau 6.1 – Matrice des données $\mathcal{M}$

D'après ce tableau 6.1 ci-dessus, on voit que $d^{\circ}_{in}(1) = d^{\circ}_{in}(4) = d^{\circ}_{in}(5) = 0$, i.e. le coefficient $\mathcal{M}[v][u]$ pour chacun de ces 3 sommets vaut nul ($\mathcal{M}[,][1] = \mathcal{M}[,][4] = \mathcal{M}[,][5] = 0$), ce qui montre qu'ils sont des sources, donc $\mathcal{E}_{SRC} = \{1, 4, 5\}$.

Génération d'un ensemble de sommets et d'arêtes orientées Cette étape consiste à générer l'ensemble S de sommets successeurs et celui d'arêtes orientées $\mathcal{A}$. C'est une étape la plus coûteuse du fait de la taille de sommets et d'arêtes. L'idée générale est de générer, à partir de l'ensemble $\mathcal{E}_{SRC}$ de sommets sources, tous les sommets et arêtes satisfaisant la distance minimale. A cet effet, pour chaque sommet source de $\mathcal{E}_{SRC}$, on sélectionne tous les sommets successeurs immédiats, i.e. partant d'un sommet source de cet ensemble $\mathcal{E}_{SRC}$, on sélectionne tous les voisins (sommets successeurs immédiats) en récupérant parmi eux l'arête qui admet la distance minimale du sommet source vers son successeur. Les sommets ayant une distance minimale sont ensuite ajoutés dans l'ensemble S et on met à jour l'ensemble $\mathcal{A}$ en y ajoutant l'arête ainsi obtenue. Autrement dit, si l'on considère $\mathcal{M}[v][u] = 0$ et $\mathcal{M}[u][v] = 1$ qui signifie u est source ($u \in \mathcal{E}_{SRC}$) et v successeur ($v \in S \backslash \mathcal{E}_{SRC}$), alors l'ensemble de somméts S (resp. $\mathcal{A}$) devient $S = S \cup \{v\}$ (resp. $\mathcal{A} = \mathcal{A} \cup \{(u, v)\}$). Puis, nous mettons à jour les restes sommets successeurs, qui ne sont pas eux-même des sommets déjà marqués. Ce processus est répété jusqu'à ce que tous les sommets (resp. arêtes) soient établies. Ce qui permet de découvrir tous les sommets accessibles du graphe et les arêtes associées, de trouver également les distances minimales et les chemins associés depuis un sommet source vers n'importe quel sommet du graphe.

Nous synthétisons ces différentes stratégies dans l'algorithme 25 ci-dessous, qui parcourt une recherche en largeur de l'espace mémoire.

6.4 Présentation de l'algorithme

Nous présentons notre algorithme M_{GK}-ImplicativeGraph qui prolonge nos travaux (Bemarisika et Totohasina [BT14a]). L'algorithme prend en argument un ensemble des règles valides stocké matriciellement dans $\mathcal{M}$, et retourne un ensemble des graphes implicatifs $\mathcal{G} = (S, \mathcal{A})$ de cet ensemble des règles valides. D'un point vue technique, M_{GK}-ImplicativeGraph parcourt de sommet en sommet l'espace de recherche. Sa principale originalité réside d'une part du fait qu'il construit, à partir d'une matrice des données $\mathcal{M}$, les graphes implicatifs de l'ensemble $\mathcal{E}_{PNR}$ des règles positives et négatives valides ; et d'autre part dans l'utilisation de la mesure plus discriminante, M_{GK}. Rappelons que $\mathcal{E}_{SRC}$ représente l'ensemble des sommets sources, $S_{\mathcal{M}}$ l'ensemble des sommets de la matrice $\mathcal{M}$, et d_{uv} la distance minimale de l'arête $u \rightarrow v$. Son pseudo-code est reporté dans l'algorithme 25.

Algorithm 25 M_{GK}-ImplicativeGraph

Require: Matrice de données $\mathcal{M} \subseteq \mathcal{E}_{PNR}$
Ensure: Graphe implicatif $\mathcal{G} = (S, \mathcal{A})$ construit
1: $\mathcal{E}_{SRC} = \emptyset$; // Initialisation de l'ensemble $\mathcal{E}_{SRC}$
2: **for all** $(u \in S_{\mathcal{M}})$ **do**
3: **for all** $(v \in S_{\mathcal{M}})$ **do**
4: **if** $(\mathcal{M}[u][v] = 1 \ \wedge \ \mathcal{M}[v][u] = 0)$ **then**
5: $\mathcal{E}_{SRC} \leftarrow \mathcal{E}_{SRC} \cup \{u\}$;
6: **end if**
7: **end for**
8: **end for**
9: $\mathcal{A} = \emptyset$; $S = \mathcal{E}_{SRC}$; // Initialisation de S et de $\mathcal{A}$
10: **repeat**
11: **for all** $(u \in \mathcal{E}_{SRC})$ **do**
12: **for all** $(v \in S_{\mathcal{M}} \backslash \mathcal{E}_{SRC})$ **do**
13: **if** (v is adjacent) **then**
14: $d_{uv} = min\{d_{uv}, d_{p(u)u} + M_{GK}(u \rightarrow v)\}$;
15: **else**
16: $d_{uv} = \infty$;
17: **end if**
18: **if** (d_{uv} is minimal) **then**
19: $S \leftarrow S \cup \{v\}$; $\mathcal{A} \leftarrow \mathcal{A} \cup \{(u, v)\}$;
20: **end if**
21: **end for**
22: **end for**
23: **until** $(|S| = |S_{\mathcal{M}}|)$
24: Return $\mathcal{G}$;

Détaillons maintenant chacune des lignes de cet algorithme. A cet effet, l'algorithme 25 parcourt de deux étapes récursives. La première consiste à générer l'ensemble des sommets sources $\mathcal{E}_{SRC}$ (lignes 1 à 8) à partir de la matrice $\mathcal{M}$. Elle commence par initialiser cet ensemble $\mathcal{E}_{SRC}$ (ligne 1). Puis, pour chaque u et v de l'ensemble $S_{\mathcal{M}}$, l'algorithme explore

en largeur tous les sommets sources du graphe (lignes 2 à 8). Cette étape est élaguée par rapport aux coefficients matriciels $\mathcal{M}[u][v]$ et $\mathcal{M}[v][u]$ (lignes 4 à 6). A cet effet, si les deux coefficients prennent respectivement les valeurs 1 et 0, s'agissant le sommet u est source (n'a plus de prédécesseur) et v successeur, alors l'algorithme met à jour l'ensemble $\mathcal{E}_{SRC}$ en y ajoutant ce sommet source u ainsi identifié (ligne 5).

Le deuxième procédé qui constitue la boucle principale (lignes 10 à 23) de l'optimisation de notre algorithme génère l'ensemble des sommets et celui d'arêtes orientées à partir de l'ensemble des sources $\mathcal{E}_{SRC}$. Ce procédé est élagué par rapport à la distance minimale d_{uv} et prend fin quand les sommets de l'ensemble $S_{\mathcal{M}}$ sont tous dans l'ensemble S du graphe à construire. Il commence par initialiser à vide l'ensemble $\mathcal{A}$, et à $\mathcal{E}_{SRC}$ l'ensemble des sommets S (ligne 9). Pour chaque sommet u de $\mathcal{E}_{SRC}$ et v de $S_{\mathcal{M}} \backslash \mathcal{E}_{SRC}$, l'algorithme génère les sommets (resp. arêtes) qui ne sont pas encore marqués dans l'ensemble S (resp. $\mathcal{A}$) (lignes 11 à 22), en calculant tout d'abord leurs distances minimales (lignes 13 à 17). A cet effet, si le sommet v sélectionné est adjacent (ligne 13), sa distance d_{uv} est la quantité définie dans l'expression de la ligne 14, c'est la distance minimale qui mène de la source u au sommet v ainsi visité. Sinon, cette distance est attribuée provisoirement de la valeur ∞ (ligne 16). Ensuite, si la contrainte d_{uv} est vérifiée, c'est-à-dire d_{uv} est minimale (ligne 18), alors l'algorithme met à jour l'ensemble des sommets S (resp. d'arêtes orientées $\mathcal{A}$), en y intégrant le nouveau sommet v (resp. l'arête (u, v)) (ligne 19). En répétant ce processus jusqu'à ce que les sommets de $S_{\mathcal{M}}$ sont tous dans S (ligne 23). L'algorithme retourne l'ensemble de graphes $\mathcal{G}$ (ligne 24) et s'arrête lorsqu'il n'y a plus des sommets qui puissent être générés.

6.4.1 Complexité de l'algorithme 25

La complexité de l'algorithme M_{GK}-ImplicativeGraph, est définie dans la proposition 25 ci-dessous.

Proposition 25. *La complexité de l'algorithme 25 est, au pire des cas,* $\mathcal{O}(|S|^3 \log |S|)$.

Démonstration. La boucle imbriquée for (lignes 2 à 8) est itérée au plus $|S_{\mathcal{M}}|$ fois. La complexité de cette boucle est donc $\mathcal{O}(|S_{\mathcal{M}}|^2) = \mathcal{O}(|S|^2)$, car $S_{\mathcal{M}} \equiv S$ au pire cas. Ensuite, il y a au plus $|S| - 1$ arêtes dans l'ensemble S. De ce fait, la boucle for (lignes 11 à 22) est itérée $(|S| - 1)^2$ fois et à chaque passage il faut parcourir la liste des successeurs (fils du nœud prédécesseur), en tenant compte que le calcul de la distance d'arcs (lignes 15 et 16) est au temps constant. Ici, le nombre de feuilles de l'arbre est borné par $|S|$, et la somme des nombres de fils de tous les nœuds se trouvant entre racine et puits est à son tour bornée par $(|S| - 1)$. Et, on effectue pour chaque nœud une recherche dans l'arbre trie en $\log |S|$. Cela veut dire que pour trier une liste $|S|$ de sommets, le nombre de comparaisons entre sommets de la liste qu'effectue cette recherche est majoré par une constante fois $\log |S|$. Les opérations des lignes 10 à 23 se repètent, au pire des cas, $|S| - 1$ fois. Dans ce cas, c'est le coût de parcours des sommets qui domine, car les arêtes sont construites incrémentalement à partir de ces sommets. Par conséquent, la complexité de ces opérations est, au pire des cas, $\mathcal{O}((|S| - 1)[(|S| - 1)^2 + (|S| - 1)^2] \log |S|) = \mathcal{O}(|S|^3 \log |S|)$. Finalement, la complexité en temps pour générer l'ensemble de graphes implicatifs de notre approche est, au pire des cas, $\mathcal{O}(|S|^2 + |S|^3 \log |S|) = \mathcal{O}(|S|^3 \log |S|)$. □

6.4.2 Exemple d'exécution de l'algorithme 25

Considérons un exemple fictif (tableau 6.2) d'un ensemble des règles valides selon M_{GK}.

$D \overset{0.15}{\to} F\ B \overset{0.25}{\to} E\ D \overset{0.20}{\to} E\ E \overset{0.10}{\to} G\ F \overset{0.20}{\to} G\ A \overset{0.15}{\to} B$
$B \overset{0.10}{\to} D\ D \overset{0.30}{\to} G\ C \overset{0.20}{\to} F\ A \overset{0.20}{\to} C\ C \overset{0.15}{\to} D$

Tableau 6.2 – Ensemble fictif des règles valides selon M_{GK}

Sa matrice d'adjacence est représentée dans le tableau 6.3 ci-après.

	A	B	C	D	E	F	G
A	0	1	1	0	0	0	0
B	0	0	0	1	1	0	0
C	0	0	0	1	0	1	0
D	0	0	0	0	1	1	1
E	0	0	0	0	0	0	1
F	0	0	0	0	0	0	1
G	0	0	0	0	0	0	0

Tableau 6.3 – Matrice d'adjacence des données

Tout d'abord, notons par S l'ensemble des sommets définitivement marqués, par $p(.)$ le prédécesseur du sommet marqué, par trait "$-$" le prédécesseur non encore connu.

A l'étape 0, S ne contient qu'un sommet source A qui est un sommet de degré entrant nul (cf. tableau 6.3). Partant de A, ses successeurs immédiats sont B et C (cf. tableau 6.3), ce qui donne, d'après la définition 54, $d_{AB} = M_{GK}(A \to B) = 0.15$ et $d_{AC} = M_{GK}(A \to C) = 0.2$. Comme D, E, F et G ne sont pas voisins de A, ils gardent encore leurs satuts. Dans les tableaux qui suivent, sans nuire à la compréhension du lecteur, notons tout simplement par d_ζ la distance entre un sommet nouvellement inscrit et son successeur ζ.

S	$\mathcal{A}$	$d_B p(B)$	$d_C p(C)$	$d_D p(D)$	$d_E p(E)$	$d_F p(F)$	$d_G p(G)$
$\{A\}$	$\emptyset$	0.15,A	0.20,A	$+\infty, -$	$+\infty, -$	$+\infty, -$	$+\infty, -$

Tableau 6.4 – Résultat de l'étape 0

A l'étape 1, on voit d'après ce résultat que d_{AB} est minimale, ce qui permet d'intégrer B dans S et de mettre à jour $\mathcal{A}$ en y ajoutant l'arête AB. Partant de B nouvellement intégré, ses voisins immédiats (cf. tableau 6.3) sont D et E, ce qui donne :

$$d_{BD} = min\{d_{BD}, d_{AB} + M_{GK}(B \to D)\} = min\{+\infty, 0.15 + 0.10\} = 0.25$$
$$d_{BE} = min\{d_{BE}, d_{AB} + M_{GK}(B \to E)\} = min\{+\infty, 0.15 + 0.25\} = 0.40$$

F et G ne sont pas voisins (cf. tableau 6.3), ils conservent encore leurs satuts.

S	$\mathcal{A}$	$d_B p(B)$	$d_C p(C)$	$d_D p(D)$	$d_E p(E)$	$d_F p(F)$	$d_G p(G)$
$\{A\}$	$\emptyset$	0.15,A	0.20,A	$+\infty, -$	$+\infty, -$	$+\infty, -$	$+\infty, -$
$S \cup \{B\}$	$\mathcal{A} \cup \{AB\}$		0.20,A	0.25,B	0.40,B	$+\infty, -$	$+\infty, -$

Tableau 6.5 – Résultat de l'étape 1

A l'étape 2, on voit d'après ce résultat que $d_{AC} = 0.2$ est minimale, ce qui permet d'intégrer C dans S et de mettre à jour $\mathcal{A}$ en y ajoutant l'arête AC. Partant de C ainsi intégré, on voit que D et F sont successeurs immédiats, ce qui donne donc :

$$d_{CD} = min\{d_{CD}, d_{AC} + M_{GK}(C \to D)\} = min\{0.25, 0.20 + 0.15\} = 0.25$$
$$d_{CF} = min\{d_{CF}, d_{AC} + M_{GK}(C \to F)\} = min\{+\infty, 0.20 + 0.20\} = 0.40.$$

Le sommet G n'est pas successeur immédiat (cf. tableau 6.3), il garde encore son statut.

S	$\mathcal{A}$	$d_B p(B)$	$d_C p(C)$	$d_D p(D)$	$d_E p(E)$	$d_F p(F)$	$d_G p(G)$
$\{A\}$	$\emptyset$	0.15,A	0.20,A	$+\infty$, –	$+\infty$, –	$+\infty$, –	$+\infty$, –
$S \cup \{B\}$	$\mathcal{A} \cup \{AB\}$		0.20,A	0.25,B	0.40,B	$+\infty$, –	$+\infty$, –
$S \cup \{C\}$	$\mathcal{A} \cup \{AC\}$			0.25,B(C)	0.40,B	0.40,C	$+\infty$, –

Tableau 6.6 – Résultat de l'étape 2

A l'étape 3, on trouve d'après ce résultat que $d_{CD} = 0.25$ est minimale, ce qui permet d'intégrer D dans S et de mettre à jour $\mathcal{A}$ en y ajoutant l'arête CD. Partant de D nouvellement inscrit, ses successeurs immédiats (cf. matrice d'adjacence du tableau 6.3) sont E, F et G, ce qui nous donne :

$$d_{DE} = min\{d_{DE}, d_{BD} + M_{GK}(D \to E)\} = min\{0.40, 0.25 + 0.30\} = 0.40$$
$$d_{DF} = min\{d_{DF}, d_{BD} + M_{GK}(D \to F)\} = min\{0.40, 0.25 + 0.30\} = 0.40$$
$$d_{DG} = min\{d_{DG}, d_{BD} + M_{GK}(D \to G)\} = min\{\infty, 0.25 + 0.30\} = 0.55.$$

S	$\mathcal{A}$	$d_B p(B)$	$d_C p(C)$	$d_D p(D)$	$d_E p(E)$	$d_F p(F)$	$d_G p(G)$
$\{A\}$	$\emptyset$	0.15,A	0.20,A	$+\infty$, –	$+\infty$, –	$+\infty$, –	$+\infty$, –
$S \cup \{B\}$	$\mathcal{A} \cup \{AB\}$		0.20,A	0.25,B	0.40,B	$+\infty$, –	$+\infty$, –
$S \cup \{C\}$	$\mathcal{A} \cup \{AC\}$			0.25,B(C)	0.40,B	0.40,C	$+\infty$, –
$S \cup \{D\}$	$\mathcal{A} \cup \{CD\}$				0.40,B(D)	0.40,C(D)	0.55,D

Tableau 6.7 – Résultat de l'étape 3

A l'étape 4, on voit d'après ce résultat que $d_{DE} = d_{DF} = 0.4$, qui est minimale, ce qui permet d'intégrer E ou F dans S, mais nous choisissons arbitrairement E. Puis, nous mettons à jour alors $\mathcal{A}$ en y intégrant l'arête DE. Partant de E ainsi passé, on trouve que G est son seul successeur (cf. matrice d'adjacence du tableau 6.3), ce qui donne :

$$d_{EG} = min\{d_{EG}, d_{DE} + M_{GK}(E \to G)\} = min\{0.55, 0.40 + 0.10\} = 0.50$$

S	$\mathcal{A}$	$d_B p(B)$	$d_C p(C)$	$d_D p(D)$	$d_E p(E)$	$d_F p(F)$	$d_G p(G)$
$\{A\}$	$\emptyset$	0.15,A	0.20,A	$+\infty$, –	$+\infty$, –	$+\infty$, –	$+\infty$, –
$S \cup \{B\}$	$\mathcal{A} \cup \{AB\}$		0.20,A	0.25,B	0.40,B	$+\infty$, –	$+\infty$, –
$S \cup \{C\}$	$\mathcal{A} \cup \{AC\}$			0.25,B(C)	0.40,B	0.40,C	$+\infty$, –
$S \cup \{D\}$	$\mathcal{A} \cup \{CD\}$				0.40,B(D)	0.40,C(D)	0.55,D
$S \cup \{E\}$	$\mathcal{A} \cup \{DE\}$					0.40,C(D)	0.50,E

Tableau 6.8 – Résultat de l'étape 4

A l'étape 5, on s'ensuit que $d_{DF} = 0.4$ est minimale, ce qui permet d'intégrer F dans S. Puis, nous mettons à jour $\mathcal{A}$ en y ajoutant l'arête EF. De F ainsi inscrit, son successeur immédiat est G (cf. tableau 6.3), ce qui donne :

$$d_{FG} = min\{d_{FG}, d_{DF} + M_{GK}(F \rightarrow G)\} = min\{0.50, 0.40 + 0.20\} = 0.50.$$

S	$\mathcal{A}$	$d_Bp(B)$	$d_Cp(C)$	$d_Dp(D)$	$d_Ep(E)$	$d_Fp(F)$	$d_Gp(G)$
$\{A\}$	$\emptyset$	0.15,A	0.20,A	$+\infty$, –	$+\infty$, –	$+\infty$, –	$+\infty$, –
$S \cup \{B\}$	$\mathcal{A} \cup \{AB\}$		0.20,A	0.25,B	0.40,B	$+\infty$, –	$+\infty$, –
$S \cup \{C\}$	$\mathcal{A} \cup \{AC\}$			0.25,B(C)	0.40,B	0.40,C	$+\infty$, –
$S \cup \{D\}$	$\mathcal{A} \cup \{CD\}$				0.40,B(D)	0.40,C(D)	0.55,D
$S \cup \{E\}$	$\mathcal{A} \cup \{DE\}$					0.40,C(D)	0.50,E
$S \cup \{F\}$	$\mathcal{A} \cup \{FG\}$						0.50,E(F)

Tableau 6.9 – Résultat de l'étape 5

A l'étape 6, le minimum entre 0.5 et 0.5 est évidemment $0.5 = d_{FG}$, ce qui permet d'intégrer G dans S, et de mettre à jour $\mathcal{A}$ en y ajoutant l'arête FG. Puisque G est puits (cf. tableau 6.3), donc il n'y a plus de chemins depuis G. Les traitements s'arrètent alors pour G, c'est la fin de l'algorithme.

S	$\mathcal{A}$	$d_Bp(B)$	$d_Cp(C)$	$d_Dp(D)$	$d_Ep(E)$	$d_Fp(F)$	$d_Gp(G)$
$\{A\}$	$\emptyset$	0.15,A	0.20,A	$+\infty$, –	$+\infty$, –	$+\infty$, –	$+\infty$, –
$S \cup \{B\}$	$\mathcal{A} \cup \{AB\}$		0.20,A	0.25,B	0.40,B	$+\infty$, –	$+\infty$, –
$S \cup \{C\}$	$\mathcal{A} \cup \{AC\}$			0.25,B(C)	0.40,B	0.40,C	$+\infty$, –
$S \cup \{D\}$	$\mathcal{A} \cup \{CD\}$				0.40,B(D)	0.40,C(D)	0.55,D
$S \cup \{E\}$	$\mathcal{A} \cup \{DE\}$					0.40,C(D)	0.50,E
$S \cup \{F\}$	$\mathcal{A} \cup \{EG\}$						0.50,E(F)
$S \cup \{G\}$	$\mathcal{A} \cup \{FG\}$						0.50,E(F)

Tableau 6.10 – Résultat de l'étape 6

6.5 Evaluation expérimentale

L'objectif de cette section est d'évaluer la faisabilité de notre approche et son comportement. Dans ce cas, nous étudions l'efficacité de notre algorithme en nous intéressant au temps d'exécution.

Protocole expérimental Nous avons implémenté notre algorithme 25 en R, et nous l'avons testé avec les mêmes bases de données que nous avons utilisées dans le chapitre 5 précédent, telles que `Adult`, `German`, `Income` et `Iris`, issues de l'UCI Machine Learning Repository [1]. A titre de rappels, la base `Adult` est composée de 48842 objets et de 115 attributs. La base `German` est composée de 1000 objets et de 71 attributs. La base `Income` est composée de 6876 objets et de 50 attributs, et la base `Iris` est composée de 150 et de 15 attributs. Par ailleurs, toutes les expériences ont été réalisées sur un PC de 4 Go de RAM tournant sous système Windows.

1. URL http ://www.ics.uci.edu/MLRepository.html

Temps d'exécution Le tableau 6.11 rapporte les résultats expérimentaux (nombre de sommets et d'arêtes, et temps d'exécution) de notre modèle, à un risque d'erreur $\alpha \in [0, 1]$ fixé. Pour chacune des bases de données, la colonne $|S|$ correspond au nombre total des sommets, la colonne $|\mathcal{A}|$ représente le nombre total d'arêtes orientées du graphe construit, et la colonne cpu indique le temps total de réponse.

	Bases de données de tests											
	`Adult`			`German`			`Income`			`Iris`		
η_α	$\|S\|$	$\|\mathcal{A}\|$	cpu	$\|S\|$	$\|\mathcal{A}\|$	cpu	$\|S\|$	$\|\mathcal{A}\|$	cpu	$\|S\|$	$\|\mathcal{A}\|$	cpu
75%	2523	25400	27	1454	17323	20	102	1200	11	97	1125	10
80%	1642	18925	25	1125	15392	14	96	1005	8	85	1019	7
85%	1533	18800	18	985	11185	12	75	650	6	68	684	5
90%	951	8376	15	95	984	10	61	576	5	56	584	4
95%	275	1298	4	25	70	3	45	299	2	45	382	2

Tableau 6.11 – Résultats expérimentaux en fonction du seuil η_α

Nous constatons que pour chacune des bases de données, le temps mis pour proposer cet ensembles de graphes diminuent au fur et à mesure que le seuil η_α augmente, et n'excèdent pas 30 secondes même si à un ensemble à plusieurs chemins (plus de 2500 sommets et 25000 arêtes). Par ailleurs, pour les jeux de données `Adult` et `German`, l'algorithme donne de temps de réponse assez supérieurs. Ces performances peuvent être expliquées par le fait que ces deux contextes sont relativement denses, contiennent en effet un grand nombre de sommets et d'arêtes, donc de plusieurs chemins implicatifs à parcourir. Ceci complique la tâche de l'algorithme pour la génération. Pour les jeux de données `Income` et `Iris`, l'algorithme, pour tous les seuils, restitue des meilleurs temps de réponse. Autrement dit, le temps de réponse obtenus avec `Income` et `Iris` sont nettement inférieurs à ceux obtenus avec les deux autres bases. Cela peut être expliqué par le fait que ces deux bases sont moins corrélées, ce qui rend facile la tâche de l'algorithme pour l'élaboration.

De manière plus générale, les expérimentations montrent la faisabilité de notre approche en terme de construction de graphes implicatifs.

6.6 Conclusion partielle et perspectives

Jusqu'à présent, le graphe implicatif de Gras [Gra79] est le seul graphe utilisé dans la communauté de l'ASI, reposant sur l'indice *intensité d'implication*, basé sur une approximation gaussienne. Or, cette approche est assez critiquable : l'indice utilisé a tendance à ne plus être discriminant en présence de données denses, ce qui n'est donc pas à l'abri de perte d'information. De plus, seules les règles positives sont étudiées, les règles négatives ne sont pas intégrées, ce qui ne suffit donc pas pour garantir la qualité des résultats obtenus. Afin d'y remédier, nous avons proposé un nouveau modèle intégrant à la fois les règles positives et négatives à l'aide de l'autre mesure plus sélective, M_{GK}. Les expérimentations menées sur quelques bases de données de référence montrent la faisabilité notable de ce modèle.

Toutefois, notre modèle pose quelques limites. Nous n'avons pas pu la comparer à des travaux existants. Des études comparatives à d'autres travaux représentatifs seraient donc une piste envisagée. Nous souhaitons aussi poursuivre notre approche en nous penchant sur la représentation concise en graphe les règles d'association négatives.

Chapitre 7

Outil CHIC-M_{GK}, Applications en didactique des mathématiques

7.1 Introduction et motivations

Il est communément connu que l'extraction de connaissances à partir de données (ECD) est issue de différents domaines connexes à la statistique. Elle nécessite la mise en œuvre de méthodes statistiques classiques (composantes principales, correspondances multiples, etc.) ou moins classiques (graphe implicatif, arbres de classification, etc.). Cette technique est aujourd'hui un thème de recherche qui est au cœur du traitement des grandes masses de données générées par toutes sortes d'applications industrielles ou scientifiques, dont nous en citerons, entre autres, Planification commerciale (Fayyad et al., 1996 [FPSSU96] ; Brühl et al., 2009 [BHB+09]), Finance et Télécommunication (Hätönen et al., 1996 [HKM+96] ; Szczerba et Ciemski, 2009 [SC09]), Médecine (Aguilera et Subero, 2009 [AS09] ; Villerd et al., 2010 [VTL10]), Agriculture (Ruß, 2009 [Ruß09] ; Urtubia et Pérez-Correa [UPC09]), Contrôle statistique (Wang et al., [WWR+09] ; Serrano, 2014 [Ser14]), et Procédé industriel (Baqueiro et al., 2009 [BWMC09] ; Ruiz, 2014 [Rui14]).

Dans la lignée de ce travail, nous nous intéressons à un domaine assez particulier, *didactique des mathématiques*. Ce domaine nécessite des méthodes éducatives probantes. Très peu de pratiques intègrent cependant des moyens adéquats d'évaluation de l'apprentissage. En revanche, la didactique de précision (Lindsley, [Lin56, Lin90, Lin91]) propose une technique plus représentative de l'évolution d'un apprentissage. Bien qu'elle soit efficace, cette technique présente néanmoins certaines limites qui découragent certains utilisateurs. Des tentatives d'amélioration pour faciliter son utilisation ont échoué. Par conséquent, cette pratique n'a pas changé depuis deux dernières décennies.

Schuessler (Schuessler, 2008 [Sch08]) reprend et prolonge cette même technique de Lindsley en mettant en place une version informatisée en vue de visualiser l'évolution des performances d'élèves ayant un trouble envahissant du développement (TED), et d'en évaluer sa convivialité. Un autre modèle conceptuel *Pedagogical Content Knowledge-PCK* (Shulman, [Shu86, Shu87]) a été introduit, afin d'étudier l'impact des disciplines (dimension disciplinaire) et des connaissances pédagogiques (dimension pédagogique) sur les pratiques d'enseignement. Ce modèle est notablement efficace, mais critiqué par de nombreux travaux, tels que (Cochran et al., 1993 [CKR93] ; Van Deriel et Devos, 1998 [VDVD98] ; Segall, 2004

[Seg04]) ; Shulman lui-même (Shulman, 2007 [Shu07]) qui ajout les connaissances du contenu et pédagogiques ; Berthiaume (Berthiaume, 2007 [Ber07]) qui propose un outil spécifique aux enseignants universitaires capturant le phénomène du *savoir pédagogique disciplinaire-SPD* où il ajoute la dimension de l'épistémologie personnelle. Les modèles sont relativement efficaces, mais n'intègrent pas la dimension technologique.

Plus récemment, Bachy (Bachy, 2014 [Bac14]) propose un nouveau modèle-outil *Savoir technopédagogique disciplinaire-STPD*, où il ajoute la dimension technologique basée sur les réflexions de Shulman. Un nouveau support didactique numérique FORSE a été proposé (Braga, 2009 [Bra09]). Ce modèle présente des modestes perfomances, mais n'intègre pas le problème de variables implicatives, ce qui nécessite donc un poste intermédiaire pour celui-ci.

Ce rapide survol état de l'art permet de nous constater que l'élaboration des didacticiels adéquats d'évaluation d'un apprentissage pour la prise de décision pédagogique demeure encore un défi majeur au sein de la didactique des mathématiques. Nous proposons en ce sens un nouvel outil `CHIC-`M_{GK}, séquence du logiciel CHIC (version Couturier, R. 2008 [Cou08]), d'élaboration de graphes implicatifs des règles d'association positives et négatives (type non abordé à notre connaissance au sein de l'ASI-analyse statistique implicative, notamment dans le logiciel CHIC (Gras et al., 1996 [GAB+96])) potentiellement pertinentes. Alors que ce type peut être riche en information importante pour l'utilisateur.

Le verrou principal que nous allons lever ici est d'une part d'implémenter ce nouvel outil en vue d'automatiser son utilisation, et d'autre part de définir un nouveau modèle sur un problème de didactique des mathématiques à Madagascar. L'outil proposé est particulièrement évalué sur un problème réel de didactique de la statistique à Madagascar, faisant intervenir les difficultés et les obstacles de nos étudiants normaliens inscrits en L1 de l'ENSET-Ecole Normale Supérieure pour l'Enseignement Technique, Université d'Antsiranana, lors d'une résolution d'un exercice proposé. Pour ce faire, nous décrivons brièvement les principales fonctionnalités de cet outil `CHIC-`M_{GK}, qui sera suivi d'une étape de discrétisation des règles à représenter et celle d'importation des données, afin d'élaborer les graphes implicatifs et arbres hiérarchiques. Nous présentons par ailleurs notre modèle d'identification de ces difficultés et obstacles liés à l'enseignement-apprentissage de la statistique à Madagascar. Notre démarche consiste à expliciter, à partir des programmes scolaires et des sujets de baccalauréat au fil des réformes qui se sont succédé à Madagascar depuis les années 1970, certaines des contraintes qui pèsent sur cet enseignement, en mettant à jour les enjeux didactiques de ce choix. Nous y examinons, à l'aide d'un graphe obtenu, l'enchaînement d'idées de ces étudiants en fonction de leurs capacités explicatives.

Le reste de ce chapitre est organisé comme suit. Dans la section 7.2, nous présentons la description du nouvel outil `CHIC-`M_{GK}. La section 7.3 présente notre modèle sur les problèmes de la didactique des mathématiques, notamment les difficultés et les obstacles liés à l'enseignement-apprentissage de la statistique à Madagascar depuis les années 1970. Celle-ci constitue la richesse et l'une des originalités de cet outil `CHIC-`M_{GK} évalué sur un problème réel de didactique de la statistique, faisant intervenir les difficultés de nos étudiants en L1. Enfin, la section 7.4 donne une conclusion et des perspectives des travaux futurs.

7.2 Outil CHIC-M_{GK}

Le prototype CHIC-M_{GK} est développé dans un objectif de représentation graphique des règles d'association valides. Pour ce faire, plusieurs critères doivent être pris en compte pour garantir l'interprétabilité des graphes : le nombre de sommets et d'arêtes doit être limité, le nombre de croisements entre arêtes doit être minimal, différentes formes et couleurs doivent être utilisées pour retranscrire plus d'information du graphe. L'outil doit être interactif et conviviale pour que l'utilisateur puisse adapter la vue à ses besoins. CHIC-M_{GK} est un outil d'aide à l'analyse des données, dédié principalement au problème des graphes implicatifs des règles d'association valides. Il implémente aussi deux vues graphiques supplémentaires, à savoir, l'arbre hiérarchique de similarités et l'arbre hiérarchique cohésitif. A cet effet, plusieurs approches sont possibles pour implémenter ces différents éléments. Le graphe implicatif repose sur notre algorithme M_{GK}-ImplicativeGraph (Bemarisika et Totohasina, 2014 [BT14a]), prolongé dans le chapitre 6. La hiérachie de similarité quant à elle repose sur l'algorithme de vraisemblance du lien-AVL (Lerman [Ler81]), la hiérarchie cohésitive repose à son tour sur l'algorithme de classification hiérarchique orientée (Gras et Kuntz [GK05]), mais avec notre propre programmation. Le présent outil, inspiré du logiciel CHIC de Gras (version Couturier, R. 2008 [Cou08]), est développé en C++ et R (en collaboration étroite avec Couturier, R., Laboratoire d'Informatique de l'Université de Franche-Comté). D'un point de vue applicatif, R est un logiciel robuste, efficace et libre de droit, afin de pouvoir être utilisé par les praticiens de différentes communautés. A travers la possibilité de diffuser ses propres packages, il est à mon sens un logiciel idéal.

7.2.1 Fonctionnalités

CHIC-M_{GK} s'inscrit dans le paradigme actuel du diagramme de traitements dans le sens où les séquences d'opérations sur les données sont visualisées à l'aide d'un graphe. Il permet d'étudier les chaînes implicatives entre les règles d'association potentiellement pertinentes sur le jeu de données de l'utilisateur. Plus précisément, CHIC-M_{GK} est une boîte à outil conçue pour aider graphiquement l'utilisateur analyste à repérer dans ses données les meilleures règles. Il permet à un décideur de visualiser en graphe ces règles potentiellement intéressantes. Ce progiciel, de type d'aide à la décision, permet in fin de guider l'utilisateur d'identifier les meilleurs règles dans son corpus. Il fonctionne sous multiples fonctionnalités, mais nous ne pouvons donner le détail, en voici les principales :

- `Préparation de données`. Il s'agit principalement de la sélection d'individus, de variables, et de la construction de variables ou de règles. Ils sont rassemblés dans les onglets *InstanceSelection*, *FeatureSelection*, et *FeatureConstruction*.
- `Traitement`. Les composants implémentés constituent le cœur de l'outil. Ils effectuent un traitement sur les données et produisent un modèle que l'on peut visualiser dans la page de résultats. C'est dans ce module qu'on calcule les paramètres de traitement en vue d'apprécier la qualité des règles d'association étudiées. Ces composants sont rassemblés dans les onglets *ImplicativeGraph*, *SimilarityTree*, et *HierarchyTree*, qui génèrent respectivement le graphe implicatif, l'arbre des similarités, et l'arbre cohésitif.
- `Evaluation`. Elle est associée à un seul onglet *EvaluatError* permettant d'évaluer le taux d'erreur de classement du modèle effectué. Celui-ci inscrit les résultats de chaque exécution dans un fichier qui recense l'ensemble des évaluations réalisées.

7.2.2 Discrétisation de règles d'association

Rappelons qu'une règle d'association exprime la co-occurrence de deux motifs disjoints ($A, B \subseteq \mathcal{I}$ et $A \cap B = \emptyset$) de la forme $A \rightarrow B$, où $\mathcal{I}$ dénote l'ensemble des motifs, et A et B sont respectivement appelés la prémisse et le conséquent de la règle. Une flèche est utilisée pour représenter l'implication statistique entre deux motifs (ou variables). L'extraction de telles règles d'association (Agrawal et al. [AIS93]) valides est devenue une tâche très classique en fouille de données. L'idée est de dégager des relations intelligibles entre des attributs. Comme nous le savons, la complexité des travaux existants est exponentielle du fait du nombre prohibitif des règles extraites dont la plupart sont non pertinentes.

Afin d'y éviter, de nombreuses optimisations ont été intégrées dans l'outil proposé. Nous avons utilisé les algorithmes GenPNR (Bemarisika et Totohasina, 2014 [BT14b]) et Apriori (Agrawal et Srikant, 1994 [AS94]). Il n'est pas ici question d'optimiser directement les règles, mais simplement de trouver les motifs fréquents, pour ensuite générer les règles selon la méthode utilisée dans l'algorithme. Pour les règles contenant des conjonctions dans la prémisse, nous avons utilisé la méthode dévéloppée dans (Couturier, R. 2008 [Cou08]). Prenons un exemple avec 5 variables A, B, C, D et E et cherchons les règles composées de 3 variables, pouvant être de la forme $AB \rightarrow C$. Pour cela, l'algorithme va déterminer les occurrences des triplets de variables, tels que ABC, ABD, ABE, ACD, ACE, ADE, BCD, BCE et CDE. Pour chacun de ces triplets, il a fallu déterminer les occurrences des couples AB, AC, AD, AE, BC, BD, BE, CD, CE et DE. A partir de ces couples et triplets, il est possible de calculer de nombreuses règles. Par exemple avec les occurrences de ABC, de AB, BC et AC, nous pouvons générer les règles $AB \rightarrow C$, $BC \rightarrow A$ et $AC \rightarrow B$.

Dans le cadre des règles d'association négatives, nous ne générons que de type assez classiques suivants, à savoir $\overline{A} \rightarrow B$ (négative à gauche), $A \rightarrow \overline{B}$ (négative à droite), et $\overline{A} \rightarrow \overline{B}$ (bilatéralement négative). Pour ce faire, l'algorithme va déterminer les occurrences des 2-motifs, tels que $\overline{A}B$, $A\overline{B}$, $\overline{A}\ \overline{B}$. Pour chacun de ces couples, il a fallu déterminer les occurrences des motifs A, B, $\overline{A}$, et $\overline{B}$. Comme nous l'avons déjà signalé, la prémisse et la conclusion peuvent être composées de plusieurs attributs, mais un attribut ne peut pas figurer simultanément dans les deux parties de la règle, par exemple $\overline{AB} \rightarrow \overline{CD}$ et $\overline{AB} \rightarrow C$. Celles-ci, grâce aux propriétés 16 et 18 de notre approche, peuvent être inférées à partir des règles respectives $AB \rightarrow CD$ et $AB \rightarrow \overline{C}$. D'un point de vue général, nous avons pour le moment limité sur ces cas assez classiques, nous n'avons pas pu optimiser les règles d'association ayant, en prémisse ou en conclusion, une conjonction des motifs négatifs et positifs, de type par exemple $\overline{A}B \rightarrow \overline{D}$, $A\overline{B} \rightarrow C\overline{D}$, ou $A\overline{B} \rightarrow \overline{C}\ \overline{D}$.

Dans la plupart de cas, le nombre des règles d'association produites par les conjonctions peut s'avérer très élevé si le nombre de variables initiales est relativement grand. De plus, le fait d'utiliser des conjonctions peut être une source de redondance des règles. Pour cela, nous utilisons le critère d'orginalité dans (Couturier, R. 2008 [Cou08]). Nous pouvons en effet sélectionner uniquement les conjonctions des règles présentant un critère d'originalité. Par exemple la règle $AB \rightarrow C$ est originale, si elle a une forte valeur et si les règles d'association $A \rightarrow C$ et $B \rightarrow C$ ont une faible valeur. En outre, d'après nos propriétés 14, 15, 17, 18 et 19, si la règle $AB \rightarrow CD$ n'est pas valide, alors les règles $A \rightarrow BCD$ et $B \rightarrow ACD$ ne seront pas valides non plus. Et si les règles $AB \rightarrow C$ et $AC \rightarrow B$ sont valides, alors la règle $A \rightarrow BC$ le sera également.

7.2.3 Importation des données

Ce module est consacré à la collecte et au prétraitement des données qui seront ensuite manipulées par les modules d'analyse et de représentation. En effet, les données sont initialement décrites par un tableau de données numériques où chaque colonne correspond à un attribut (ou variable) de $\mathcal{I}$, chaque ligne quant à elle correspond à une transaction (ou objet) de $\mathcal{T}$. Comme dans CHIC, le nouvel outil CHIC-M_{GK} prend en charge des fichiers au format CSV, avec un point virgule comme séparateur. Les données sont disposées sous forme d'un tableau de contingence, c'est-à-dire qu'à chaque variable que nous souhaitons évaluer, nous faisons correspondre le résultat de l'évaluation de chaque objet à cette variable. Les variables sont disposées dans la première ligne (ici de FOA1 à FOA10, cf. figure 7.1) et les objets sont dans la première colonne (ici de o1 à o14). Toutes les cases doivent être remplies. Il est préférable d'utiliser le format binaire. Une variable est présente dans un objet, si elle a une valeur 1 et 0 sinon. Une binarisation permet alors d'obtenir des attributs binaires qui s'organisent en contexte $(\mathcal{I}, \mathcal{R}, \mathcal{T})$ où chaque objet de $\mathcal{T}$ est en relation avec $\mathcal{R}$ et un ensemble d'attributs $\mathcal{I}$. La figure 7.1 ci-dessous montre comment sont disposées les données avec le logiciel Excel.

	A	B	C	D	E	F	G	H	I	J	K
1		FOA1	FOA2	FOA3	FOA4	FOA5	FOA6	FOA7	FOA8	FOA9	FOA10
2	o1	0	1	0	1	1	0	0	0	0	1
3	o2	0	0	0	0	0	1	0	0	0	1
4	o3	0	0	1	1	0	1	0	1	0	0
5	o4	0	0	0	1	1	0	1	0	1	1
6	o5	0	1	1	0	1	1	0	0	1	0
7	o6	0	0	1	0	0	1	0	1	1	1
8	o7	0	1	0	0	0	0	0	0	0	0
9	o8	1	1	0	0	1	1	0	0	0	0
10	o9	1	0	0	0	0	1	0	0	1	0
11	o10	0	0	0	0	0	0	0	0	1	0
12	o11	0	0	1	1	0	0	0	0	0	0
13	o12	0	0	0	1	0	0	0	0	0	0
14	o13	0	0	1	1	0	0	0	0	0	1
15	o14	0	1	1	1	0	0	0	0	1	0

FORMAT

Prêt

100 %

Figure 7.1 – Format des données sous Excel

7.2.4 Représentation d'un graphe implicatif

La représentation d'un graphe implicatif est un problème algorithmique ardu. La conception d'un programme offrant les fonctionnalités est une tâche de longue haleine qui requiert de fortes compétences en mathématiques et algorithmique. Nous appelons graphe implicatif sur l'ensemble des variables à une valeur critique ξ_α, le graphe de la relation $\mathcal{R}$ définie par : $A\mathcal{R}B$ tel que $M_{GK}(A \to B) \geq \xi_\alpha$, où $\alpha \in [0, 1]$ est un seuil fixé par l'utilisateur.

Afin de rendre tel graphe plus lisible, l'outil utilise un algorithme de graphes qui essaie de minimiser le nombre de croisements entre les règles à représenter. Dans ce cas, l'utilisateur peut visualiser ses résultats à l'aide du graphe implicatif. Chaque nœud représente les

variables d'implication. L'arête reliant deux nœuds représente le flux des données vers l'opérateur suivant. Comme dans CHIC, le nouvel outil permet, en même temps, de sélectionner des seuils différents, et propose des codes couleurs différents (cyan, vert, bleu et rouge) pour les identifier. Chaque couleur est associée à une plage de valeurs de [50 - 100%] selon le choix de l'utilisateur, comme l'indique la figure 7.2 ci-dessous.

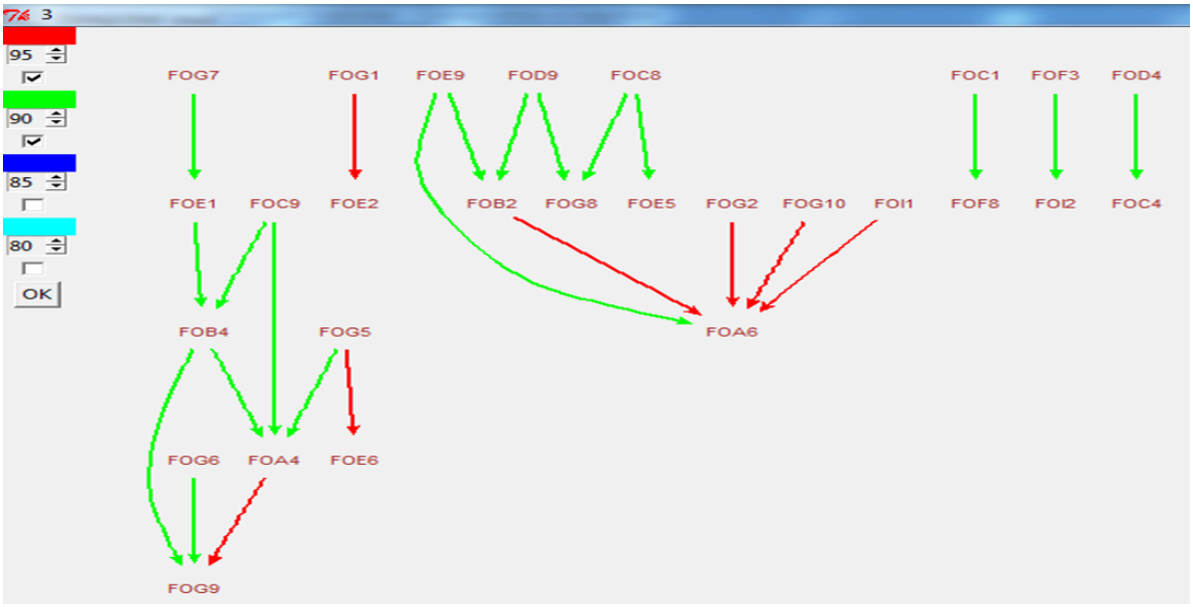

Figure 7.2 – Un exemple de graphe implicatif selon M_{GK}, aux seuils 90% et 95%

Le problème de graphes implicatifs n'est pas trivial, car sa complexité croît avec la taille des règles d'association à représenter. Cette dernière peut être importante, ce qui nous dispose d'une grande surface de travail pouvant être largement supérieure à la taille de l'écran. L'utilisateur peut donc se rendre compte que seules certaines variables lui semblent utiles à interpréter. Il peut aussi supprimer temporairement les variables désirées grâce à une boîte de dialogue prévu à cet effet, puis l'outil met à jour à nouveau le graphe construit, sans re-importer les données. A tout moment, il est possible d'ajouter ou de supprimer des variables dans l'analyse que l'on effectue. Seules les règles d'association impliquant les variables présentes sont représentées, ceci réduit notablement le nombre des règles et rend plus lisible l'ensemble des graphes. A cela, l'outil ne nécessite pas de refaire les calculs puisqu'il les mémorise. Par défaut, les fermetures transitives ne sont pas affichées sur le graphe implicatif, car elles ne présentent aucun intérêt pour l'interprétation. Ensuite, même si l'utilisateur sélectionne ou désélectionne certaines variables, et change le seuil de tests pour les règles, le nouvel outil affiche le graphe sans aucun calcul supplémentaire. Cela permet à l'utilisateur de mettre en évidence les caractéristiques importantes de ses données. Néanmoins, l'utilisation de cette procédure est coûteuse en temps de calculs, c'est pourquoi il n'est pas souhaitable de l'utiliser systématiquement.

7.2.5 Arbre de similarités et arbre cohésitif

Dans cette section, nous présentons dans un premier temps l'élaboration de l'arbre hiérachique de similarités, et dans un second temps celle de l'arbre hiérarchique cohésitif. Comme dans le graphe implicatif, les deux arbres sont obtenus à partir de l'ensemble des règles d'association valides. Elles peuvent s'apparenter à une méthode de classification orientée ou non en fonction du type de calcul choisi "implication ou similarité", mais les manières de construction comportent certaines similitudes. A chaque niveau de la classification, cet outil choisit la classe qui possède la plus grande cohésion en termes de similarité ou d'implication. A chaque étape, il calcule ensuite un ensemble de nouvelles classes à partir des classes présentes dans la hiérarchie. Pour créer une nouvelle classe, on agrège une classe existante avec, soit une variable n'ayant pas été agrégée pour l'instant, soit avec une autre classe de la hiérarchie. Dans ce cas, chaque couple de variables, lors de l'agrégation de deux classes, doit être valide au sens de M_{GK}. Par exemple, la formation de la classe $((A, B), C)$ nécessite que les classes (A, B) et (B, C) aient une bonne cohésion ou soient similaires. La classe $((A, B), C)$ représente la règle $(A \rightarrow B) \rightarrow C$, cela signifie que cette classe admet une bonne cohésion, et que la classe (A, B) est similaire à C. Le détail de cette notion est disponible dans (Lerman, 1981 [Ler81] ; Gras et al., 1996 [GAB+96]). La figure 7.3 ci-après représente un exemple de l'arbre hiérachique des similarités, ce qui s'avère très intéressant en terme de classifications des données.

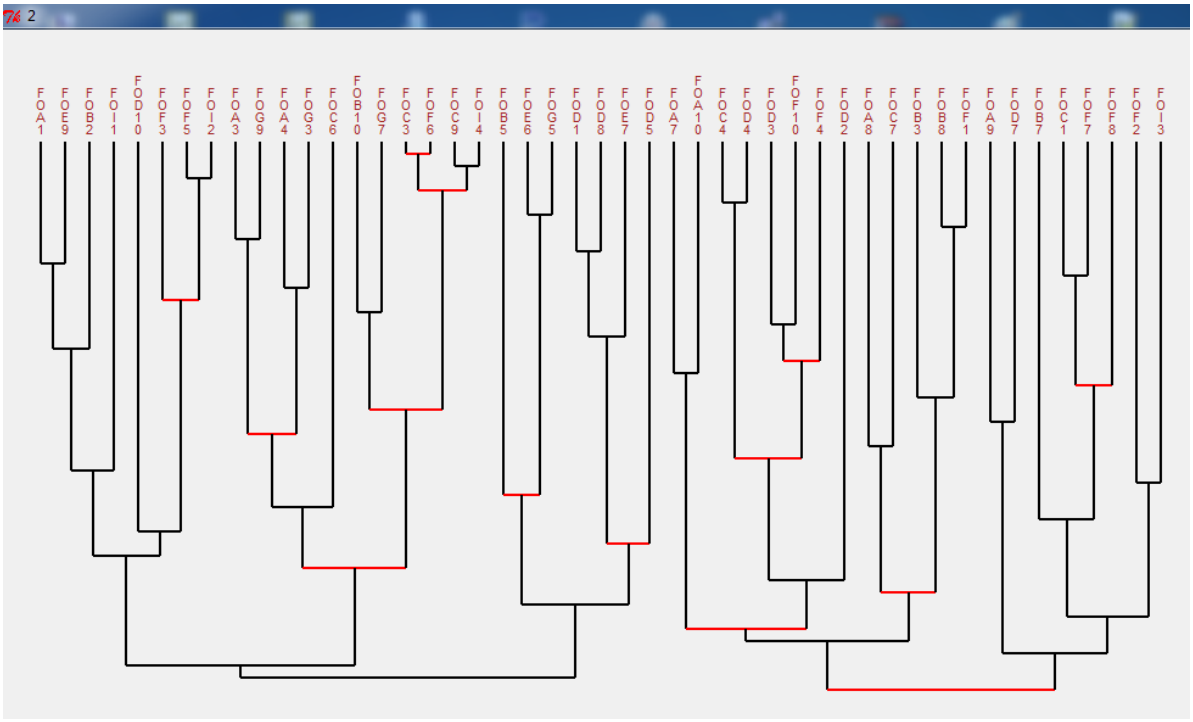

Figure 7.3 – Arbre hiérachique de similarités selon I.C Lerman

L'arbre hiérarchique de similarités montre les ressemblances calculées entre des couples de variables, ou de classes de variables. A cela, plus le niveau du regroupement est élevé, plus il y a de proximité entre les éléments agrégés. En outre, les niveaux identifiés par un trait

rouge gras sont des niveaux significatifs dans la mesure où ceux-ci ont plus de signification classifiante que les autres niveaux. Comme nous avons vu dans le graphe implicatif, l'utilisateur peut déselectionner les variables qu'il ne souhaite pas à représenter. Malheureusement, une petite modification portant sur la présence des variables implique une reconstruction totale de la hiérarchie. Cette étape dépend fortement du nombre de variables concernées dans le calcul. L'algorithme, comme nous l'avons mentionné dans le chapitre précédent, admet une complexité qui dépend de la factorielle du nombre de variables dans le pire des cas. Le processus de construction de telle hiérarchie nécessite donc un temps d'exécution relativement suffisant.

Le second type de classification que nous avons élaboré est l'arbre hiérarchique cohésitif. Dans cet arbre, des classes des règles (ou variables) sont constituées à partir des implications entre celles-ci. A chaque étape de calculs, l'algorithme agrège les variables conduisant à la cohésion la plus forte. La procédure est toujours la même que dans l'arbre des similarités : dériver quelques classes à partir des prototypes existants. La hiérarchie interne des classes respecte le découpage en famille des composants, il est aisé d'identifier rapidement la classe ancêtre adéquate. La figure 7.4 représente l'arbre cohésitif obtenu avec les mêmes données que l'exemple de l'arbre de similarités.

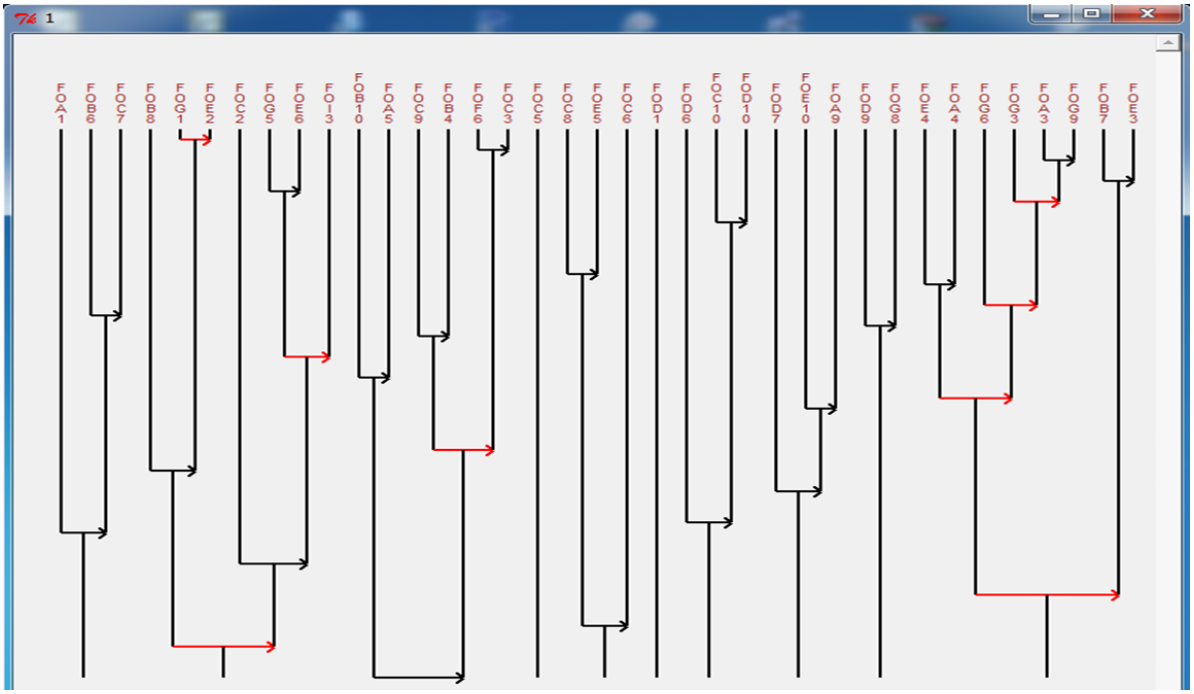

Figure 7.4 – Arbre hiérachique cohésitif selon Gras

Prenons par exemple le deuxième arbre à partir de la gauche de la figure, nous observons au premier niveau de la hiérarchie que la classe $\mathcal{C}_1 = (FOG1, FOE2)$ est créée. Cette classe représente le fait que la variable $FOG1$ implique la variable $FOE2$ avec une forte cohésion que tous les autres couples. Ce premier niveau de la hiérarchie est d'ailleurs significatif comme l'indique la flèche rouge. Au second niveau, $FOG5$ et $FOE6$ sont les variables qui sont les plus semblables, elles forment donc la deuxième classe $\mathcal{C}_2 = (FOG5, FOE6)$

de l'arbre hiérarchique. Au troisième niveau, cette classe est ensuite agrégée à la variable $FOI3$ pour former la classe $\mathcal{C}_3 = (\mathcal{C}_2, FOI3)$. Elle a la plus forte cohésion parmi celles de toutes les classes possibles à trois composantes, elle est aussi plus significative. Au quatrième niveau, la variable $FOB8$ est groupée à la première classe $\mathcal{C}_1$ pour donner la quatrième classe $\mathcal{C}_4 = (FOB8, \mathcal{C}_1)$. Puis, au cinquième niveau, la variable $FOC2$ est réunie à la classe $\mathcal{C}_3$ pour fabriquer la classe $\mathcal{C}_5 = (FOC2, \mathcal{C}_3)$. Finalement, la classe $\mathcal{C}_4$ et la classe $\mathcal{C}_5$ sont regroupées en une classe $\mathcal{C}_6 = (\mathcal{C}_4, \mathcal{C}_5)$. Cette dernière est significative comme l'indique la flèche rouge de la figure. Contrairement à l'algorithme de l'arbre de similarité, l'algorithme de l'arbre cohésitif constitue, de manière quasi systématique, plusieurs classes et arêtes. Son processus de construction, dès que la cohésion entre variables, devient faible. D'un point de vue pédagogique, il nous semble indispensable d'utiliser cet outil pour l'enseignement-apprentissage.

7.3 Applications en didactique des mathématiques

Cette section présente une application de notre nouveau outil CHIC-M_{GK} à la didactique des mathématiques. Ce travail s'inscrit dans le cadre de l'ingéniérie didactique, pour créer des outils et/ou des stratégies didactiques, pour franchir de diffultés et d'obstacles liés à l'appropriation du concept de la statistique par les élèves mais également par les enseignants.

La statistique est aujourd'hui considérée comme l'un des domaines les plus importants des mathématiques, très utilisée pour résoudre certains problèmes de la vie courante. Elle joue aussi un rôle indéniable dans de nombreux domaines scientifiques comme la médecine, la biologie, l'économie, la finance, l'astronomie, la météorologie, etc. Par exemple, en météorologie, en économie ou en astronomie, les ingénieurs effectuent souvent des modèles statistiques pour prévoir l'éventualité qu'un fait précis puisse se produire.

L'enseignement de cette discipline semble aussi essentiel pour la formation des jeunes adolescents dans la mesure où il leur permet d'avoir des outils nécessaires pour décrire certaines expériences statistiques, mais aussi d'avoir un regard critique et de pouvoir prendre des décisions scientifiques. Le module statistique introduit récemment dans le programme scolaire malagasy pose d'énormes problèmes aux élèves mais aussi aux professeurs.

C'est dans ce cadre que nous avons mené une étude portant sur un sujet assez particulier touchant l'enseignement-apprentissage de la statistique à Madagascar. Notre stratégie consiste en la description du problème de l'enseignement-apprentissage de la statistique à partir des programmes scolaires et des sujets de baccalauréat au fil des réformes qui se sont succédées à Madagascar depuis 1970. Il s'agit de faire un état des lieux de difficultés et d'obstacles sur cet enseignement-apprentissage de la statistique, puis de conjecturer, voire proposer, des prescriptions pédagogiques.

Pour ce faire, nous commençons par présenter tout d'abord un préambule sur le pourquoi de l'enseignement de la statistique (Jacobsen, et Robert Morris [Mor94]). Nous exposons ensuite notre nouveau modèle d'identification de ces difficultés et obstacles de cet enseignement-apprentissage de la statistique à Madagascar, à travers les contextes éducatifs. Nous présentons enfin des expérimentations menées sur une population de nos 180 étudiants en L1 lors d'une résolution d'un exercice proposé. Dégageons enfin une synthèse de notre travail en guise de conclusion avant de présenter des propositions de solutions en rapport avec les principaux paradigmes décelés de ce modèle.

7.3.1 Pourquoi enseigner la statistique ?

Prenons un quotidien de notre choix et cherchons-y les éléments d'information dont la compréhension ou l'interprétation fait appel aux mathématiques. Nous n'y trouverons guère de rubrique comportant des équations, des raisonnements géométriques ou trigonométriques, des mises en facteur de polynômes. Par contre, nous rencontrerons des courbes et des graphiques, et abondance de mots comme "moyenne", "tendance", "projection", "estimation", "corrélation", "improbable", "chances", "amélioration", etc. Tous ces mots relèvent du domaine de la statistique. Les pages sportives sont pleines de tableaux et de graphiques retraçant les performances des équipes et des champions. Dans les pages financières, courbes et graphiques illustrent les fluctuations des taux d'intérêt et du cours des devises, ainsi que l'évolution des facteurs de l'inflation, les variations des indices boursiers, la croissance des sociétés et le volume des transactions. Les journaux sont pleins de sondages d'opinion sur la popularité des hommes politiques, sur l'opportunité de construire par exemple une nouvelle école ou un nouvel aéroport, sur les risques que les fumeurs estiment encourir, etc.

Pourquoi la statistique tient-elle tant de place dans la presse ? C'est que nombre de décisions que nous avons à prendre se fondent sur des informations incomplètes ou incertaines, et la statistique peut nous aider à déterminer le parti à prendre. En outre, nos choix comportent souvent un risque de conséquences secondaires néfastes. Tel est le cas de nombreuses décisions collectives, par exemple, choix de l'énergie à utiliser pour la production d'électricité, avec, d'un côté, les risques liés à une irradiation accidentelle et à l'évacuation des déchets nucléaires et, de l'autre, ceux que présentent les pluies acides résultant de la combustion du charbon ; limitation de la vitesse sur les autoroutes ; niveau acceptable d'irradiation des produits alimentaires pour assurer leur préservation ; niveau de radioactivité maximal dans les aliments, dans l'eau, sur les lieux de travail ; teneur des aliments en résidus de pesticides, en hormones, en engrais et en additifs chimiques ; ampleur des effets secondaires admissibles des médicaments, etc. Nous devons être à même de mettre en balance les craintes et les espoirs en mesurant les risques respectifs des options offertes. Nous définissons le risque comme la probabilité que survienne un mal mais nous laissons de côté la question de l'intensité du mal ; autrement dit, nous ne tenons pas compte du fait que les conséquences de deux événements dont la probabilité est identique peuvent être d'une gravité bien différente.

La plupart des gens ont du mal à apprécier la grandeur relative d'un risque. Les compagnies d'assurance, qui sont évidemment expertes en la matière, comparent le nombre de victimes de tel ou tel sinistre au nombre de personnes qui ont encouru le même risque en agissant exactement comme les victimes, mais en sont sortis indemnes. Les médias parlent des victimes, mais ne disent rien de tous ceux qui, avec un même comportement, n'ont subi aucun dommage. Puisqu'il nous est impossible de vivre dans une sécurité absolue, nous devons avoir une intelligence suffisante des probabilités de danger pour mesurer l'importance relative des risques. Les cours de science traditionnels nous apprennent à raisonner dans la certitude, ce qui est injustifié. *La didactique du risque* doit être interdisciplinaire car, comme le fait observer Dickson [Dic85], l'origine des risques peut être scientifique, juridique, politique, financière, sociale, technologique, ou combiner plusieurs de ces facteurs. Tous les citoyens doivent apprendre à mesurer le risque et à mettre en balance les risques et les avantages. La statistique permet de prendre des décisions dans une situation d'incertitude. Chacun doit donc être à l'aise dans le domaine de la statistique.

7.3.2 Identification des difficultés et des obstacles

Cette sous-section décrit l'identification proprement dite des difficultés et des obstacles liés à cet enseignement-apprentissage de la statistique à Madagascar à travers, comme nous l'avons déjà signalé, les programmes scolaires et les sujets de baccalauréat au fil des réformes qui se sont succédées à Madagascar depuis les années 1970.

L'enseignement secondaire malagasy accueille les élèves au sortir des cinq années passées en école primaire. Il se répartit en quatre années de collège dénommé chronologiquement (6è, 5è, 4è, 3è) sanctionné au diplôme BEPC-Brevet d'Etudes du Premier Cycle, et trois années de lycée (seconde, première et terminale) menant quant à lui au diplôme Baccalauréat. Quant à l'organisation/évaluation, il existe différents niveaux d'application et de mise en œuvre des programmes. Sous la tutelle du ministère de l'Education Nationale, la DREN-Direction Régionale de l'Education Nationale, et CISO-Circonscriptions Scolaires assurent entre autres le suivi de l'application des contenus des programmes dans les examens officiels.

A cet effet, le focus de notre travail se centre sur l'analyse critique de l'enseignement-apprentissage de la statistique à Madagascar, en s'appuyant sur l'analyse des programmes officiels et des sujets du baccalauréat série D, là où les programmes de statistiques sont en principe plus avancés.

Une lecture rapide des programmes des mathématiques nous revèle que la statistique ne fait pas partie du programme des mathématiques du secondaire durant la période de *mathématiques traditionnelles* (avant 1970). Par rapport à d'autres domaines (géométrie, algèbre ou analyse), la statistique est donc apparue très tardivement dans le cursus du secondaire. Ce n'est qu'il y a une quarantaine d'années, à l'occasion de la réforme de 1970 dite des *mathématiques modernes*, que la statistique descriptive n'occupe qu'une moindre place. La statistique n'est apparue qu'une seule fois dans les cursus du secondaire. La classe de troisième est la seule classe, parmi 7, ayant l'occasion de voir la notion de statistique.

Aujourd'hui, à l'occasion de la réforme de 1999, la statistique figure dans les programmes de troisième, seconde, première et terminales A et D, mais absente dans la classe de terminale C, alors que celle-ci est la classe scientifique avec un programme renforcé des mathématiques. Tout se passe ainsi comme si la statistique ne fait pas encore partie des mathématiques ! Cela apparaît absurde ! Car c'est la série C qui prépare la pépinière des futurs mathématiciens dont les probabilistes et les statisticiens !

Quant à l'évaluation, la statistique figure dans les sujets d'examen au BEPC, mais cette place est très discutable au baccalauréat. A cet effet, la statistique y apparaît à titre optionnel, et est rarement figurée dans les examens de baccalauréat. Certains professeurs sont découragés par cette situation, alors ils vont jusqu'à la rejeter en fin de programme, voire la bâcler tout simplement. Ils sont souvent déconcertés par cette partie où ils ne sont pas bien à l'aise. Les élèves, de leur côté, sont déroutés par les types de raisonnement qu'ils rencontrent en statistique, qui leur apparaissent très différents de ceux utilisés en analyse, en algèbre ou en géométrie. D'ailleurs, le fait de ne pas pouvoir classer la statistique dans l'un de ces trois domaines pose de réelles difficultés aux élèves.

Pour déterminer ces difficultés et mieux comprendre leurs causes, nous avons analysé les programmes des mathématiques et les sujets de baccalauréat en série D, sur deux périodes, période des mathématiques modernes et celle qui a suivi la réforme de 1999.

Analyse didactique sur les programmes scolaires Nous entrons dans le cadre des programmes scolaires (Collège et Lycée) malagasy depuis les différentes réformes qui se sont succédées. Pour ce faire, nous effectuons notre analyse en deux temps : avant et après la réforme de 1999. Le tableau 7.1 ci-après présente le programme scolaire de la statistique du collège, en classe de troisième.

Objectifs généraux	`Les mathématiques doivent amener l'élève à:` développer des habiletés intellectuelles et psychomotrices ; acquérir les concepts fondamentaux dans le domaine de la numération, de la géométrie et de la mesure ; maîtriser les stratégies et les automatismes de calcul ; acquérir une bonne méthodologie dans la recherche des solutions à des exercices ou problèmes ; conjecturer, s'efforcer de prouver et contrôler des résultats obtenus ; développer les qualités d'expression écrite et orale (clarté du raisonnement, soin apporté à la présentation et à la rédaction) ; acquérir une formation scientifique lui permettant de poursuivre des études et/ou de s'intégrer dans vie active et professionnelle.
Objectifs spécifiques	`L'élève doit être capable de (d'):` observer des données sur une population, représenter une distribution statistique par un histogramme ou un diagramme, calculer les effectifs ou les fréquences, la moyenne, la médiane, la variance et l'écart-type.
Contenus	`Traitement des données:` Tableau statistique, Classe modale, Calcul de la moyenne, de la médiane, de la variance et de l'écart-type. `Diagrammes:` Bâton, Bandes, Rectangle, Circulaire.
Stratégies	`Vocabulaires statistiques:` population statistique, échantillon, caractères qualitatifs et quantitatifs, intervalle et amplitude, mode et classe modale. Histogramme, Notions d'effectifs et de fréquences cumulées.

Tableau 7.1 – Programme du collège sur la statistique avant la réforme de 1999

Comme il a souligné, le programme de la statistique a pour objectif d'amener l'élève à développer des habiletés intellectuelles et psychomotrices, et acquérir les concepts fondamentaux au sens du terme. Il se contente particulièrement à la représentation graphique des séries statistiques à une seule variable et à la formulation des calculs d'un effectif, d'une fréquence, d'une moyenne, d'une médiane, d'une variance ou d'un écart-type, notamment le regroupement des données en classe, et l'indicateur de position. Il vise aussi à initier l'élève à se conformer aux techniques élémentaires pour la prise de décicion scientifique.

Nous voyons dans cette réforme une volonté noosphérienne de renforcer la place de la statistique, mais cette place demeure encore précaire, voire insuffisante pour la formation chez les jeunes adolescents, car après la classe de troisième, la statistique disparaît, ne figure nul part dans les cursus de formations de ces jeunes adolescents. Ces derniers n'ont donc pas l'occasion d'assimiler cette discipline dans leurs cursus. De plus, l'interprétation statistique est absente, c'est-à-dire l'idée d'interpréter les résultats à la signification statistique au sens du terme est ignorée. Cela entraîne un frein incontestable, chez les jeunes adolescents voire la Nation, au développement de la statistique, malgré que la statistique est le domaine qui offre une bonne opportunité de se conformer à l'objectif général d'entraîner ces jeunes de

résoudre des problèmes concrets. Le tableau 7.2 ci-après reporte le nouveau programme du collège en classe de troisième, après la réforme de 1999.

Objectifs généraux	`Les mathématiques doivent amener l'élève à:` développer des habiletés intellectuelles et psychomotrices ; acquérir les concepts fondamentaux dans le domaine de la numération, de la géométrie et de la mesure ; maîtriser les stratégies et les automatismes de calcul ; acquérir une bonne méthodologie dans la recherche des solutions à des exercices ou problèmes ; conjecturer, s'efforcer de prouver et contrôler des résultats obtenus ; développer les qualités d'expression écrite et orale (clarté du raisonnement, soin apporté à la présentation et à la rédaction) ; acquérir une formation scientifique lui permettant de poursuivre des études et/ou de s'intégrer dans vie active et professionnelle.
Objectifs spécifiques	`L'élève doit être capable de (d'):` observer des données sur une population, représenter une distribution statistique par un histogramme ou un diagramme, lire et interpréter des informations d'une série statistique, calculer les effectifs ou les fréquences, la moyenne, la médiane, la variance et l'écart-type.
Contenus	`Traitement des données:` Tableau statistique, Classe modale, Calcul de la moyenne, de la médiane, de la variance et de l'écart-type. `Diagrammes:` Bâton, Bandes, Rectangle et Circulaire. `Initiation à l'utilisation de tableurs-grapheurs ou calculatrice en statistique.`
Stratégies	`Vocabulaires statistiques:` population statistique, échantillon, caractères qualitatifs et quantitatifs, intervalle et amplitude, mode et classe modale. Histogramme, Notions d'effectifs et de fréquences cumulées.

Tableau 7.2 – Programme du collège sur la statistique après la réforme de 1999

L'ancien programme des mathématiques au collège était jugé déjà trop inadapté ; et la nouvelle version n'en a rien corrigé. Ainsi, la statistique n'occupe encore qu'une moindre place, elle ne se trouve qu'en classe de troisième seulement. A ce niveau, le nouveau programme innove peu par rapport à l'ancien. Il reprend l'immense majorité de l'ancien en leur adjoignant un seul point, tel que l'initiation à l'utilisation de tableurs-grapheurs ou calculatrice. L'usage de cet outil permet d'observer dynamiquement les effets des modifications des données. Ceci par ailleurs offre à l'élève une opportunité à l'apprentissage des technologies statistiques, mais diminue très vite de la culture scientifique (aspects psychopédagogiques sur les processus de calculs) chez les jeunes adolescents ; ce qui nécessite donc une certaine pédagogie bien adaptée.

Au Lycée, le programme de mathématiques a connu de nouvelles réformes en introduisant la statistique en seconde de détermination, en première et en terminales A et D. Le nouveau programme est entré en vigueur l'année scolaire 1998-1999, marquant aussi sa volonté de renouveler l'enseignement de la statistique (cf. tableaux 7.3 et 7.4 ci-dessous). Il consiste largement en une actualisation des capacités que les élèves ne sont pas parvenus à acquérir antérieurement, en insistant sur celles qui sont les plus indispensables au développement scientifique et économique de la Nation. Ce nouvel édifice est donc cohérent, voire important chez les jeunes lycéens, mais sa réalisation pose d'énormes problèmes : son application se heurte au fait que, simultanément, le programme d'analyse a été fort réduit et donc nombre

d'outils manquent pour l'étayer, en particulier s'agissant de l'étude des fonctions. Cette situation aggrave chez nombre de professeurs de mathématiques leur réticence face à cette branche du programme.

Objectifs généraux	`Les mathématiques doivent amener l'élève à:` développer des habiletés intellectuelles et psychomotrices ; acquérir les concepts fondamentaux dans le domaine de la numération, de la géométrie et de la mesure ; maîtriser les stratégies et les automatismes de calcul ; acquérir une bonne méthodologie dans la recherche des solutions à des exercices ou problèmes ; conjecturer, s'efforcer de prouver et contrôler des résultats obtenus ; développer les qualités d'expression écrite et orale (clarté du raisonnement, soin apporté à la présentation et à la rédaction) ; acquérir une formation scientifique lui permettant de poursuivre des études et/ou de s'intégrer dans vie active et professionnelle.
Objectifs spécifiques	`L'élève doit être capable de (d'):` faire la distinction entre caractère qualitatif et caractère quantitatif et en donner des représentations graphiques ; lire et interpréter des informations d'une série statistique (représentation graphique ou sous forme de tableau) ; énoncer la définition de quartile, décile d'une série statistique et en donner une signification pratique ; dépouiller des données statistiques à deux variables et les représenter dans un tableau ; étudier et interpréter un tableau de contingence ; représenter une série statistique par un nuage de points ; déterminer les coordonnées du point moyen d'un nuage de points ; faire un ajustement linéaire graphique ; utiliser une droite d'ajustement à des problèmes simples de la vie quotidienne (évolution de prix, de revenus, de la population, etc.)
Contenus	`Caractères qualitatifs et quantitatifs` ; `Représentations graphiques` ; `Caractéristiques de position:` mode, médiane, moyenne, variance, écart-type, quartile et décile ; `Etude conjointe de deux caratctères d'une population:` nuage de points, point moyen ; `Initiation à l'ajustement linéaire par:` méthode graphique, méthode de Mayer.

Tableau 7.3 – Programmes du lycée en classe de terminale A après la réforme de 1999

Les programmes de seconde de détermination [1] et de première, dans le domaine de mathématiques, contiennent bel et bien de la statistique, mais aucune différence par rapport à ceux de la classe de troisième. Ils reprennent exactement ceux de la troisième, si ce n'est, dans la rubrique proportionnalité, des items intitulés : calculer une moyenne, lire et interpréter. Un flagrant phénomène montre que cette discipline est trop rarement traitée. L'articulation entre les classes de seconde, première et terminale, en matière de statistique, se fait donc dans de moins bonnes conditions, car les programmes de seconde et de première n'étant pas en général traités de manière suffisamment différenciée [2], voire non traités tout simplement dans la plupart de cas. En conséquence, les élèves entrant en terminale se trouvent peu préparés à un accroissement subi et important des exigences. Une remarque plus étonnante à propos de cette réforme, comme nous l'avons signalé, est que l'absence de la statistique en terminale C, alors que celle-ci est la série la plus scientifique (où dominent mathématiques ou sciences physiques), censée donner de bonne base en mathématiques. Là, jusqu'à présent, la statistique n'a eu aucune place dans les programmes.

1. Les éléments de statistique qui y figurent doivent être considérés comme faisant désormais partie d'un fonds commun partagé pratiquement par tous les futurs citoyens de ce pays.

2. Les programmes de seconde et de première incluent la *statistique*, qui devrait jouer un rôle important dans la préparation aux diverses classes de terminales, mais cette discipline est trop rarement traitée. Et il n'est pas possible de décider, dans les difficultés relatives aux transitions seconde-première-terminale, ce qui viendrait de cette lacune de ce qui vient d'autres facteurs.

De ce présent programme, l'étude des deux premières parties (caractères qualitatifs et quantitatifs, et représentations graphiques) portera sur l'approfondissement des notions antérieurement acquises. La partie « *caractéristiques de position* » permet d'initier l'élève à l'utilisation du symbole $\sum$ (sigma) pour alléger les écritures de la somme. Le présent programme introduit les notions des statistiques à deux variables, en mettant l'accent sur les droites de régression et l'ajustement linéaire. Constatons également que la notion d'échantillonnage est absente, c'est-à-dire les données statistiques ne sont jamais considérées comme celles d'un échantillon d'une population mère, mais toujours comme celles d'une population toute entière. L'idée d'élargir les résultats obtenus sur une sous-population de la population entière étudiée et l'idée de comparer à d'autres populations ne sont pas exploitées. Synthétisons ci-après le programme de terminale D après la réforme de 1999.

Objectifs généraux	`Les mathématiques doivent amener l'élève à:` développer des habiletés intellectuelles et psychomotrices ; acquérir les concepts fondamentaux dans le domaine de la numération, de la géométrie et de la mesure ; maîtriser les stratégies et les automatismes de calcul ; acquérir une bonne méthodologie dans la recherche des solutions à des exercices ou problèmes ; conjecturer, s'efforcer de prouver et contrôler des résultats obtenus ; développer les qualités d'expression écrite et orale (clarté du raisonnement, soin apporté à la présentation et à la rédaction) ; acquérir une formation scientifique lui permettant de poursuivre des études et/ou de s'intégrer dans vie active et professionnelle.
Objectifs spécifiques	`L'élève doit être capable de (d'):` représenter graphiquement une série statistique simple. Calculer la moyenne, la variance et l'écart-type d'une série statistique simple par application directe de formules appropriées $\overline{x} = \frac{\sum n_i x_i}{N}$, $V(x) = \frac{\sum n_i x_i^2}{N} - \overline{x}^2$, $\sigma(x) = \sqrt{V(x)}$, où $N = \sum n_i$. Représenter graphiquement un nuage de points et déterminer les coordonnées $(\overline{x}, \overline{y})$ du point moyen G. Définir une droite d'ajustement (ou droite de régression) de y en x (resp. de x en y). Calculer une covariance $cov(x, y)$. Déterminer l'équation de la droite de régression de y en x (resp. de x en y). Calculer et interpréter le coefficient de corrélation linéaire d'une série à deux variables x et y $r = \frac{cov(x,y)}{\sigma(x).\sigma(y)}$.
Contenus	`Série statistique à une variable:` Représentation graphique , caractéristiques de position et de dispersion ; `Série statistique à deux variables:` Représentation d'un nuage de points (points pondérés et points moyens), Ajustement linéaire par la méthode des moindres carrés (droite de régression), Corrélation linéaire (coefficient r de corrélation, interprétation de ce coefficient r).

Tableau 7.4 – Programmes du lycée en classe de terminale D après la réforme de 1999

A la lumière de ce tableau 7.4, le programme de terminale D n'a aucune sérieuse différence par rapport à celui de terminale A. Une seule différence se trouve sur la méthode des moindres carrés. La première partie se concentre sur la révision pour maîtriser les notions acquises dans la classe antérieure sur les séries statistiques à simple variable. En seconde partie, la statistique à deux variables (représentation d'un nuage de points, ajustement linéaire, coefficient de corrélation linéaire) est plus accentuée. Cependant, quelques éléments d'indicateur de dispersion comme notions de quartiles, déciles, intervalle interquartile, coefficient de variation, et box-plot n'ont pas été abordés. Il y a donc un retrait par rapport à la terminale A. Ceci semble regrettable, si on veut faire prendre conscience aux élèves du "regard" que l'on peut porter sur une série statistique, d'autant plus que l'on trouve souvent dans des données socio-économiques. Par exemple, pour faire prendre conscience d'inégalités économiques entre régions d'un pays ou entre pays.

La notion d'échantillonnage est également absente. En effet, les connaissances d'un savoir sur le risque à prendre quand on veut tirer la valeur d'un paramètre de la population mère connue de celle d'un échantillon sont ignorées. De plus, les technologies statistiques, qui sont déjà initiées en troisième, sont absentes. La méthode de simulation d'un modèle de la loi mère est également absente, alors que la comparaison de la distribution d'un échantillon avec un modèle de celle de sa population mère est l'objet du test d'adéquation à une loi, qui est une partie très intéressante chez les jeunes adolescents pour connaître le lien Statistique-Probabilité [3]. L'idée de programmer ou de simuler un modèle statistique, même pour une simple initiation, est ignorée, alors que, comme nous le savons, la technologie facilite grandement la tâche de l'apprenant en cette ère.

Un constat essentiel après ce survol d'analyse est que l'enseignement de la statistique dans les lycées malagasy, qui accueillent les élèves en seconde, première et terminale, est assez sommaire et purement descriptif. La statistique à double variables est bien étudiée, mais l'idée de faire prendre conscience aux jeunes lycéens sur les caractéristiques essentielles de la statistique (fluctuation des données, par exemple) et l'idée de mobiliser ces élèves à la notion d'intervalle de confiance ne sont pas explicitées. La statistique inférentielle est donc totalement absente. En effet, la prise de décision à partir d'un échantillon ouvrant la fenêtre vers la problématique des tests d'hypothèses n'est pas exploitée.

L'une des grandes difficultés sur l'enseignement-apprentissage de cette notion est la carrence en connaissances théoriques parce que derrière, il y a des concepts beaucoup plus forts. Ces difficultés ne se limitent pas seulement aux élèves, mais aussi aux professeurs. De nombreux enseignants, bien qu'ils soient diplômés en mathématiques, n'ont jamais fait de la statistique pendant leurs cursus ; ils se sentent bien dépourvus devant la statistique inférentielle. Il est important d'en voir l'impact sur la vie scientifique et économique de la Nation. C'est par ces sections que passent pour leur énorme majorité les responsables à venir de l'enseignement des mathématiques (inspecteurs, professeurs de ces mêmes classes) ainsi qu'une fraction importante des professeurs du lycée de plus haut niveau (professeurs certifiés de l'Ecole normale supérieure). Le risque est donc important que la statistique continue de leur apparaître largement comme un "mal nécessaire" dans les programmes de mathématiques, mal relié à l'ensemble de la discipline et par rapport auquel leur recul restera souvent insuffisant. Cette situation est peu favorable au développement d'une pensée statistique, et on peut craindre que cette situation dommageable ne soit pas fondamentalement modifiée avec les programmes en vigueur.

Notons qu'à aucun moment n'ont été abordés les outils bidimensionnels, ainsi que le jeu de langage "statistique-statistiques", alors que ce langage est souvent dénué d'ambiguïté non seulement aux élèves mais également chez nombre de professeurs. Il est aussi essentiel de remarquer que la volonté de centrer l'enseignement moins sur des concepts que sur des actions (exploiter des données ; identifier, classer, hiérarchiser l'information ; interpréter un résultat statistique) n'est pas contrecarrée par la nécessité de s'adapter aux contraintes inhérentes aux examens de fin de scolarité. Cette situation, en ce qui concerne les épreuves de la statistique au baccalauréat, sera étudiée dans le paragraphe ci-dessous.

3. Par exemple la ressemblance entre fréquence et probabilité, entre variable statistique et variable aléatoire, la moyenne et l'espérance mathématique.

Analyse didactique sur les sujets de baccalauréat L'évaluation est un élément crucial du processus d'apprentissage. Elle est un moyen ce que les élèves ont appris à la fin d'une unité de formation pour s'assurer qu'ils ont le niveau requis pour obtenir un diplôme de fin d'études ou pour sélectionner les élèves à l'entrée de l'enseignement supérieur. Nous synthétisons dans ce paragraphe l'analyse didactique des sujets de baccalauréat série D, dans les 16 dernières années, par rapport aux objectifs généraux du programme officiel. Une épreuve de baccalauréat, vitrine d'une discipline au niveau du lycée, doit être un modèle de qualité scientifique. L'épreuve de statistique au baccalauréat malagasy a été marquée en 1999 par suite de la réforme des programmes. La mise en œuvre de cette épreuve se heurte toutefois à d'importants obstacles, notamment les choix et la conception des épreuves, ainsi qu'un manque de cohérence et de vision systématique entre les démarches d'évaluation des systèmes et des classes.

A Madagascar, les sujets du baccalauréat sont conçus par les professeurs de lycée qui détiennent des classes d'examen. Les sujets sont obligatoirement centralisés dans une banque au niveau de l'Office du baccalauréat. Le choix des sujets est confié à un professeur d'enseignement secondaire chevronné, travaillant seul ou en équipe. La personne ou l'équipe qui choisit expérimente les sujets sur elle-même et apportera les correctifs nécessaires avant de signer les différents textes tels que les candidats les auront pour l'examen. Le sujet doit donc se présenter de telle sorte que le candidat puisse par un effort de transposition faire la preuve non seulement de connaissances et de démarches acquises, mais aussi de leur opérationalité dans des situations nouvelles.

La rédaction de tels sujets se heurte aux contraintes qui pèsent sur le baccalauréat : contraintes liées à la limitation du nombre d'épreuves, au temps alloué à chacune, et à leur normalisation d'une part, et d'autre part, à la société du fait que cet examen est le couronnement d'un cursus sanctionné par un diplôme reconnu. Cette rédaction est surtout difficile : rédiger de tels exercices à partir de documents exige une bonne maîtrise des contenus scientifiques, des raisonnements statistiques rigoureux et des méthodologies démonstratives, mais aussi, et ce n'est pas le moindre, une bonne connaissance des différentes techniques d'évaluation. C'est donc un travail tellement délicat que dans bon nombre de pays, est confié à des instituts spécialisés dans la recherche, la confection des épreuves.

Pour être plus au fait des difficultés rencontrées, nous étudions au paragraphe ci-dessous les sujets de baccalauréat malagasy depuis la réforme de 1999, toujours dans l'optique didactique, c'est le véritable objet de notre analyse.

La réforme pédagogique des lycées malagasy, mise en œuvre il y a 16 ans, avait pour objectif principal de rénover les programmes scolaires et d'améliorer la qualité des examens dans toutes les séries. Le baccalauréat, diplôme national dont l'importance est primordiale pour le futur scolaire des élèves, ne peut échapper à cette règle. À l'issue de la mise en place à partir de l'année 1998-1999 de nouveaux programmes dans les séries générales, le baccalauréat 1999, comme nous l'avons signalé, a vu la première évaluation de la réforme. Nous avons tenté d'observer dans quelle mesure l'épreuve de statistique au baccalauréat malagasy avait pris en compte la réforme effective, et dans quelle mesure cette évaluation avait apportée des effets sur les apprentissages. La première question que l'on peut se poser concerne la cohérence des épreuves de statistique par rapport aux objectifs généraux du

programme officiel. Pour être plus explicite, nous effectuons notre analyse en deux temps : période de 1999 à 2002, et celle de 2003 à 2015.

Baccalauréat Série D, Partie Statistique-Session 1999

Exercice 2. ***NB. On exprimera les résultats sous forme décimale à*** 10^{-2} ***près.***
Le tableau ci-dessous donne en milliards de francs malgache (FMG) les importations d'une société, de 1993 à 1998.

Année	1993	1994	1995	1996	1997	1998
Rang de l'année : x_i	1	2	3	4	5	6
Importations : y_i	5	6.5	7	6.5	10	12

1. Représenter le nuage de points associé à cette série statistique (x_i, y_i) dans un repère orthogonal. L'unité graphique sera prise égale à 1cm sur l'axe des abscisses x, et 1cm pour 1 milliards sur l'axe des ordonnées y.
2. Calculer le coefficient de corrélation linéaire.
3. Par la méthode des moindres carrés, déterminer une équation de la droite de régression de y en x et représenter cette droite dans le même repère défini ci-dessus.
4. A l'aide de cette droite de régression de y en x, quelle estimation peut-on faire du montant des importations en l'an 2004 ?

Baccalauréat Série D, Partie Statistique-Session 2000

Exercice 2. ***NB. On exprimera les résultats sous forme décimale à*** 10^{-2} ***près.***
Le tableau suivant indique les variations du chiffre d'affaires y_i d'une entreprise selon les frais de publicité x_i (x_i et y_i sont exprimés en millions de francs malagasy) de 1992 à 1999.

Année	1992	1993	1994	1995	1996	1997	1998	1999
x_i	2	2.3	2.6	2.9	3.2	3.5	3.8	4.1
y_i	52	59	60	65	70	72	73	75

On donne : $\sum_{i=1}^{8} x_i = 24.40$; $\sum_{i=1}^{8} y_i = 526$; $\sum_{i=1}^{8} x_i^2 = 78.20$; $\sum_{i=1}^{8} y_i^2 = 35048$; $\sum_{i=1}^{8} x_i y_i = 1645.10$.

1. (a) Représenter le nuage de points $M_i(x_i, y_i)$. Unités graphiques :
   - 2cm représente 1 million de francs malagasy sur l'axe des abscisses.
   - 1cm représente 10 millions de francs malagasy sur l'axe des ordonnées.

   (b) Calculer les coordonnées du point moyen G et placer ce point.
2. (a) Montrer que le coefficient de corrélation linéaire associé à cette série statistique est $r \approx 0.98$.

   (b) Interpréter ce résultat.

   (c) Par la méthode des moindres carrés, donner l'équation de la droite de régression (D) de y en x et tracer cette droite.
3. (a) Montrer que $x_1, \ldots, x_8$ constituent les 8 premiers termes d'une suite arithmétique (X_n) dont on précisera la raison.

   (b) Donner une estimation du chiffre d'affaire de cette entreprise en 2002.

Depuis 1999 jusqu'en 2002, les épreuves de mathématiques au baccalauréat acceptent et apprécient les nouveaux programmes de la réforme. L'épreuve de statistique continue cette

ouverture (cf. bacc 1999, 2000 ci-dessus, et bacc 2001, 2002 ci-dessous). A cette époque, la statistique figure dans la même place que les autres modules des mathématiques, tant au volume qu'aux points alloués. Les sujets de mathématiques étaient composés de 4 exercices (géométrie, analyse, probabilité, et statistique) de même poids, mettant en jeu les notions liées aux différentes parties du programme. Cette ouverture semble avoir suscité plus de travaux didactiques sur les séries à double variables et quelques éléments de la statistique à simple variable. On y trouve également une modeste exigence de « littératie [4] statistique ». La discipline statistique tient donc une place assez stable.

Baccalauréat Série D, Partie Statistique-Session 2001

Exercice 2. ***NB. On exprimera les résultats sous forme décimale à*** 10^{-2} ***près.***
Le tableau suivant indique les variations des dépenses annuelles y_i de la famille Rakoto lors de 7 premier mois de l'année 2000. (x_i désigne le rang du mois et y_i est exprimé en milliers de francs malagasy).

Mois	Janvier	Février	Mars	Avril	Mai	Juin	Juillet
x_i	1	2	3	4	5	6	7
y_i	375	387	385	393	400	410	415

On donne : $\sum_{i=1}^{7} x_i = 28$; $\sum_{i=1}^{7} y_i = 2765$; $\sum_{i=1}^{7} x_i^2 = 140$; $\sum_{i=1}^{7} y_i^2 = 1093393$; $\sum_{i=1}^{7} x_i y_i = 11241$.

1. (a) Représenter le nuage de points $M_i(x_i, y_i)$ associé à cette série statistique dans un repère orthogonal.
   - Sur l'axe des abscisses, choisir 1cm pour unité graphique.
   - Sur l'axe des ordonnées, placer 370 à l'origine puis choisir 1cm pour représenter 10 000 francs.

   (b) Calculer les coordonnées du point moyen G.
2. (a) Calculer le coefficient de corrélation linéaire r.

   (b) Interpréter ce résultat.
3. (a) Par la méthode des moindres carrés, donner l'équation de la droite de régression (D) de y en x. Tracer cette droite.

   (b) En utilisant la droite (D), donner une estimation des dépenses de la famille Rakoto pour le mois d'octobre 2000.

Toutefois, les épreuves n'obéissent dans la plupart de cas qu'à un seul critère, celui de leur faisabilité par le professeur et ses élèves. Elles sont souvent critiquées pour leur aspect stéréotypé. Constatons qu'il y a aussi un manque de vision systématique entre les éléments d'un même ensemble. Il y a une fréquentation de quelques parties du programme, ceux qui tombent presque chaque année (*représentation d'un nuage de points, ajustement linéaire par la méthode des moindres carrés, corrélation linéaire*), ceux qui tombent de temps en temps (*paramètres de position et de dispersion*), et les parties qui sont presque abandonnées (*interprétation des résultats, problème de proportionnalité et d'échantillonnage*). Par ailleurs, la

4. Littératie : terme d'usage récent, mais rapidement croissant, en français, parfois orthographié "littéracie" ou "litéracie", issu de l'anglais "literacy" lequel, désignant à l'origine le fait de savoir lire et écrire, a vu son sens se généraliser pour désigner *la capacité d'emploi, pour un public donné, d'un niveau de culture assigné comme objectif dans un domaine donné.*

quasi-totalité des sujets, comme nous l'avons déjà signalé, abordent timidement une approche de caractéristiques de dispersion.

Baccalauréat Série D, Partie Statistique-Session 2002

Exercice 2. Le tableau suivant indique, pour une même distance, les variations des quantités y_i d'essence consommées de certaines voitures suivant leurs puissances x_i (x_i est exprimé en chevaux et y_i en litres).

x_i	3	4	5	5	6	7	8	10
y_i	10	12	20	25	28	30	32	35

On donne : $\sum_{i=1}^{8} x_i = 48$; $\sum_{i=1}^{8} y_i = 192$; $\sum_{i=1}^{8} x_i^2 = 324$; $\sum_{i=1}^{8} y_i^2 = 5202$; $\sum_{i=1}^{8} x_i y_i = 1287$.

1. (a) Représenter le nuage de points $M_i(x_i, y_i)$ associé à cette série statistique dans un repère orthogonal.
   - 1cm sur l'axe des abscisses représente 1 cheval.
   - 1cm sur l'axe des ordonnées représente 5 litres.

   (b) Calculer les coordonnées du point moyen G et placer ce point.
2. (a) Calculer le coefficient de corrélation linéaire associé à cette série.

   (b) Interpréter ce résultat.
3. (a) Par la méthode des moindres carrés, donner l'équation de la droite de régression (D) de y en x. Tracer cette droite.

   (b) Donner une estimation de la quantité d'essence consommée par une voiture de puissance de 12 chevaux.

La situation s'aggrave de plus en plus depuis 2003. La place accordée à la statistique (cf. bacc 2008, 2011, 2013, 2015 ci-dessous) diminue terriblement. La formulation de sujet se dégrade : la notion de « problème » est mise en cause, le contenu effectivement inclus dans ces problèmes formés de plusieurs parties enchaînées ne laissant plus guère de place à l'expression de la créativité ou de l'inventivité des candidats. Les listes de « compétences » sont purement opératoires, impliquant des activités aisées et peu mathématisées, souvent vues comme de simples mises en jeu d'opérations stéréotypées. Son épreuve, à part des contenus [5], est très spéctaculaire. L'épreuve de statistique est souvent absente dans plusieurs sessions de baccalauréat : depuis 16 années de la réforme, la statistique, sans tenir compte des contenus, n'est apparue que 8 fois seulement, donc déjà 8 fois d'absence ! Les causes de cette baisse brutale consécutive peuvent être multiples.

Comme dans bon nombre de pays d'Afrique, la politique de l'enseignement est beaucoup plus influencée par la volonté politique du gouvernement, alors que la plupart des décideurs politiques sont souvent non spécialistes en la matière (ceux qui tombent par hasard par la décision politique, dans la plupart de cas). La politique de l'enseignement à Madagascar n'échappe pas à ce contexte. La crise politique nationale de 2002 modifie notablement l'enseignement de statistique à Madagascar, malgré les efforts déployées. Un rapide constat à travers des sessions de baccalauréat montre que l'épreuve de statistique ne figure nul part

5. L'épreuve de statistique connaît des modifications considérables : il y a des incohérences notoires entre les programmes officiels, et une limitation du volume des questions conduisant à écarter des sérieux problèmes.

dans les sujets de baccalauréat durant les sessions 2003-2007 (5 années d'absences consécutives), alors qu'elle figure bel et bien dans le programme scolaire, sans aucune modification, ni des objectifs, ni des contenus. Ceci incontestablement paralyse, voire entrave la pensée statistique dans le regard scientifique chez les jeunes esprits de la Nation.

La conjugaison de ces facteurs plutôt négatifs a profondement affecté le système éducatif. Après ce long silence, la statistique n'est intégrée qu'en 2008. Et depuis, sa place au baccalauréat reste encore très critiquable : la statistique ne tient qu'une moindre place aux sujets de mathématiques (cf. bacc 2008, 2011, 2013, 2015 ci-dessous). Elle est actuellement associée à la partie probabilité, et n'apparaît que de façon alternative, i.e. son évaluation au baccalauréat est en dents de scie, attribuée à un moindre poids.

Baccalauréat Série D, Partie Statistique-Session 2008

Exercice 2. B. Lors d'un test, les notes obtenues par 4 candidats, aux epreuves de chant et de musique, sont indiquées dans le tableau suivant.

Musique (x_i)	a	3	6	9
Chant (y_i)	2	4	5	b

1. On sait que le point associé à cette série statistique a pour coordonnées $\overline{x} = 5$ et $\overline{y} = 4.5$; déterminer les notes a et b respectivement obtenue par deux candidats différent en musique et en chant.
2. Déterminer le coefficient de corrélation linéaire de cette série. Interpréter le résultat obtenu.
3. Déterminer l'équation de la droite de régression de y en x.

Autrement dit, l'épreuve de statistique, comme nous l'avons signalé, connait des modifications considérables tant des contenus (il y a des incohérences notoires par rapport aux objectifs affichés par les programmes officiels, et une limitation du volume des questions conduisant à éloigner des vrais problèmes) que d'organisation (jusqu'ici 2015, l'épreuve de statistique reste encore irrégulière dans les sessions de baccalauréat). La plupart des épreuves de statistique au baccalauréat donnent aux candidats un problème plutôt formel que concret de la vie courante. L'interprétation des résultats et la sortie des routines sont moins évoquées dans l'ensemble des épreuves. L'introduction de questions donnant l'esprit de modélisation statistique, afin de donner la possibilité de valoriser l'initiative et l'imagination des candidats, est également moins exploitée.

Bien que les programmes en vigueurs affichent les 3 catégories d'objectifs visés par l'enseignement moderne de la statistique : les connaissances, les méthodes, et les savoir-faire (ou attitudes). Cependant, jusqu'ici, seul le premier point fait l'objet d'une évaluation à l'examen de statistique à Madagascar, les deux derniers points sont souvent absents. Nous entendons par « connaissances » tous les éléments factuels, les savoirs déclaratifs, les définitions qui sont véhiculées par le programme et que chaque élève doit garder en mémoire. Les « méthodes » sont toutes les activités intellectuelles qui dépassent le seul niveau de la remémoration, et font appel à une mobilisation des concepts pour déboucher sur une production structurée, par exemple : formuler des hypothèses, analyser et interpréter des résultats. Les « savoir-faire » sont en rapport avec une pratique beaucoup plus concrète sur des points assez précis et plus techniques. Ils sont mesurés par les capacités d'analyse, plus précisément analyse

critique des résultats de la démarche scientifique, et d'exploitation de documents voisins de ceux étudiés en classe.

Baccalauréat Série D, Partie Statistique-Session 2011

Exercice 2. II. On donne sur le tableau ci-dessous le nombre d'élèves d'un lycée ayant le Baccalauréat durant 4 années successives.

Année	2007	2008	2009	2010
Rang de l'année x_i	1	2	3	4
Nombre d'élèves en centaine y_i	3	5	6	9

1. Représenter le nuage des points $M_i(x_i, y_i)_{1\leq i\leq 4}$ associé à cette série statistique. Echelle : sur l'axe des abscisses, prendre 1cm pour représenter une unité. Sur l'axe des ordonnées, placer à l'origine des axes puis 1cm pour représenter 100 élèves.
2. Déterminer le point moyen G.
3. Calculer le coefficient de corrélation r et interpréter.
4. Ecrire l'équation de la droite de régression de y en x.
5. Combien de réussites peut-on espérer en 2014 ?

Les résultats seront donnés à 10^{-2} *près.*

Baccalauréat Série D, Partie Statistique-Session 2013

Exercice 1. B. Etant donnée une série statistique à deux variables (X, Y) dont la droite de régression de Y en X est : $y = 0.12x + 7.88$. Sachant que la moyenne $\overline{X} = 51$ et le coefficient de corrélation $r = 0.93$.

1. Déterminer la moyenne arithmétique $\overline{Y}$.
2. Peut-on avoir un ajustement linéaire par moindre carrée ? Expliquer. Déterminer une équation de la droite de régression de X en Y.

On donnera le résultat à 10^{-2} *près.*

Observant qu'il y a également un confinement de l'épreuve de statistique au baccalauréat malagasy, de manière plus ou moins inconsciente, dans les objectifs les plus faciles à évaluer comme ceux de connaissances ou d'analyse. Et c'est cela qui explique du coup de la hiérarchie que nous avions constatée dans la fréquentation des chapitres, mentionnés ci-dessus. Ce sont les parties qui offrent le plus de facilité de conception d'exercices à partir de documents qui auront les préférences des concepteurs de sujet d'examen.

A vrai dire, les concepteurs se sont heurtés à un problème de *renouvellement des sujets*. Si nous prenons l'exemple des données étudiées, les professeurs avaient d'après le programme la possibilité d'étudier des données *réelles, riches et variées* de leur choix reflétant des exemples concrets, et de faire acquérir à l'élève la méthodologie démonstrative. On considère alors que l'élève arrive au baccalauréat avec ce bagage qu'il devra transposer ; c'est-à-dire que pour éviter la répétition dogmatique, on doit proposer à l'examen un exemple non étudié en classe. C'est ainsi que l'épreuve s'est progressivement standardisée dans la plupart des chapitres du programme avec un certain nombre de situations types ; avec des documents passe-partout. Il y avait là un problème qui se posait avec acuité et qu'il fallait résoudre si l'on ne voulait

pas retomber dans la répétition dogmatique et le bachotage. C'est alors que va s'opérer un tournant décisif face à ce problème de la réforme sur la formulation des sujets.

Baccalauréat Série D, Partie Statistique-Session 2015

Exercice 2. II. Les chiffres d'affaires d'une entreprise de l'année 2008 à 2012 sont représentées dans le tableau suivant : x_i désigne le rang de l'année et y_i le chiffre d'affaire en million d'ariary.

Année	2008	2009	2010	2011	2012
Rang de l'année x_i	0	1	2	3	4
Chiffre d'affaire en million d'ariary y_i	504	580	664	y_3	735

L'équation de la droite de régression (D) de y en x est : $y = 57.3x + 516.2$

1. Calculer les coordonnées du point moyen G.
2. En déduire la valeur de y_3.
3. En quelle année, l'entreprise pourra-t-elle atteindre le chiffre d'affaire de un milliard quatre cent trois millions d'Ariary ?

Notre commentaire du point de vue didactique à ce sujet 2015 est le suivant. A signaler que cette analyse particulière de ce sujet est exemplaire pour l'ensemble des sujets que nous avons étudiés, car on se retrouve qu'ils ont au même aspect de construction.

Sur le fond, l'auteur souhaite probablement évaluer l'état de compréhension des candidats sur le concept de régression, son utilisation en prévision, ainsi que leurs capacités à résoudre un problème concret. Ce problème permet de valider certaines connaissances : moyennes marginales, utilisation des résultats obtenus au problème d'estimation, acquisition de techniques à la prise de décicion scientifique. Sur la formulation du sujet et de question, par rapport à l'échelle taxonomique de Bloom, ces trois questions s'avèrent quelque peu mal présentées pour les deux premières, mais la troisième est bien positionnée. Pour la deuxième question, le candidat pourrait être tenté de comprendre qu'il faut déduire de 1) la valeur de y_3. Alors que son calcul se fait directement à partir de l'équation de la droite de régression. Il conviendrait donc de ranger comme ceci : 2) $\rightarrow$ 1) $\rightarrow$ 3). Bien que le calcul dans la troisième question soit élémentaire, il est jugé qu'une erreur dans ce calcul compromettrait le travail du candidat et serait en conséquence chargée d'un poids assez élevé. Il aurait été plus intéressant de demander aux candidats de : (i) Représenter le nuage de points ou diagramme de dispersion associé. (ii) Est-il opportun de considérer la droite de régression de y en x pour ajuster la dépendance entre les 2 variables en jeux. (iii) Calculer le coefficient de régression linéaire. (iv) En déduire la confirmation de réponse à (ii).

Nous continuons par la suite notre analyse un peu plus globale. Pour élaborer des sujets originaux, les concepteurs ne vont pas hésiter à emprunter des documents qui ont effectué des progrès fulgurants dans des revues scientifiques spécialisées. C'est ainsi que les exercices au baccalauréat vont être construits à partir de documents extraits de revues spécialisées et de publications récentes. Malheureusement, ce travail s'est révélé plein d'obstacles, car la plupart de professeurs n'avaient pas appris l'informatique (ou technologie) et la statistique durant leurs cursus universitaires. L'évocation de ces techniques dans les sujets est souvent incomplète et trop rapide ; ce qui pose évidemment d'énormes problèmes de compréhension de l'énoncé aux pauvres candidats, surtout qu'il y a des fois des erreurs scientifiques et des

simplifications exagérées et des contradictions. Actuellement, le système s'est essoufflé et la tendance est de plus en plus au retour à des documents déjà vus en classe. C'est donc le second piège, celui du renouvellement des sujets et de l'originalité qui n'a pas été résolu.

Constatons qu'il y a aussi un problème de *questionnement et modalités de raisonnement*. L'analyse de ces différents sujets révèle que le questionnement actuel n'est pas aussi clair et précis que le laissent croire les apparences. Nous avons relevé quelques verbes utilisés dans les questions pour voir leurs définitions exactes dans les dictionnaires avant d'étudier l'usage qu'en font les concepteurs des sujets : `Représenter` désigne étymologiquement *replacer devant les yeux de quelqu'un*. Représenter apparaît comme une présentification : il s'agit de rendre quelques chose d'absent présent. Cette notion d'origine latine garde tout son sens étymologique, mais revêt des acceptations sensiblement distinctes suivant le contexte dans lequel elle est utilisée. `Calculer`, du latin *calcular* signifie déterminer, réfléchir, conjecturer. En mathématiques, un *calcul* est une opération ou un ensemble d'opératiions effectuées sur des grandeurs. `Montrer`, du latin *monstrare* signifie, dans la pratique de l'enseignement des mathématiques, prouver, vérifier, ou expliquer. `Déduire`, du latin *deducere* signifie faire comprendre nettement en développant, commenter, interpréter, élucider. `Interpréter`, du latin *interpretari* signifie expliquer, traduire, commenter.

Ceci dégage une liste de quasi-synonymie des termes utilisés dans l'ensemble des sujets que nous avons étudiés. Nous voyons qu'il y a des situations dans lesquelles ces termes sont employés en couple ou même en triplet. Par exemple, commenter et interpréter le résultat. Dans ce cas, il est logique de se demander s'il s'agit toujours de mots qui appellent le même travail intellectuel chez le candidat. C'est-à-dire, où commence le commentaire, et où finit l'interprétation ? L'analyse que nous avons menée à travers ces huit sujets de baccalauréat (1999-2015) révèle en fait qu'aujourd'hui ce vocabulaire du questionnement est un artifice qui cache mal une demande de répétition. Comme nous l'avons signalé ci-dessus, la tendance est de plus en plus à un retour à des documents déjà vus en classe. La situation est telle que les documents proposés à l'examen sont tout à fait similaires à ceux qui ont été expliqués, commentés, ou interprétés en classe par le professeur, de telle sorte que les questions posées sont à la limite des questions de cours déguisées, donc à un niveau très bas de l'échelle de la taxonomie de Bloom : ces sujets d'évaluation normative n'évaluent pas vraiment les objectifs anoncés dans le programme.

En bref, il y a belle lurette, l'épreuve de statistique au baccalauréat de Madagascar a perdu son pouvoir de tester beaucoup plus l'intelligence que la mémoire des élèves. La tendance est plutôt inversée ! La plupart des épreuves de statistique ont tendance à faire appel à la seule mémoire. Très souvent, les sujets ne provoquent pas, au delà des qualités d'analyse de la manifestation, d'autres qualités, et n'exigent pas toujours l'élaboration d'un raisonnement logique et bien structuré. Ils génèrent un conformisme de la pensée au lieu de révéler les sujets créatifs et inventifs, et sont peu propices à déceler les élèves susceptibles de devenir de bons mathématiciens ou de bons statisticiens, de devenir créatif et capable de modéliser un problème concret. Au niveau des programmes, les chapitres privilégiés au baccalauréat le sont en classe. Les techniques et connaissances nouvelles drainées par le baccalauréat arrivent en classe. Au niveau des enseignants, la tâche des professeurs est de plus en plus démesurée : enseigner et faire aimer les mathématiques, en l'occurrence la statistique, dans les classes de terminales apparaît aujourd'hui très éprouvant. Les professeurs ignorent,

voire suppriment, les parties statistiques pour pouvoir terminer les programmes. Au niveau des apprenants, les élèves apprennent le plus souvent leurs cours par cœur et réfléchissent peu devant les situations nouvelles. Si on peut inférer de l'ordre dans lequel sont présentés les items des programmes, nous remarquons que la partie statistique est placée, pour chaque année scolaire, en dernier temps après algèbre, analyse, et géométrie. Ce qui incite nombre d'enseignants à reléguer la statistique en fin d'année scolaire, alors ils la traitent comme supplémentaire, et y consacrent très peu de temps.

7.3.3 Expérimentations

Contexte et population A notre connaissance, la plupart des recherches en didactique de disciplines se sont intéressées aux jeunes adolscents du primaire et du secondaire, très peu d'études concentrent aux étudiants universitaires. Nous relevons de plus que la majorité des travaux concernent l'algèbre, la géométrie, la biologie, la physique ou encore les langues. La statistique demeure rarement la cible des recherches, alors qu'elle peut se révéler un vrai problème de la didactique. Ce n'est qu'à partir de la dernière décénnie, grâce aux contributions d'un certain nombre d'auteurs spécialisés (Bascou, 2001 [Bas01] ; Régnier, 2002 [Rég02] ; Oriol et Régnier, 2003 [OR03] ; Calmant et al., 2011 [CDS11] ; Régnier, 2012 [Rég12] ; Raoult, 2013 [Rao13]), que la statistique trouve sa notable visibilité au sein de la didactique des mathématiques. Ce rapide état des lieux, qui nous a permis de situer notre problématique, nous a amené à nous pencher sur les difficultés et obstacles rencontrés dans l'enseignement-apprentissage des concepts statistiques par nos jeunes étudiants en L1 de l'ENSET, Université d'Antsiranana, Madagascar.

La recherche a été conduite au cours de l'année universitaire 2014-2015 auprès comme nous l'avons signalé de nos étudiants normaliens en L1. Les cursus dans ladite école se déroulent en 5 ans, à la suite d'un concours d'entrée en première année, réservé aux élèves issus de baccalauréat scientifique (séries C, D et Technique Industrielle). La stratégie éducative s'appuie de façon significative sur des scientifiques et pédagogies, visant entre autres à former les étudiants dans la professionnalisation des métiers d'enseignement et d'entreprise de pointes. En première année, les étudiants reçoivent des cours en tronc-commun, un socle de formation de bases scientifiques de remise à niveau. Sur le plan d'apprentissage, ces étudiants avaient été initiés au concept de la statistique descriptive et celui de l'inférentiel.

Notre tâche est de décrire et de comprendre comment se caractérisent les stratégies des élèves dans le cas où ils sont confrontés à un problème dans la résolution d'exercices proposés. Il s'agit d'identifier les difficultés rencontrées dans les concepts de tests paramétriques. Pour ce faire, nous avons décelé dans un premier temps leurs difficultés en identifiant les causes. Nous avons proposé dans un second temps des solutions pour y faire face.

Méthodologie de recueil de données Nos objectifs de recherche se centrent sur les difficultés et obstacles liés à : *l'identification des paramètres statistiques ; la formulation des hypothèses nulle et alternative ; la compréhension du niveau de signification du test ; l'interprétation de la latéralité du test ; l'acquisition du raisonnement inductif en soi ; l'émission de la conclusion du test.* Nous tenterons de décrire les difficultés et obstacles suscités par chacun de ces concepts, et d'élucider les conceptions erronées sous-jacentes. La méthodologie

de recueil de données serait une instrumentation double, celle de l'épreuve écrite (cf. annexe A), et celle des informations complémentaires (cf. annexe B). Ces deux instruments nous apparaissent idoines et complémentaires. Les épreuves écrites permettent de recueillir des données sur un grand nombre de sujets, tandis que les questions supplémentaires produisent quant à elles des informations plus fines et permettent un retour sur les raisonnements des étudiants. Pour la construction de ces dispositifs, nous nous sommes fondés sur les considérations méthodologiques de quelques travaux (Ottaviani et Zannoni, 2001 [OZ01] ; Oriol et Régnier, 2003 [OR03] ; Gueye, 2012 [Gue12] ; Ramanantsoa et Totohasina, 2014 [RT14]).

Elaboration des énoncés L'étape de construction des énoncés et des questions constitue un point stratégique dans la recherche en didactique. L'épreuve écrite joue dans notre dispositif de fournir sur l'ensemble de nos 180 étudiants. La passation de cette épreuve se déroule après que les étudiants, comme nous l'avons déjà signalé, aient suivi le cours de la statistique descriptive et de la statistique inférentielle. Nous y avons tenté de les construire avec pertinence, en étant vigilant sur le contexte, l'intelligibilité et l'absence d'ambiguïtés des énoncés, ainsi que sur les conditions d'application des tests soumis. Pour les deux protocoles, nous avons choisi des énoncés contextualisés dans notre population d'expérimentation. Nous prenons cette précaution afin d'éviter d'éventuelles interprétations erronées des problèmes ou des tâches causées par une incompréhension du thème ou par un excès d'abstraction. La construction et la sélection des énoncés se sont basées aussi sur les aspects familiers et abordables de la thématique. Nous avons pris soin d'éviter des thèmes trop neutres pour les étudiants, trop choquants, trop banaux afin que leurs attentions et leurs motivations ne soient pas détournées de la tâche proposée.

Sur l'intelligibilité des énoncés, nous avons eu le souci de rendre plus intelligibles chacun des éléments des énoncés. Prenons par exemple le cas du test unilatéral, qui est trop vite d'avoir une confusion lorsqu'on compare les valeurs du paramètre. Pour éviter ce type de confusion qui brouillerait notre analyse, nous avons fait le choix de prendre une variable dont la valeur soit directement proportionnelle à l'intensité du phénomène mesuré. Compte tenu de toutes ces contraintes de contexte et d'intelligibilité, notre choix s'est porté sur un thème qui nous apparaît susceptible d'intéresser et de motiver les étudiants. Cependant, même avec un thème compréhensible et accessible, il nous faut prendre le soin de l'exprimer avec clarté.

En effet, il est indispensable de formuler les énoncés des tâches et des problèmes de la façon la plus claire possible en évitant toute ambiguïté. Nous nous sommes donnés aussi comme consigne d'éliminer toute information non utile pour la résolution du problème, afin de limiter les sources collatérales de difficultés de décryptage. Les sujets n'auront pas non plus à effectuer les calculs des moyennes et des écarts-types échantillonnaux, qui seront fournis dans l'énoncé. Nous avons fait ce choix pour écarter le stress et l'embarras fomentés par la manipulation de la calculatrice, et limiter les conclusions erronées dûes à des simples méprises de l'arithmétique.

Sur les conditions d'application, il nous est apparu plus pertinent, pour l'ultérieure analyse des données, de ne proposer aux sujets que des situations de tests relevant des conditions d'application identiques, dans un but de simplification et d'homogénéisation des données, mais aussi dans le but pédagogique de ne pas engendrer d'éventuelles sources supplémentaires d'erreurs chez les sujets. Pour tous les énoncés de tests, nous avons fixé la grandeur

de la taille des échantillons et le statut de l'écart-type σ de la population parente (connu ou inconnu). Nous avons décidé de ne soumettre que des échantillons de grande taille ($n \geq 30$) pour homogénéiser la problématique, et éviter les doutes des étudiants concernant l'application, soit de la loi Normale, soit de la loi de Student. Avec les échantillons de grande taille, la distribution d'échantillonnage appropriée correspondra, pour tous les tests, à une loi gaussienne. En ce qui concerne l'écart-type de la population échantillonnée, nous avons fait le choix de ne considérer que les cas où σ est inconnu, car il correspond à une situation plus courante. Pour les tests de comparaison, nous avons fait le choix de donner l'exemple des échantillons indépendants, car il nous a semblé plus pertinent pour observer avec plus de précision. Tout au long, nous sommes restés sur un test de comparaison.

De ce fait, le premier protocole s'articule autour de problèmes de difficulté croissante portant sur des tests de comparaison à une moyenne : un test bilatéral, un test unilatéral à droite et un test unilatéral à gauche. Dans ce protocole, il s'agira d'un test de comparaison d'une moyenne à une norme (un seul échantillon est impliqué dans le raisonnement) ; la difficulté supplémentaire vient du fait d'avoir à transformer les statistiques échantillonnales. Le deuxième protocole concerne les informations supplémentaires. Ces informations complètent et participent au traitement qui conduit aux graphes de ces difficultés. Elles sont ensuite utilisées pour discriminer les catégories d'étudiants offrant la contribution la plus forte à la constitution d'un chemin du graphe implicatif.

Description des résultats Bien sûr, le problème de l'enseignement-apprentissage de la statistique n'est pas un problème spécifique de Madagascar. Ce sujet a fait l'objet de quelques études didactiques dans certains pays, nous en avons trouvé en France, au Canada, au Sénégal, au Vietnam, où plusieurs difficultés sont relevées aussi bien du côté de l'enseignement que de l'apprentissage.

Dans cette section, nous présentons les réponses formulées par nos étudiants à l'issu d'une part de l'exercice proposé portant sur les tests paramétriques, et d'autre part aux questionnaires sur ce sujet durant les cursus du lycée. Nos expérimentations se focalisent sur la méthode d'analyse statistique implicative (Gras et al. [GAB+96]), en utilisant le nouvel outil `CHIC-`M_{GK}. L'analyse implicative entre les variables ou groupes de variables rend compte d'une structure dynamique au sein de ceux-ci, et dégage les chemins quasi-transitifs orientés par les implications d'un nœud du graphe vers ses successeurs. Elle n'a concerné, dans cette étude, que des variables binaires. Ainsi, chaque modalité est considérée comme binaire qui prend la valeur 1 quand elle se manifeste chez un étudiant et 0 dans le cas contraire, de sorte que l'on obtient une matrice présence-absence de dimension $n \times m$, où n est le nombre des sujets (ici $n = 180$) et m est le nombre des variables binaires. Le terme variable fera ensuite référence à une variable binaire.

Comme dans CHIC, le graphe implicatif de notre outil `CHIC-`M_{GK} est parmi les informations qui semblent les plus importantes. Il donne, comme nous l'avons déjà signalé, une vision immédiate et une lecture aisée des relations implicatives entre les règles d'association valides (ou un grand nombre de couples de variables). La figure 7.5 ci-dessous reporte le graphe implicatif des hiérarchies de réponses de nos étudiants de l'exercice du protocole 1, et des informations supplémentaires du protocole 2, au seuil 5%. C'est une figure complexe qui

représente, en même temps, les variables utilisées, et les chaînes implicatives entre elles. Elle donne une vision globale des modalités de réponses de ces étudiants. A cet effet, le premier graphe repose sur les réponses du protocole 1, le deuxième repose quant à lui sur les réponses du protocole 2.

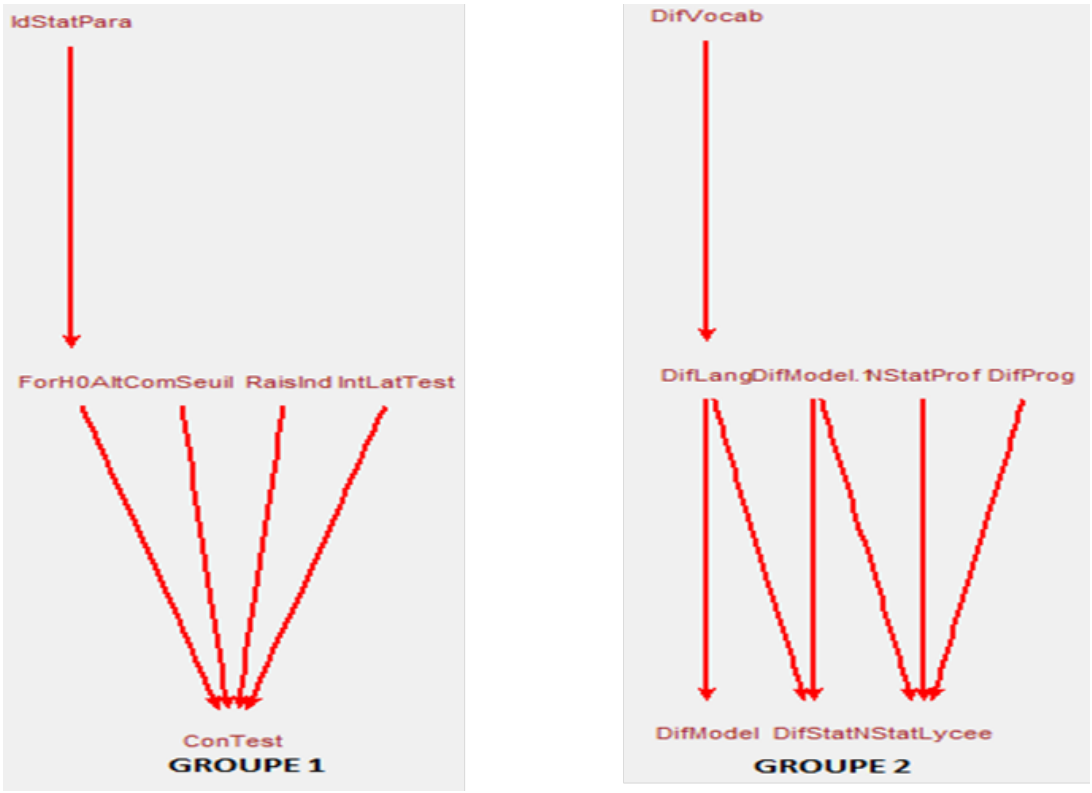

Figure 7.5 – Graphe implicatif pour les difficultés des étudiants au risque de 5%

A la lumière de cette figure 7.5, les étudiants qui ont mal identifié les statistiques et paramètres du test n'arrivent pas à mettre en œuvre successivement les hypothèses du test et la conclusion correcte du sujet, probablement en raison de la moindre familiarité à la statistique inférentielle. La réponse correcte pour formuler les hypothèses du test (hypothèses nulles H_0 et alternatives) à partir des statistiques et des paramètres du test est trop moyenne (58% des étudiants seulement). Les 42% des étudiants n'ont pas pu répondre correctement cette question. Pour la conclusion du test, seuls les 43% des étudiants ont répondu correctement. L'examen des questionnaires à l'aide de ce graphe implicatif 7.5 montre que les erreurs sont également en partie de l'incompréhension du niveau de signification du test (57% des étudiants n'ont pas pu comprendre correctement la signification statistique de ce seuil de confiance), de l'interprétation de la latéralité du test (56% des étudiants n'ont pas pu interpréter correctement leur résultat), et du problème de raisonnement inductif en soi (58% des étudiants n'ont pas du raisonnement correct à cette question).

Les difficultés supplémentaires que nous avons conçues tournent à notre sens autour de quelques points suivants : construction d'une nouvelle statistique à partir d'une moyenne observée, choix du sens de cette différence, et maintien de la cohérence de ce choix pour la

construction de la statistique du test et pour la formulation de la conclusion. Nous relevons de plus que les étudiants se confrontent au problème de la mise en œuvre des étapes du test : comparaison des moyennes (cas du test bilatéral), comparaison des moyennes (cas du test unilatéral à droite), comparaison des moyennes (cas du test unilatéral à gauche).

Le deuxième sous-graphe, entièrement relatif aux réponses aux questions du deuxième protocole, montre que les étudiants rencontrent un problème de compréhension du vocabulaire statistique, notamment du vocabulaire ensembliste. Du fait que le concept des structures algébriques qui sont riches en théorie des ensembles a été supprimé dans les programmes du lycée malagasy depuis la réforme de 1998, le vocabulaire ensembliste est assez étranger chez les jeunes étudiants. En conséquence, ces derniers font souvent la confusion entre réunion et intersection, entre contraire et opposé, etc.

Observant, toujours de la figure 7.5 ci-dessus, que ces difficultés sont ensuite liées à la langue française et à la terminologie utilisée en statistique. La langue maternelle (malagasy) est différente de la langue d'enseignement de la statistique (français), c'est-ce qu'on appelle *ethnostatistique* ou *ethnomathématique* comme l'a étudié dans (Ramanandraisoa, 2013 [Ram13]). La sortie statistique de nos expériences montre que 98% environ des étudiants ayant des difficultés au vocabulaire statistique ont également un problème du langage statistique. Le langage de la statistique étant assez particulier et très subtil, il faut un niveau acceptable en français pour pouvoir suivre. Or, la majeure partie des étudiants ont un niveau faible en français, ils ne maitrisent pas par exemple, les connecteurs logiques et la traduction litérale des concepts statistiques. De ce fait, ils ont du mal à comprendre les textes qu'on leur présente et à les traduire correctement en langage statistique.

De plus, les étudiants ont des difficultés pour modéliser la réalité (cf. figure 7.5 ci-dessus). Ils sont habitués à la résolution d'équations ou d'inéquations, à l'application permanente de façon mécanique, des règles de calcul. Les situations concrètes leur sont rarement proposées, ce qui fait qu'ils ne parviennent pas à comprendre la réalité dont la statistique doit rendre compte, en plus des principes et concepts étrangers ou nouveaux. En outre, les modèles choisis par l'enseignant ne tiennent pas compte, très souvent, de l'environnement de l'élève.

Enfin, une difficulté qui n'est pas la moindre est le manque de la discipline statistique dans plusieurs établissements du lycée malagasy (cf. figure 7.5). Cette difficulté est liée à plusieurs facteurs. Parmi ces derniers, il y a d'abord un problème lié à la formation des professeurs de mathématiques soulevé par 90% des étudiants. Il y a également, comme nous l'avons signalé, des difficultés de construire un modèle adéquat à partir de données réelles et de distinguer la réalité du modèle choisi pour la représenter : la majorité des étudiants sont en difficulté devant les questions donnant l'esprit de modélisation (manque d'initiative et d'imagination). Il y a enfin des difficultés liées au programme : la sortie statistique montre que les 99% des difficultés sont dûes au problème de programme scolaire, c'est-à-dire le programme introduit tardivement l'enseignement de statistique, manque d'articulation et de cohérence.

L'arbre hiérarchique (figure 7.6 ci-dessous) confirme et complète notre précédente étude portant sur les graphes implicatifs. Signalons que cet arbre hiérarchique est sortie du logiciel CHIC (version Couturier, R. 2008 [Cou08]) fondé sur la mesure Intensité d'implication (Gras et al., 1996 [GAB+96]), réposant sur l'algorithme de classification hiérarchique orien-

tée (Gras et Kuntz, 2005 [GK05]). La version avec la mesure de qualité M_{GK} est en cours de développement par un autre collègue du laboratoire.

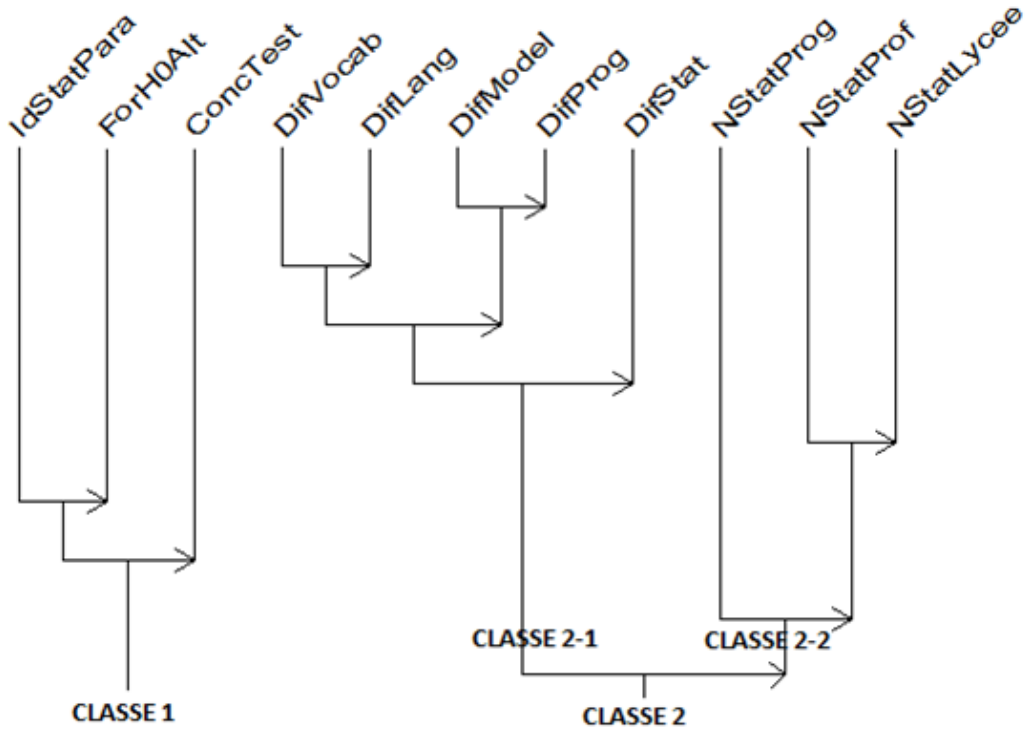

Figure 7.6 – Arbre hiérarchique pour les difficultés des étudiants

On voit que les deux classes ne se lient pas entre elles, ni a fortiori à des niveaux inférieurs de la cohésion, confirmant ainsi ce qui a été mentionné en ce qui concerne l'absence de la statistique inférentielle, notamment les concepts des tests paramétriques, durant les cursus du lycée malagasy.

Dans la première classe, la contribution des deux premières questions à la dernière question est relativement basse (43% c'est-ce que nous avons déjà dit ci-dessus). Ce constat montre bien que l'identification des statistiques et des paramètres du test, et la formulation des hypothèses nulles et alternatives sont, sur le plan cognitif, le fondement des réponses positives à la dernière question (émission de la conclusion du test) puisque le transfert est opérant dans la dernière. Nous pouvons dire que le facteur de réussite tient donc d'une part à la nature algorithmique des tâches, et d'autre part à la familiarisation des étudiants avec les situations antérieures dans l'enseignement secondaire.

Dans la deuxième classe, deux sous-classes 2-1 et 2-2, à des niveaux très inférieurs, apparaissent. La première sous-classe présente les difficultés de l'enseignement-apprentissage de la statistique, tandis que la deuxième expose les principales difficultés pouvant entrainer l'absence de la statistique dans les cursus du lycée malagasy. De la première sous-classe, l'analyse du groupe optimal et de la catégorie contributive à la constitution de cette sous-classe montre, comme nous l'avons déjà dit, que les couples des variables difficultés du vocabulaire/langage

mathématique, et du modèle adéquat/programme scolaire ont une contribution importante (respectivement 98% et 99%) aux desdits difficultés. Pour la deuxième sous-classe, l'absence de la statistique dans les programmes scolaires n'a une contribution importante (0.08%) vis à vis de son enseignement-apprentissage au niveau supérieur, c'est-à-dire l'immense mojorité des étudiants se trouvent dans les mêmes conditions en raison, sans doute, de cette absence. Ceci confirme à ce que nous l'avons signalé : la statistique figure bel et bien dans les programmes scolaires malagasy, mais la mise en vre pose encore d'énormes problèmes. La difficulté liée aux professeurs (cf. figure 7.6 ci-dessus) est parmi les principales causes de l'absence de cette discipline dans plusieurs lycées malagasy, elle constitue les 78%.

7.4 Conclusion partielle et suggestions

Notre premier abord dans ce présent chapitre a été centré sur l'élaboration d'un nouvel outil, CHIC-M_{GK}. D'un point de vue technique, le prototype est développé en C++ et R. Il offre une interface de visualisation interactive portable, et prend en charge pour l'instant des fichiers de données au format CSV (Excel et SAS).

Le deuxième abord se focalise sur la didactique de mathématiques. Nous y avons conçu un nouveau modèle permettant l'identification des difficultés de l'enseignement-apprentissage de la statistique à Madagascar. Au terme de notre étude, nous pouvons conclure qu'il existe effectivement des différentes difficultés et qu'elles sont liées en partie à la carence des concepts, au vocabulaire, et au langage. De plus, elles sont liées à notre sens au manque de manuels conforme aux programmes officiels. Et en grande partie à la modélisation de la réalité, au problème de formation des professeurs de mathématiques et au programme scolaire.

Pour aider à contourner cette situation, il nous semble que, tout d'abord, les concepteurs des programmes de mathématiques devraient réviser le programme et l'étaler sur les trois années du lycée afin que les professeurs ne bâclent plus cette partie et que les élèves puissent avoir un temps suffisant d'assimilation.

Ensuite, les professeurs devraient motiver les élèves et les initier aux langages statistiques en leur présentant des activités concrètes tirées de la vie quotidienne. Par ailleurs, les Inspecteurs, les concepteurs des programmes et les professeurs de mathématiques du secondaire devraient être en collaboration avec les acteurs de l'enseignement des mathématiques du supérieur (Ecole normale supérieure), et devraient aussi organiser régulièrement des ateliers dans lesquels ils feraient diverses activités de modélisation en rapport avec la statistique.

Enfin, une réflexion sur l'enseignement de statistique à Madagascar devrait prendre en compte d'une part la définition claire des objectifs des programmes scolaires, et d'autre part la transition lycée-université. A cet effet, il faut une véritable instance de coordination qui fera en sorte que les différentes structures officielles telles que l'enseignement secondaire (Commission des programmes, DREN) et l'enseignement supérieur (Office du baccalauréat, et Ecole normale) ne travailleront plus de manière séparée, mais de façon concertée et complémentaire au service de la qualité durable. La balle est sans doute dans le camp du ministère de l'Education Nationale, mais aussi dans celui des enseignants, des chercheurs et de tous ceux qui réfléchissent aux modalités de l'enseignement scientifique.

Chapitre 8

Conclusion générale et perspectives

Les travaux de recherche présentés dans ce mémoire ont porté sur l'extraction des règles d'association dans un contexte transactionnel. Au cours de ce travail, nous nous sommes attachés à ne pas nous enfermer uniquement autour de l'extraction des motifs fréquents et des règles d'association positives et négatives pertinentes. Nous avons privilégié en outre la construction des graphes implicatifs à l'aide de la mesure de qualité plus séléctive, M_{GK}. Puis, nous avons implémenté un nouvel outil, `CHIC-`M_{GK}, d'analyse de données, et conçu par ailleurs un nouveau modèle pour ses applications en didactique des mathématiques, particulièrement en didactique de la statistique à Madagascar.

Pour ce faire, nous avons tout d'abord présenté un état de l'art autour d'une part de la problématique de la fouille des données, notamment de l'extraction des règles d'association à sémantique d'implication logique et statistique, et d'autre part de quelques algorithmes permettant la recherche des motifs fréquents et des règles d'association pertinentes. En nous appuyant sur cette étude, nous avons proposé une nouvelle approche permettant l'extraction optimisée des motifs fréquents, fondée sur une nouvelle structure de données `MatriceSupport` et sur les concepts des générateurs minimaux. Nous y avons défini un nouvel algorithme, baptisé Eomf, qui étend l'algorithme Apriori (Agrawal et Srikant [AS94]). L'algorithme ainsi défini est validé par des expérimentations menées sur quelques jeux de données de référence, et comparé aux algorithmes sémantiquement proches et représentatifs de la luttérature. Nos résultats expérimentaux confirment l'efficacité notable de notre approche.

Dans la deuxième contribution, nous nous sommes intéressés à l'extraction des règles d'association valides. La plupart des travaux existants se sont intéressés à l'extraction des règles positives, assez peu d'approches traitent les règles négatives. Par ailleurs, ces travaux requièrent très souvent un grand nombre des règles, dû aux mesures de qualité utilisées. Ils se limitent en effet à l'utilisation du couple classique support-confiance qui produit facilement un nombre prohibitif des règles dont plusieurs sont non pertinentes. A cela, nous avons proposé une nouvelle approche générant à la fois les règles d'association positives et négatives à l'aide du nouveau couple plus discriminant, support-M_{GK}. Egalement, nous avons proposé un nouvel algorithme afin d'automatiser l'extraction. Le choix de ce couple est motivé par le fait que le degré de confiance accordé à la sélection des règles est supérieur. Le modèle ainsi obtenu est validé par des expérimentations menées sur quelques bases des données de référence, et comparé à des approches sémantiquement proches. Les résultats applicatifs démontrent l'efficacité considérable de notre approche.

La troisième contribution a été centrée sur l'élaboration des graphes implicatifs des règles valides afin d'observer leurs enchaînements. Le concept de tels graphes trouve actuellement

une utilisation courante dans le cadre de l'ASI-analyse statistique implicative, mais la majorité des travaux se limitent à l'utilisation unique de l'approche de Gras [Gra79], reposant sur la mesure *intensité d'implication*, basée sur une approximation gaussienne, ce qui n'est donc pas à l'abri de perte d'information. De plus, seules les règles positives sont considérées de cette approche, les règles négatives qui peuvent se révéler une source d'information pertinente sont ignorées, ce qui pose donc un réel problème pour qualifier les résultats obtenus. Pour cela, nous avons proposé une nouvelle méthode d'élaboration de graphes implicatifs en utilisant l'autre mesure plus sélective, M_{GK}. Ainsi, le modèle obtenu intègre à la fois les règles d'association positives et négatives. Puis, il est validé par des expérimentations menées sur quelques bases de données représentatives de la littérature. Les expérimentations ont montré la faisabilité considérable de notre approche.

La quatrième contribution a été focalisée d'une part sur l'implantation d'un nouvel outil `CHIC-`M_{GK}, une séquence du logiciel CHIC (version Couturier, R. 2008 [Cou08]) de Gras, et d'autre part à l'analyse didactique des mathématiques à travers du contexte éducatif à Madagascar depuis 1970. L'originalité de cet outil vient enrichir ce logiciel CHIC, dans le cadre de constructions des graphes implicatifs des règles valides. Sur le plan didactique, nous avons conçu un nouveau modèle, permettant l'identification des difficultés et obstacles de l'enseignement-apprentissage de la statistique à Madagascar. Ce modèle donne entre autres, des prescriptions pédagogiques pertinentes sur le thème des mathématiques, particulièrement de la statistique. L'outil ainsi élaboré a été évalué sur un problème réel de didactique de la statistique, faisant intervenir les difficultés de nos étudiants en L1, lors d'une résolution d'un exercice de tests paramétriques et des questions suplémentaires qui pèsent sur cet enseignement-apprentissage. Nos expériences montrent l'efficacité notable de notre prototype `CHIC-`M_{GK} en regard la stabilité et la qualité des résultats obtenus.

Terminons en dégageant quelques perspectives supplémentaires. De nombreuses perspectives sont apparues au cours de nos travaux de recherche. Nous présenterons une liste *à court terme* s'inscrivant dans la poursuite immédiate, et *à moyen terme*.

A cours terme, nous souhaitons poursuivre l'optimisation de notre algorithme Eomf en combinant les concepts des bordures négatives et des générateurs minimaux. La combinaison de ces deux concepts pourrait encore réduire le coût de l'extraction des motifs fréquents. Une fois le modèle établi, son extension à la génération des règles d'association valides serait bénéfique pour l'optimisation de notre algorithme GenPNR. Dans notre approche M_{GK}-ImplicativeGraph, la portée effective actuelle est limitée puisque nous n'avons pas pu la comparer expérimentalement aux autres travaux. Des études comparatives pour positionner ses performances à celles de la littérature seraient donc une piste des travaux futurs.

A moyen terme, on pourrait étendre notre approche GenPNR sur deux pistes : (*i*) une étude plus poussée sur les règles redondantes, problème rarement abordé à notre connaissance sur l'extraction optimisée des règles négatives et positives ; (*ii*) une optimisation sur les règles ayant, en prémisse/conclusion, une conjonction des motifs négatifs et positifs. On pourrait aussi penser améliorer la robustesse de notre algorithme M_{GK}-ImplicativeGraph. Nous n'avons à l'heure actuelle pas de véritable solution à proposer pour ce problème. Une piste pourrait consister à essayer de trouver les notions de partitionnement. La technologie des graphes implicatifs est assez récente. Nous pouvons donc nous attendre à diverses améliorations d'ordre logiciel ou matériel. Faire passer à l'échelle notre outil `CHIC-`M_{GK} est donc naturel. Cet outil intègre déjà le problème des règles négatives, type non abordé à notre connaissance dans la communauté de l'ASI, ce qui méritrait d'être développé.

Annexes

Nous présentons brièvement ici la méthodologie de recueil/exploitation de données d'expérimentations de la partie didactique. L'expérimentation proprement dite, qui porte sur une population de nos 180 étudiants normaliens inscrits en L1, vise à identifier les difficultés et les obstacles liés à l'enseignement-apprentissage de la statistique à Madagascar. Les informations sont reccueillies à l'issu de l'épreuve écrite (cf. annexe A) et du questionnaire complémentaire (cf. annexe B). Elles sont codées dans les tableaux respectifs 8.2 et 8.3 de l'annexe C ci-dessous.

Annexe A : Epreuve écrite

Une étude a été conduite chez 81 patients dont le dosage saguin du cholestérol total était supérieur à $2g/l$. Après un an de traitement hypocholestérolémient, un second dosage a été réalisé. Les résultats sont les suivants :

Cholestérol total (g/l)	moyenne	écart-type de la moyenne
Avant traitement $(n = 81)$	2.27	0.21
Après un an de traitement $(n = 81)$	2.12	0.15
Différence $(n = 81)$	0.15	0.06

Tableau 8.1 – Statistiques des résultats du test

1. Analyser ces résultats, en répondant les questions suivantes :
   (a) Identifier les statistiques et les paramètres du test.
   (b) Formuler les hypothèses du test.
   (c) Interpréter la latéralité du test.
   (d) Comment comprenez-vous le niveau de signification du test ?
   (e) Faire une acquisition du raisonnement inductif en soi.
2. Quelle conclusion tirez-vous de l'analyse ?

Annexe B : Informations complémentaires

1. Avez-vous suivi le cours de statistique dans le lycée ? Si NON, nous vous prions de bien vouloir répondre aux questions suivantes.
   (a) Les défauts sont-ils liés au problème de professeurs ? Pourqoui ?
   (b) Les défauts sont-ils liés à l'insuffisance du volume horaire ? Pourquoi ?
   (c) Les défauts sont-ils liés au programme scolaire ? Pourqoui ?
2. Avez-vous rencontré des difficultés dans l'apprentissage de statistique ? Si OUI, nous vous prions de bien vouloir répondre aux questions suivantes.
   (a) Les difficultés sont-elles liées au problème de langage ? Pourqoui ?
   (b) Les difficultés sont-elles liées à l'incompréhension du vocabulaire ? Pourquoi ?
   (c) Les difficultés sont-elles liées au programme scolaire ? Pourqoui ?
   (d) Les difficultés sont-elles liées au problème de modélisation de la réalité ?

Annexe C : Codage du questionnaire

QUESTION 1	CODE
Identification des statistiques et des paramètres du test	IdStatPara
Formulation des hypothèses (nulle et alternative) du test	ForH0Alt
Interprétation de la latéralité du test	IntLat
Compréhension du niveau de signification du test	CompSeuil
Acquisition du raisonnement inductif en soi	RaisInduc
Emmission de la conclusion du test	ConcTest

Tableau 8.2 – Codage de l'épreuve écrite

QUESTION 2	CODE
Difficultés de l'enseignement-apprentissage de la statistique	DifStat
Difficultés liées au problème de vocabulaires	DifVocab
Difficultés liées au problème de langage	DifLang
Difficultés liées au problème de modélisation	DifModel
Difficultés liées au programme scolaire	DifProg
Difficultés liées au professeur de mathématiques	DifProf
Absence de la statistique dans le cursus du lycée	NStatLycée
Absence de la statistique liée au professeur	NStatProf
Absence de la statistique liée au porogramme	NStatProg

Tableau 8.3 – Codage des informations supplémentaires

Bibliographie

[AGI+92] R. Agrawal, S. Ghosh, T. Imielinski, B. Iyer, and A. Swami. An intervall classifier for database mining applications. pages 560–573, 1992. 10

[AGM04] F. Afrati, A. Gionis, and H. Mannila. Approximating a collection of frequent sets. In *KDD'04 : Proceedings of the tenth ACM SIGKDD*, pages 12–19, 2004. 2, 46

[AIS93] R. Agrawal, T. Imielinski, and A. N. Swami. Mining association rules between sets of items in large databases. In *Proceedings of the ACM SIGMOD Conference*, volume 22, pages 207–216, Washington, DC, 1993. 1, 10, 17, 18, 19, 21, 43, 46, 48, 59, 91

[AS94] R. Agrawal and R. Srikant. Fast algorithms for mining association rules. In *Proceedings of the 20th VLDB Conference*, pages 487–499, Santiago, Chile, 1994. 1, 2, 3, 21, 22, 26, 38, 41, 45, 46, 47, 48, 49, 52, 55, 58, 59, 91, 119

[AS09] A. Aguilera and A. Subero. *Knowledge Representation in Difficult Medical Diagnosis*. P. Perner (Ed.) : Industrial Conference on Data Mining, ICDM 2009, LNAI 5633, 2009. 19, 88

[AZ04] M. L. Antonie and O. R. Zaïane. Mining positive and negative association rules : An approch for confined rules. In *Proc. 8th Int. Conf. on Principal and Practice of Knowledge Discovery in Databases (PKDD'04)*, pages 27–38. Springer-Verlag, 2004. 59

[Bac14] S. Bachy. Un modèle-outil pour représenter le savoir technopédagogique disciplinaire des enseignants. *Revue internationale de pédagogie de l'enseignement supérieur*, pages 1–23, 2014. 89

[Bas01] N. Bascou. Des statistiques à la pensée statistique. Institut de Recherche pour l'Enseignement des Mathématiques IREM de Montpellier, 2001. 112

[Bay98] R. J. Bayardo. Efficiently mining long patterns from databases. In *Proc. of the ACM SIGMOD Conference*, pages 85–93, Washington, U.S.A., June 1998. 26, 34, 35

[BBBG09] M. Berlingerio, F. Bonchi, B. Bringmann, and A. Gionis. Mining graph evolution rules. *European Conf. on Machine Learning and Princ. and Pract. of Knowl. Disc. in Databases (ECML/PKDD)*, pages 115–130, 2009. 75

[BBJ00] J.-F. Boulicaut, A. Bykowski, and B. Jeudy. Towards the tractable discovery of association rules with negations. *Conference on FQAS'00*, pages 425–434, 2000. 59

[BD13] P. Bertrand and J. Diatta. *Prepyramidal clustering and Robinsonian dissimilarities : one-to-one correspondences*. WIREs. Data Mining and Knowledge Discovery, 2013. 1

[Ber07] D. Berthiaume. Une description empirique du savoir pédagogique disciplinaire des professeurs d'université. *Dans Actes du colloque de l'AIPU : regards sur l'innovation la collaboration et la valorisation*, pages 179–181, 2007. 89

[BGS12] B. Boden, S. Günnemann, and T. Seidl. Tracing clusters in evolving graphs with node attributes. *CIKM*, pages 2331–2334, 2012. 76

[BGYS07] S. Bouker, G. Gasmi, S. B. Yahia, and Y. Slimani. Extraction des bases génériques des règles généralisées. *ECG 2007*, 2007. 60

[BHB+09] B. Brühl, M. Hulsmann, D. Borscheid, C. M. Friedrich, and D. Reith. *A Sales Forecast Model for the German Automobile Market Based on Time Series Analysis and Data Mining Methods*. P. Perner (Ed.) : Industrial Conference on Data Mining, ICDM 2009, LNAI 5633, 2009. 19, 88

[BKSS90] N. Becmann, H. P. Kriegel, R. Schneider, and B. Seeger. The R*-tree : An efficient and robust access method for points and rectangles. In *ACM'90*, pages 322–331, 1990. 10

[BL09] E. Bahri and S. Lallich. Introduction de l'élagage pour l'extraction de règles d'association de classe sans génération de cardinalité. *QDC*, pages 26–32, 2009. 33

[BMS97] S. Brin, R. Motwani, and C. Silverstein. Bayond Market Baskets : Generalizing Association Rules to Correlations. In *Proceedings of the ACM SIGMOD*, pages 265–276, 1997. 59, 60, 61, 62

[Bor86] J. P. Bordat. Calcul pratique du treillis de Galois d'une correspondance. *Math. Sci. Humaines*, 96 :31–47, 1986. 36

[Bor03] C. Borgelt. Efficient Implementations of Apriori and Eclat. In *FIMI'03 Workshop on Frequent Item Set Mining Implementations*, Aachen, Germany, CEUR Workshop Proceedings 90, November 2003. 56

[Bra09] E.M. Braga. *Enseignement apprentissage de la statistique, TICE et environnement numérique de travail : Etude des effets de supports didactiques numériques, médiateurs dans la conceptualisation en statistique*. PhD thesis, Université Lumière, Lyon 2, 2009. 89

[BRB04] J. Besson, C. Robardet, and J-F. Boulicaut. Mining a new faulttolerant pattern type as an alternative to formal concept discovery. *Proc. ICCS*, pages 144–157, 2004. 47

[BRTR12] P. Bemarisika, H. Ramanantsoa, A. Totohasina, and L. Ramifidisoa. Enseignement et apprentissage de la résolution d'équations polynomiales par l'utilsation de TIC au niveau secondaire. In *Colloque international sur les TIC*, ENS Ampefiloha, Antananarivo, 2012. 4

[BT14a] P. Bemarisika and A. Totohasina. Elaboration of implicative graph according to measure M_{GK}. *IJCSI International Journal of Computer Science Issues*, Vol. 11, No 1, Issue 4 :52–59, July 2014. 4, 82, 90

[BT14b] P. Bemarisika and A. Totohasina. A Novel Algorithm for Mining Negative and Positive Association Rules. *International Journal of Computer and Information Technology*, Volume 03-Issue 04 :792–798, July 2014. 3, 60, 91

[BT14c] P. Bemarisika and A. Totohasina. Apport des règles négatives à l'extraction des règles d'association. *Actes de la 21ème Rencontre de la Société Francophone de Classification*, pages 99–104, Sep. 2014. 3, 60, 61

[BT16] P. Bemarisika and A. Totohasina. EOMF : Un algorithme d'extraction optimisé des motifs fréquents. In *Proceedings of AAFD & SFC 2016 : Francophone International Conference on Data science, Mathematical and algorithmic challenges*, pages 198–203, Marrakech Maroc, Mai 2016. 3, 47, 55

[BTP+00] Y. Bastide, R. Taouil, N. Pasquier, G. Stumme, and L. Lakhal. Mining frequent patterns with counting inference. *ACM-SIGKDD Explorations*, 2(2) :66–75, 2000. 16, 48

[BTP+02] Y. Bastide, R. Taouil, N. Pasquier, G. Stumme, and L. Lakhal. Pascal : un algorithme d'extraction des motifs fréquents. *Technique et sciences informatiques*, 21 :65–95, 2002. 46, 52, 55, 58

[Bur98] P. Burmeister. Formal concept analysis with conimp : Introduction to the basic features. 1998. 36

[BWMC09] O. Baqueiro, Y. J. Wang, P. McBurney, and F. Coenen. *Integrating Data Mining and Agent Based Modeling and Simulation*. P. Perner (Ed.) : Industrial Conference on Data Mining, ICDM 2009, LNAI 5633, 2009. 1, 20, 88

[CDS11] P. Calmant, M. Ducarme, and M. Schneider. Obstacles a priori à l'apprentissage de l'analyse statistique inférentielle. *Statistique et Enseignement, Société Française de Statistique (SFdS)*, 2(1) :43–59, 2011. 112

[CK07] V. Chandola and V. Kumar. Summarization Compressing data into an informative representation. *Knowl. Inf. Syst.*, 12 :355–378, 2007. 46

[CKR93] K. F. Cochran, R. A. King, and J. A. De Ruiter. Pedagogical content knowledge : an integrative model for teacher preparation. *Journal of teacher Education*, 44(4) :263–272, 1993. 88

[CLRS01] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein. Introduction to algorithms. *Second Edition, The MIT Press*, 2001. 75

[Cou08] R. Couturier. Statistical implicative analysis, CHIC : Cohesive Hierarchical Implicative Classification. *Studies in Computational Intelligence*, 127 :41–52, 2008. 4, 89, 90, 91, 116, 120

[CPRC09] T. Charnois, M. Plantevit, C. Rigotti, and B. Crémilleux. Fouille de données séquentielles pour l'extraction d'information dans les textes. *TAL*, 50, 2009. 47

[CR93] C. Carpineto and G. Romano. Galois : an order theoretic approach to conceptual clustering. In *Proceedings of the Machine Learning Conference*, pages 33–40, 1993. 36

[CRB04] T. Calders, C. Rigotti, and J-F. Boulicaut. A survey on condensed representations for frequent sets. *Springer*, 3848 :64–80, 2004. 2, 47

[CYH08] H. Cheng, P. S. Yu, and Han. Approximate frequent itemset mining in the presence of random noise. *Soft Computing for Knowledge Discovery and Data Mining*, pages 363–389, 2008. 47

[CYZC06] C. Cornelis, P. Yan, X. Zhang, and G. Chen. Mining positive and negative association rules from large databases. *Proceedings of the IEEE*, pages 613–618, 2006. 59

[Ded00] R. Dedekind. über zelengungen von zahlen durch ihre grosten gemeinsamen teiler. 2 :103–148, 2000. 11

[DGG+09] M. Deodhar, G. Gupta, J. Ghosh, H. Cho, and I. S Dhillon. A scalable framework for discovering coherent co-clusters in noisy data. *Proc. ICML*, 2009. 47

[Dic85] G.C.A Dickson. Risky businesses. times higher education supplement. 1985. 97

[Dom02] F. Domenach. *Structures latticielles, correspondances de* G*alois contraintes et classification symbolique*. PhD thesis, Université Paris 1 Panthéon-Sorbone, France, 2002. 12

[DPB14] E. Desmier, M. Plantevit, and J-F. Boulicaut. Granularité des motifs de covariations dans des graphes attribués dynamiques. *ECG*, pages 431–442, 2014. 76

[EKX95] M. Ester, H. P. Kriegel, and X. Xu. A database interface for clustering in large spatial databsaes. pages 94–99, 1995. 10

[EP96] J. Elder and D. Pergibon. A statistical perspective on knowledge discovery in databases. *AAAI Press*, pages 83–115, 1996. 10

[Fau07] C. Fauré. *Découvertes de motifs pertinents par l'implémentation d'un réseau bayésien : application à l'industrie aéronautique*. PhD thesis, Institut National des Sciences Appliquées de Lyon, 2007. 20

[Fen07] D. R. Feno. *Mesures de qualité des règles d'association : normalisation et caractérisation de bases*. PhD thesis, Université de La Réunion, France, 2007. 12, 60

[Fis87] D. Fisher. Knowledge acquisition via incremental conceptual clustering. *Machine Learning*, 2 :139–172, 1987. 10

[FPSM92] W. J. Frawley, G. Piatetsky-Shapiro, and C. J. Matheus. Knowledge Discovery in Databases. volume 13, pages 57–70, 1992. 8

[FPSS96a] U. Fayyad, G. Piatetsky-Shapiro, and P. Smyth. *From data mining to knowledge discovery in databases*. AI Magazine, Potland, 1996. 9

[FPSS96b] U. M. Fayyad, G. Piatetsky-Shapiro, and P. Smyth. Knowledge discovery and data mining : towards a unifying framework. In *Proceedings of the second International Conference on Knowledge Discovery and Data Mining*, pages 82–88, Portland, OR, 1996. 8

[FPSSU96] U. M. Fayyad, G. Pietetsky-Shapiro, P. Smyth, and R. Uthurusamy. Advances in knowledge discovery and data mining. 1996. 10, 19, 88

[GAB+96] R. Gras, S. A. Almouloud, M. Bailleul, A. Lahrer, M. Polo, H. Ratsimba-Rajohn, and A. Totohasina. L'implication statistique : Nouvelle méthode exploratoire de données. *La Pensée Sauvage*, 1996. 3, 4, 76, 79, 89, 94, 114, 116

[GH07a] F. Guillet and H. Hamilton. Quality measures in data mining. *Springer-Verlag*, 2007. 1, 60

[GH07b] F. Guillet and H.J. Hamilton. *Quality Measures in Data Mining*. Fabrice Guillet, Howard J. Hamilton (Eds.), Springer-Verlag Berlin Heidelberg, 2007. 1

[GK05] R. Gras and P. Kuntz. Discovering r-rules with a directed hierarchy. *Soft Computing. Sous presse*, 2005. 4, 5, 90, 117

[GP13] S. Guillaume and P-A. Papon. Extraction optimisée de règles d'association positives et négatives (RAPN). *RNTI*, 2013. 60, 69

[Gra79] R. Gras. *Contribution à l'étude expérimentale et à l'analyse de certaines acquisitions cognitives et de certains objectifs didactiques en mathématiques*. PhD thesis, Université de Rennes I, 1979. 3, 76, 79, 87, 120

[GSK05] R. Gras, E. Suzuki, and P. Kuntz. Règle et R-règle d'exception en Analyse Statistique Implicative. *Actes de la 3è Rencontre ASI*, 2005. 3, 76

[Gue12] K. Gueye. Difficultés de l'enseignement et de l'apprentissage des probabilités dans le secondaire. In *Enseignement des mathématiques et contrat social : Enjeux et défis pour le 21e siècle. Actes du colloque EMF2012*, pages 1604–1615, UCAD. Dakar, Sénégal, 2012. 113

[Gui00] S. Guillaume. *Traitement des données volumineuses. Mesures et algorithmes d'extraction des règles d'association et règles ordinales*. PhD thesis, Universté de Nantes, France, 2000. 62, 76

[Gui10] S. Guillaume. Améliorations de la mesure d'intérêt M_{GK}. *Actes de la XVIIè Rencontre de la Société Francophone de Classification*, pages 41–45, 2010. 60, 63

[GW99] B. Ganter and R. Wille. *Formal concept analysis, Mathematical foundations*. Springer Verlag, Berlin, 1999. 15

[HCC92] J. Han, Y. Cai, and N. Cercone. Knowledge discovery in databases : An attribute oriental approach. 1992. 10

[HHC66] P. Hajek, Havel, and Chytil. The guha method of automatic hypotheses determination. In *Computing*, pages 293–308, 1966. 1

[HK01] J. Han and M. Kamber. Data mining : Concepts et techniques. 2001. 10

[HKM+96] K. Hätönen, M. Klemettinen, H. Mannila, P. Ronkainen, and H. Toivonen. Knowledge discovery from telecommunication netwok alarm databases. In *ICDE'96*, pages 115–122, 1996. 19, 88

[HKP12] J. Han, M. Kamber, and J. Pei. *DATA MINING : Concepts and Techniques*. Third Edition, 2012. 1

[HN11] B. Hanczar and M. Nadif. Using the bagging approach for biclustering of gene expression data. *Neurocomputing*, 75(10) :1595–1605, 2011. 47

[HPYM00] J. Han, J. Pei, Y. Yin, and R. Mao. Mining frequent patterns without candidate generation : A frequent-pattern tree approach. pages 53–87, 2000. 2, 26, 31, 46, 47

[HR99] P. Hajek and J. Rauch. Logics and statistics for association rules and beyond. In *Tutorial PKDD'99, Prague*, 1999. 1

[HYN11] T. Hamrouni, S. Ben Yahia, and E. Mephu Nguifo. Construction efficace du treillis des motifs fermés fréquents et Extraction simultannée des bases génériques de règles. *Math. Sci. hum. Mathematics and Social Sciences (49è année, nº 195)*, 3 :5–54, 2011. 48

[Idi13] B. Idiri. *Méthodologie d'extraction de connaissances spatio-temporelles par fouille de données pour l'analyse de comportements à risques - Application à la surveillance maritime*. PhD thesis, 2013. 20

[IPB+09] A. Ivannikov, M. Pechenizkiy, J. Bakker, T. Leino, M. Jegoroff, T. Kärkkäinen, and S. Äyrämo. *Online Mass Flow Prediction in CFB Boilers*. P. Perner (Ed.) : Industrial Conference on Data Mining, ICDM 2009, LNAI 5633, 2009. 20

[JAAXR08] R. Jin, M. Abu-Ata, Y. Xiang, and N. Ruan. Effective and efficient itemset pattern summarization : Regression-based approaches. In *KDD'08 : Proceedings of the 14th ACM SIGKDD*, pages 399–407, 2008. 47

[JMA07] R. Jin, S. McCallen, and E. Almaas. Trend Motif : A graph mining approach for analysis of dynamic complex networks. *ICDM,IEEE*, pages 541–546, 2007. 75, 76

[JPS+06] Liu J., S. Paulsen, X. Sun, A. B. Nobel, and J. Prins. Mining approximate frequent itemsets in the presence of noise : Algorithm and analysis. *Proc. SDM*, pages 405–416, 2006. 47

[KLR11] M. Karima, L. Létocart, and C. Rouveirol. Itemset mining in noisy contexts : A hybrid approach. *Proc. ICTAI*, pages 33–40, 2011. 47

[KLR12] M. Karima, L. Létocart, and C. Rouveiro. Extraction de biclusters contraints dans des contextes bruités. *CAp2012*, 2012. 47

[KMT97] M. Klemettinen, H. Mannila, and H. Toivonen. A data-mining methology and its application to semi-automatic knowledge acquisition. In *DEXA'97*, pages 670–677, 1997. 19

[KNZ01] Y. Kodratoff, A. Napoli, and D. Zighed. extraction de connaissances dans des bases de données. In *Bulletin de lassociation française dintelligence artificielle*, 2001. 8

[Lat13] C. Latiri. *Extraction de Connaissances à partir de Textes : Méthodes et Applications*. PhD thesis, Université de LORRAINE, 2013. 16, 17

[LBW10] M. Lahiri and T. Berger-Wolf. Periodic subgraph mining in dynamic networks. *KAIS*, 24(3) :467–497, 2010. 75

[LD12] R. Lucia and V. Dusan. L'évaluation d'un problème géométrique de l'application de l'analyse statistique implicative. *Conférence Internationale, ASI*, 2012. 3, 20, 76

[Ler81] I-C. Lerman. *Classification et analyse ordinale des donnés*. Dunod, 1981. 4, 5, 90, 94

[Lin56] O. R. Lindsley. Operant conditioning methods applied to research in chronic schizophrenia. *Psychiatric Research Reports*, 5 :118–139, 1956. 88

[Lin90] O. R. Lindsley. Precision teaching : By teachers for children. *Teaching Exceptional Children, Spring*, 5 :10–15, 1990. 88

[Lin91] O. R. Lindsley. Precision teaching's unique legacy from b. f. skinner. *Journal of Behavioral Education*, 1 :253–266, 1991. 88

[Lip87] R. P. Lippmann. An introduction to computing with neural networks. 4(2) :4–22, April 1987. 10

[LLP08] D. H. Li, A. Laurent, and P. Poncelet. Découverte de motifs séquentiels et de règles inattendus. *RNTI*, 2008. 47

[LLW+06] J. Li, H. Li, L. Wong, J. Pei, and G. Dong. Minimum description length principle : Generators are preferable to closed patterns. *AAAI Press*, pages 409–414, 2006. 2, 47

[LLW08] G. Liu, J. Li, and L. Wong. A new concise representation of frequent itemsets using generators and a positive border. *Knowl. Inf. Syst*, 17 :35–56, 2008. 2, 47

[LP09] I. C. Lerman and K. Pascal. Directed binary hierarchies and directed ultrametrics. 2009. 3, 20, 76

[LPBT99] L. Lakhal, N. Pasquier, Y. Bastide, and R. Taouil. Efficient mining of associationrules using closed itemset lattices. *Information Systems*, 24 :25–46, 1999. 19

[LSL95] H. Lu, R. Setino, and H. Liu. A connectionist approach to data mining. pages 478–489, September 1995. 10

[LT04] S. Lallich and O. Teytaud. Evaluation et validation de mesures d'intérêt des règles d'association. *RNTI-E-1*, spécial :193–217, 2004. 11, 60, 61, 62

[MAQ03] D. MAQUIN. *Eléments de Théorie des graphes*. 2003. 75

[MCRE09] F. Moser, R. Colak, A. Rafiey, and M. Ester. Mining cohesive patterns from graphs with feature vectors. *SIAM Data Mining Conf.(SDM)*, pages 593–604, 2009. 75

[MDT14] Z. Makhlouf, Y. Dupont, and I. Tellier. Caractériser l'acquisition d'une langue avec des patrons d'étiquettes morpho-syntaxiques. *JADT 2014 : 12è Journées internationales d'Analyse statistique des Données Textuelles*, pages 447–458, 2014. 47

[MM70] B. Marc and Bernard M. Ordre et classification, algèbre et combinatoire. *Hachette*, 1970. 36

[MM03] T. Mielikäinen and H. Mannila. The pattern ordering problem. In *Proceedings of the 7th European Conference on Principles of Data Mining and Knowledge Discovery*, pages 327–338, Springer-Verlag, 2003. 2, 46

[MO09] H. Motoda and K. Ohara. *The Top Ten Algorithms in Data Mining*. SERIES EDITOR, Xindong Wu and Vipin Kumar, 2009. 2

[Mon03] B. Monjardet. The presence of lattice theory in discrete problems of mathematical social sciences. *Mathematical Social Sciences*, 46 :103–144, 2003. 12

[MOO09] Y. Miyoshi, T. Ozaki, and T. Ohkawa. Frequent pattern discovery from a single graph with quantitative itemsets. *IEEE Int. Conf. on Data Mining (ICDM) Workshops*, pages 527–532, 2009. 75

[Mor94] Robert Morris. *Etude sur l'enseignement des mathématiques, L'enseignement de la statistique*. Editions UNESCO, 1994. 96

[MPR+10] P-N. Mougel, M. Plantevit, C. Rigotti, O. Gandrillon, and J.-F. Boulicaut. Constraint-based mining of sets of cliques sharing vertex properties. *Workshop on Analysis of Complex NEtworks (ACNE'10) PKDD*, 2010. 75

[MPTM07] F. Masseglia, P. Poncelet, M. Teisseire, and A. Marescu. Web usage mining : extracting unexpected periods from web logs. *Data Mining and Knowledge Discovery, Springer Verlag (Germany)*, pages 039–065, 2007. 1

[MR10] S. Matthias and G. Ritschard. Une analyse statistique implicative des résultats d'une fouille de texte. 2010. 3, 20, 76

[MRG12] P-N. Mougel, C. Rigotti, and O. Gandrillon. Finding collections of k-clique percolated components in attributed graphs. *PAKDD*, pages 181–192, 2012. 75

[MS83] R. S. Michalski and R. E. Stepp. Learning from observation : Conceptuel clustering. 1 :331–363, 1983. 10

[MST94] D. Michie, D. J. Spiegelhalter, and C. C. Taylord. Machine learning, neural, and statistical classification. 1994. 10

[MT96] H. Mannila and H. Toivonen. Multiple uses of frequent sets and condensed representations (extended abstract). *KDD'96*, pages 189–194, 1996. 46

[MYGS91] M. McLeach, P. Yao, M. Garg, and T. Stirzinger. Discovery of medical diagnostic information : An overview of method and results. pages 477–499, 1991. 19

[NCCC12] B. Nicolas, P. Cellier, T. Charnois, and B. Cremilleux. Fouille de motifs séquentiels pour la découverte de relations entre gènes et maladies rares. *IC 2012*, 2012. 47

[NDG+10] M. Ndiaye, C. T. Diop, A. Giacometti, P. Marcel, and A. Soulet. Construction et exploration de résumé de grands ensembles de règles d'association. 2010. 2

[NR07] M. Nanni and C. Rigotti. Extracting trees of quantitative serial episodes. In *Proc. Of KDID'07*, pages 170–188, 2007. 47

[OEC06] Carlos O., N. Ezquerra, and A. Cesar. Constraining and summarizing association rules in medical data. *KNOWL. Inf. Syst.*, 9(3) :259–283, 2006. 2

[OO98] C. Ordonez and E. Omiecinski. Image mining : A new approach for data mining. 1998. 19

[OR03] J.C. Oriol and J.C Régnier. Fonctionnement didactique de la simulation en statistique. exemple de l'enseignement du concept d'intervalle de confiance. In *Actes des XXXVèmes Journées de Statistique*, pages 743–750, Lyon, France, 2003. 112, 113

[OZ01] M.G. Ottaviani and S. Zannoni. Implication statistique et recherche en didactique. utilisation d'un outil non symétrique d'analyse de données pour l'introduction des résultats d'un test d'évaluation. *Math. et Sci. hum.*, 39e année, n° 154-155 :61–79, 2001. 113

[Pas00] N. Pasquier. *Algorithme d'Extraction et de Réduction des Règles d'Association dans la base de données*. PhD thesis, Université de Clermont-Ferrand II, France, 2000. 41

[PBG11] T. Piton, J. Blanchard, and F. Guillet. Une méthodologie de recommandations produits fondée sur l'actionnabilité et l'intérêt Économique des clients. In *Actes des onzièmes journées Extraction et Gestion des Connaissances EGC'2011*, volume E-20, pages 203–214, Brest France, 2011. 19

[PBTL98] N. Pasquier, Y. Bastide, R. Taouil, and L. Lakhal. Pruning closed itemset lattices for association rules. *BDA'98*, pages 177–196, 1998. 26, 37, 46

[PBTL99a] N. Pasquier, Y. Bastide, R. Taouil, and L. Lakhal. Discovering frequent closed itemsets for association rules. In *Proc. ICDT conf., LNCS 1540*, pages 398–416, 1999. 26, 37, 38, 46

[PBTL99b] N. Pasquier, Y. Bastide, R. Taouil, and L. Lakhal. Efficient mining of association rules using closed itemset lattices. *Information Systems*, 24 :25–46, 1999. 37

[PBTL99c] N. Pasquier, Y. Bastide, R. Taouil, and L. Lakhal. Efficient mining rules using closed itemsets lattices. *Inforation Systems*, 24 :25–46, 1999. 16, 26, 37, 46

[PCW12] X. Peng, P. Cheng, and M. Wang. A study of negative association rules mining algorithm based on multi-database. *Proceedings of the 2012 2nd International Conference on Computer and Information Application (ICCIA 2012)*, 2012. 60

[Pea00] K. Pearson. On the Criterion that a given System of Deviations from the Probable in the Case of a Correlated System of Variables. 1900. 59

[PG09] A. K. Poernomo and V. Gopalkrishnan. Towards efficient mining of proportional fault-tolerant frequent itemsets. *Proc. KDD*, pages 697–706, 2009. 47

[PJFD13] A. Prado, B. Jeudy, É. Fromont, and F. Diot. Mining spatiotemporal patterns in dynamic plane graphs. *IDA Journal*, 17(1) :71–92, 2013. 75

[PPRB13] A. Prado, M. Plantevit, C. Robardet, and J-F. Boulicaut. Mining graph topological patterns : Finding covariations among vertex descriptors. *IEEE TKDE*, 25(9) :2090–2104, 2013. 75

[PS91] G. Piatetsky-Shapiro. Knowledge discovery in real data bases : A report on the ijcai-89 workshop. *AI Magazine*, 11(5) :68–70, 1991. 8

[PSF91] G. Piatetsky-Shapiro and W. Frawley. Knowledge Discovery in Databases. *AAAI Press*, 1991. 8

[PTM07] P. Poncelet, M. Teisseire, and F. Massaglia. *Data Mining Paterns : New Methods and Applications*. Information Science Reference, 2007. 1

[QCC14] S. Quiniou, P. Cellier, and T. Charnois. Fouille de données pour associer des noms de sessions aux articles scientifiques. *12è Traitement Automatique des Langues Naturelles*, 2014. 47

[Ram13] M. L. M. Ramanandraisoa. *Articulation transdisciplinaire des connaissances de mathématiques et sciences physiques. Le cas de la proportionnalité en fin d'Ecole primaire et début du Collège à Madagascar. Approches didactiques, interactionnistes et ethnomathématiques*. PhD thesis, Université de Lyon 2, France, 2013. 116

[Rao13] J.-P. Raoult. La statistique dans l'enseignement secondaire en france. *Statistique et Enseignement, Société Française de Statistique (SFdS)*, 4(1) :55–69, 2013. 112

[RBATR12] H. Ramanantsoa, P. Bemarisika, A. A. Totohasina, and L. Ramifidisoa. Enseignement de limite d'une fonction et TIC au niveau secondaire. In *Colloque international sur les TIC*, ENS Ampefiloha, Antananarivo, 2012. 4

[Rég02] J-C Régnier. A propos de la formation en statistique. approches praxéologiques et épistémologiques de questions du champ de la didactique de la statistique. *Revue du centre de recherche en éducation, Didactique des mathématiques*, pages 157–201, 2002. 112

[Rég09] J-C. Régnier. Conceptualisation de l'analyse statistique implicative. Université de Lyon 2, 2009. 3, 20, 76

[Rég12] J-C. Régnier. Enseignement et apprentissage de la statistique : entre un art pédagogique et une didactique scientifique. In *Enseignement et apprentissage de la statistique*, volume 3(1), pages 19–36, Université de Lyon 2, 2012. 112

[RGR11] B. Ramasubbareddy, A. Govardhan, and A. Ramamohanreddy. Classification based on positive and negative association rules. *International Journal of Data Engineering, (IJDE)*, 2 : Issue 2 :84–92, 2011. 60

[RMM09] G. Ritschard, S. Matthias, and O. Michel. Applications à la sociologie. 2009. 3, 20, 76

[RT14] H. Ramanantsoa and A. Totohasina. Une stratégie d'intégration pédagogique des TIC dans l'enseignement des mathématiques à madagascar. *Frantice.net*, n° 9, Nov. 2014 :74–85, 2014. 113

[RT15] H. Ramanantsoa and A. Totohasina. Notes sur les bases des règles valides au sens de la mesure m_{GK}. In *Actes de la 22è Rencontre de la Société Francophone de Classification (SFC'15)*, 2015. 63

[Ruß09] Georg Ruß. *Data Mining of Agricultural Yield Data : A Comparison of Regression Models*. P. Perner (Ed.) : Industrial Conference on Data Mining, ICDM 2009, LNAI 5633, 2009. 88

[Rui14] P. A. Ruiz. *Génération de connaissances à l'aide du retour d'expérience : Application à la maintenance industrielle*. PhD thesis, Institut National Polytechnique de Toulouse, France, 2014. 1, 20, 60, 88

[SA96] R. Srikant and R. Agrawal. Mining sequential patterns : Generalizations and performance improvements. In *Proceedings of 5th Biennial International Conference on Extending Database Technology (EDBT'96)*, pages 3–17, Avignon, 1996. 47

[SC09] M. Szczerba and A. Ciemski. *Credit Risk Handling in Telecommunication Sector*. P. Perner (Ed.) : Industrial Conference on Data Mining, ICDM 2009, LNAI 5633, 2009. 19, 88

[Sch08] K. Schuessler. *Didactique de précision en version informatisée : sa description, son utilisation et sa convivialité*. PhD thesis, Université de Québec, Montréal, 2008. 88

[Seg04] A. Segall. Revisiting pedagogical content knowledge : the pedagogy of content/the content of pedagogy. *Teaching and Teacher Education*, 20 :489–504, 2004. 89

[Ser14] L. Serrano. *Vers une capitalisation des connaissances orientée utilisateur, Extraction et structuration automatiques de l'information issue de sources ouvertes*. PhD thesis, Université de Caen Basse-Normandie, 2014. 1, 20, 88

[SFP+13] J. Sanhes, F. Flouvat, C. Pasquier, N. Selmaoui, and J-F. Boulicaut. Extraction de motifs condensés dans un unique graphe orienté acyclique attribué. *IJCAI*, 2013. 75

[SHFM09] K. Shinozawa, N. Hagita, M. Furutani, and R. Matsuoka. *A Data Mining Method for Finding Hidden Relationship in Blood and Urine Examination Items for Health Check*. P. Perner (Ed.) : Industrial Conference on Data Mining, ICDM 2009, LNAI 5633, 2009. 19

[Shu86] L. S. Shulman. Those who understand : Knowledge growth in teaching. *Educational researcher*, 15(2) :4–14, 1986. 88

[Shu87] L. S. Shulman. Knowledge and teaching : foundations of the new reform. *Harvard Educational review*, 57 :1–22, 1987. 88

[Shu07] L. S. Shulman. Ceux qui comprennent : Le développement de la connaissance dans l'enseignement. *Education et Didactique*, 1(1) :97–114, 2007. 89

[SJZ12] A. Silva, W. M. Jr, and M. J. Zaki. Mining attribute-structure correlated patterns in large attributed graphs. *PVLDB*, 5(5) :466–477, 2012. 75

[SLS15] B. R. Sourour, C. Latiri, and Y. Slimani. Vers des méta-règles de contexte appréciées par la IIE pour la RI. *CORIA 2015*, pages 1–15, 2015. 60

[SM04] J. K. Seppänen and H. Mannila. Dense itemsets. *Proc. KDD*, pages 683–688, 2004. 47

[SMZ10] A. Silva, J. W. Meira, and M. J. Zaki. Structural correlation pattern mining for large graphs. *8th Workshop on Mining and Learning with Graphs*, 2010. 75

[SN10] T. Sébastien and EM Normandie. Le web 2.0 comme nouveau paradigme de l'entreprise ? *JEL : L22, M15*, 2010. 19

[SON95] A. Savasere, E. Omiecinski, and S. Navathe. An efficient algorithm for mining association rules in large databases. In *Proc. of the 21th VLDB Conference*, pages 432–444, Zurich,Switzerland, September 1995. 2, 26, 27, 46

[SON98] A. Savasere, E. Omiecinki, and S. Navathe. Mining for strong negatives associations in a large database of customer transactions. In *Proceedings of the 14th ICDE'98*, pages 494–502, 1998. 59

[SPJ+05] B. Serge, V. Philippé, P. Jacques, R. Gras, and O. Guilhaume. L'analyse implicative pour l'élaboration de référentiels comportementaux. *Troisièmes Rencontres Internationales, ASI*, 2005. 3, 20, 76

[SR14] A. Soulet and F. Rioult. Extraire les motifs minimaux efficacement et en profondeur. *Revue des Nouvelles Technologies de l'Information (RNTI)*, pages 383–394, 2014. 2, 47

[SSP+06] L. Szathmary, M. Sandy, P. Petronin, T. Yannick, and N. Amedeo. Vers l'extraction de motifs rares. *RNTI-E-6*, pages 499–510, 2006. 61

[STB+02] G. Stumme, R. Taouil, Y. Bastide, N. Pasquier, and L. Lakhal. Computing iceberg concept lattices with TITANIC. *Data and Knowledge Engineering*, 42 :189–222, 2002. 13

[SVNG09] L. Szathmary, P. Valtchev, A. Napoli, and R. Godin. Efficient vertical mining of frequent closures and generators. *IDA*, 5772 :393–404, 2009. 2, 47

[Sza06] L. Szathmary. *Méthodes symboliques de fouille de données avec la plate-forme Coron*. PhD thesis, Université de Henri Poincaré, Nancy 1, 2006. 50

[TF08] A. Totohasina and D. R. Feno. De la qualité des règles d'association : Etude comparative des mesures M_{GK} et confiance. *CARI'2008*, pages 561–568, 2008. 43, 63

[THC02] W. G. Teng, M. J. Hisieh, and M. S. Chen. On the mining of subtitution rules for statiscally dependent items. *Second IEEE International Conference on Data Mining (ICDM'02)*, pages 442–449, 2002. 59

[Tot03] A. Totohasina. Normalisation de mesures probabilistes de la qualité des règles. In *Proc. SFDS'03, XXXVème Journée de Statistiques*, volume 2, pages 985–988, Lyon 2, France, 2003. 62, 76

[Tot08] A. Totohasina. *Contribution à l'étude des mesures de qualité des règles d'association : Normalisation sous cinq contraintes et cas de M_{GK} ; Propriété, base composite des règles d'association, et extension en vue d'applications en statistique et en sciences physiques*. PhD thesis, Université d'Antsiranana, Madagascar, 2008. HDR. 43, 60, 63, 65

[Tou11] Y. Toussaint. *Fouille de textes : des méthodes symboliques pour la construction d'ontologies et l'annotation sémantique guidée par les connaissances*. PhD thesis, 2011. HDR. 1, 8, 60

[UPC09] A. U. Urtubia and J. R. Pérez-Correa. *Study of Principal Components on Classification of Problematic Wine Fermentations*. P. Perner (Ed.) : Industrial Conference on Data Mining, ICDM 2009, LNAI 5633, 2009. 88

[VDVD98] J. H. Van Driel, N. Verloop, and W. Devos. Developing science teachers' pedagogical content knowledge. *Journal of research in science Teaching*, 35(6) :673–695, 1998. 88

[VMHG03] P. Valtchev, R. Missaoui, M. R. Hacene, and R. Godin. Incremental maintenance of association rule bases. In *Proceedings of the 2nd Workshop on Discrete Mathematics and Data Mining*, San Francisco, 2003. 19

[VTL10] J. Villerd, Y. Toussaint, and A. Lillo-Le Louët. *Adverse Drug Reaction Mining in Pharmacovigilance data using Formal Concept Analysis*. J.L. Balcézar et al. (Eds.) : ECML PKDD 2010, Part III, LNAI 6323, 2010. 19, 88

[WKQ+07] X. Wu, V. Kumar, J.R. Quinlan, J. Ghosh, Q. Yang, H. Motoda, G.J. McLachlan, N. Angus, B. Liu, P.S. Yu, Z.-H. Zhou, M. Steinbach, D.J. Hand, and D. Steinberg. Top 10 algorithms in data mining. *Springer-Verlag London Limited*, pages 1–37, 2007. 2

[WLH06] Y. Wang, E. P Lim, and S. Y. Hwang. Efficient mining of group patterns from user movement data. *Data Knowl*, 57 :240–282, 2006. 46

[WWR+09] W. Wang, Y. J. Wang, B.-A. René, Z. Cui, and F. Coenen. *Application of Classification Association Rule Mining for Mammalian Mesenchymal Stem Cell Differentiation*. P. Perner (Ed.) : Industrial Conference on Data Mining, ICDM 2009, LNAI 5633, 2009. 20, 88

[WZZ04] X. Wu, C. Zhang, and S. Zhang. Efficient mining of both positive and negative association rules. *ACM Transactions on Information Systems*, 22 :381–405, 2004. 59, 62, 69, 76

[YCHX05] X. Yan, H. Cheng, J. Han, and D. Xin. Summarizing itemset patterns : A profilebased approach. *KDD'05*, pages 314–323, 2005. 2, 46

[Zak01] M. J. Zaki. Spade : An efficient algorithm for mining frequent sequences. pages 31–60, 2001. 47

[Zak04] M. J. Zaki. Mining non-redundant association rules. *Knowledge Discovery and Data Mining*, 9 :223–248, 2004. 18, 19

[ZH02] M. J. Zaki and C-J. Hsiao. CHARM : An Efficient Algorithm for Closed Itemset Mining. *SIAM International Conference on Data Mining SDM'02*, pages 33–43, 2002. 46, 50

[ZPOL97] M. J. Zaki, S. Parthasarathy, M. Ogihara, and W. Li. New algorithms for discovery association rules. In *Knowledge Discovery and Data Mining*, pages 283–296, 1997. 2, 26, 30, 34, 46