Contribution à l'extraction des règles d'association basée sur des préférences

Slim Bouker

UNIVERSITE DE CLERMONT-FERRAND II

LIMOS - CNRS UMR 6158

Laboratoire d'Informatique, de Modélisation et d'Optimisation des Systèmes

THÈSE

Soutenue et présentée le 30 Juin 2015 pour l'obtention du

Diplôme de Docteur en Informatique

par

Slim BOUKER

Contribution à l'extraction des règles d'association basée sur des préférences

Composition du jury

Fred KORICHE	Professeur, Université d'Artois	Rapporteur
Jérôme AZÉ	Professeur, Université de Montpellier	Rapporteur
Philippe LENCA	Professeur, Télécom Bretagne	Rapporteur
Souhila KACI	Professeur, Université Montpellier	Présidente
Sylvie GUILLAUME	Maître de conférences, Université d'Auvergne	Examinatrice
Engelbert MEPHU NGUIFO	Professeur, Université de Clermont-Ferrand II	Directeur de thèse
Sadok BEN YAHIA	Professeur, Université El Manar	Co-directeur de thèse

Remerciements

Mes éternels remerciements et louanges á :

Dieu, le miséricordieux, de m'avoir orienté et aidé à réaliser cette thèse. Je le remercie de m'avoir entourer des personnes qui ont chacune à sa façon et à différentes étapes de mon acheminement, contribué à la réalisation de cette thèse de doctorat.

Ma gratitude et mes profonds remerciements à :

Pr. Engelbert Mephu Nguifo, Professeur à l'Université de Clermont-Ferrand II, pour la confiance qu'il a manifesté à mon égard en acceptant de m'encadrer. Je le remercie pour sa patience, ses directives et ses encouragements qui m'ont permis de réaliser cette thèse. Je lui suis infiniment reconnaissant de m'avoir aidé à surmonter les moments les plus difficiles que j'ai dûs affronter.

Pr. Sadok Ben Yahia, Professeur à l'Université El Manar, pour son soutien, sa grande disponibilité et la richesse de ses réflexions sur ma thèse. Je lui suis très reconnaissant de m'avoir aidé à développer mes aptitudes pour la recherche et l'enseignement, dès l'année de mon master.

Tous les mots ne suffisent pas pour exprimer ma reconnaissance envers vos efforts mes chers professeurs. Outre l'encadrement scientifique, vous m'avez transmis des valeurs que rien à y penser je suis déjà très fortuné : la générosité, l'aide inconditionnelle,...

Mes respectueux remerciements à :

Pr. Souhila Kaci, Professeur à l'Université de Montpellier, d'avoir accepté de présider le jury de cette thèse.

Pr. Philippe Lenca, Professeur à Telecom Bretagne, Pr. Jérôme Azé, Professeur à l'Université de Montpellier et Pr. Fred Koriche, Professeur à l'Université d'Artois, d'avoir

accepté de rapporter mon travail. Je les remercie pour le temps qu'ils ont consacré et pour leurs remarques constructives qui m'ont permis d'améliorer le manuscrit.

Pr. Alain Quilliot, Professeur à l'Université de Clermont-Ferrand II, pour avoir accepté d'être examinateur de la thèse. Je le remercie également pour m'avoir accueilli au sein du laboratoire LIMOS et permis de travailler dans de très bonnes conditions.

Mc. Sylvie Guillaume, Maître de conférences à l'Université d'Auvergne, pour avoir accepté d'être examinatrice de la thèse.

Cette thèse est aussi le fruit de collaborations scientifiques avec des équipes de recherche tunisiennes dans le cadre d'échanges scientifiques Tuniso-français et de projets CMCU.

Ma reconnaissance et mon affection à :

Ma mère Jouda, mon père Romdhane Bouker et mes deux frères, dont le soutien ne m'a jamais fait défaut. Je les remercie pour leur amour infini qu'ils m'ont toujours accordé.

Ma femme Mariem, pour toute sa compréhension, sa patience et son amour durant les années de la thèse : nous nous sommes serrés les coudes quand la pente était trop raide.

J'ai longtemps chercher les mots qui seraient les plus justes pour vous remercier d'être toujours présents et de m'épauler quoi qu'il arrive, mais après tant de réflexions je ne vois rien d'aussi fort que : je vous aime.

Mes remerciements à :

Ma belle famille pour sa compréhension et son soutient inconditionnel.

Une pensée amicale à :

Ghada Gasmi, Sami Zghal, Mohamed Ali ben Hassine, Ines Bouzouita, Wajdi Dhifli, Marie Favre, Rabie Saidi, Sabeur Aridhi, Tarek Hamrouni et Djelloul Mameri pour tous les bon moments que nous avons vécus.

Les membres du Laboratoire LIMOS qui m'ont toujours chaleureusement accueilli pendant ces années de thèse.

Table des matières

1 Introduction **1**

1.1 Contexte général 1

1.2 Motivations et contributions 3

1.3 Organisation du manuscrit 7

2 Règles d'association **9**

2.1 Introduction 9

2.2 Problème d'extraction de règles d'association 10

2.2.1 Formalisme de base 10

2.2.2 Décomposition du problème d'extraction des règles d'association . . 13

2.3 Extraction des règles d'association à partir des motifs fréquents 13

2.3.1 Algorithme Apriori 14

2.3.2 Génération des règles d'association 17

2.3.3 Autres algorithmes 18

2.4 Réduction des règles d'association redondantes 19

2.4.1 Analyse formelle des concepts 20

2.4.2 Bases génériques pour les règles d'association 24

2.4.3 Synthèse des différentes bases génériques 36

2.5 Extraction de règles d'association sous contraintes 37

2.6 Conclusion 39

3 Fouille de données et Préférences **41**

3.1 Introduction 41

3.2 Approches d'agrégation des préférences 42

3.2.1 Systèmes de vote 43

3.2.2 Dominance de Pareto 51

3.2.3 Synthèse des différentes approches d'agrégation de préférences . . . 52

3.3 Recherche des motifs basée sur les préférences 54
3.3.1 Motifs les plus informatifs 54
3.3.2 Motifs et dominance de Pareto 58
3.3.3 Graphes et dominance de Pareto 62
3.3.4 Synthèse des approches de recherche des motifs basées sur les préférences 64
3.3.5 Agrégation de mesures d'intérêt de règles d'association 65
3.4 Conclusion 66

4 Sélection des règles d'association basée sur la relation de dominance 68
4.1 Introduction 69
4.2 Travaux sur les mesures de qualité 70
4.2.1 Caractérisation d'une bonne mesure de qualité 70
4.2.2 Classification des mesures de qualité 72
4.2.3 Limites et motivations 73
4.3 Sélection des règles non dominées 75
4.3.1 Règles non dominées 75
4.3.2 Formalisation pour la sélection des règles non dominées 77
4.3.3 Algorithme SkyRule 80
4.3.4 Expérimentations 82
4.4 Sélection des k meilleures règles d'association 87
4.4.1 Ordonnancement des règles selon plusieurs mesures 88
4.4.2 Algorithme RankRule 90
4.4.3 Dualité 91
4.4.4 Expérimentations 92
4.5 Conclusion 98

5 Le modèle des règles d'association représentatives 101
5.1 Introduction 101
5.2 Famille des règles d'association représentatives 103
5.2.1 Motivation 103
5.2.2 Règles d'association représentatives 104
5.3 Sélection des règles d'association représentatives 106
5.3.1 Cas de comparabilité transitive 107
5.3.2 Cas de comparabilité non-transitive 109

5.3.3 Compacité des règles représentatives 119
5.4 Conclusion 121

6 Conclusion et perspectives **123**
6.1 Conclusion 123
6.2 Perspectives 125

Table des figures

1.1 Environnement coopératif 5
1.2 Exemple d'un expert 5

2.1 Treillis d'inclusion 14
2.2 Treillis de l'iceberg de Galois 24
2.3 Le couple ($\mathcal{BG}$, $\mathcal{RI}$) 34

3.1 (a) Le graphe pondéré associé à l'exemple d'évaluation du tableau 3.1, (b) Le graphe pondéré sans cycle associé à l'exemple d'évaluation du tableau 3.1. 49
3.2 Exemple de sous-graphes. 63

4.1 Résultat de l'application de RankRule sur la table relationnelle Ω donnée par le Tableau 4.1(b). 91
4.2 Approche duale appliquée sur Ω donnée dans le tableau 4.1(b). 92
4.3 Temps d'exécution de RankRule suite à la variation du nombre de règles. 94
4.4 Temps d'exécution de RankRule suite à la variation de k 95

5.1 Diagramme de dominance associé à la table relationnelle Ω illustrée par la table 5.1(b). 111
5.2 Règles représentatives dans le diagramme de dominance associé à Ω illustré par la table 5.1(b). 114

Liste des tableaux

1.1 Exemple d'évaluation de quatre règles 4

2.1 Exemple de relation binaire. 11
2.2 Notations utilisées dans l'algorithme Apriori 15
2.3 Notations utilisées dans l'algorithme Gen-Rules 17
2.5 Base $\mathcal{BGD}$ des règles d'association 27
2.6 Base $\mathcal{BGD}$ des règles d'association 28
2.7 Base $\mathcal{BR}$ des règles d'association 30
2.8 Base générique $\mathcal{BNR}$ 31
2.9 Base $\mathcal{IGB}$ des règles d'association 35
2.10 Comparaison des principales bases génériques dans la littérature 36

3.1 Exemple d'évaluation de quatre alternatives 43
3.2 Évaluation des deux alternatives a_1 et a_2 45
3.3 Évaluation des trois alternatives a_1, a_2 et a_3 46
3.4 Évaluation des deux alternatives a_1 et a_3 47
3.5 Duels majoritaires entre les alternatives 48
3.6 Caractéristiques des principales approches d'agrégation de préférences. . . 53
3.7 Exemple de contexte d'extraction. 54
3.8 Valeurs de *support* et d'*area* des motifs 57
3.9 Skypatterns extraits à partir du contexte d'extraction du tableau 3.7 . . . 59
3.10 Caractéristiques des principales approches de sélection des motifs basées sur les préférences 65

4.1 Exemple de contexte d'extraction. 76
4.2 Caractéristiques des données de test. 83
4.3 Définitions des mesures de qualité 84

4.4 Variation des Règles non dominées *Sky-R vs* règles optimales sous contraintes ($ROSC$) et toutes les règles AR 84
4.5 Nombre moyen des règles non dominées, Nombre moyen des ROSC et Gain moyen des règles non dominées 85
4.6 Temps d'extraction des règles non dominées. 86
4.7 Valeurs de degré de ressemblance entre les mesures $Confiance$, $Loevinger$, $Zhang$ et les classes de mesures {Conf, Loev}, {Conf, Zhang}, {Conf, Loev, Zhang} 97

5.1 Exemples d'un contexte d'extraction $\mathcal{T}$ et d'une table relationnelle $\Omega = (\mathcal{AR}, \mathcal{M})$ 102
5.2 Table relationnelle $\Omega' \subset \Omega$ 109
5.3 Temps d'extraction des règles représentatives. 119
5.4 Règles représentatives *vs* règles non dominées, $\mathcal{ROSC}$ et $\mathcal{AR}$ 120

Liste des algorithmes

1 Algorithme Apriori 16
2 Algorithme Apriori-Gen 16
3 Algorithme Gen-Rules 18
4 Algorithme Ap-Genrules 19
5 SkyRule 81
6 RankRule 90
7 TransRule 108
8 RprRule_1 112
9 RprRule_2 117

Chapitre 1

Introduction

Sommaire

1.1 Contexte général 1

1.2 Motivations et contributions 3

1.3 Organisation du manuscrit 7

1.1 Contexte général

La fouille de données est une discipline à la frontière de plusieurs autres disciplines comme les bases de données, les statistiques, l'analyse de données, l'intelligence artificielle, la visualisation de données, etc. Elle vise à découvrir des connaissances préalablement inconnues et potentiellement utiles, pour les experts, à partir de grandes bases de données [Fay96].

Aujourd'hui avec les outils automatiques de collecte de données et les coûts de plus en plus réduits de stockage et de traitement, nous assistons à une croissance exponentielle du volume de données collectées. Ces bases de données, qui sont à la fois très larges et hétérogènes, peuvent cacher de précieuses informations utiles pour l'aide à la décision, l'optimisation de requêtes, etc. Devant le besoin scientifique et économique d'extraire les connaissances dispersées dans ces masses de données, la fouille de données se propose de fournir un certain nombre d'outils et de techniques pour l'extraction de connaissances nouvelles, potentiellement utiles et compréhensibles à partir de données préalablement sélectionnées et préparées.

Les techniques de fouille de données peuvent être classées en deux catégories [HK00]. La première regroupe les techniques de vérification pour lesquelles l'expert fournit des hypo-

thèses que les systèmes doivent valider. La seconde regroupe les techniques de recherche qui permettent d'extraire automatiquement de nouvelles informations pertinentes. Parmi les techniques de recherche, nous trouvons l'extraction des associations, le regroupement par similarité, la classification, la prédiction, etc. Dans le cadre de notre travail, nous nous intéressons à l'extraction des connaissances et plus particulièrement à l'extraction des règles d'association.

L'extraction des règles d'association est un problème fondamental de la fouille de données et a été largement étudiée depuis son introduction par *Agrawal et al.* [AIS93b]. A l'origine, l'étude des règles d'association a été motivée par le fait qu'elles peuvent être utilisées pour exprimer des corrélations entre les différents attributs d'une base de données. Étant donnée une table dans une base de données relationnelle, les règles d'association sont des implications conditionnelles de la forme *prémisse* $\rightarrow$ *conclusion* où la prémisse et la conclusion sont des expressions qui portent sur les attributs de la table. Une règle signifie que les enregistrements qui contiennent les attributs de la prémisse ont tendance à contenir les attributs de la conclusion. L'une des premières applications de l'extraction des règles d'association a été l'étude du panier de la ménagère où chaque enregistrement comprend une liste d'articles (ou produits) achetés par un client dans un supermarché. L'objectif de cette étude consiste à découvrir des corrélations cachées, potentiellement utiles, entre les articles (par exemple, "65% des clients qui achètent du fromage achètent aussi du pain").

Les règles d'association ont été ensuite utilisées dans de nombreux domaines tels que l'économie, la recherche médicale, l'analyse des données spatiales, la prédiction des événements séquentiels [RLM13], etc. Dans le cas général, étant donné un ensemble d'items (ou attributs) et un ensemble d'objets (ou enregistrements), l'extraction des règles d'association consiste à déterminer les règles dont le support et la confiance sont au moins égaux, respectivement, à un seuil minimal de support *minsup* et un seuil minimal de confiance *minconf*, prédéfinis par l'expert. Le *support* désigne le nombre d'objets qui vérifient la prémisse et la conclusion, tandis que la *confiance* est la proportion d'enregistrements qui vérifient la conclusion parmi ceux qui vérifient la prémisse. Toutefois, cette technique produit en pratique un nombre prohibitif de règles qui sont très difficiles à gérer par l'expert. Ce problème s'amplifie au fur et à mesure qu'on diminue le seuil du support, ce qui est essentiel pour découvrir les spécificités des données. Ainsi, deux problèmes se posent : le premier est de nature quantitative puisque le nombre de règles est très élevé. Le second est de nature qualitative puisque beaucoup de règles sont redondantes et sans valeur ajoutée.

1.2 Motivations et contributions

Différentes approches, ont été proposées pour traiter ces problèmes et aider l'expert à explorer les règles intéressantes. Une première approche consiste à introduire des contraintes permettant à l'utilisateur de spécifier les caractéristiques des règles d'association recherchées [FC07, JB02, SVA97, JAG99]. Une seconde approche consiste à extraire un sous-ensemble réduit de règles, appelé *bases génériques* [Zak00, Kry98, BPT+00, YGN09], en se basant sur l'opérateur de fermeture de la correspondance de Galois utilisé dans l'Analyse Formelle de Concepts [GW99]. L'objectif des bases génériques est de diminuer le nombre de règles générées tout en assurant que les règles peuvent être retrouvées ainsi que leurs supports et leurs confiances. Une autre approche intéressante consiste à utiliser d'autres mesures de qualité pour compléter le support et la confiance qui, utilisés seuls, ne permettent d'évaluer que certains aspects de la qualité des règles [GH06]. Il existe deux catégories de mesures : les mesures subjectives qui nécessitent les connaissances de l'utilisateur *a priori* sur les données [ST95, LHCM00] et les mesures objectives qui ne dépendent que des propriétés intrinsèques aux règles. Dans cette thèse, nous nous focalisons sur les mesures objectives de la qualité de règles.

L'idée principale d'une mesure est d'associer une valeur numérique à une règle permettant de quantifier son intérêt. Elle permet ainsi d'évaluer les règles, d'éliminer celles qui sont non pertinentes (en utilisant un seuil minimal) et d'ordonner celles qui sont retenues par ordre de pertinence. Dans les deux dernières décennies, une panoplie de mesures de qualité obéissant à différentes sémantiques ont été proposées dans la littérature (environ une soixantaine). Cependant, très hétérogènes, les mesures de qualité peuvent évaluer différemment les règles. Face à l'abondance et l'hétérogénéité des mesures de qualité, de nombreux travaux [TKS02, VLL04, HGB06, GGN11, BGG+11] se sont intéressés à analyser le comportement des mesures de qualité afin de les regrouper d'une manière homogène et de faire ressortir leurs ressemblances et leurs divergences. D'autres auteurs [JA99, LMPV03, BGGB05, LMVL08] se sont intéressés à définir des propriétés sur les mesures dans le but de caractériser "une bonne mesure" pour l'utilisateur. En effet, après que l'utilisateur ait exprimé ses préférences sur ces propriétés, une ou plusieurs mesures lui sont recommandées.

Néanmoins, le problème est loin d'être résolu pour les raisons suivantes :

- La pertinence d'une mesure de qualité depend de l'idée que l'expert se fait d'une règle intéressante pour son application et sur les données. Par conséquent, le choix de la mesure de qualité à appliquer sur les règles doit être effectué par l'expert en

fonction de ses préférences. Il se pose ainsi clairement le problème de la sélection des meilleures règles dans un environnement coopératif où plusieurs experts interviennent dans la décision ayant chacun une préférence pour une mesure de qualité. Prenons

	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8
r_1	1.00	1.00	1.00	0.20	0.20	0.40	0.40	0.40
r_2	0.70	0.70	0.70	1.00	1.00	0.20	0.20	0.20
r_3	0.40	0.40	0.40	0.70	0.70	1.00	1.00	0.70
r_4	0.20	0.20	0.20	0.40	0.40	0.70	0.70	1.00

TABLE 1.1 – Un exemple d'évaluation de quatre règles r_1, r_2, r_3 et r_4 par huit mesures de qualité m_1, m_2, m_3, m_4, m_5, m_6, m_7, et m_8

l'exemple d'une évaluation de quatre règles selon huit mesures, illustré par le tableau 1.1, et d'un environnement coopératif qui comprend quatre experts e_1, e_2, e_3 et e_4 chacun ayant choisi d'utiliser respectivement m_1, m_4, m_6 et m_8. Supposons que tous les experts fixent un seuil minimal de 0.6, alors l'expert e_1 obtient les règles r_1 et r_2, l'expert e_2 obtient les règles r_2 et r_3 et les experts e_3 et e_4 obtiennent les règles r_3 et r_4 (*c.f.,* la figure 1.1). Le même résultat est obtenu lorsqu'un même expert e opte pour les mesures m_1, m_4, m_6 et m_8 (*c.f.,* la figure 1.2). Ainsi, la question qui se pose : comment synthétiser les préférences partielles modélisées par chaque mesure en un tout cohérent, *i.e.,* une préférence globale, pouvant servir de base à déterminer les meilleures règles ?

– L'utilisateur peut avoir des difficultés dans la spécification d'un seuil pour une mesure. En effet, fixer un seuil élevé peut entraîner la perte de certaines règles pertinentes et un seuil faible peut produire des règles redondantes (par exemple, des règles similaires). Ce problème devient plus complexe lorsque plusieurs mesures sont sélectionnées. En effet, l'utilisateur doit faire face à deux problèmes : fixer un seuil fiable pour chaque mesure et prendre une décision concernant les règles qui ne vérifient que certains seuils.

L'objectif que nous nous sommes fixé consiste à résoudre ces problèmes en sélectionnant un ensemble réduit de règles d'association selon plusieurs mesures. Dans ce contexte, *les systèmes d'agrégation de préférences* tels que les systèmes de vote [Bra07] et la dominance de Pareto [KLP75, Mat91], peuvent être utiles pour résoudre ces problèmes. En effet, ces systèmes présentent des méthodes permettant d'agréger les avis exprimés par un ensemble d'individus concernant différentes alternatives de façon à déterminer une alternative "ga-

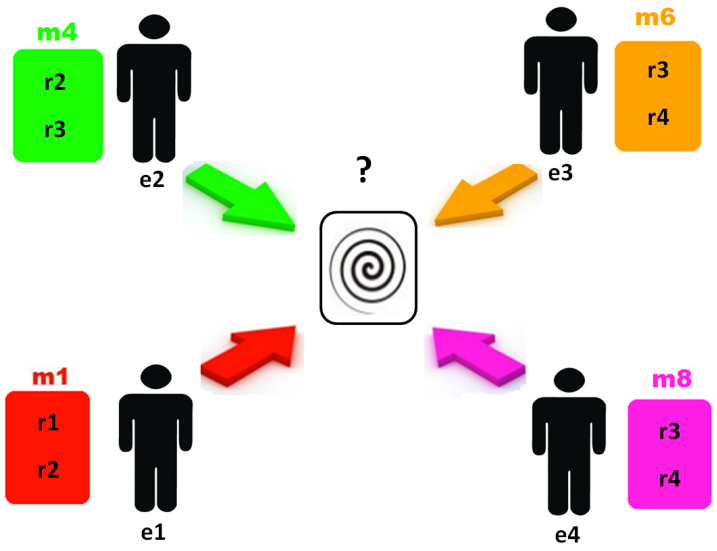

FIGURE 1.1 – Environnement coopératif de quatre experts e_1, e_2, e_3 et e_4 chacun ayant choisi d'utiliser respectivement m_1, m_4, m_6 et m_8 pour la sélection de règles à partir de l'exemple d'évaluation du tableau 1.1.

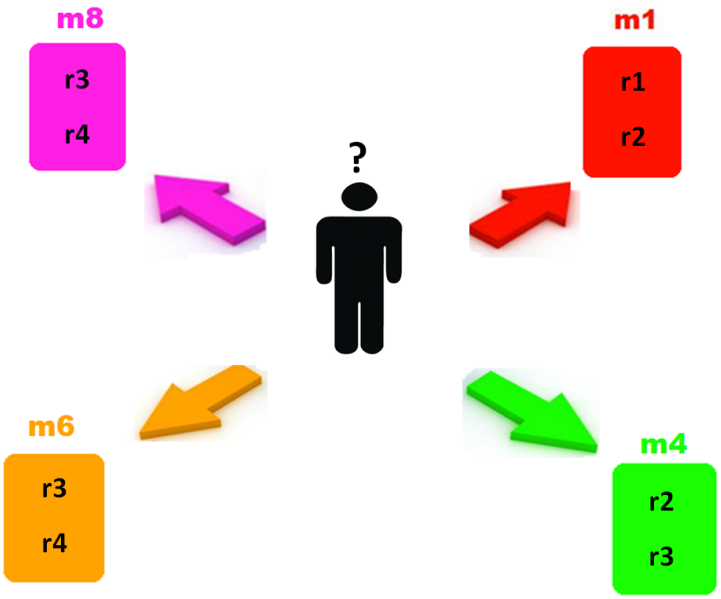

FIGURE 1.2 – Exemple d'un expert ayant choisi d'utiliser respectivement m_1, m_4, m_6 et m_8 pour la sélection de règles à partir de l'exemple d'évaluation du tableau 1.1.

gnante" ou encore à classer par ordre de préférence les différentes alternatives. Il est aisé de voir le lien entre l'agrégation de préférences et la détermination des meilleures règles selon plusieurs mesures grâce aux correspondances suivantes : les individus peuvent être

référés aux experts (aussi aux mesures) et les alternatives peuvent être référées aux règles d'association.

Ainsi, nos contributions concernent l'exploitation d'un système d'agrégation de préférences pour la sélection des règles d'association en considérant plusieurs mesures simultanément. Ainsi, dans cette perspective, nos contributions sont les suivantes :

- **Règles non dominées :** Notre première contribution consiste à proposer une approche permettant de sélectionner les règles pertinentes selon plusieurs mesures en se basant sur la notion de dominance de Pareto [BSYN12]. Une règle r est dite dominée par une autre règle r' si r' est au moins aussi bonne que r pour **toutes** les mesures. La règle r est écartée du résultat, non pas parce qu'elle n'est pas pertinente pour l'une des mesures, mais parce qu'elle n'est pas pertinente selon la combinaison de toutes les mesures. En d'autres termes, il existe au moins une meilleure règle pour les experts qui, elle, sera retenue. Une telle règle est appelée *règle non-dominée*.

- **Sélection des Top-k règles d'association :** En fouille de données, la recherche des k meilleures règles selon une mesure de qualité se révèle très utile pour trouver les règles d'association les plus significatives par rapport aux préférences de l'expert. Dans la littérature, plusieurs approches ont été proposées et qui permettent de sélectionner les k meilleures règles (*top-k*) selon une seule mesure, où k est fixé par l'expert [FVWT12, Web11]. Cependant, aucune de ces approches ne permet de résoudre le problème en présence de plusieurs experts ayant chacun une préférence pour une mesure de qualité. Ainsi, notre deuxième contribution consiste à proposer une approche *top-k* permettant de sélectionner les k meilleures règles selon plusieurs mesures [BSYN14].

- **Règles représentatives :** Bien que la relation de dominance permettait de sélectionner des règles intéressantes selon plusieurs mesures, elle n'accorde aucune importance à la similarité structurelle et sémantique qui peuvent exister entre les règles. En effet, une règle non dominée r peut éliminer une autre règle r' légèrement moins intéressante alors qu'elles véhiculent des informations totalement différentes. Dans ce cas, l'information contenue dans la règle r' ne sera pas présentée à l'expert. Ainsi, la négligence du côté sémantique, peut entraîner une perte d'information. Notre troisième contribution consiste, alors, à proposer une approche permettant de sélectionner un ensemble réduit de règles selon plusieurs mesures, appelé "*règles représentatives*", tout en tenant compte de l'aspect sémantique. L'objectif de cet ensemble est en effet de produire un ensemble de règles qui soient à la fois les plus représentatives des don-

nées (*i.e.,* véhiculant le maximum d'informations utiles) et non redondantes (*i.e.,* ne véhiculant pas la même information). Afin de mettre en valeur notre contribution, nous avons mené des expérimentations sur des bases benchmark, qui ont permis d'illustrer le gain qu'apporte les règles représentatives en termes de compacité par rapport au nombre total des règles.

1.3 Organisation du manuscrit

Le reste de manuscrit est organisé comme suit :

Le deuxième chapitre présente les règles d'association et les algorithmes qui permettent de les extraire. Par ailleurs, une étude des principales approches dédiées à l'extraction des bases génériques ainsi que les approches d'extraction sous-contraintes est présentée.

Le troisième chapitre présente les différentes approches d'agrégation des préférences et montre leur connexion avec l'approche de recherche de règles d'association. Il présente également une étude des approches dédiées à la sélection de motifs en utilisant les préférences.

Dans le quatrième chapitre, nous proposons une nouvelle approche permettant de sélectionner les règles d'association selon plusieurs mesures de qualité. A cet effet, nous introduisons la notion de règles non dominées et nous proposons un algorithme pour les sélectionner. Une étude expérimentale sur les règles non dominées est réalisée confirmant qu'elles sont bien moins nombreuses que certaines générées par une approche de sélection sous contraintes. Nous proposons également, une approche permettant de sélectionner les k meilleures règles selon plusieurs mesures en tirant profit de la notion de dominance de Pareto.

Dans le cinquième chapitre, nous introduisons une nouvelle famille de règles qualifiées de *règles représentatives*. L'objectif de cette famille est de sélectionner un ensemble réduit de règles, selon plusieurs mesures, qui soient à la fois les plus représentatives des données (*i.e.,* véhiculant le maximum d'informations utiles) et peu redondantes (*i.e.,* ne véhiculant pas la même information). Nous proposons également une approche permettant de sélectionner les *règles représentatives* lorsque la similarité structurelle entre les règles est transitive, puis une deuxième approche dans le cas contraire (*i.e.,* lorsque la similarité structurelle entre les règles est non-transitive). Nous illustrons à travers une série d'expé-

rimentations le gain qu'apporte ces règles en termes de compacité par rapport au nombre total des règles.

Le chapitre six présente une conclusion générale qui rappelle l'ensemble de nos travaux, puis elle adresse quelques perspectives de prolongement de nos travaux en fonction de ce que nous avons obtenu comme résultats.

Chapitre 2

Règles d'association

Sommaire

2.1 Introduction 9
2.2 Problème d'extraction de règles d'association 10
2.2.1 Formalisme de base 10
2.2.2 Décomposition du problème d'extraction des règles d'association 13
2.3 Extraction des règles d'association à partir des motifs fréquents 13
2.3.1 Algorithme Apriori 14
2.3.2 Génération des règles d'association 17
2.3.3 Autres algorithmes 18
2.4 Réduction des règles d'association redondantes 19
2.4.1 Analyse formelle des concepts 20
2.4.2 Bases génériques pour les règles d'association 24
2.4.3 Synthèse des différentes bases génériques 36
2.5 Extraction de règles d'association sous contraintes 37
2.6 Conclusion 39

2.1 Introduction

L'extraction des connaissances à partir des données (ECD) désigne le processus de découverte des informations implicites précédemment inconnues et potentiellement utiles à partir des bases de données volumineuses [FPSM92]. Introduite par *Agrawal et al.* [AIS93a] l'extraction de règles d'association est l'un des principaux problèmes de l'ECD. À l'origine, ce problème a concerné l'analyse des bases de transactions de vente. En effet, l'extraction

des règles d'association permet d'analyser les achats des clients afin de comprendre leurs comportements de consommation, ranger les rayons du magasin, proposer des promotions, gérer les stocks, etc, dans le but d'augmenter le profit. Dans la base de données de vente, une transaction contient un ensemble d'articles achetés par un client, appelés items ou attributs. Dans ce contexte, les règles d'association sont des implications conditionnelles entre des ensembles d'articles dans une base transactionnelle. En d'autres termes, l'objectif de l'extraction des règles d'association est d'identifier si la présence d'un ensemble d'articles dans une transaction est associée à la présence d'un autre ensemble d'articles.

Afin de déterminer les règles d'association significatives ou valides, deux mesures sont très souvent utilisées, à savoir le support et la confiance, permettant de mesurer, respectivement, la fréquence et la force de validité d'une règle. Ainsi, seules les règles d'association, ayant un support et une confiance au moins égaux à un seuil minimal de support *minsup* et à un seuil minimal de confiance *minconf* prédéfinis par l'utilisateur, seront extraites. Ce problème est loin d'être évident car le nombre d'attributs de la base transactionnelle peut être très grand.

Dans ce chapitre, nous présentons le problème d'extraction des règles d'association basée sur les motifs fréquents. Nous rappelons aussi, les éléments fondamentaux de l'Analyse Formelle de Concepts (AFC) [GW99] qui sont utilisés pour la dérivation de sous-ensembles de règles non redondantes, appelé *bases génériques*.

2.2 Problème d'extraction de règles d'association

Dans ce qui suit, nous présentons le formalisme de base et la décomposition du problème d'extraction de règles d'association.

2.2.1 Formalisme de base

Dans son formalisme initial, l'extraction des règles d'association considère un ensemble $\mathcal{T} = \{t_1, t_2,...,t_n\}$ de n transactions décrites par un ensemble $\mathcal{I} = \{i_1, i_2,...,i_m\}$ de m attributs, appelés aussi items, selon une relation binaire $\mathcal{R} \subseteq \mathcal{T} \times \mathcal{I}$. Un exemple d'une telle relation binaire est représenté par le tableau 2.1. Les colonnes représentent les attributs de a à e alors que les lignes représentent les transactions de t_1 à t_6. Une croix indique quels sont les items qui sont en relation avec une transaction. La donnée de cette relation binaire permet de déterminer l'ensemble des transactions présentant un ensemble donné d'items.

$\mathcal{R}$	a	b	c	d	e
t_1	×	×	×	×	×
t_2	×	×		×	×
t_3	×	×	×		×
t_4		×	×	×	
t_5	×	×		×	×
t_6		×	×		×

TABLE 2.1 – Exemple de relation binaire.

Définition 1 ***(motifs et support)*** *:*

1. *Soit I un sous-ensemble d'items, contenant k items, alors I est appelé k-motif.*
2. *Un motif $I \subseteq \mathcal{I}$ décrit une transaction de $\mathcal{T}$ si cette transaction est en relation avec tous les items de I*
3. *Le support d'un motif I est le nombre de transactions contenant I : $support(I) = |\{t \in \mathcal{T} | \ \forall \ i \in I, \ i\mathcal{R}t\}|$*
4. *La fréquence d'un motif I est donnée par $frequence(I) = \frac{support(I)}{|\mathcal{T}|}$*

Propriété 1 ***(antimonotonicité)*** *Soit un motif $I \subseteq \mathcal{I}$.*

$$\forall \ I' \subseteq I \ \text{on a} \ support(I') \geq support(I)$$

La preuve de cette propriété est triviale, car toutes les transactions qui contiennent I contiennent également I'. Cette propriété est visible sur le diagramme de l'ordre des motifs représenté sur la figure 2.1. On peut observer la décroissance du support des motifs indiqué entre parenthèses, le long de tout chemin ascendant du sommet représentant le motif vide.

Définition 2 ***(motifs fréquent ou non fréquent)*** *:*
Un motif I est dit fréquent lorsque son support est supérieur ou égal à un seuil minimal minsup, fixé par l'utilisateur. Autrement il est dit non fréquent.

Exemple 1 *Considérons le motif abc de la relation binaire donnée par le tableau 2.1. Les deux transactions 1 et 3 contiennent ce motif. Ainsi, $support(abc) = 2$. La fréquence de abc est alors égale à 1/3. Si $minsup = 1$, alors abc est fréquent puisque $support(abc) = 2 \geq 1$*

Propriété 2 *[AS94] Étant donné un seuil de support minsup fixé,*

- *tout sous-ensemble d'un motif fréquent est fréquent,*
- *tout sur-ensemble d'un motif non fréquent est non fréquent.*

Preuve 1 *Soit minsup un seuil de support fixé.*
Soit I' un sous-ensemble d'un motif fréquent I. D'après la propriété 1, on a $support(I') \geq support(I)$. I étant fréquent, $support(I) \geq minsup$. Ainsi, $support(I') \geq minsup$, I' est donc fréquent.
De par cette même propriété, I'_1 étant un sur-ensemble d'un motif I_1 non fréquent, on déduit que : $support(I_1) \geq support(I'_1)$. I_1 étant non fréquent, $support(I_1) < minsup$. Ainsi, $support(I'_1) < minsup$, I'_1 est donc non fréquent.

Définition 3 ***(règle d'association, support, confiance)*** *:*
Une règle d'association est définie comme une implication entre deux motifs. Elle est de la forme suivante : $R : X \Rightarrow Y - X$ tel que $X \subseteq \mathcal{I}$, $Y \subseteq \mathcal{I}$ et $X \subset Y$. La règle R est dite basée sur le motif Y et les deux motifs X et $Y - X$ sont appelés, respectivement, prémisse et conclusion de R.
Afin de vérifier la validité d'une règle associative R, deux mesures sont communément utilisées :

- **Le support** *: il correspond au nombre de transactions qui contiennent le motif Y. Le support de la règle R, noté $support(R)$ est donné par $support(Y)$.*
- **La confiance** *: elle exprime la probabilité conditionnelle qu'une transaction contienne $Y - X$ sachant qu'elle contient X. La confiance de la règle R, notée $confiance(R)$, est mesurée par le ratio $\frac{support(Y)}{support(X)}$.*

Notons que la confiance de R est toujours supérieure ou égale à sa fréquence : $confiance(R) \geq frequence(R) = \frac{support(R)}{|\mathcal{T}|}$. En effet, $support(X) \leq |\mathcal{T}|$.

Définition 4 ***(règle d'association, valide, exacte, approximative)*** *:*
Une règle R est dite valide si :

- *Son support, $support(R)$, est supérieur ou égal à un seuil minimal minsup, fixé par l'utilisateur,*
- *Sa confiance, $confiance(R)$, est supérieure ou égale à un seuil minimal $minconf$, fixé par l'utilisateur.*

Si $confiance(R) = 1$, alors R est appelée règle d'association exacte. Sinon, elle est appelée règle d'association approximative.

Exemple 2 *Soient les deux règles d'association $a \Rightarrow b$ et $a \Rightarrow c$ extraites à partir de la relation binaire donnée par le tableau 2.1. Si $minsup = 1$ et $minconf = 0.5$, alors :*

- *$a \Rightarrow b$ est une règle valide exacte, puisque $support(a \Rightarrow b) = 4 \geq 1$ et $confiance(a \Rightarrow b) = 1 \geq 0.5$.*
- *$a \Rightarrow c$ est une règle valide approximative, puisque $support(a \Rightarrow c) = 2 \geq 1$ et $0.5 \leq confiance(a \Rightarrow c) = 0.5 < 1$.*

2.2.2 Décomposition du problème d'extraction des règles d'association

L'extraction des règles d'association consiste à déterminer l'ensemble de règles valides *i.e.*, dont le support et la confiance sont au moins égaux à un seuil minimal de support *minsup* et à un seuil minimal de confiance *minconf* prédéfinis par l'utilisateur. Une décomposition de ce problème a été présentée dans [AIS93b] :

1. Extraction des motifs fréquents : ce sont tous les motifs ayant un support au moins égal à *minsup*.
2. Génération des règles d'association valides basées sur l'ensemble des motifs fréquents préalablement extraits : ces règles sont de la forme $R : X \Rightarrow Y$ et $confiance(R) \geq minconf$.

2.3 Extraction des règles d'association à partir des motifs fréquents

Au niveau de la première phase *i.e.,* l'extraction des motifs fréquents, les 2^m motifs potentiellement fréquents, extraits d'un ensemble $\mathcal{I}$ de m items, constituent l'ensemble des parties de l'ensemble des items $\mathcal{I}$. L'ensemble ordonné $\mathcal{L}_{\mathcal{I}} = (\mathcal{P}(\mathcal{I}),\leq)$ est un treillis d'inclusion appelé aussi treillis des parties de $\mathcal{I}$, ayant une hauteur égale à $(m + 1)$ (c.f., figure 2.1). La découverte des motifs fréquents revient alors à extraire, à partir du treillis des parties de $\mathcal{I}$, les motifs dont le support est au moins égal au support minimal *minsup*.

Les premiers travaux dédiés à la découverte des motifs fréquents [BMU97, GPW98, HPY00, BTP+00] ont montré l'intérêt de définir un ordre total sur l'ensemble des motifs. En effet, ceci permet d'une part d'éviter les calculs de supports redondants lors de la recherche, en parcourant chaque motif du treillis au plus une fois. D'autre part, il permet

FIGURE 2.1 – Treillis d'inclusion associé au contexte illustré par le tableau 2.1

de construire des structures de données permettant d'exécuter, de manière efficace, les opérations relatives aux motifs. La plupart des travaux ont utilisé l'ordre lexicographique, constituant un ordre total sur l'ensemble des motifs.

En plus de l'aspect combinatoire de la recherche des motifs fréquents, des balayages répétitifs de la base de transactions doivent être effectués afin de déterminer les supports des motifs et de déterminer ceux qui sont fréquents. En effet, les opérations d'entrée/sortie liées aux lectures de transactions sont très coûteuses. Ceci a pour conséquence d'augmenter le temps global nécessaire à l'extraction des règles d'association. Pour pallier cet inconvénient, de nombreux travaux se sont concentrés sur la réduction du nombre d'accès à la base de transactions [BMU97, GPW98].

2.3.1 Algorithme APRIORI

Plusieurs algorithmes permettant de réduire le coût d'extraction des motifs fréquents ont été proposés (*cf.* les travaux [BTP+02, H.T96, HPY00, SON95, PCY95]). Parmi ceux-

ci, nous présentons l'algorithme APRIORI introduit par Agrawal *et al.* [AIS93b] qui est considéré comme l'archétype des algorithmes de recherche par niveau.

Cet algorithme est itératif et adopte une stratégie d'exploration par niveaux de l'espace de recherche, appelée "*Tester-et-générer*". À chaque itération k, l'ensemble de tous les motifs candidats de taille k (motifs fréquents potentiels) est généré. Ensuite, un accès à la base de transactions est réalisé afin de supprimer les candidats non fréquents. L'ensemble des k-motifs fréquents retenu est utilisé lors de l'itération $(k+1)$ suivante pour générer les candidats de taille $(k+1)$. Dans ce qui suit, nous présentons l'algorithme APRIORI, dont le pseudo-code est donné par l'algorithme 1. Les notations utilisées sont présentées dans le Tableau 2.2.

$\mathcal{C}_k$	l'ensemble des k-motifs candidats.
$\mathcal{IF}_k$	l'ensemble des k-motifs fréquents.
$\mathcal{S}_t$	Sous-ensemble de $\mathcal{C}_k$ contenu dans une transaction t.

TABLE 2.2 – Notations utilisées dans l'algorithme APRIORI

L'algorithme APRIORI prend en entrée un contexte $\mathcal{K}=(\mathcal{T}, \mathcal{I}, \mathcal{R})$ constitué d'une relation binaire $\mathcal{R}$ entre un ensemble de transactions $\mathcal{T}$ et un ensemble d'items $\mathcal{I}$, le seuil minimal de support *minsup* et retourne en sortie l'ensemble des motifs fréquents $\mathcal{IF}$.

Dans la première itération, l'ensemble des 1-motifs est initialisé avec les items de la base de transactions. Un balayage de la base est réalisé afin de déterminer l'ensemble des 1-motifs fréquents $\mathcal{IF}_1$ (ligne 2). Durant une itération k, APRIORI construit l'ensemble $\mathcal{C}_k$ des motifs de taille k candidats à être fréquents en utilisant la phase combinatoire d'APRIORIGEN (*c.f,* 2) appliquée aux motifs fréquents de taille k-1 (ligne 4). Cette phase combinatoire consiste à joindre les motifs fréquents de taille k-1 partageant k-2 items. Parmi les candidats de taille k, seuls sont retenus les motifs dont tous les sous-motifs de longueur k-1 se sont révélés fréquents (lignes 5-8), d'après la propriété d'anti-monotonie des motifs fréquents (*c.f.,* 2). Les supports de ces motifs candidats sont calculés suite à un accès à la base de transactions (*c.f,* fonction SUBSET, lignes 10-12). Les k-motifs fréquents sont finalement insérés dans l'ensemble $\mathcal{IF}$(ligne 13). Seuls les motifs candidats fréquents sont conservés et la procédure est réitérée pour les motifs de longueur k+1. L'algorithme prend fin lorsqu'il n'y a plus de candidats à générer.

Données : Une base de transactions $\mathcal{T}$, seuil minimal de support *minsup*
Résultats : Ensemble des motifs fréquents $\mathcal{IF}$

```
1  début
2      IF_1 = {1-motifs fréquents} ;
3      pour (k=2 ; IF_{k-1} ≠ ∅ ; k++) faire
4          C_k = AprioriGen(IF_{k-1}) ;
5          pour chaque candidat c ∈ C_k faire
6              pour chaque sous-ensemble c_1 de c de taille k-1 faire
7                  si (c_1 ∉ IF_{k-1}) alors
8                      C_k = C_k - c ;
9          pour chaque transaction t ∈ T faire
10             S_t = Subset(C_k, t) ;
11         pour chaque candidat c ∈ S_t faire
12             c.support++ ;
13         IF_k = {c ∈ C_k | c.support ≥ minsup} ;
14     retourner IF = ⋃_k IF_k ;
15 fin
```

Algorithme 1 : Algorithme APRIORI

Données : Ensemble des $k-1$-motifs fréquents $\mathcal{IF}_{k-1}$
Résultats : Ensemble des k-motifs candidats $\mathcal{C}_k$

```
1  début
2      C_k = ∅
3      pour chaque x ∈ IF_{k-1} faire
4          pour chaque y ∈ IF_{k-1} faire
5              c = x ∪ y
6              si |c| = k alors
7                  C_k = C_k ∪ c
8      retourner C_k ;
9  fin
```

Algorithme 2 : Algorithme APRIORI-GEN

2.3.2 Génération des règles d'association

Après avoir présenté la première phase du processus de génération de règles d'association, à savoir l'extraction des motifs fréquents, nous présentons dans cette section, la seconde phase de ce processus, à savoir la génération des règles d'association. Le problème de génération des règles d'association est un problème exponentiel en fonction de la taille des motifs fréquents [PBTL98]. À partir d'un motif fréquent de taille $k \geq 2$, 2^k-2 règles d'association peuvent être générées. Toutefois, cette génération est réalisée de manière directe, sans accéder à la base de transactions. Ainsi, les temps d'exécution de cette étape sont faibles, par rapport au temps nécessaire à l'extraction des motifs fréquents. Le principe de la génération de règles d'association valides est le suivant : pour chaque motif fréquent I_k de taille $k \geq 2$, chaque sous-ensemble I_s de I_k est déterminé et la valeur du rapport $\frac{support(I_k)}{support(I_s)}$ est calculée. Si cette valeur est supérieure ou égale au seuil de confiance minimale $minconf$, alors la règle d'association $I_s \Rightarrow (I_k$ - $I_s)$ est générée. Afin de réduire le nombre d'opérations réalisées pour la génération, un algorithme optimisé a été proposé par *Agrawal et al.* [AS94]. Cet algorithme se base sur la propriété suivante :

Propriété 3 *Soit I un motif fréquent :*
$\forall I_1 \subset I$, $I_1 \neq \emptyset$, si la règle $I_1 \Rightarrow I - I_1$ est valide, alors $\forall I_2 \subset I_1$, $I_2 \neq \emptyset$, la règle $I_2 \Rightarrow I - I_2$ est valide.

Dans ce qui suit, nous présentons l'algorithme de génération des règles d'association, appelé Gen-Rules, proposé par Agrawal *et al.* [AIS93b].

$\mathcal{IF}$	Ensemble de motifs fréquents.
$\mathcal{H}_m$	m-motifs qui figurent dans la partie conclusion des règles valides générées à partir du motif I_k.

Table 2.3 – Notations utilisées dans l'algorithme Gen-Rules

Le pseudo-code de l'algorithme est présenté dans l'algorithme 3. Les notations utilisées sont présentées dans le Tableau 2.3. L'algorithme fonctionne de la manière suivante. Pour chacun des motifs fréquents I_k contenant au moins deux items, l'algorithme génère les règles d'association ayant un seul item dans la conclusion (ligne 3). Ensuite, la procédure Ap-Genrules est appelée afin d'insérer dans $\mathcal{AR}$ les règles valides générées à partir de I_k, dont la conclusion contient plus d'un item (ligne 4). Cette procédure, dont le pseudo-code est donné par l'algorithme 4, prend en entrée un motif fréquent I_k de taille k, un ensemble $\mathcal{H}_m$ qui contient les motifs de taille m, qui sont les conclusions de règles valides

générées à partir de I_k et un seuil minimal de confiance *minconf* comme paramètres. Cette procédure réutilise APRIORI-GEN pour combiner les motifs de $\mathcal{H}_m$ afin de générer l'ensemble $\mathcal{H}_{m+1}$ des motifs de taille $m+1$ qui peuvent être des conclusions de règles valides générées à partir de I_k (ligne 3). Chaque règle, dont la conclusion est un motif de $\mathcal{H}_{m+1}$, est alors testée : si la règle satisfait la contrainte *minconf*, alors elle est ajoutée au résultat (ligne 6). Autrement, la conclusion $(m+1)$-motif est supprimée de $\mathcal{H}_{m+1}$ (ligne 8). AP-GENRULES est appelée récursivement pour $\mathcal{H}_{m+1}$ jusqu'à ce que la taille des motifs de $\mathcal{H}_{m+1}$ dépasse celle du motif courant I_k.

Données : Ensemble des motifs fréquents $\mathcal{IF}$, seuil minimal de confiance *minconf*
Résultats : Ensemble des règles d'association valides $\mathcal{AR}$

```
1 début
2   pour chaque k-motif fréquent I_k ∈ IF tel que card(I_k)≥2 faire
3     H_1 ← {{i} | confiance(I_k \ i → i) ≥ minconf} ;
4     AR ← AR ∪ Ap-Genrules(I_k, H_1, minconf) ;
5   retourner AR ;
6 fin
```

Algorithme 3 : Algorithme GEN-RULES

2.3.3 Autres algorithmes

Dans la littérature, de nombreux algorithmes ont été développés en vue de réduire la complexité exponentielle de l'algorithme APRIORI. En particulier, les algorithmes qui adoptent la stratégie de l'exploration en profondeur de l'espace de recherche tel que ECLAT [ZPOL97] qui consiste à énumérer les motifs fréquents dans un ordre prédéfini. Ce parcours en profondeur est réalisé en ordonnant les items des motifs selon un ordre lexicographique afin d'éviter des générations redondantes de motifs. Par exemple, le motif *aecb* est représenté par le motif *abce*. Le motif *abce* ne sera généré qu'une seule fois par son parent *abc*, et il sera généré seulement si *abc* est fréquent. Les algorithmes en profondeur se sont focalisés essentiellement sur la réduction des entrées/sorties et la minimisation du coût de l'étape de calcul du support d'un motif I en sauvegardant la liste des transactions contenant I. En utilisant un codage vertical des données associant chaque item i à la liste des transactions contenant cet item, il est possible de calculer rapidement le support du motif $I \cup i$ résultant de l'extension du motif I par l'item i. Enfin, l'algorithme FP-GROWTH

Données : k-motif fréquent I_k, ensemble $\mathcal{H}_m$ de m-motifs qui figurent dans la partie conclusion des règles valides générées à partir de I_k, seuil minimal de confiance *minconf*.

Résultats : Ensemble $\mathcal{AR}$ de règles associatives valides augmenté des règles valides générées à partir de I_k dont la partie conclusion est un $(m+1)$-motif.

```
1  début
2      si (k > m+1) alors
3          H_{m+1} ← Apriori-Gen(H_m) ;
4          pour chaque h_{m+1} ∈ H_{m+1} faire
5              si confiance(I_k \ h_{m+1} → h_{m+1}) ≥ minconf alors
6                  AR ← AR ∪ {I_k \ h_{m+1} → h_{m+1}} ;
7              sinon
8                  H_{m+1} ← H_{m+1} \ h_{m+1} ;
9          AR ← AR ∪ Ap-Genrules(I_k, H_{m+1})
10     retourner AR ;
11 fin
```

Algorithme 4 : Algorithme Ap-Genrules

[HPY00] qui opère sur une structure de données compacte, appelée *FP-tree*, construite en parcourant en profondeur l'ensemble des motifs. Cette structure permet de compresser l'ensemble des transactions en mémoire pour pouvoir calculer les supports des motifs rapidement.

2.4 Réduction des règles d'association redondantes

Le problème de l'exploitation du résultat présenté à l'utilisateur est devenu primordial, étant donné que le nombre de règles d'association extraites à partir d'ensembles de données réels est très important. Ce constat est notamment renforcé pour les bases de données denses, dans lesquelles les items sont fortement corrélés [PBTL99b]. Ceci est dû à la présence de règles redondantes *i.e.,* véhiculant la même information. Par conséquent, l'utilisateur ne peut plus gérer et exploiter efficacement les connaissances utiles qui lui sont présentées. Ainsi, l'objectif de la fouille de données ne consiste plus seulement, à extraire les connaissances utiles pour l'utilisateur mais aussi, à :

1. déterminer l'ensemble minimal de règles d'association (ou base générique) présenté à l'utilisateur tout en maximisant la quantité d'informations utiles véhiculées ;
2. Disposer d'un mécanisme d'inférence qui, suite à la demande de l'utilisateur, permet de retrouver le reste des règles d'association tout en déterminant avec exactitude leurs supports et leurs confiances sans accéder à la base de données.

Ceci a suscité l'intérêt des chercheurs pour proposer des approches de recherche des règles, qui véhiculent le maximum de connaissances utiles en se basant sur les travaux issus de la théorie de l'analyse formelle des concepts (AFC) dont l'objectif est d'étudier le problème de l'extraction des connaissances sous l'angle des treillis de Galois [BM70]. L'AFC introduite par *Wille* [Wil82] a initialement trouvé des applications en intelligence artificielle pour la représentation et l'acquisition automatique des connaissances [Wil89] [LN90] [Ngu94], puis dans la génération des règles à partir du treillis [GD86]. Wille propose de considérer chaque élément du treillis, perçu comme une hiérarchie, comme un concept formel, et le graphe associé comme une relation de généralisation/spécialisation. Ainsi, cette hiérarchie de concepts met en évidence de façon exhaustive les regroupements potentiellement intéressants par rapport aux observations.

Dans ce qui suit, nous présentons quelques résultats clefs provenant de l'analyse formelle des concepts (AFC) et nous montrons leur connexion avec la technique d'extraction des règles d'association.

2.4.1 Analyse formelle des concepts

Nous commençons par présenter la correspondance de Galois utilisée pour faire le lien entre l'ensemble des parties de $\mathcal{T}$ et l'ensemble des parties de $\mathcal{I}$ associés respectivement à l'ensemble des transactions $\mathcal{T}$ et l'ensemble des items $\mathcal{I}$.

Correspondance de Galois

Soit un contexte d'extraction $\mathcal{K} = (\mathcal{T},\mathcal{I},\mathcal{R})$. Soit l'application ϕ, de l'ensemble des parties de $\mathcal{T}$(*i.e.,* l'ensemble de tous les sous-ensembles de $\mathcal{T}$), noté par $2^{\mathcal{T}}$, dans l'ensemble des parties de $\mathcal{I}$, $2^{\mathcal{I}}$. L'application ϕ associe à un ensemble de transactions $T \subseteq \mathcal{T}$ l'ensemble des items $i \in \mathcal{I}$ communs à tous les objets $t \in T$:

$$\phi : 2^{\mathcal{T}} \rightarrow 2^{\mathcal{I}}$$
$$\phi(T) = \{i \in \mathcal{I} \mid \forall\, t \in T, (t,i) \in \mathcal{R}\}$$

Soit l'application ψ, de l'ensemble des parties de $\mathcal{I}$ dans l'ensemble des parties de $\mathcal{T}$. Cette application associe à tout motif $I \subseteq \mathcal{I}$ l'ensemble des transactions $t \subseteq \mathcal{T}$ contenant tous les items $i \in I$:

$$\psi : 2^{\mathcal{I}} \rightarrow 2^{\mathcal{T}}$$
$$\psi(I) = \{t \in \mathcal{T} \mid \forall\, i \in I, (t,i) \in \mathcal{R}\}$$

Le couple d'applications (ϕ,ψ) définit une *correspondance de Galois* entre l'ensemble des parties de $\mathcal{T}$ et l'ensemble des parties de $\mathcal{I}$. Étant donné une correspondance de Galois, les propriétés suivantes sont vérifiées quelques soient I, I_1, $I_2 \subseteq \mathcal{I}$ et T, T_1, $T_2 \subseteq \mathcal{T}$:

1. $I_1 \subseteq I_2 \Rightarrow \psi(I_2) \subseteq \psi(I_1)$;
2. $T_1 \subseteq T_2 \Rightarrow \phi(T_2) \subseteq \phi(T_1)$;
3. $T \subseteq \psi(I) \Leftrightarrow I \subseteq \phi(T) \Leftrightarrow (I,T) \in \mathcal{R}$.

Opérateur de fermeture

Soit un ensemble partiellement ordonné (E,$\leq$). Une application γ de (E,$\leq$) dans (E,$\leq$) est appelée un *opérateur de fermeture*, si et seulement si elle possède les propriétés suivantes pour tous sous-ensembles $x, y \subseteq E$:

1. *Isotonie* : $x \leq y \Rightarrow \gamma(x) \leq \gamma(y)$
2. *Extensivité* : $x \leq \gamma(x)$
3. *Idempotence* : $\gamma(\gamma(x)) = \gamma(x)$

Étant donné un opérateur de fermeture γ sur un ensemble partiellement ordonné (E,$\leq$), un élément $x \in E$ est un élément *fermé* si l'image de x par l'opérateur de fermeture γ est égale à lui-même, *i.e.*, $\gamma(x) = x$ [PBTL99a].

Fermeture de la correspondance de Galois

Considérons les ensembles des parties $2^{\mathcal{I}}$ et $2^{\mathcal{T}}$ dotés de la relation d'inclusion $\subseteq$, *i.e.*, les ensembles partiellement ordonnés $(2^{\mathcal{I}}, \subseteq)$ et $(2^{\mathcal{T}}, \subseteq)$. Les opérateurs $\gamma = \phi \circ \psi$ de $(2^{\mathcal{I}}, \subseteq)$ dans $(2^{\mathcal{I}}, \subseteq)$ et $\omega = \psi \circ \phi$ de $(2^{\mathcal{T}}, \subseteq)$ dans $(2^{\mathcal{T}}, \subseteq)$ sont des *opérateurs de fermeture de la correspondance de Galois*. Étant donné une correspondance de Galois, les propriétés suivantes sont vérifiées quelques soient I, I_1, $I_2 \subseteq \mathcal{I}$ et T, T_1, $T_2 \subseteq \mathcal{T}$:

1. $I \subseteq \gamma(I)$	1'. $T \subseteq \omega(T)$
2. $\gamma(\gamma(I)) = \gamma(I)$	2'. $\omega(\omega(T)) = \omega(T)$
3. $I_1 \subseteq I_2 \Rightarrow \gamma(I_1) \subseteq \gamma(I_2)$	3'. $T_1 \subseteq T_2 \Rightarrow \omega(T_1) \subseteq \omega(T_2)$
4. $\gamma(\psi(I)) = \psi(I)$	4'. $\gamma(\phi(O)) = \phi(O)$

Concept formel

Une paire $c = (T, I) \in \mathcal{T} \times \mathcal{I}$ est appelée un *concept formel* si et seulement si $\phi(T) = I$ et $\psi(I) = T$. L'ensemble T est appelé l'*extension* du concept c et I est appelé son *intention*. La relation d'ordre partiel entre des concepts formels est définie comme suit : $\forall\ c_1 = (T_1, I_1)$ et $c_2 = (T_2, I_2)$ deux concepts formels, $c_1 \leq c_2$ si et seulement si $T_2 \subseteq T_1$ ($\Leftrightarrow$ $I_1 \subseteq I_2$).

Exemple 3 *La paire (12356, be) est un concept à partir du contexte d'extraction donné par le tableau 2.1.*

Opérateurs *sup* et *inf*

Soient (T_1, I_1) et (T_2, I_2) deux concepts formels. Les opérateurs *sup* ($\vee$) et *inf* ($\wedge$) sont définis respectivement comme suit :

- $(T_1, I_1) \vee (T_2, I_2) = (\omega(T_1 \cup T_2), I_1 \cap I_2)$
- $(T_1, I_1) \wedge (T_2, I_2) = (T_1 \cap T_2, \gamma(I_1 \cup I_2))$

Ainsi, les opérateurs *sup* et *inf* permettent d'obtenir, respectivement, la plus petite borne supérieure et la plus grande borne inférieure d'un couple de concepts formels.

Les concepts formels partiellement triés selon une relation d'inclusion ensembliste forment une structure appelée *treillis de Galois*.

Treillis de Galois

Étant donné un contexte d'extraction $\mathcal{K}$, l'ensemble des concepts formels $\mathcal{C}_\mathcal{K}$ est un treillis complet $\mathcal{L}_\mathcal{C} = (\mathcal{C}, \leq)$, appelé *treillis de Galois* ou *treillis de concepts formels*, quand l'ensemble $\mathcal{C}_\mathcal{K}$ est considéré avec la relation d'inclusion entre extensions (ou intentions) des concepts.

Motif fermé fréquent

Un motif $I \subseteq \mathcal{I}$ est appelé *motif fermé* si $\gamma(I) = I$. Un motif fermé est donc un ensemble maximal d'items communs à un ensemble de transactions. Un motif fermé I est fréquent si seulement si son support est supérieur ou égal à un seuil minimal de support *minsup*.

Exemple 4 *Soit le contexte d'extraction illustré par la figure 2.1. Pour minsup = 2, le motif abce est fermé fréquent de support 2.*

Dans ce qui suit, nous présentons trois propriétés relatives aux motifs fermés :

1. Tous les sous-ensembles d'un motif fermé fréquent sont fréquents.
2. Tous les super-ensembles d'un motif fermé non fréquent sont non fréquents.
3. Le support d'un motif I est égal au support de sa fermeture $\gamma(I)$, qui est le plus petit motif fermé contenant I, *i.e.*, support(I) = support$(\gamma(I))$.

Générateur minimal

Un motif $I_1 \subseteq \mathcal{I}$ est dit *générateur minimal* d'un motif fermé I, si et seulement si $\gamma(I_1) = I$ et, $\forall\ I_2 \subseteq \mathcal{I}$, si $\gamma(I_2) = I$, alors $I_1 = I_2$ [BPT$^+$00]. L'ensemble GM_I des générateurs minimaux d'un motif fermé I est défini comme suit :

$$GM_I = \{\ I_1 \subseteq \mathcal{I} \mid \gamma(I_1){=}I, \nexists\ I_2 \subset I_1, \gamma(I_2) {=}I\ \}.$$

Classe d'équivalence

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 partionné en des sous-ensembles disjoints, appelés aussi *classes d'équivalence*. Dans chaque classe, tous les éléments possèdent la même valeur de support et la même fermeture. Les générateurs minimaux d'une classe sont les éléments incomparables (selon la relation d'inclusion) les plus petits, tandis que le motif fermé est l'élément le plus large de cette classe.

Treillis de l'iceberg de Galois

Soit $\mathcal{MFF}$ l'ensemble des motifs fermés fréquents extraits du contexte formel $\mathcal{K} = (\mathcal{T}, \mathcal{I}, \mathcal{R})$. Lorsque l'ensemble $\mathcal{MFF}$ est ordonné partiellement par la relation d'inclusion ensembliste, la structure obtenue $\mathcal{L}_F{=}(\mathcal{MFF}, \leq)$ préserve seulement l'opérateur *sup*. Cette structure forme un semi-treillis supérieur et elle est désignée par *treillis de l'iceberg de Galois* [STB$^+$02].

Exemple 5 *Soit le contexte d'extraction illustré par le tableau 2.1. Le treillis de l'iceberg de Galois correspondant est présenté dans la figure 2.2 pour minsup* = 3. *Chaque nœud du treillis représente le couple* (*motif fermé, support*) *et il est décoré par la liste de ses générateurs minimaux.*

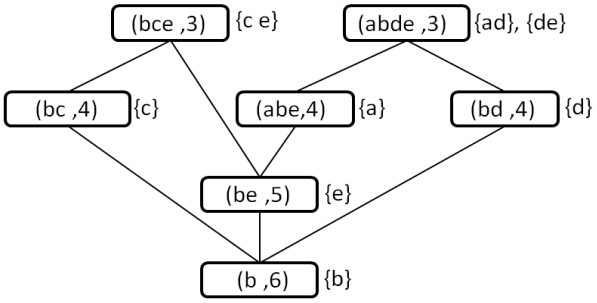

FIGURE 2.2 – Treillis de l'iceberg de Galois associé au contexte d'extraction présenté par le tableau 2.1 pour *minsup*=3.

2.4.2 Bases génériques pour les règles d'association

En pratique, l'extraction des règles d'association valides produit un nombre très important de règles conduisant à deux problèmes majeurs : d'une part l'exploration des règles par l'expert devient une tâche complexe et d'autre part l'information véhiculée par chaque règle s'en trouve réduite du fait de la redondance d'information entre les règles. Cette constatation a motivé la substitution de l'ensemble des règles d'association valide par une base de règles compacte plus facile à interpreter par l'expert. Ainsi, de nombreuses approches ont été proposées pour répondre à ce problème. Ces approches reposent sur l'extraction d'un sous-ensemble générique de toutes les règles d'association, appelé base générique, tout en satisfaisant les conditions suivantes [Kry02b] :

1. ***Dérivabilité*** : La base générique doit être accompagnée d'un mécanisme d'inférence permettant la dérivation des règles valides redondantes. Le mécanisme d'inférence doit être valide, *i.e.,* il ne permet de dériver que les règles d'association valides. Il doit être également complet, *i.e.,* il permet de retrouver l'ensemble de toutes les règles valides.

2. ***Informativité*** : la base générique des règles d'association doit permettre de retrouver avec exactitude le support et la confiance des règles valides redondantes.

Ces approchent sont basées principalement sur les opérateurs de fermeture de la correspondance de Galois utilisés dans l'analyse formelle des concepts (AFC). Dans ce qui suit, nous allons présenter et discuter les limites des différents travaux dédiés à l'extraction des bases génériques.

2.4.2.1 Mécanismes d'inférence de règles redondantes

Dans ce qui suit, nous présentons les différents mécanismes d'inférence de règles d'association proposés dans la littérature. Ces mécanismes se définissent comme étant les méthodes de dérivation (ou d'inférence) de règles à partir d'autres règles [KG02].

Les axiomes d'Armstrong ($\mathcal{AA}$)

Le système axiomatique d'Armstrong [Arm74] est utilisé uniquement dans le cadre des règles exactes et il est constitué des axiomes suivants :

- Augmentation : si $confiance(I \Rightarrow I_1) = 1$ alors, $confiance(I \cup I_2 \Rightarrow I_1) = 1$.
- Transitivité : si $confiance(I \Rightarrow I_1) = 1$ et $confiance(I_1 \Rightarrow I_2) = 1$ alors, $confiance(I \Rightarrow I_2) = 1$.

Transitivité de la confiance ($\mathcal{TC}$)

Contrairement à l'axiome de transitivité d'Armstrong, la transitivité de la confiance, introduite par Luxenburger [Lux91], est utilisée aussi bien pour les règles exactes que pour les règles approximatives.
Soient I, I_1 et I_2 trois motifs fréquents tels que $I \subset I_1 \subset I_2$. Le support et la confiance de la règle $I \Rightarrow (I_2 - I)$ peuvent être déterminés à partir des mesures de validité des règles $I \Rightarrow (I_1 - I)$ et $I_1 \Rightarrow (I_2 - I_1)$ comme suit :

- $support(I \Rightarrow I_2 - I) = support(I_1 \Rightarrow I_2 - I_1)$.
- $confiance(I \Rightarrow I_2 - I) = confiance(I \Rightarrow I_1 - I) \times confiance(I_1 \Rightarrow I_2 - I_1)$.

Ainsi, la confiance d'une règle dérivée est toujours inférieure ou égale aux confiances respectives des règles à partir desquelles elle a été inférée.

Opérateur de couverture ($\mathcal{C}$)

L'opérateur de couverture est introduit par Kryszkiewicz pour la dérivation des règles d'association à partir d'une règle donnée. La couverture $\mathcal{C}$ d'une règle $I \Rightarrow I_1$, $I_1 \neq \emptyset$ est défini comme suit [Kry02a] :

$$\mathcal{C}(I \Rightarrow I_1) = \{I \cup I_2 \Rightarrow I_3 | I_2, I_3 \subseteq I_1,\ I_2 \cap I_3 = \emptyset,\ I_3 \neq \emptyset\}$$

Soit R une règle dérivée, alors le support et la confiance de R sont supérieures ou égales respectivement, aux supports et aux confiances de toutes les règles qui couvre R. Formellement :

- $support(R) \geq max\{support(R')|R \in \mathcal{C}(R'))$.
- $confiance(R) \geq max\{confiance(R')|R \in \mathcal{C}(R'))$.

Inférence de règles basée sur la fermeture ($\mathcal{IRF}$)

Introduit par Pasquier [Pas00], ce mécanisme d'inférence est basé principalement sur le fait que le support d'un motif fréquent I est égal à celui de sa fermeture $\gamma(I)$. Soient I et I_1 deux motifs tels que $I \subset I_1$:

- $support(I \Rightarrow (I_1 - I)) = support(\gamma(I_1))$.
- $confiance(I \Rightarrow (I_1 - I)) = \frac{support(\gamma(I_1))}{support(\gamma(I))}$.
- Une règle $I \Rightarrow (I_1 - I)$ est valide si et seulement si $\gamma(I) \Rightarrow (\gamma(I_1) - \gamma(I))$ est valide.

2.4.2.2 Bases de règles d'implication

Les bases des règles d'implication entre deux ensembles d'items ont été étudiées essentiellement dans les domaines d'analyse de données et d'analyse formelle des concepts. Nous présentons deux bases de règles d'implication, à savoir la base de Guigues-Duquenne [GD86] pour les implications globales [1], et la base de Luxenburger [Lux91] pour les implications partielles [2] et leur adaptation dans le cadre de l'extraction des règles d'association exactes et approximatives.

Base de Guigues-Duquenne

La base de Guigues-Duquenne pour les implication globale est définie à partir de l'ensemble des motifs pseudo-fermés :

Définition 5 *Un motif $I \in \mathcal{I}$ est un motif pseudo-fermé s'il n'est pas fermé et s'il contient les fermetures de tous ses sous-ensembles qui sont des motifs pseudo-fermés. Ainsi, l'ensemble $\mathcal{PF}$ des motifs pseudo-fermés est défini comme suit :*

$$\mathcal{PF} = \{I \in \mathcal{I} \mid I \neq \gamma(I), \forall I_1 \subset I \text{ tel que } I_1 \in \mathcal{PF}, \gamma(I_1) \subset I \}$$

Ainsi, la base de Guigues-Duquenne pour les implications globales est définie comme suit :

1. Les règles d'implications globales sont des règles d'association exactes
2. les règles d'implication partielles sont des règles d'association approximatives

Définition 6 $\mathcal{BGD} = \{I \Rightarrow \gamma(I) \mid I \in \mathcal{PF}\}$

Telle qu'elle a été définie, la base $\mathcal{BGD}$ de règles d'implications ne tient pas compte de la mesure de support. Afin d'adapter la base $\mathcal{BGD}$ au cadre des règles d'association, Pasquier a pris en considération les éléments suivants [Pas00] :

- Le support des règles doit être déterminé ;
- La conclusion de la règle doit être diminuée de sa prémisse.

La base de Guigues-Duquenne adaptée au cadre des règles d'association est définie comme suit [Pas00] :

Définition 7 *Soit l'ensemble $\mathcal{PF}$ des motifs pseudo-fermés extraits d'un contexte d'extraction $\mathcal{K}$. La base de Duquenne-Guigues des règles d'association exactes est donnée par :* $\mathcal{BGD} = \{I \Rightarrow \gamma(I) - I \mid I \in \mathcal{PF}, I \text{ est fréquent}\}$.

Exemple 6 *Pour minsup = 3, la base de Guigues-Duquenne des règles d'association extraite du contexte du tableau 2.1 est donnée par le tableau 2.5.*

Règle	*support*	*confiance*
$\emptyset \Rightarrow b$	6	1
$ab \Rightarrow e$	4	1
$bde \Rightarrow a$	3	1

TABLE 2.5 – Base $\mathcal{BGD}$ des règles d'association associée au contexte du tableau 2.1 pour $minsup = 3$.

Afin de dériver l'ensemble des règles valides exactes à partir de la base $\mathcal{BGD}$, Guigues et Duquenne utilisent les axiomes d'Armstrong. Par exemple, la règle $c \Rightarrow b$ est dérivée à partir de la règle générique $\emptyset \Rightarrow b$. Toutefois, il n'y a aucune possibilité de déterminer le support de la règle $c \Rightarrow b$. En effet, à partir de l'ensemble des motifs fermés fréquents, le support de n'importe quel motif fréquent peut être déterminé. Toutefois, la base $\mathcal{BGD}$ ne contient que les motifs fermés fréquents b, abe et $abde$ qui ne sont pas des super-ensembles du motif bc. Par conséquent, le support du motif bc ne peut pas être déterminé. Ainsi, aucune information n'est disponible pour vérifier la validité de la règle $c \Rightarrow b$.

Ainsi, la base Guigues-Duquenne n'est pas informative. La dérivabilité est assurée par les axiomes d'Armstrong ($\mathcal{AA}$) permettant de retrouver toutes les règles exactes. Néanmoins,

elles ne sont pas toujours valides. Ainsi, ($\mathcal{AA}$) est un mécanisme d'inférence complet mais il n'est pas valide. Par conséquent, l'approche d'extraction de la base $\mathcal{BGD}$ est qualifiée d'approche avec perte d'information.

Base de Luxenburger

Dans [Lux91], Luxenburger montre que l'ensemble des règles d'implication, dont la prémisse est un motif fermé I et la conclusion est un motif fermé I_1 tels que $I \subset I_1$, constitue une base pour l'ensemble des règles d'implication partielles du contexte. Ces règles possèdent une confiance strictement inférieure à 1. Tout comme pour la base Guigues-Duquenne, l'adaptation de la base de Luxenburger dans le cadre des règles d'association nécessite la prise en considération du support des motifs fermés. La base adaptée est définie par Pasquier comme suit [Pas00] :

Définition 8 *Soit $\mathcal{MFF}$ l'ensemble des motifs fermés fréquents. La base de Luxenburger est donnée par : $\mathcal{BL} = \{I \Rightarrow I_1 \mid I, I_1 \in \mathcal{MFF},\ I \subset I_1,\ confiance(I \Rightarrow I_1) \geq minconf\}$*

Exemple 7 *Pour $minsup = 3$ et $minconf = 2/3$, la base de de Luxenburger des règles d'association extraite du contexte du tableau 2.1 est donnée par le tableau 2.6.*

Règle	*support*	*confiance*
$b \Rightarrow d$	4	2/3
$b \Rightarrow c$	4	2/3
$b \Rightarrow e$	5	5/6
$b \Rightarrow ae$	4	2/3
$bc \Rightarrow e$	3	3/4
$bd \Rightarrow ae$	3	3/4
$abe \Rightarrow d$	3	3/4
$be \Rightarrow a$	4	4/5

TABLE 2.6 – Base $\mathcal{BGD}$ des règles d'association associée au contexte du tableau 2.1 pour $minsup = 3$ et $minconf = 2/3$.

Dans [Pas00], Pasquier utilise le mécanisme d'inférence $\mathcal{IRF}$, afin de dériver l'ensemble des règles approximatives redondantes ainsi que leurs valeurs exactes de support et de confiance. En effet, il a été montré que pour toute règle approximative valide $I \Rightarrow (I_1 - I)$, il

existe une règle de $\mathcal{BL}$, $I' \Rightarrow (I'_1 - I')$ telle que $\gamma(I) = \gamma(I')$ et $\gamma(I_1) = \gamma(I'_1)$. Ceci s'explique par fait qu'il existe une règle de $\mathcal{BL}$ pour chaque paire de motifs fermés fréquents. Ainsi, les supports et les confiances des règles approximatives valides peuvent être calculés à partir des supports et des confiances des règles de $\mathcal{BL}$. Par exemple, le support et la confiance de la règle X peuvent être dérivés du support et de la confiance de la règle Y.

2.4.2.3 Bases de règles avec perte d'information

Une base générique est dite *sans perte d'information*, si elle satisfait les deux conditions suivantes : la dérivabilité et l'informativité. Ainsi, une base générique est dite *avec perte d'information*, si elle ne satisfait pas l'une de ces deux conditions. Dans ce qui suit, nous présentons les bases génériques des règles d'association *avec perte d'information.*

Base Representative $\mathcal{BR}$

En utilisant l'opérateur de couverture $\mathcal{C}$, Kryszkiewicz [Kry98] introduit une base générique des règles d'association, appelée base représentative $\mathcal{BR}$, définie comme suit :

Définition 9 *Soit $\mathcal{AR}$ l'ensemble de toutes les règles d'association valides pouvant être extraites à partir d'un contexte d'extraction $\mathcal{K}$ pour un seuil de support minsup et un seuil de confiance minconf. La base représentative est donnée par :* $\mathcal{BR} = \{R \in \mathcal{AR} \mid \nexists R' \in \mathcal{AR},\ R \neq R',\ R \in \mathcal{C}(R)\}$

Exemple 8 *Pour minsup = 3 et minconf = 3/4, la base représentative $\mathcal{BR}$ des règles d'association extraite du contexte donné par le tableau 2.1, est donnée par le tableau 2.7.*

Dans [Kry98], Kryszkiewicz montre que l'application de l'opérateur de couverture $\mathcal{C}$ sur les règles de $\mathcal{BR}$ permet de dériver l'ensemble des règles valides. Par exemple, les règles valides $\emptyset \Rightarrow d$, $\emptyset \Rightarrow b$, $b \Rightarrow d$ et $d \Rightarrow b$ peuvent être dérivées à partir de la règle $\emptyset \Rightarrow bd$, suite à l'application de l'opérateur de couverture $\mathcal{C}$. Bien que la base $\mathcal{BR}$ permettait de dériver la totalité des règles d'association valides, elle n'est pas informative. En effet, il n'est pas toujours possible de determiner avec exactitude le support et la confiance de toutes les règles dérivées. Par exemple, il n'y a aucune possibilité de calculer la confiance de la règle $b \Rightarrow d$ puisque le support de l'item b n'est pas déterminé.

Règles d'association Non Redondantes $\mathcal{BNR}$

Afin d'extraire un sous ensemble générique des règles d'association, Zaki s'est basé sur la définition de redondance suivante [Zak00] :

Règle	*support*	$confiance$
$a \Rightarrow bde$	3	3/4
$d \Rightarrow abe$	3	3/4
$c \Rightarrow be$	3	3/4
$\emptyset \Rightarrow bc$	4	2/3
$\emptyset \Rightarrow bd$	4	2/3
$\emptyset \Rightarrow abe$	4	2/3

TABLE 2.7 – Base $\mathcal{BR}$ des règles d'association associée au contexte du tableau 2.1 pour $minsup = 3$ et $minconf = 2/3$.

Définition 10 *Soit $\mathcal{AR}$ l'ensemble de toute les règles d'association valides pouvant être extraites à partir d'un contexte d'extraction $\mathcal{K}$ pour un seuil de support minsup et un seuil de confiance $minconf$. Une règle $R' : I' \Rightarrow I'_1 \in \mathcal{AR}$ est redondante par rapport à une règle $R : I \Rightarrow I_1 \in \mathcal{AR}$, notée $R \preceq R'$, si et seulement si :*

1. *$support(R) = support(R')$ et $confiance(R) = confiance(R')$;*
2. *$I \subseteq I'$ et $I_1 \subseteq I'_1$.*

Ainsi, une règle R est non redondante si et seulement si il n'existe pas une règle R' tel que $R' \preceq R$. Par conséquent, R' peut être dérivée en ajoutant des items à la partie prémisse et/ou à la partie conclusion de la règle R.

Exemple 9 *Considérons les règles valides $e \Rightarrow a$, $e \Rightarrow ab$ et $be \Rightarrow a$ extraites à partir du contexte d'extraction $\mathcal{K}$. Les règles $e \Rightarrow ab$ et $be \Rightarrow a$ sont redondantes par rapport à $e \Rightarrow a$ puisque $support(ae) = support(abe) = 4$ et $confiance(e \Rightarrow a) = confiance(be \Rightarrow a) = confiance(ab \Rightarrow e) = 4/5$.*

En se basant sur la notion de redondance, Zaki introduit une base générique des règles d'association, appelé $\mathcal{BNR}$ [Zak00].

Définition 11 *La base $\mathcal{BNR}$ est donnée par : $\mathcal{BNR}$= { $R \in \mathcal{AR} \mid \nexists R' \in \mathcal{AR}, R' \preceq R$}*

Puisque les règles $I \Rightarrow I_1$ et $\gamma(I) \Rightarrow \gamma(I_1)$ ont le même support et la même confiance, Zaki montre dans [Zak00] qu'il est suffisant de considérer les règles entre les motifs fermés fréquents. Il montre aussi qu'il suffit de considérer uniquement les règles entre les motifs fermés fréquents adjacents, puisque les autres règles peuvent être inférées en utilisant la transitivité de la confiance $\mathcal{TC}$.

Règles exactes minimales		
Règle $[I^{\top}, I^{\perp}]$	*support*	*confiance*
$de \Rightarrow a$ $[b, \emptyset]$	3	1
$a \Rightarrow e$ $[b, \emptyset]$	4	1
$e \Rightarrow b$ $[\emptyset, \emptyset]$	5	1
$d \Rightarrow b$ $[\emptyset, \emptyset]$	4	1
$c \Rightarrow b$ $[\emptyset, \emptyset]$	4	1

Règles approximatives minimales		
Règle $[I^{\top}, I^{\perp}]$	*support*	*confiance*
$b \Rightarrow e$ $[\emptyset, \emptyset]$	5	5/6
$e \Rightarrow a$ $[b, \emptyset]$	4	4/5
$c \Rightarrow e$ $[b, \emptyset]$	4	3/4
$a \Rightarrow d$ $[be, \emptyset]$	3	3/4
$d \Rightarrow a$ $[b, e]$	4	3/4
$d \Rightarrow e$ $[b, a]$	4	3/4
$b \Rightarrow d$ $[\emptyset, \emptyset]$	4	2/3
$b \Rightarrow c$ $[\emptyset, \emptyset]$	4	2/3

TABLE 2.8 – Base générique $\mathcal{BNR}$ extraite du contexte d'extraction illustré par le tableau 2.1, pour *minsup*=2 et *minconf* =2/3

La base $\mathcal{BNR}$ contient deux types de règles à savoir :

- **La règle exacte minimale** : elle traduit une corrélation entre deux générateurs minimaux disjoints d'un même motif fermé fréquent ou une corrélation entre un générateur minimal d'un motif fermé fréquent I et un générateur minimal d'un motif fermé fréquent I' successeur immédiat de I.
- **La règle approximative minimale** : elle traduit une corrélation entre un générateur minimal d'un motif fermé fréquent I et un générateur minimal d'un motif fermé fréquent I' prédécesseur immédiat de I.

Exemple 10 *Pour minsup* $= 3$ *et minconf* $= 2/3$*, la base* $\mathcal{BNR}$ *des règles d'association extraite du contexte du tableau 2.1 est donnée par le tableau 2.8.*

Afin de dériver l'ensemble de règles redondantes valides, Zaki utilise la transitivité de la confiance $\mathcal{TC}$ ainsi que l'axiome d'augmentation. Cependant, l'augmentation d'une règle de la base $\mathcal{BNR}$ avec n'importe quel item peut mener à la dérivation d'une règle non valide. Par exemple en augmentant la conclusion de la règle $c \Rightarrow e$ par l'item a, la règle obtenue est $c \Rightarrow ae$. Toutefois, cette règle n'est pas valide car le motif ace n'est pas fréquent. Ainsi, dans [ZP03], Zaki propose des règles non redondantes de la forme $I_1 \Rightarrow I_2[I^{\top}, I^{\perp}]$, où $I^{\top}$ est l'ensemble d'items qui peuvent être ajoutés à I_1 ou à I_2 pour donner une règle redondante. Cependant, $I^{\perp}$ est l'ensemble d'items qui peuvent être ajoutés seulement à I_1 lorsque la règle est exacte ou seulement à I_2 lorsque la règle est approximative. Par exemple, l'ensemble $I^{\top}$ associé à la règle $c \Rightarrow e$, contient l'item b alors

que $I^{\perp} = \emptyset$, ce qui signifie l'item b peut être ajouté à la prémisse ou à la conclusion de la règle $c \Rightarrow e$.

Une telle présentation de règle garantit à la base $\mathcal{BNR}$ d'être informative. En effet, les valeurs de support et de confiance d'une règle R dérivée suite à l'application de l'axiome d'augmentation sur une règle R' sont identiques à celles de R'. Par ailleurs, une règle $R : I \Rightarrow I_2$ dérivée des deux règles $R' : I \Rightarrow I_1$ et $R'' : I_1 \Rightarrow I_2$, suite à l'application de la transitivité de confiance (TC) a une valeur de support identique à celle de R'' et une valeur de confiance égale au produit des confiances de R' et R''. Ainsi, pour chaque règle redondante le support et la confiance peuvent être déterminées avec exactitude. Cependant, la base $\mathcal{BNR}$ ne permet pas de générer toutes les règles redondantes valides. Par exemple, la règle valide $ce \Rightarrow b$ ne peut être dérivée ni par l'axiome d'augmentation ni par la transitivité de confiance. Ainsi, la base $\mathcal{BNR}$ ne satisfait pas la condition de dérivabilité.

2.4.2.4 Bases de règles sans perte d'information

Dans ce qui suit, nous présentons les bases génériques des règles d'association *sans perte d'information* (*i.e.,* elle satisfont la dérivabilité et l'informativité).

Règles d'association informatives

Dans [BPT+00], Bastide *et al.,* proposent un couple de sous-ensemble générique de règles en se basant sur la définition de redondance suivante :

Définition 12 *Soit $\mathcal{AR}$ l'ensemble de toute les règles d'association valides pouvant être extraites à partir d'un contexte d'extraction $\mathcal{K}$ pour un seuil de support minsup et un seuil de confiance minconf. Une règle $R' : I' \Rightarrow I'_1 \in \mathcal{AR}$ est redondante par rapport à une règle $R : I \Rightarrow I_1 \in \mathcal{AR}$, noté $R \preceq R'$, si et seulement si :*

1. *$I \subseteq I'$ et $I'_1 \subset I_1$;*
2. *$support(R) = support(R')$ et $confiance(R) = confiance(R')$.*

Exemple 11 *Considérons les deux règles $R : ab \Rightarrow de$ et $R' : ab \Rightarrow d$ extraites à partir du contexte d'extraction illustré par le tableau 2.1. Le support du motif abde est égal au support du motif abe puisqu'ils appartiennent à la même classe d'équivalence. Ainsi, $support(R) = support(R')$. De plus, $confiance(R) = confiance(R')$, puisqu'elles ont la même prémisse. En effet, $confiance(R) = \frac{support(R)}{support(ab)} = \frac{support(R')}{support(ab)} = confiance(R')$. Par conséquent, R' est redondante par rapport à R puisqu'elles ont les mêmes valeurs de support et de confiance, et $d \subset de$*

Par conséquent, une règle $R : I \Rightarrow I_1$ est dite *non redondante*, s'il n'existe pas une règle $R' : I' \Rightarrow I'_1$ tel que $I \subseteq I'$, $I'_1 \subset I_1$, $support(R) = support(R')$ et $confiance(R) = confiance(R')$. Ainsi, une règle *non redondante* possède une prémisse minimale et une conclusion maximale.

En se basant sur la définition 12 de redondance, Bastide *et al* définissent le couple suivant [BPT+00] :

1. La base générique des règles exactes est définie comme suit :

   Définition 13 *Soit $\mathcal{MFF}$ l'ensemble des motifs fermés fréquents extraits d'un contexte d'extraction $\mathcal{K}$. Soit MG_I l'ensemble des générateurs minimaux d'un motif fermé fréquent $I \in \mathcal{MFF}$. La base générique des règles exactes $\mathcal{BG}$ est donnée par :*
   $\mathcal{BG}= \{R : I_1 \Rightarrow I - I_1 \mid I \in \mathcal{MFF},\ I_1 \in MG_I \text{ et } I_1 \neq I\}$ [3]

2. La réduction transitive de la base informative des règles approximatives est définie comme suit :

   Définition 14 *Soient $\mathcal{MFF}$ l'ensemble des motifs fermés fréquents et $\mathcal{G}$ l'ensemble des générateurs minimaux extraits d'un contexte d'extraction $\mathcal{K}$. La réduction transitive $\mathcal{RI}$ est donnée par :*
   $\mathcal{RI}= \{R : I_1 \Rightarrow I - I_1 \mid I \in \mathcal{MFF},\ I_1 \in \mathcal{G},\ \gamma(I_1) \subset I,\ \nexists I' \in \mathcal{MFF},\ I' \subset I,\ et\ confiance(R) \geq minconf\}$

A partir du *treillis de l'Iceberg de Galois*, dans lequel chaque motif fermé est décoré par la liste de ses générateurs minimaux, la base ($\mathcal{BG}$, $\mathcal{RI}$) peut être obtenue directement. En effet, les règles génériques exactes représentent des corrélations "intra-nœud" avec une confiance égale à 1. Par exemple, en se référant au *treillis de l'Iceberg de Galois* présenté dans la figure 2.2, la règle exacte $ad \Rightarrow be$ est générée à partir du motif fermé *abde*. Inversement, les règles génériques approximatives représentent des corrélations "inter-nœud" avec une confiance supérieure ou égale à $minconf$. Par exemple, la règle approximative $c \Rightarrow be$ est générée à partir des deux classes d'équivalence associées respectivement aux motifs fermés fréquents, *bc* ayant *c* comme générateur minimal, et *bce*.

Exemple 12 *Pour $minsup = 3$ et $minconf = 2/3$, le couple ($\mathcal{BG}$, $\mathcal{RI}$) généré à partir du contexte illustré par le tableau 2.1 est donnée par le tableau 2.3.*

Dans [BPT+00], Bastide *et al.,* ont montré qu'en appliquant le mécanisme d'inférence $\mathcal{IRF}$ sur les règles de la base $\mathcal{BG}$, toutes les règles valides exactes redondantes peuvent

3. La condition $I_1 \neq I$ permet l'elimination des règles non informatives de la forme $I_1 \Rightarrow \emptyset$.

Règles $\mathcal{BG}$		
Règle	support	confiance
$ad \Rightarrow be$	3	1
$de \Rightarrow ab$	3	1
$ce \Rightarrow b$	3	1
$c \Rightarrow b$	4	1
$e \Rightarrow b$	5	1
$d \Rightarrow b$	4	1
$a \Rightarrow be$	4	1

Règles $\mathcal{RI}$		
Règle	support	confiance
$b \Rightarrow e$	5	5/6
$d \Rightarrow abe$	3	3/4
$c \Rightarrow be$	4	3/4
$a \Rightarrow bde$	3	3/4
$b \Rightarrow d$	4	2/3
$b \Rightarrow c$	4	2/3

FIGURE 2.3 – Le couple ($\mathcal{BG}$, $\mathcal{RI}$) associé au contexte d'extraction donné par le tableau 2.1 pour $minsup$=3 et $minconf$=2/3

être dérivées avec leurs supports ainsi que leurs confiances. En effet, tous les motifs fermés fréquents peuvent être reconstruits en joignant la prémisse et la conclusion des règles génériques de $\mathcal{BG}$. Ainsi, le support de tout motif fréquent peut être déterminé. Il a été également montré qu'en appliquant l'opérateur de couverture $\mathcal{C}$ sur les règles de $\mathcal{RI}$ toutes les règles valides approximatives redondantes peuvent être dérivées avec leurs supports ainsi que leurs confiances. Par exemple, en appliquant l'opérateur de couverture $\mathcal{C}$ sur la règle $c \Rightarrow be$, les règles suivantes sont dérivées $c \Rightarrow e$, $ce \Rightarrow b$ et $bc \Rightarrow e$ ayant un support égal 3 et confiance 3/4.

Ainsi, le couple ($\mathcal{BG}$, $\mathcal{RI}$) vérifie les contraintes de l'informativité et de la dérivabilité. Par conséquent, la base ($\mathcal{BG}$, $\mathcal{RI}$) est qualifiée de base sans perte d'information.

Base générique informative

Dans [YGN09], *Ben Yahia et al.,* proposent une base générique $\mathcal{IGB}$ en considérant d'une part qu'une règle dérivée ne peut pas présenter une prémisse plus petit que celle de sa règle générique associée, *i.e.,* à partir de laquelle elle peut être dérivée. D'autre part, une règle dérivée ne peut pas présenter une conclusion plus grande que celle de sa règle générique associée. La base générique $\mathcal{IGB}$ est définie comme suit :

Définition 15 *Soient $\mathcal{MFF}$ l'ensemble des motifs fermés fréquents et $\mathcal{G}$ l'ensemble des générateurs minimaux extraits d'un contexte d'extraction $\mathcal{K}$. La base générique informative $\mathcal{IGB}$ est donnée par :*
$\mathcal{IGB}$= $\{R : I_1 \Rightarrow I - I_1 \mid I \in \mathcal{MFF}, I \neq \emptyset, I_1 \in \mathcal{G}, \gamma(I_1) \subseteq I, confiance(R) \geq minconf$ *et* $\nexists I_2 \mid I_2 \subset I_1, confiance(I_2 \Rightarrow I - I_2) \geq minconf\}$

Exemple 13 *Pour $minsup = 3$ et $minconf = 2/3$, la base $\mathcal{IGB}$ des règles d'association extraite du contexte du tableau 2.1 est donnée par le tableau 2.9.*

Règle	*support*	*confiance*
$a \Rightarrow bde$	3	3/4
$d \Rightarrow abe$	3	3/4
$c \Rightarrow be$	3	3/4
$\emptyset \Rightarrow bc$	4	2/3
$\emptyset \Rightarrow abe$	4	2/3
$\emptyset \Rightarrow bd$	5	5/6
$\emptyset \Rightarrow be$	5	5/6
$\emptyset \Rightarrow b$	6	1

TABLE 2.9 – Base $\mathcal{IGB}$ des règles d'association associée au contexte d'extraction illustré par le tableau 2.1 pour $minsup = 3$ et $minconf = 2/3$.

Afin de dériver l'ensemble de toutes les règles redondantes valides, *Ben Yahia et al.,* proposent le système axiomatique suivant [YGN09] :

- Réfléxivité conditionnelle : Si $I \Rightarrow I' \in \mathcal{IGB}$ et $I \neq \emptyset$ alors $I \Rightarrow I'$ est valide.
- Augmentation : Si $I \Rightarrow I' \in \mathcal{IGB}$ et $I'' \subset I'$, alors $I \cup I'' \Rightarrow I'$ est valide et $confiance(I'' \Rightarrow I') \geq confiance(I \Rightarrow I')$.
- Décomposition : Soit $I \Rightarrow I'$ une règle dérivée en appliquant la réfléxivité conditionnelle ou l'augmentation sur une règle de $\mathcal{IGB}$. Si $I'' \subset I'$ et $\gamma(I \cup I'') = I \cup I'$ alors, la règle $I \Rightarrow I'$ est valide.

Dans [YGN09], *Ben Yahia et al.,* montrent que l'application de ce système axiomatique sur l'ensemble des règles de $\mathcal{IGB}$ ne permet de dériver que les règles redondantes valides. Ainsi, la base $\mathcal{IGB}$ satisfait la condition de dérivabilité. Il a été montré également que $\mathcal{IGB}$ est informative. En effet, chaque règle d'association de la base $\mathcal{IGB}$ représente une corrélation entre un motif fermé fréquent non vide I et les plus petits générateurs minimaux associés aux motifs fermés fréquents inclus dans I. Ainsi, l'ensemble de tous les motifs fermés fréquents peut être retrouvé en joignant la prémisse et la conclusion des règles de la base $\mathcal{IGB}$. Puisque, le support d'un motif fréquent est égal à celui du motif fermé fréquent le plus petit qui le contient, alors la fermeture et le support de chaque motif

peuvent être calculés à partir de la base $\mathcal{IGB}$. Par conséquent, le support et la confiance de toutes les règles redondantes peuvent être déterminés avec exactitude.

2.4.3 Synthèse des différentes bases génériques

A la lumière de ce qui a été présenté précédemment, nous remarquons que les bases génériques présentent des différences que nous organisons selon les critères suivants :

- **Forme des règles :** Ce critère décrit la nature de la prémisse et celle de la conclusion des règles de la base générique.
- **Mécanisme d'inférence :** Ce critère indique le mécanisme d'inférence qui doit être appliqué sur la base générique afin de générer l'ensemble des règles d'association valides.
- **Dérivabilité :** Ce critère permet de vérifier la dérivabilité de la base générique, *i.e.,* vérifier si l'application du mécanisme d'inférence sur la base générique permet de dériver toutes et seulement les règles d'association valides.
- **Informativité :** Ce critère permet de vérifier si la base générique permet de retrouver avec exactitude le support et la confiance des règles d'association valides.
- **Règles dérivées :** Ce critère décrit la nature des règles d'association qui peuvent être dérivées à partir de la base générique.

Base	Forme des règles	Mécanisme d'inférence	Dérivabilité	Informativité	Règles dérivées
$\mathcal{BGD}$	pseudo fermé → fermé	$\mathcal{AA}$	non	non	règles exactes
$\mathcal{BL}$	fermé → fermé	$\mathcal{IRF}$	oui	oui	règles approximatives
$\mathcal{BR}$	gén.minimal → fermé	$\mathcal{C}$	oui	non	règles exactes et règles approximatives
$\mathcal{BNR}$	gén.minimal → fermé	$\mathcal{TC}$ + Augmentation	non	oui	règles exactes et règles approximatives
$(\mathcal{BG}, \mathcal{RI})$	gén.minimal → fermé	$\mathcal{IRF}$	oui	oui	règles exactes et règles approximatives
$\mathcal{IGB}$	gén.minimal → fermé	Réfléxivité + Augmentation + Décomposition	oui	oui	règles exactes et règles approximatives

TABLE 2.10 – Comparaison des principales bases génériques dans la littérature

Le tableau 2.10 présente une étude comparative entre les différentes base génériques selon les critères décrits précédemment. Nous remarquons :

1. Les règles des bases génériques $\mathcal{BR}$, $\mathcal{BNR}$, $(\mathcal{BG}, \mathcal{RI})$ et $\mathcal{IGB}$ possèdent une prémisse minimale et une conclusion maximale puisqu'elles traduisent des corrélations entre les générateurs minimaux et les motifs fermés fréquents.
2. Seules les bases $\mathcal{BL}$, $\mathcal{BR}$, $(\mathcal{BG}, \mathcal{RI})$ et $\mathcal{IGB}$ vérifient la dérivabilité. Cependant, la base $\mathcal{BL}$ ne permet pas la dérivation des règles exactes valides.
3. Seules les bases $\mathcal{BL}$, $\mathcal{BNR}$, $(\mathcal{BG}, \mathcal{RI})$ et $\mathcal{IGB}$ sont informatives. Cependant, la base $\mathcal{BL}$ permet de retrouver seulement le support et la confiance des règles approximatives valides.
4. Seules les bases $(\mathcal{BG}, \mathcal{RI})$ et $\mathcal{IGB}$ permettent la dérivation de toutes les règles valides exactes et approximatives sans perte d'information. La non perte d'information est due d'une part au fait que ces deux deux bases sont informatives et d'autre part, au fait que les mécanismes d'inférence utilisés pour la dérivation des règles d'association sont valides et complets.

2.5 Extraction de règles d'association sous contraintes

La recherche des règles d'association sous contraintes [JAG99, JB02] est une approche complémentaire aux approches basées sur les deux mesures support et confiance, qui introduit des contraintes particulières dans le processus de fouille de données, afin de sélectionner un sous-ensemble restreint de règles pertinentes. Avec les contraintes, l'utilisateur a la possibilité de spécifier les caractéristiques des règles d'association recherchées. Une contrainte est généralement un prédicat appliqué sur les règles. Seules les règles vérifiant le prédicat sont retenues dans le résultat final. Cette vérification se fait après la génération des règles lorsque les contraintes ne peuvent être traduites sur les motifs (par exemple la contrainte de confiance minimale). Dans ce cas, les contraintes sont utilisées pour filtrer l'ensemble des règles extraites. Cependant, lorsque les contraintes sur les règles peuvent être traduites sur les motifs (par exemple la contrainte de support minimal), la vérification de la satisfaction des contraintes se fait lors de l'extraction des motifs afin d'élaguer certaines parties de l'espace de recherche. Dans ce cas, un nombre important d'opérations de calcul peuvent être évitées permettant l'amélioration du temps d'exécution et la diminution des ressources utilisées. Plusieurs familles de contraintes peuvent être intégrées lors de l'extraction des motifs. Les deux principales familles de contraintes qui ont été

étudiées sont les contraintes *anti-monotones* et *monotones* [BJ10, JB02].

1. *anti-monotonicité* : étant donné un motif I, une contrainte $\mathcal{C}_{AM}$ est anti-monotone si $\forall I' \subseteq I, \mathcal{C}_{AM}(I) \Rightarrow \mathcal{C}_{AM}(I')$. La contrainte $anti-monotone$ la plus connue est la contrainte de fréquence (support). Elle est intégrée dans l'algorithme APRIORI avec l'interprétation suivante : si un motif ne vérifie pas $\mathcal{C}_{sup}$ alors, tous ses sur-ensembles ne vérifient pas $\mathcal{C}_{sup}$.
2. *monotonicité* : étant donné un motif I, une contrainte $\mathcal{C}_M$ est monotone si $\forall I \subseteq I'$, $\mathcal{C}_M(I) \Rightarrow \mathcal{C}_M(I')$. Ainsi, une contrainte monotone est la négation d'une contrainte anti-monotone.

Dans ce qui suit, nous présentons les principales approches proposées pour la recherche des règles d'association sous contraintes. Ng *et al.,*[NLHP98] et Srikant *et al.,* [SVA97] ont proposé des contraintes syntaxiques qui imposent ou empêchent la présence d'un ensemble d'items dans les motifs extraits. Bayardo *et al.*[JAG99] ont développé un algorithme, appelé DENSE-MINER, qui intègre deux contraintes. La première contrainte impose la présence d'un motif, fixé par l'utilisateur, dans les conclusions des règles générées. La deuxième contrainte est basée sur une mesure de qualité appelée "*amélioration*" [JAG99] :

Définition 16 *Soient $R : I \Rightarrow I_1$ et $R' : I' \Rightarrow I_1$ deux règles d'association. Si $I' \subset I$ alors, R' est une généralisation de R. L'amélioration de la règle R est la différence minimale entre sa confiance et la confiance de toute ses règles générales. Formellement,* $amélioration(I \Rightarrow I_1) = min(\forall I' \subset I,\ confiance(I \Rightarrow I_1) - confiance(I' \Rightarrow I_1))$

Selon Bayardo *et al.,* [JAG99], si la valeur de l'$amélioration(R)$ est positive alors, R est intéressante car elle apporte plus d'informations que l'ensemble de ses règles générales. Cependant, si la valeur de l'*amélioration* de R est négative alors, il existe au moins une règle R' plus générale (par conséquent plus intéressante) que R telle que $confiance(R') \geq confiance(R)$.

Dans [JB02], les auteurs montrent que la recherche sous contraintes des règles d'association est liée aux bases de données inductives, dont le langage de requête spécifie d'une manière déclarative des contraintes permettant d'extraire des données particulières mais aussi des motifs présents dans ces données. Ainsi, les algorithmes de génération de règles d'association doivent être intégrés dans les bases inductives afin de pouvoir sélectionner les règles sous une conjonction de contraintes anti-monotones et monotones. En pratique, le nombre de règles extraites sous contraintes reste important. Une des manière de réduire ce nombre de règles est d'extraire les *top-k* règles d'association avec k un nombre fixé

par l'utilisateur. Cette approche dite des *top-k* règles revient à détecter les k meilleures règles qui optimisent une mesure de qualité [SC07, FVWT12, Web11] outre la satisfaction d'éventuelles autres contraintes.

2.6 Conclusion

Introduite par Agrawal *et al.,* [AIS93b], l'extraction des règles d'association est l'un problème traditionnel de la fouille de données. Étant donné le nombre exorbitant des règles qui peuvent être extraites, de nombreux approches de sélection de règles ont été proposées. Dans ce chapitre, nous avons discuté les caractéristiques des approches dédiées à l'extraction des bases génériques ainsi que les approches d'extraction sous-contraintes. Les approches dédiées à l'extraction des bases génériques se sont basées essentiellement sur la théorie de l'analyse formelle des concepts afin d'éliminer les règles redondantes. Le nombre de règles est alors passé de quelques millions à des milliers. Avec un tel nombre, l'expert reste toujours incapable d'interpréter et exploiter les règles extraites. Ainsi, de nouveaux filtres sont nécessaires pour retenir un ensemble réduit de règles pertinentes et interprétables par l'expert. Parmi ces filtres, nous citons les mesures de qualité qui servent essentiellement à filtrer et ordonner les règles selon les préférences des experts. Cependant, les mesures de qualité peuvent évaluer différemment les règles. Il se pose ainsi clairement le problème d'agrégation des différentes évaluations, lorsqu'un ou plusieurs experts utilisent des mesures différentes. Dans le chapitre suivant, nous présentons les travaux dédiés à l'agrégation des préférences et nous montrons leur connexion avec l'approche d'extraction de règles d'association.

Points clés

- Nous avons présenté les notions de bases associées aux règles d'association et le problème de leurs extraction en utilisant les deux mesures : *support* et *confiance*.
- Nous avons passé en revue et analysé les différentes approches qui permettent de réduire le nombre de règles générées en se basant sur l'analyse formelle de concepts (AFC).
- Dans la conclusion, nous avons mentionné l'intérêt de compléter le support et la confiance par d'autres mesures de qualité afin de réduire davantage l'ensemble des règles présentées à l'expert.

Chapitre 3

Fouille de données et Préférences

Sommaire

3.1 Introduction 41

3.2 Approches d'agrégation des préférences 42

3.2.1 Systèmes de vote 43

3.2.2 Dominance de Pareto 51

3.2.3 Synthèse des différentes approches d'agrégation de préférences . 52

3.3 Recherche des motifs basée sur les préférences 54

3.3.1 Motifs les plus informatifs 54

3.3.2 Motifs et dominance de Pareto 58

3.3.3 Graphes et dominance de Pareto 62

3.3.4 Synthèse des approches de recherche des motifs basées sur les préférences 64

3.3.5 Agrégation de mesures d'intérêt de règles d'association 65

3.4 Conclusion 66

3.1 Introduction

Dans le chapitre précédent, nous avons étudié la technique d'extraction des règles d'association basée sur les deux mesures *support* et $confiance$. Toutefois, cette technique produit en pratique un nombre prohibitif de règles qui sont très difficiles à gérer et dont la plupart sont redondantes et parfois sans intérêt. Ce problème s'amplifie lorsque l'on diminue le seuil du support, ce qui est essentiel pour découvrir les spécificités des données. Bien que des techniques telles que l'extraction des bases génériques, permettaient de ré-

duire de manière conséquente le nombre de règles, elles ne garantissent pas que les règles sélectionnées soient pertinentes pour l'expert. En effet, l'utilisation du support peut éliminer des règles ayant un faible support alors que certaines peuvent avoir une forte confiance présentant un réel intérêt [Azé03]. Ainsi, le recours exclusif au couple (*support, confiance*) n'est pas suffisant pour déterminer les règles d'association intéressantes et a été remis en cause dans plusieurs travaux [SM02]. Pour pallier les faiblesses de ces deux mesures, diverses caractéristiques ont été proposées pour concevoir d'autres mesures de qualité dont l'objectif est d'associer une valeur numérique à une règle permettant de quantifier son intérêt. En conséquence, dans les deux dernières décennies, une panoplie de mesures de qualité obéissant à différentes sémantiques a été proposée dans la littérature (environ une soixantaine) permettant de filtrer et d'ordonner les règles d'association.

Néanmoins, le problème est loin d'être résolu car les mesures proposées sont très hétérogènes. En effet, une règle peut être considérée pertinente selon une mesure et non pertinente selon une autre. Ce genre d'observations permet de soulever de nouvelles questions : Comment déterminer les meilleures règles en présence de plusieurs experts simultanément, ayant chacun une préférence pour une mesure de qualité ? Comment déterminer les règles les plus pertinentes lorsqu'un expert a des préférences pour différentes mesures ?

Dans ce contexte, *les systèmes d'agrégation de préférences* [Kac11], [Rol13], [Roy72] peuvent être utiles pour répondre à de telles questions. En effet, ces systèmes présentent des méthodes permettant d'agréger les avis exprimés par un ensemble d'individus concernant différentes alternatives de façon à déterminer une alternative "gagnante" ou encore à classer par ordre de préférence les différentes alternatives. Il est aisé de voir le lien entre l'agrégation de préférences et la détermination des meilleures règles selon plusieurs mesures grâce aux correspondances suivantes : les individus peuvent être référés aux mesures (aussi aux experts) et les alternatives peuvent être référées aux règles d'association.

Ce chapitre est organisé en deux sections. Dans la première section, nous allons présenter les différentes approches d'agrégation des préférences et nous montrons leur connexion avec l'approche d'extraction de règles d'association. Dans la deuxième section, nous allons présenter et discuter les approches de sélection de motifs utilisant les préférences.

3.2 Approches d'agrégation des préférences

Une approche très générale pour agréger les préférences des individus pour un ensemble d'alternatives est la suivante : demander aux individus de classer les alternatives par ordre de préférences, puis choisir, en fonction des classements produits par ces individus,

la meilleure alternative.

Exemple 14 *Le tableau 3.1, illustre un exemple de classement de quatres alternatives par huit individus selon leurs préférences. Chaque colonne x_i indique les rangs attribués aux alternatives par un individu x_i. Cet exemple sera utilisé tout au long de cette section.*

Cette approche peut être référée à une opération de vote sur les alternatives en considérant les individus comme des votants et les alternatives comme des candidats. Elle peut faire référence aussi à la notion de *dominance de Pareto* [Mat91, KLP75] dont le principe consiste à préférer une alternative a_1 à une autre a_2 si pour tous les individus, a_1 est préférée à a_2.

	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8
a_1	1	1	1	4	4	3	3	4
a_2	2	2	2	1	1	4	4	3
a_3	3	3	3	3	3	1	1	2
a_4	4	4	4	2	2	2	2	1

TABLE 3.1 – Un exemple d'évaluation de quatre alternatives a_1, a_2, a_3 et a_4 par huit individus x_1, x_2, x_3, x_4, x_5, x_6, x_7, et x_8

Dans ce qui suit, nous allons passer en revue les principales approches d'agrégation des préférences à savoir *les systèmes de vote* et la notion de *dominance de Pareto*.

3.2.1 Systèmes de vote

L'objectif principal des systèmes de vote est de répondre à la question suivante : comment agréger des préférences individuelles en un choix collectif ? Cette question posée initialement dans le cadre des sciences politiques par des mathématiciens français comme Borda [Bor] et Condorcet [Con85] conduit naturellement à l'étude des systèmes de vote. Au cours des années cinquante, ce sont les économistes qui ont fait revivre ces systèmes, afin de trouver une méthode optimale pour les allocations des ressources dont l'objectif était de distribuer les ressources financières sur un ensemble d'individus. Les systèmes de vote décrivent la manière dont les avis exprimés par un ensemble de votants concernant divers candidats sont agrégés afin de déterminer un élu ou de classer les candidats. Une correspondance entre la décision dans les systèmes de vote et la décision dans l'extraction

des règles d'association selon plusieurs mesures est en effet possible en considérant les règles comme des candidats et les mesures comme des votants.

Les systèmes de vote peuvent être classés en deux catégories :

1. Systèmes de vote par majorité.
2. Systèmes de vote par scorage.

3.2.1.1 Systèmes de vote par majorité

La référence principale des systèmes de vote est la méthode majoritaire, élaborée à la fin du 18ème siècle, qui est considérée par beaucoup comme l'origine des systèmes de vote. Ces systèmes traitent seulement les candidats, qui occupent le premier rang. Un candidat c_1 est majoritairement préféré à un candidat c_2 si le nombre de votants qui préfère c_1 est supérieur au nombre de votants qui préfère c_2. En correspondance avec l'extraction des règles selon plusieurs mesures, une règle r_1 est majoritairement préférée à une règle r_2, si le nombre de mesures qui classent r_1 avant r_2 est supérieur au nombre de mesures qui classent r_2 avant r_1.

Système de vote majoritaire simple

Le système de vote majoritaire simple (*c.f.,* [FG82]), consiste à déclarer le candidat ayant obtenu le plus de voix sur un territoire donné vainqueur. Une majorité relative de voix suffit pour gagner une élection, *i.e.,* qu'il est possible que le candidat élu recueille moins de la moitié des voix exprimées. En appliquant ce système sur notre exemple illustré par le tableau 3.1, nous obtenons :

- L'alternative a_1 est la plus pertinente parmi toutes les alternatives puisqu'elle occupe la première position trois fois contre deux fois pour les alternatives a_2 et a_3 et une seule fois pour l'alternative a_4.
- Chacune des deux alternatives a_2 et a_3 atteint deux fois le haut du classement. En effet, l'alternative a_2 est en première position pour les deux individus x_4 et x_5, et l'alternative a_3 se trouve en première position pour les individus x_6 et x_7.
- L'alternative a_4 est la moins pertinente parmi toutes les alternatives puisqu'elle occupe la première position une seule fois.

Par conséquent, l'alternative a_1 occupe la première place, les alternatives a_2 et a_3 occupent la deuxième place, alors que l'alternative a_4 occupe la dernière place.

Système de vote majoritaire à deux tours

Le système de vote majoritaire (*c.f.,* [FG82]) consiste à déclarer un candidat vainqueur s'il obtient la majorité absolue des votes, *i.e.,* plus que 50% des voix. Si au contraire, aucun candidat n'obtient la majorité absolue, un deuxième tour est organisé entre les deux candidats qui ont obtenu le plus de voix au premier tour. Lors de ce second tour, le candidat ayant récolté le plus de voix est déclaré vainqueur, ce qui est inévitablement une majorité, car il n'y a que deux candidats en lice. En supposant qu'il y a un seul et même vote pour les deux tours, nous pouvons appliquer ce système sur notre exemple illustré par le tableau 3.1 de la manière suivante :

- **Au 1^{er} tour :** Dans notre exemple, pour déclarer une alternative gagnante au premier tour, elle doit être classée en première position par au moins cinq individus, ce qui n'est pas le cas. Ainsi, un second tour est nécessaire. Les alternatives a_1 et a_2 sont retenues pour le second tour et les alternatives a_3 et a_4 sont écartées puisque a_1 et a_2 sont préférées aux deux autres alternatives selon l'ensemble des individus.
- **$2^{\text{ème}}$ tour :** Il s'agit de déterminer l'alternative la plus pertinente entre a_1 et a_2. Ainsi, il suffit de considérer l'évaluation des deux alternatives illustrée par le tableau 3.2.

	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8
a_1	1	1	1	2	2	1	1	2
a_2	2	2	2	1	1	2	2	1

TABLE 3.2 – L'évaluation des deux alternatives a_1 et a_2 par huit individus x_1, x_2, x_3, x_4, x_5, x_6, x_7, et x_8

A la lecture du tableau 3.2, nous pouvons déduire que l'alternative a_1 est plus pertinente que l'alternative a_2 puisque l'alternative a_1 est préférée à l'alternative a_2 pour les cinq individus x_1, x_2, x_3, x_4, x_6 et x_7, alors que a_2 est préférée à a_1 pour seulement les trois individus x_4, x_5 et x_8.

Par conséquent, les alternatives a_1 et a_2 occupent les deux premières places alors que les alternatives a_3 et a_4 occupent les deux dernières places.

Système de vote majoritaire à plusieurs tours

Dans le système de vote majoritaire à plusieurs tours (*c.f.,* [FG82]), le votant ne se contente pas de sélectionner un candidat préféré, mais il donne un ordre de préférence

pour l'ensemble des candidats. Ce système consiste à réaliser $n-1$ tours pour n candidats à moins d'avoir avant une majorité absolue pour un candidat. A chaque tour i, chaque votant est supposé avoir un candidat préféré parmi les candidats en course. Le candidat qui obtient le moins de voix est éliminé et ne passe pas au tour $i+1$. Si un candidat obtient la majorité absolue, il est déclaré vainqueur. Dans le cas contraire, le tour $i+1$ est organisé entre les candidats qui ont obtenu le plus de voix au tour précédent. En considérant les individus comme des votants et les alternatives comme des candidats pour le tableau 3.1, nous obtenons :

- **Au 1^{er} tour :** Aucune alternative n'a la majorité absolue puisque aucune d'entre elles n'est classée en première position pour au moins cinq individus. En effet,
  - L'alternative a_1 est classée en première position par les trois individus x_1, x_2 et x_3.
  - L'alternative a_2 est classée en première position par les deux individus x_4 et x_5.
  - L'alternative a_3 est classée en première position par les deux individus x_6 et x_7.
  - L'alternative a_4 est classée en première position par seulement l'individu x_8.

  Ainsi, l'alternative a_4 est éliminée et elle ne passe pas au tour suivant.
- **Au $2^{\text{ème}}$ tour :** En supprimant l'alternative a_4 du tableau 3.1, nous obtenons le tableau d'évaluation donnée par la table 3.3.

	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8
a_1	1	1	1	3	3	2	2	3
a_2	2	2	2	1	1	3	3	2
a_3	3	3	3	2	2	1	1	1

TABLE 3.3 – L'évaluation des trois alternatives a_1, a_2 et a_3 par huit individus x_1, x_2, x_3, x_4, x_5, x_6, x_7, et x_8

A la lecture du tableau 3.3, nous constatons que
- L'alternative a_1 est classée en première position par les individus x_1, x_2 et x_3.
- L'alternative a_2 est classée en première position par les individus x_4 et x_5.
- L'alternative a_3 est classée en première position par les individus x_6, x_7 et x_8.

Ainsi, aucune alternative n'obtient la majorité absolue. Par conséquent, un autre tour est nécessaire afin d'évaluer la pertinence des deux alternatives a_1 et a_3. L'alternative a_2 est éliminée puisqu'elle est la moins classée en première position.

- **Au 3ème tour :** En supprimant l'alternative a_2 du tableau 3.3, nous obtenons le tableau d'évaluation donné par la table 3.4.

	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8
a_1	1	1	1	2	2	2	2	2
a_3	2	2	2	1	1	1	1	1

TABLE 3.4 – L'évaluation des deux alternatives a_1 et a_3 par les huit individus x_1, x_2, x_3, x_4, x_5, x_6, x_7, et x_8

A la lecture du tableau 3.4, nous constatons que l'alternative a_3 obtient la majorité absolue. Il n'est plus alors utile de réaliser un autre tour.

Par conséquent, les alternatives a_3 et a_1 occupent les deux premières places alors que les alternatives a_2 et a_4 occupent les deux dernières places.

3.2.1.2 Systèmes de vote par scorage

Les systèmes par scorage se sont historiquement opposés aux systèmes majoritaires. Ils sont construits autour de l'idée qui consiste à prendre en compte l'intensité des préférences des votants en associant un score à chaque candidat. Dans ces systèmes, un candidat c_1 est préféré à un candidat c_2 si le score de c_1 est supérieur au score c_2. Le candidat qui obtient le plus grand score est déclaré vainqueur.

Système de Condorcet

Dans le système de Condorcet [Con85], chaque votant range les candidats par ordre de préférence. Afin de révéler l'opinion collective, Condorcet (1785) a proposé de comparer les candidats par paires en utilisant la méthode suivante : un candidat c_1 est préféré collectivement à un candidat c_2 si et seulement si le nombre de votants ayant classé c_1 avant c_2 est supérieur au nombre de votants ayant classé c_2 avant c_1. Un candidat est déclaré vainqueur s'il est préféré collectivement à chacun des autres candidats. Ce candidat est appelé le vainqueur de Condorcet et il est nécessairement unique. En appliquant le système de Condorcet sur notre exemple donné par le tableau 3.1, nous obtenons les duels majoritaires entre les alternatives illustrés par le tableau 3.5.

Il est à noter que ce système n'est pas toujours opérationnel puisqu'il peut conduire à des cycles au niveau collectif. Par exemple a_1 est préférée à a_2 (5 contre 3), a_2 est préférée

a_1	a_2
5	3

a_1	a_3
3	5

a_1	a_4
3	5

a_2	a_3
5	3

a_2	a_4
5	3

a_3	a_4
7	1

TABLE 3.5 – Duels majoritaires entre les alternatives

à a_3 (5 contre 3), a_3 est préférée à a_4 (7 contre 1) et a_4 est préférée à a_1 (5 contre 3). Ainsi, il n'existe pas une alternative gagnante. Pour résoudre ce problème, trois méthodes ont été proposées :

1. **Règle de Copeland [MS] :** elle consiste à associer pour un candidat c le score suivant :

$$Cop(c) = \sum_{\substack{c_i \in \mathcal{C} \\ c_i \neq c}} P(c, c_i) \text{ tel que } P(c, c_i) = \begin{cases} 1 & \textit{si la majorité préfère } c \text{ à } c_i \\ -1 & \textit{si la majorité préfère } c_i \text{ à } c \\ 0 & \textit{sinon} \end{cases}$$

   Dans notre exemple,
   - $Cop(a_1) = P(a_1, a_2) + P(a_1, a_3) + P(a_1, a_4) = +1$ -1 -1 = -1.
   - $Cop(a_2) = P(a_2, a_1) + P(a_2, a_3) + P(a_2, a_4) =$ -1 $+1$ $+1 = +1$.
   - $Cop(a_3) = P(a_3, a_1) + P(a_3, a_2) + P(a_3, a_4) = +1$ -1 $+1 = +1$.
   - $Cop(a_4) = P(a_4, a_1) + P(a_4, a_2) + P(a_4, a_3) = +1$ -1 -1 = -1.

   Par conséquent, les alternatives a_2 et a_3 occupent la première place et les alternatives a_1 et a_4 occupent la dernière place.

2. **Règle de Kramer-Simpson :** elle consiste à associer pour un candidat c le score suivant :
   $Simp(c) = min\{N(c, c_i) | c_i \in \mathcal{C}$ et $N(c, c_i) =$ nombre de votants qui préfèrent c à $c_i\}$

   Dans notre exemple,
   - $Simp(a_1) = min\{N(a_1, a_2), N(a_1, a_3), N(a_1, a_4)\} = min\{5, 3, 3\} = 3$.
   - $Simp(a_2) = min\{N(a_2, a_1), N(a_2, a_3), N(a_2, a_4)\} = min\{3, 5, 5\} = 3$.
   - $Simp(a_3) = min\{N(a_3, a_1), N(a_3, a_2), N(a_3, a_4)\} = min\{5, 3, 7\} = 3$.
   - $Simp(a_4) = min\{N(a_4, a_1), N(a_4, a_2), N(a_4, a_3)\} = min\{5, 3, 1\} = 1$.

Par conséquent, les alternatives a_1, a_2 et a_3 occupent la première place et l'alternative a_4 occupe la dernière place.

3. **Méthode de Tideman [Tid06] :** Cette méthode consiste à établir un graphe orienté pondéré où les sommets représentent les candidats et entre chaque paire de candidats (c, c_i) un arc orienté de c vers c_i si c est collectivement préféré à c_i, auquel une valeur est attribuée et est égale à la difference entre le nombre de votants de c et le nombre de votant de c_i. La figure 3.1 (a) présente le graphe associé à l'exemple d'évaluation du tableau 3.1. Le graphe est ensuite parcouru par ordre croissant du poids attribué en éliminant les arcs qui créent un cycle. Au terme des opérations, un graphe sans cycles est obtenu. Le vainqueur est le sommet vers lequel n'arrive aucune flèche. Par exemple, nous éliminons l'arc (a_3, a_4) puis l'arc (a_1, a_2). Ainsi, nous obtenons un graphe sans cycles donné par la figure 3.1 (b).

   Par conséquent, l'alternative a_2 est l'alternative Condorcet, elle occupe alors la première place. Les deux alternatives a_3 et a_4 occupent la deuxième place puisqu'une seule flèche arrive à chacun de ces alternatives. L'alternative a_1 occupe la dernière place puisqu'il existe deux flèches qui arrivent à cette alternative.

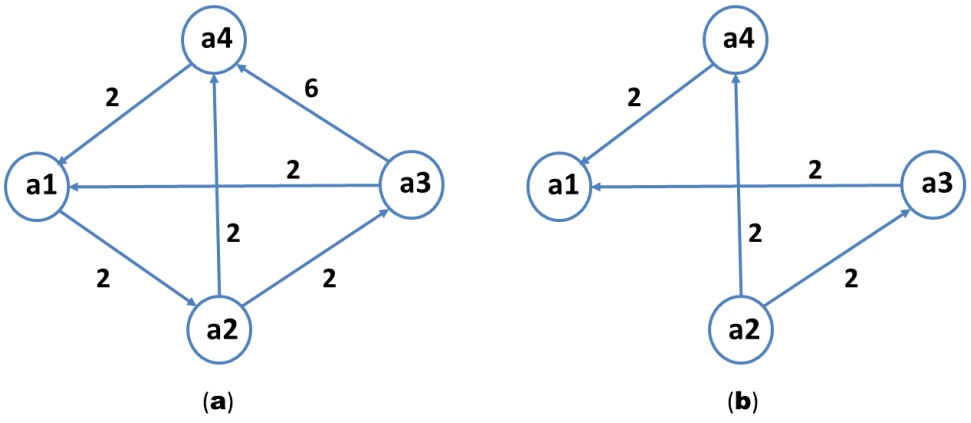

FIGURE 3.1 – (a) Le graphe pondéré associé à l'exemple d'évaluation du tableau 3.1, (b) Le graphe pondéré sans cycle associé à l'exemple d'évaluation du tableau 3.1.

Système de Borda

Tout comme pour le système de Condorcet, le système de Borda [Bor] considère qu'un classement de n candidats par ordre de préférence est donné par chaque votant. On attribue alors au candidat classé premier n points, au candidat classé deuxième n-1, etc.

Le candidat classé dernier se voyant attribuer un point. On additionne alors le nombre de points obtenus par les candidats dans les classements des votants. Le candidat qui obtient le plus de points est déclaré vainqueur. En appliquant ce système sur notre exemple, on obtient :

- L'alternative a_1 est classée à la première position par les individus x_1, x_2 et x_3, classée à la troisième position par les individus x_6 et x_7 et classée à la dernière position par les individus x_4, x_5 et x_8. Ainsi, le score de a_1 égal à $3 \times 4 + 2 \times 3 + 3 \times 2 = 24$.
- L'alternative a_2 est classée à la première position par les individus x_4 et x_5, classée à la deuxième position par les individus x_1, x_2 et x_3, classée à la troisième position par l'individu x_8 et classée à la dernière position par les alternatives x_6 et x_7. Ainsi, le score de a_2 égal à $2 \times 4 + 3 \times 3 + 1 \times 2 + 2 \times 1 = 21$.
- L'alternative a_3 est classée à la première position par les individus x_7 et x_8, classée à la deuxième position par les individus x_4, x_5 et x_8 et classée à la troisième position par les individus x_1, x_2 et x_3. Ainsi, le score de a_3 égal à $2 \times 4 + 3 \times 3 + 3 \times 2 = 23$.
- L'alternative a_4 est classée à la première position par l'individu x_8, classée à la deuxième position par les individus x_6 et x_7, classée à la troisième position par les individus x_4 et x_5 et classée à la dernière position par les individus x_1, x_2 et x_3. Ainsi, le score de a_3 égal à $1 \times 4 + 2 \times 3 + 2 \times 2 + 3 \times 1 = 17$.

Par conséquent, l'alternative a_3 occupe la première place, les alternatives a_1 et a_2 occupent la deuxième place et l'alternative a_4 occupe la dernière place.

Vote par approbation

Le vote par approbation [BF07] permet à chaque votant d'indiquer pour chacun des candidats s'il accepte de le voir élu ou non. Le candidat qui obtient le plus grand nombre de votes est déclaré vainqueur. Pour pouvoir appliquer ce système sur notre exemple, il est possible d'attribuer à une alternative la valeur 1 (pour approuver) lorsque sa position pour un individu dépasse un rang minimal défini par l'individu et lui attribuer la valeur 0 (pour désapprouver) dans le cas contraire. Supposons que pour chaque individu le rang minimal est 2, nous obtenons :

- L'alternative a_1 est approuvée par les trois individus x_1, x_2 et x_3.
- L'alternative a_2 est approuvée par les cinq individus x_1, x_2, x_3, x_4 et x_5.
- L'alternative a_3 est approuvée par les cinq individus x_6, x_7 et x_8.
- L'alternative a_4 est approuvée par les trois individus x_4, x_5, x_6, x_7 et x_8.

Par conséquent, les deux alternatives a_2 et a_4 occupent la première place et les alternatives a_1 et a_3 occupent la dernière place.

3.2.1.3 Discussion

Dans [Arr51], Arrow montre l'impossibilité pour un système de vote de posséder de manière simultanée un petit nombre de propriétés dès qu'on souhaite agréger $n \geq 3$ préférences en une préférence globale. Dans notre contexte, ces propriétés peuvent être traduites comme suit :

- ***Non dictature :*** il n'existe pas de dictateur, *i.e.,* aucun individu ne peut dicter sa préférence à la majorité.
- ***Universalité :*** il existe toujours une préférence globale, quelles que soient les préférences de chaque individu.
- ***Transitivité :*** le système doit fournir un classement sous la forme d'un préordre complet.
- ***Unanimité :*** le résultat d'un système ne doit pas contredire une préférence unanime des individus.
- ***Indépendance :*** le résultat de la comparaison entre deux alternatives ne dépend que de leurs positions relatives dans les listes ordonnées fournies par les individus.

Nous pouvons alors énoncer le théorème d'Arrow suivant [Arr51] :

Théorème 3.2.1 *Dès lors qu'il y a au moins trois candidats, aucun système de vote ne peut satisfaire les conditions de non dictature, d'universalité, de transitivité, d'unanimité et d'indépendance.*

3.2.2 Dominance de Pareto

La dominance de Pareto [Mat91] représente un paradigme très populaire et puissant pour extraire des objets d'un ensemble de données multidimensionnel. Elle a été introduite par Börzsönyi [BKK01] dans le cadre d'extension des systèmes de base de données afin de supporter des requêtes avec plusieurs critères de préférences en parallèle. Elle est particulièrement utile en présence de plusieurs critères d'optimisation où il est difficile, voire impossible, de formuler une bonne fonction de préférence. Le principe fondamental de *la dominance de Pareto* est de filtrer un ensemble d'alternatives intéressantes à

partir d'un ensemble d'alternatives potentiellement importantes. *Une alternative est intéressante si elle n'est dominée par aucune autre alternative. La dominance de Pareto* peut être définie comme suit :

Définition 17 *Étant donné un ensemble d'alternatives $\mathcal{A}$ dans un espace de dimension d, la dominance de Pareto retourne le sous-ensemble d'alternatives $\mathcal{A}'$ de $\mathcal{A}$ qui sont intéressantes dans le sens où elles ne sont dominées par aucune autre alternative de $\mathcal{A} \setminus \mathcal{A}'$. On dit qu'une alternative p = (p[1], p[2],..., p[d]) domine une alternative q = (q[1], q[2],...,q[d]) si :*

1. *p est meilleure que q pour au moins une dimension j, i.e., le rang de p est strictement inférieur à celui de q : p[j] < q[j].*
2. *p est au moins aussi bonne que q sur toutes les dimensions, i.e., le rang de p est inférieur ou égal à celui de q : p[i] ≤ q[i], pour 1 ≤ i ≤ d.*

Exemple 15 *Dans notre exemple illustré par le tableau 3.1, aucune des quatre alternatives n'est dominée puisque chacune est classée au moins une fois en première position.*

Bien que la dominance de Pareto permettait d'identifier les alternatives intéressantes dans un ensemble de données multidimensionnel, elle présente l'inconvénient de produire un grand nombre d'alternatives lorsque la dimension est élevée ou lorsque les données sont anti-corrélées [LJZ11].

3.2.3 Synthèse des différentes approches d'agrégation de préférences

Dans le tableau 3.6, nous comparons les différentes approches d'agrégation de préférences tout en tenant compte des points suivants :

- **Donnée :** Ce point décrit l'information qui doit être fournie par chaque individu.
- **Résultat :** Ce point décrit la nature de la relation entre les alternatives fournie par l'approche. Cette relation peut être soit un ordre total, soit un ordre partiel sur les alternatives.
- **Sélection :** Ce point indique si l'approche d'agrégation de préférences permet de réaliser une sélection d'alternatives.
- **Comparaison 2 à 2 :** Ce point indique si l'approche d'agrégation de préférences compare les alternatives par paire. En d'autre termes, ce point indique si l'approche prend en considération toutes les positions de chaque alternative dans l'ordre des préférences.

Approches	Donnée	Résultat	Sélection	Comparaison 2 à 2
Système majoritaire simple	Une alternative par individu	Ordre total sur les alternatives	Non	Non
Système majoritaire à 2 tours	Ordre de préférences entre les alternatives par individu	Ordre total sur les alternatives	Non	Non
Système majoritaire à plusieurs tours	Ordre de préférences entre les alternatives par individu	Ordre total sur les alternatives	Non	Non
Système de Condorcet*	Ordre de préférences entre les alternatives par individu	Ordre total sur les alternatives	Non	Oui
Système de Borda	Ordre de préférences entre les alternatives par individu	Ordre total sur les alternatives	Non	Oui
Système d'approbation	Ensemble d'alternatives par individu	Ordre total sur les alternatives	Non	Oui
Dominance de Pareto	Ordre de préférences entre les alternatives par individu	Ordre partiel sur les alternatives	Oui	Non

TABLE 3.6 – Caractéristiques des principales approches d'agrégation de préférences.
(*) Système de Condorcet utilisant la règle de Copeland ou la règle de Kramer-Simpson ou la règle de Tideman.

À la lecture du tableau 3.6, nous constatons que :

1. Chaque individu doit fournir un ordre de préférences entre les alternatives pour les systèmes majoritaire à deux tours, majoritaire à plusieurs tours, de Condorcet, de Borda et dominance de Pareto. Cependant, le système majoritaire simple et le système d'approbation nécessitent le choix, respectivement, d'une alternative et d'un ensemble d'alternatives.
2. L'approche basée sur la dominance de Pareto est l'unique approche qui fournit un ordre partiel entre les alternatives. En effet, il est possible de trouver deux alternatives incomparables, *i.e.,* qui ne se dominent pas.
3. L'approche basée sur la dominance de Pareto est l'unique approche qui permet de sélectionner un ensemble d'alternatives. En effet, les alternatives non dominées peuvent être retenues tandis que les alternatives dominées peuvent être écartées. Cependant, toutes les autres approches fournissent un ordre total sur les alternatives. Ainsi, les individus doivent préciser un nombre k permettant de sélectionner les k

meilleures alternatives.

4. Seuls les systèmes de vote par scorage comparent les alternatives par paire.

3.3 Recherche des motifs basée sur les préférences

La sélection des motifs basée sur les préférences consiste à évaluer le motif à partir de son support et d'un ensemble de fonctions de score traduisant les préférences d'un ou plusieurs experts. Par exemple, pour une base de transactions de ventes, l'expert souhaite sélectionner les ensembles de produits qui sont les plus achetés, qui maximisent le profit et qui minimisent le coût de leur stockage. Afin d'apporter une réponse adéquate à ce type de requête, les motifs doivent être évalués selon les critères de choix. Pour résoudre ce problème, la sélection des motifs basée sur les préférences propose de projeter ces critères sur une ou plusieurs fonctions de scores permettant de quantifier l'intérêt des motifs. Le tri de ces motifs selon leurs scores permet d'aboutir à des motifs intéressants. Dans ce qui suit, nous allons présenter et discuter les limites des approches de sélection basée sur les préférences. Notons que ces approches sont illustrées à partir du contexte d'extraction donné par le tableau 3.7.

$\mathcal{R}$	a	b	c	d
t_1	×	×	×	×
t_2	×	×	×	×
t_3		×	×	×
t_4	×	×		
t_5			×	

TABLE 3.7 – Exemple de contexte d'extraction.

3.3.1 Motifs les plus informatifs

3.3.1.1 Description

Dans [PN09], les auteurs introduisent un ensemble réduit de motifs appelé *Motifs les plus informatifs*. L'objectif de cet ensemble est de sélectionner les motifs fréquents maximisant l'informativité et minimisant la redondance structurelle. L'informativité d'un motif est

évaluée selon une fonction de score, calculable à partir du support et de la structure du motif. Cette fonction de score est donnée par l'expert exprimant ses préférences pour les motifs à sélectionner. Cependant, le choix de n'importe quelle fonction de score peut entraîner la sélection de motifs redondants. En effet, les motifs présentant un support et une structure semblables peuvent avoir des scores similaires. Ces motifs sont alors classés dans le même rang dans la liste des motifs triés selon la fonction de score. Par conséquent, des motifs redondants peuvent être sélectionnés. Ainsi, les auteurs proposent d'utiliser une famille de fonctions de score réalisant un compromis entre l'informativité et la non redondance. Cette famille de fonctions de score, appelée *fonction de score informative*, est définie comme suit [PN09] :

Définition 18 *Étant donné un contexte d'extraction $\mathcal{K}$=($\mathcal{T}$, $\mathcal{I}$, $\mathcal{R}$) et l'ensemble des motifs partiellement ordonnés ($\mathcal{P}(\mathcal{I})$,$\leq_{\mathcal{P}(\mathcal{I})}$). Une* fonction de score *est une fonction $S : \mathcal{P}(\mathcal{I}) \times [0; |\mathcal{T}|] \rightarrow \sigma$, où l'ensemble σ est muni d'une relation d'ordre partiel $\leq_\sigma$. Le score d'un motif I dans le contexte $\mathcal{K}$, noté par $S(I)$, est donné par $S(I) = S(I, support(I))$, où $support(I)$ est le support de I dans $\mathcal{K}$. Une fonction de score S est informative si elle vérifie les propriétés suivantes :*

- *Pour tout motif $I \neq \emptyset$ la fonction partielle $S^I : r \mapsto S(I, r)$ est une fonction strictement croissante de $r \in [0; |\mathcal{T}|]$:*
  $\forall I \in \mathcal{I}$, $I \neq \emptyset$, $\forall (a_1, a_2) \in [0; |\mathcal{T}|]^2$, *si* $a_1 < a_2$ *alors* $S^I(a_1) <_\sigma S^I(a_2)$
- *Pour tout $r \in]O; |\mathcal{T}|]$, la fonction partielle $S^r : I \mapsto S(I, r)$ est une fonction strictement croissante de $I \in \mathcal{I}$:*
  $\forall r \in]O; |\mathcal{T}|]$, $\forall (I_1, I_2) \in \mathcal{I}^2$, *si* $I_1 <_{\mathcal{P}(\mathcal{I})} I_2$ *alors* $S^r(I_1) <_\sigma S^r(I_2)$
- *un motif de support nul ou vide ne peut avoir un score supérieur à celui d'un motif de support non nul ou non vide :*
  $\forall (I_1, I_2) \in \mathcal{I}^2$, $\nexists r > 0$, $s(I_1, r) <_\sigma s(I_2, 0)$.

Exemple 16 *La fonction area : $(I, support(I)) \mapsto support(I) \times |I|$, où $|I|$ est la taille du motif I, est une fonction de score informative. En effet, la fonction taille du motif est strictement croissante définie dans l'ordre des motifs.*

En se basant sur la notion de score informatif, Pennerath et Napoli définissent un motif plus informatif [PN09].

Définition 19 *Soient un contexte d'extraction $\mathcal{K}$=($\mathcal{T}$, $\mathcal{I}$, $\mathcal{R}$) et S une fonction de score associée à un ordre de score (σ, $\leq_\sigma$)*

- *Un motif I est un voisin du motif I' si I est prédécesseur ou successeur immédiat de I' selon l'ordre de motifs $(\mathcal{P}(\mathcal{I}),\leq_{\mathcal{P}(\mathcal{I})})$,* i.e., *il n'existe pas un motif entre I et I'.*
- *Un motif I domine un autre motif I' si et seulement si I et I' sont voisins dans $(\mathcal{P}(\mathcal{I}),\leq_{\mathcal{P}(\mathcal{I})})$ et si $S(I') \leq_\sigma S(I)$.*
- *Un motif I est plus informatif si et seulement si $support(I) \neq 0$ et si aucun motif ne domine I.*

Exemple 17 *Le tableau 3.8 illustre le support et l'area de chaque motif pouvant être extraits à partir du contexte donné par le tableau 3.7. Pour $minsup = 3$ et $S(I) = area(I) = support(I) \times |I|$, les motifs les plus informatifs fréquents figurent en gras : b avec un score égal à 6 et bcd avec un score égal à 9.*

Ainsi, les auteurs proposent d'extraire l'ensemble de motifs les plus informatifs fréquents ainsi que les supports et les scores associés. Il est à noter qu'un motif fréquent I peut être dominé par l'un de ses successeurs immédiats infréquents alors qu'il n'est dominé par aucun de ses prédécesseurs immédiats et par aucun de ses successeurs immédiats fréquents. Dans ce cas, I sera considéré comme un motif des plus informatifs alors qu'il ne l'est pas. Par exemple, pour $minsup = 4$, le motif c est fréquent et n'est dominé par aucun motif fréquent, mais il n'est pas des plus informatifs puisqu'il est dominé par le motif infréquent cd. Ainsi, lors de la recherche des motifs les plus informatifs fréquents, il est nécessaire de connaître les supports des motifs infréquents qui possèdent au moins un prédécesseur immédiat fréquent dans $(\mathcal{P}(\mathcal{I}),\leq_{\mathcal{P}(\mathcal{I})})$.

3.3.1.2 Recherche des motifs les plus informatifs

Afin d'extraire les motifs les plus informatifs fréquents, Pennerath et Napoli ont proposé deux algorithmes adoptant deux stratégies différentes [PN09]. Le premier algorithme permet d'extraire les motifs les plus informatifs fréquents en explorant directement le contexte d'extraction. Cependant, le deuxième consiste à filtrer les motifs fréquents extraits préalablement pour obtenir les plus informatifs. Dans ce qui suit, nous allons décrire brièvement les deux algorithmes.

L'algorithme d'extraction directe parcourt l'espace de recherche en profondeur partant du motif vide $\emptyset$. Pour générer les motifs candidats à être les plus informatifs, des extensions sont appliquées au motif courant I conduisant à ses successeurs immédiats. Ensuite, un accès à la base de transactions est effectué afin de calculer les supports de ses successeurs. Le score de chacun de ses successeurs (qu'ils soient fréquents ou non) est comparé à

1-$motifs$	$support$	$area$
a	3	3
b	4	4
c	4	4
d	3	3

2-$motifs$	$support$	$area$
ab	**3**	**6**
ac	2	4
ad	2	4
bc	3	6
bd	3	6
cd	3	6

3-$motifs$	$support$	$area$
abc	2	6
abd	2	6
acd	2	6
bcd	**3**	**9**

4-$motifs$	$support$	$area$
$abcd$	2	8

TABLE 3.8 – Les valeurs de $support$ et d'$area$ des motifs pouvant être extraits à partir du contexte donné par le tableau 3.7. Pour $minsup = 3$ et $S(I) = area(I)$, les motifs les plus informatifs figurent en gras

celui du motif courant I. Le motif du score inférieur est éliminé de l'ensemble des motifs candidats à être plus informatifs. Par ailleurs, chaque successeur fréquent est recherché dans une liste contenant les motifs déjà traités. Si la liste ne contient pas ce motif, cela signifie que le motif est traité pour la première fois, il est alors ajouté à la liste puis développé à son tour comme un motif courant. Les successeurs infréquents sont sauvegardés dans la liste mais ne sont pas développés.

L'idée du deuxième algorithme proposé consiste à sélectionner les motifs les plus informatifs parmi les motifs fréquents produits par un algorithme existant d'extraction de motifs fréquents. Cette sélection est réalisée en traitant les motifs fréquents niveau par niveau où chaque niveau regroupe les motifs de même taille. Le score de chaque motif fréquent du niveau $i+1$ est comparé à celui de ses prédécesseurs immédiats fréquents du niveau i. Le motif du score inférieur est éliminé de l'ensemble des motifs candidats à être des plus informatifs. Ce filtrage est appliqué pour tous les niveaux possibles. Cependant, à la fin de ce filtrage, il se peut qu'un motif fréquent retenu soit dominé par un successeur immédiat infréquent. Ainsi, les scores de tous les successeurs immédiats infréquents des motifs retenus sont calculés. Un deuxième filtrage est alors appliqué pour éliminer les

motifs fréquents qui sont dominés par au moins un successeur immédiat infréquent.

3.3.1.3 Discussion

Suite à une étude critique de l'ensemble des motifs les plus informatifs fréquents, nous avons constaté que cet ensemble présente l'avantage de produire des motifs disposant d'une interprétation sémantique qui permet une exploitation et une interprétation efficaces des connaissances présentées à l'expert. Cependant, la recherche de cet ensemble présente les limites suivantes :

- Seules les fonctions de scores informatives peuvent être utilisées pour extraire les motifs les plus informatifs. Ainsi, l'expert est contraint de projeter ses préférences dans une fonction de score qui doit être informative.
- Afin d'éliminer la redondance Pennerath et Napoli proposent de comparer un motif I seulement à ses prédécesseurs immédiats et ses successeurs immédiats qui lui sont structurellement similaires. Cependant, I peut présenter une structure similaire avec un motif qui ne lui est pas voisin. Par exemple, supposons que les motifs *ab* et *abcd* ne soient dominés par aucun de leurs motifs voisins, alors *ab* et *abcd* seront considérés comme des motifs les plus informatifs, alors qu'ils présentent une structure similaire. Ainsi, l'ensemble des motifs les plus fréquents peut présenter une forme de redondance structurelle.
- Outre l'aspect combinatoire de la recherche des motifs les plus informatifs, des balayages répétitifs du contexte d'extraction doivent être réalisés afin de déterminer les supports des motifs candidats et ceux de leurs successeurs immédiats infréquents. En effet, les opérations d'entrée/sortie liées aux lectures du contexte sont très coûteuses. Ceci a pour conséquence d'augmenter le temps global nécessaire à la recherche des motifs les plus informatifs.

3.3.2 Motifs et dominance de Pareto

Dans [SRPC11], Soulet *et al.,* ont cherché à étendre le modèle des motifs les plus informatifs afin d'y intégrer un ensemble de fonctions de scores exprimant les préférences d'un ou plusieurs experts pour les motifs à sélectionner. Les auteurs introduisent alors, un ensemble réduit de motifs appelé *Skypatterns* dont l'objectif est de sélectionner les motifs qui expriment les meilleurs compromis entre les fonctions de score données par l'expert. Cet ensemble est sélectionné à partir des motifs en se basant sur la notion de dominance

de Pareto [Mat91]. Un motif I est dominé par un autre motif I' si, pour toutes les fonctions de score s_i intéressant le décideur, $s_i(I) \leq s_i(I')$ et il existe au moins une fonction de score s_j telle que $s_j(I) < s_j(I')$. Par exemple, en considérant le contexte d'extraction illustré par le tableau 3.7, le motif *abc* est dominé par le motif *ab* pour les fonctions de score *support* et *area*. En effet, $support(abc) < support(ab)$ et $area(abc) = area(ab) = 6$. Ainsi, l'ensemble *Skypatterns* contient les motifs qui ne sont pas dominés.

Définition 20 *Soient un contexte d'extraction* $\mathcal{K}=(\mathcal{T}, \mathcal{I}, \mathcal{R})$ *et un ensemble de scores* S. *L'ensemble des* Skypatterns *qui peuvent être extraits à partir de* $\mathcal{K}$, *noté* $Sky(\mathcal{P}(\mathcal{I}), S)$, *est défini comme suit :* $Sky(\mathcal{P}(\mathcal{I}), S) = \{I \in \mathcal{P}(\mathcal{I}) | \nexists I' \in \mathcal{P}(\mathcal{I}), I' \text{ domine } I\}$

Exemple 18 *Considérons le contexte d'extraction illustré par le tableau 3.7, l'ensemble des* Skypatterns *pour les fonctions de score support et area est donné par le tableau 3.9*

Skpatterns	*support*	*area*
b	4	4
c	4	4
bcd	3	9

TABLE 3.9 – Skypatterns extraits à partir du contexte d'extraction du tableau 3.7

3.3.2.1 Recherche des Skypatterns

Dans [SRPC11], Soulet *et al.,* proposent une approche pour extraire les *Skypatterns* à partir d'un contexte d'extraction selon un ensemble de fonctions de score S. Cette approche tire profit d'une famille de fonctions de score appelée *fonctions skylinables* permettant de sélectionner un ensemble concis de motifs en vue de réduire l'espace de recherche des *Skypatterns*. Ainsi, le problème de recherche des *Skypatterns* est formulé comme suit :

1. A partir de l'ensemble des fonctions de scores données par l'utilisateur S, déterminer le sous-ensemble S' qui permet de générer un ensemble concis de motifs.
2. Générer l'ensemble concis de motifs.
3. Extraire les *Skypatterns* de l'ensemble concis de motifs.
4. A partir des *Skypatterns* de l'ensemble concis de motifs, générer la totalité des *Skypatterns*

Afin de réaliser la première étape, Soulet *et al.,* introduisent dans [SRPC11] la notion de *Skylinabilité* définie comme suit :

Définition 21 *Soient S, S' deux ensembles de fonctions de score tel que $S' \subseteq S$. L'ensemble S est dit strictement S'-Skylinable selon $\subset$ (respectivement $\supset$) si et seulement si pour tout $I \in \mathcal{I}$ et pour tout $s' \in S'$ tels que $s'(I) = s'(I_1)$ et $I_1 \subset I$ (respectivement $I \supset I_1$) on a, I domine I_1 selon l'ensemble S.*

Exemple 19 *Supposons que $S = \{support, area\}$ et $S' = \{support\}$, alors S est strictement S'-Skylinable selon $\supset$ car en augmentant la taille d'un motif I, la valeur de $area(I)$ croit lorsque le support reste constant. Dans notre contexte illustré par le tableau 3.7, le support du motif bc est égal au support du motif bcd, nous pouvons alors déduire que bcd domine bc selon S.*

En se basant sur la notion de *Skylinabilité*, Soulet *et al.,* ont montré qu'il est possible de générer l'ensemble de tous les *Skypatterns* à partir des *Skypatterns* d'un ensemble réduit de motifs. Cet ensemble réduit est caractérisé par l'opérateur *distinct* défini comme suit :

Définition 22 *Soient un contexte d'extraction $\mathcal{K}$=($\mathcal{T}$, $\mathcal{I}$, $\mathcal{R}$) et un ensemble de score S'. L'ensemble des motifs distincts qui peuvent être extraits à partir de $\mathcal{K}$ selon S', est défini comme suit : $\mathcal{D}is_\theta(\mathcal{P}(\mathcal{I}), S') = \{I \in \mathcal{P}(\mathcal{I}) | \forall I \theta I_1, s'(I) \neq s'(I_1)$ pour tout $s' \in S'\}$ où $\theta \in \{\subset, \supset\}$*

Exemple 20 *Considérons le contexte d'extraction illustré par le tableau 3.7, l'ensemble des motifs distincts qui peuvent être extraits selon la fonction de score support est donné par : $\mathcal{D}is_\supset(\mathcal{P}(\mathcal{I}), \{support\}) = \{b, c, ab, bcd, abcd\}$*

Afin de pouvoir générer la totalité des *Skypatterns* à partir de l'ensemble des motifs distincts, il est nécessaire d'utiliser l'opérateur *indistinct* défini comme suit :

Définition 23 *Soient un contexte d'extraction $\mathcal{K}$=($\mathcal{T}$, $\mathcal{I}$, $\mathcal{R}$), un ensemble $Q \subseteq \mathcal{P}(\mathcal{I})$ et un ensemble de fonctions score S. L'ensemble des motifs indistincts de Q qui peuvent être extraits à partir de $\mathcal{K}$ selon S', est défini comme suit : $\mathcal{I}nd(\mathcal{P}(\mathcal{I}), S, Q) = \{I \in \mathcal{P}(\mathcal{I}) | \exists I_1 \in Q, s(I) = s(I_1)$ pour tout $s \in S\}$*

Le théorème suivant montre comment générer la totalité des *Skypatterns* à partir de l'ensemble des motifs distincts.

Théorème 3.3.1 *Soient un contexte d'extraction $\mathcal{K}$=($\mathcal{T}$, $\mathcal{I}$, $\mathcal{R}$) et deux ensembles de fonction score S et S' tel que S est strictement S'-Skylinable selon $\theta \in \{\subset, \supset\}$, nous avons :*

$$Sky(\mathcal{P}(\mathcal{I}), S) = \mathcal{I}nd(\mathcal{P}(\mathcal{I}), S, Sky(\mathcal{D}is_\theta(\mathcal{P}(\mathcal{I}), S'), S))$$

Exemple 21 *Pour le contexte d'extraction donné par le tableau 3.7, examinons le déroulement de l'approche pour $S = \{support, area\}$. La première étape consiste à déterminer l'ensemble S' tel que S est strictement S'-Skylinable. Ainsi, nous obtenons $S' = \{support\}$. Dans la deuxième étape, l'opérateur $\mathcal{D}is_\supset$ est appliqué sur l'ensemble de motifs afin d'extraire les motifs distincts. Ainsi, nous obtenons $\mathcal{D}is_\supset(\mathcal{P}(\mathcal{I}), \{support\}) = \{b, c, ab, bcd, abcd\}$. La troisième étape consiste à trouver les* Skypatterns *des motifs distincts, nous obtenons alors $Sky(\{b, c, ab, bcd, abcd\}, S) = \{b, c, bcd\}$. Dans la dernière étape, la totalité des* Skypatterns *est généré en appliquant l'opérateur $\mathcal{I}nd$ sur les* Skypatterns *des motifs distincts. Nous obtenons finalement, l'ensemble des* Skypatterns*, $Sky(\mathcal{P}(\mathcal{I}), S) = \{b, c, bcd\}$.*

3.3.2.2 Discussion

Tout comme l'approche de recherche des motifs les plus informatifs fréquents, l'approche de recherche des *Skypatterns* présente l'avantage de considérer les préférences de l'expert à travers des fonctions de scores permettant d'évaluer l'intérêt des motifs. Toutefois, elle présente certains inconvénients :

- L'évaluation de l'intérêt des motifs ne tient pas compte des similarités structurelles entre les motifs. Les motifs présentant une structure similaires ont tendance à avoir des scores similaires. Par conséquent, l'ensemble des *Skypatterns* peut être saturé par des motifs ayant des scores élevés mais qui sont fortement redondants.
- L'ensemble des *Skypatterns* peut présenter une perte d'information. En effet, un motif I peut éliminer un autre motif I_1 légèrement moins intéressant alors qu'ils présentent des structures totalement différentes. Dans ce cas, l'information contenue dans le motif I_1 ne sera pas présenté à l'expert.
- Lors de la recherche des *Skypatterns* selon un ensemble de fonction de score S, l'espace de recherche peut être réduit lorsqu'il est possible de déterminer un sous-ensemble S' ayant une relation de *Skylinabilité* avec S. Dans le cas contraire, *i.e.,* lorsque S' est lui même S, l'application de l'opérateur $\mathcal{D}is_\theta$ devient très coûteuse puisque les

scores de chaque motif doivent être comparés à ceux de ses sous-ensembles et de ses sur-ensembles.

3.3.3 Graphes et dominance de Pareto

3.3.3.1 Description

Les graphes sont fréquemment utilisés pour modéliser des données complexes [YHC06]. En effet, toutes les données composées d'entités ayant des relations peuvent être représentées par un graphe où les entités seront représentées par des noeuds et les relations représentées par des arêtes. Formellement, un graphe est défini comme suit :

Définition 24 *Un graphe $G = (S, A)$ est la donnée :*

- *d'un ensemble S dont les éléments sont les sommets du graphe,*
- *d'un ensemble A dont les éléments, les arcs du graphe, sont des couples d'éléments de S.*

Un graphe est dit connexe si pour tout couple de sommets x, $y \in S$ il existe un arc reliant x à y.
Un sous-graphe de G est un graphe $G' = (S', A')$ tel que $S' \subset S$ et $A' \subset A$.

Une des techniques les plus puissantes pour analyser et étudier les graphes consiste à chercher les sous-graphes intéressants. Des sous-graphes sont considérés intéressants s'ils vérifient un ou plusieurs critères définis par l'expert. Ces critères peuvent être de nature structurelle et topologique, basés sur la fréquence et même de nature sémantique lorsque les graphes sont étiquetés. Récemment, une approche de fouille des sous-graphes intéressants, appelée *SkyGraph*, a été proposée dans [PLM08] adoptant la notion de dominance de Pareto. L'objectif principal de cette approche est d'extraire les sous-graphes intéressants en considérant simultanément plusieurs critères de préférence donnés par l'expert. Les sous-graphes retournés à l'expert sont ceux qui ne sont dominés par aucun autre sous-graphe. Un sous-graphe G_1 domine un autre sous-graphe G_2 si G_1 est aussi bon que G_2 pour tous les critères de préférences de l'expert. Dans [PLM08], les auteurs ont considéré deux critères de préférence à savoir le nombre de noeuds et la connectivité décrivant la cohérence d'un graphe [HYH+05]. La connectivité d'un graphe est définie comme suit :

Définition 25 *La connectivité des arêtes d'un graphe connecté est le nombre minimum d'arcs dont leur suppression permet d'obtenir deux sous-graphes connexes*

Exemple 22 *La Figure 3.2 illustre un exemple de deux sous-graphes. Le nombre de noeuds est égal à 6 pour G1 et G2, par contre la connectivité est égale à 3 pour G1 et 1 pour G2. Ainsi, G1 domine G2 pour les deux mesures : nombre de noeuds et connectivité.*

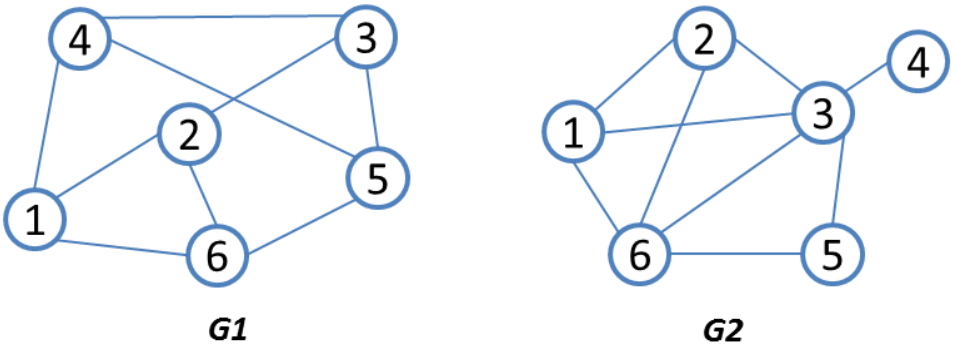

FIGURE 3.2 – Exemple de sous-graphes.

3.3.3.2 Recherche des graphes non dominés

Dans [PLM08], Papadopoulos *et al.* ont proposé un algorithme permettant d'extraire les graphes non dominés. Cet algorithme se base sur une application successive d'un algorithme de type min-cut algorithm [HS00] afin de découper chaque graphe en des sous-graphes. Pour chaque sous-graphe g, l'algorithme détermine le nombre de noeuds ansi que la connectivité, puis il effectue un test pour vérifier si g est un candidat potentiel pour être parmi les *Skygraphs*. Trois cas peuvent être distingués :

1. Si g est dominé par au moins un sous-graphe candidat déjà extrait, alors g est éliminé.
2. Si g domine un ou plusieurs sous-graphes candidats, alors ces sous-graphes sont éliminés et g est considéré un candidat potentiel.
3. Si g n'est dominé et ne domine aucun sous-graphe candidat, alors g est considéré un candidat potentiel.

L'algorithme prend fin lorsque chaque sous-graphe comprend un seul sommet.

3.3.3.3 Discussion

Dans [PLM08], une évaluation expérimentale de *SkyGraph* a été menée sur différentes bases de graphes représentant un réseau de bio-puces [YMH+07], le réseau routier de San

Francisco [1] et un réseau de co-auteurs [2] en utilisant deux critères précédemment mentionnés, à savoir le nombre de noeuds et la connectivité. Les résultats expérimentaux ont montré que la relation de dominance permet de détecter des sous-graphes importants. Cependant, *SkyGraph* présente l'inconvénient de sélectionner un seul sous-graphe dans le cas où ce dernier domine tous les autres sous-graphes pour les critères considérés. Ainsi, il serait intéressant d'intégrer une fonction d'ordre pour les sous-graphes permettant à l'expert de spécifier le nombre de sous-graphes à extraire.

3.3.4 Synthèse des approches de recherche des motifs basées sur les préférences

Dans le tableau 3.10, nous caractérisons les différentes approches de sélection des motifs basées sur les préférences tout en tenant compte des critères suivants :

- **Donnée :** Ce critère décrit la nature des motifs utilisés par l'approche de sélection.
- **Résultat :** Ce critère décrit les caractéristiques des motifs sélectionnés.
- **Extraction :** Ce critère indique si l'approche peut être appliquée lors de la phase d'extraction de motifs.
- **Sélection :** Ce critère indique si l'approche peut être appliquée lors de la phase de sélection de motifs.
- **Dépendance à l'ordre des motifs :** Ce critère indique si l'approche sélectionne les motifs en considérant leur position au sein de l'ordre des motifs.
- **Fonction :** Ce critère indique le mécanisme utilisé par l'approche afin de comparer les motifs.

A la lecture du tableau 3.10, nous constatons que :

1. Seules les approches MIPs et *SkyGraph* peuvent être appliquées lors de la phase d'extraction des motifs.
2. Seuls les *SkyPatterns* ne peuvent être retrouvés qu'en utilisant la phase de sélection.
3. L'approche de recherche des motifs les plus informatifs est l'unique approche qui prend en considération la position d'un motif au sein de l'ordre des motifs. En effet, le score de chaque motif candidat à être parmi les plus informatifs est comparé avec ceux de ses voisins selon l'ordre de motifs.

1. Valable sur http ://www.rtreeportal.org.
2. Valable sur http ://www.cs.helsinki.fi/u/tsaparas/MACN2006/data-code.html.

Approches	Donnée	Résultat	Extraction	Sélection	Dépendance à l'ordre des motifs	Mécanisme de comparaison
Motifs les plus informatifs (MIPs)	motifs d'items	motifs les plus informatifs	oui	oui	oui	comparaison selon un score informatif
Motifs et dominance de Pareto	motifs d'items	*SkyPatterns*	non	oui	non	dominance selon un ensemble de scores
Graphes et dominance de Pareto	graphe	*SkyGraph*	oui	non	non	dominance selon le nombre de noeuds et la connectivité

TABLE 3.10 – Caractéristiques des principales approches de sélection des motifs basées sur les préférences

4. Seules l'approches de recherche des *SkyPatterns* et l'approche de recherche des *SkyGraphs* tolèrent l'utilisation d'un ensemble de scores. Cependant, l'approche de recherche des motifs les plus informatifs ne tolère qu'une seule fonction de score.

3.3.5 Agrégation de mesures d'intérêt de règles d'association

Dans [BLLV06], Barthélemy *et al.,* proposent de tenir compte des différentes informations apportées par les mesures de qualité qui permettent d'évaluer la qualité des règles d'association. Très hétérogènes ces mesures produisent des évaluations très variées. Les auteurs ont alors adopté une approche permettant l'agrégation à l'aide de relations valuées dans le but de mesurer le degré d'intensité de préférence d'une règle sur une autre. Cette relation valuée repose sur l'idée suivante : Soit un ensemble de règles $AR = \{r_1, r_2, ..., r_n\}$ et un ensemble de mesures $M = \{m_1, m_2, ..., m_k\}$, la relation valuée $R_{m_k}(r_i, r_j)$ correspond à une différence normalisée entre les valeurs prises par la mesure m_k sur les règles r_i et r_j. Les auteurs ont présenté la relation valuée suivante :

$$R_{m_k}(r_i, r_j) = \begin{cases} 1 - exp(-\frac{(r_i[m_k] - r_j[m_k])^2}{2\sigma_k^2}) & \text{si } r_i[m_k] - r_j[m_k] > 0 \\ 0 & \text{sinon} \end{cases}$$

Le paramètre σ_k représente un seuil entre "les préférences faibles" et "les préférences fortes". Cette relation permet de modéliser un système de préférences sur l'ensemble des règles. Les auteurs ont ensuite appliqué un opérateur d'agrégation sur l'ensemble des

relations valuées dans le but de produire une relation de consensus.

3.4 Conclusion

Dans ce chapitre, nous avons tout d'abord exposé le problème de la sélection des règles d'association pour les deux scénarios suivants :

- Plusieurs experts interviennent dans la décision, ayant chacun une préférence pour une mesure de qualité.
- Un seul expert intervient dans la décision, ayant des préférences pour plusieurs mesures de qualité.

Ensuite, nous avons passé en revue les différentes approches d'agrégation de préférences et nous avons montré leur connexion avec le problème exposé ci-dessus. Enfin, nous avons présenté et discuté les travaux dédiés à la sélection des motifs en utilisant les préférences.

Dans le chapitre suivant, nous allons proposer une approche permettant de sélectionner les règles d'association selon les préférences d'un ou plusieurs experts.

Points clés

- Nous avons exposé le problème de sélection des règles d'association en utilisant plusieurs mesures de qualité.
- Nous avons passé en revue les approches d'agrégation des préférences et avons montré leur connexion avec la sélection des règles d'association selon plusieurs mesures grâce aux correspondances *individus/mesures* et *alternatives/règles*.
- Nous avons passé en revue et analysé les approches d'extraction des motifs basées sur les préférences.

Chapitre 4

Sélection des règles d'association basée sur la relation de dominance

Sommaire

4.1 Introduction 69
4.2 Travaux sur les mesures de qualité 70
4.2.1 Caractérisation d'une bonne mesure de qualité 70
4.2.2 Classification des mesures de qualité 72
4.2.3 Limites et motivations 73
4.3 Sélection des règles non dominées 75
4.3.1 Règles non dominées 75
4.3.2 Formalisation pour la sélection des règles non dominées 77
4.3.3 Algorithme SkyRule 80
4.3.4 Expérimentations 82
4.4 Sélection des k meilleures règles d'association 87
4.4.1 Ordonnancement des règles selon plusieurs mesures 88
4.4.2 Algorithme RankRule 90
4.4.3 Dualité 91
4.4.4 Expérimentations 92
4.5 Conclusion 98

L'objectif de ce chapitre est d'introduire une approche permettant de sélectionner, selon plusieurs mesures de qualité, un ensemble réduit de règles d'association exprimant

le meilleur compromis entre les différentes évaluations des mesures choisies par un ou plusieurs experts.

4.1 Introduction

Pour comprendre l'intérêt des règles d'association en fouille de données, il faut se rappeler le but de l'extraction de connaissances qui n'est autre que produire à un utilisateur des éléments d'analyse capables de lui fournir des connaissances préalablement inconnues et potentiellement utiles dans sa spécialité. Pour que cette analyse conduite par l'utilisateur soit efficace, seules les règles d'association intéressantes doivent être extraites. Étant donné le nombre exorbitant de règles qui peuvent être générées, les trier selon différentes mesures de qualités (Confiance, Corrélation, Rappel, etc.) pourrait être une solution permettant d'aboutir à celles qui sont pertinentes et descriptives de données. Néanmoins, un nombre important de mesures de qualité proposées dans la littérature a induit à son tour de nouveaux problèmes, tel que le problème de sélection d'une ou plusieurs mesures qui soient les mieux appropriées aux besoins de l'utilisateur. Ce problème a été largement traité par un certain nombre de travaux qui ont eu pour objet d'étudier les comportements des différentes mesures. Toutefois, ces travaux n'ont pas résolu un deuxième problème qui consiste à sélectionner les règles pertinentes en utilisant simultanément plusieurs mesures.

Dans ce chapitre, nous proposons une nouvelle approche dont l'objectif est de sélectionner un ensemble réduit de règles, appelé *règles non dominées*, selon plusieurs mesures. Cette sélection est basée essentiellement sur la notion de dominance de Pareto [BSYN12], qui va permettre de retenir seulement les règles exprimant le meilleur compromis entre les différentes évaluations de mesures choisies. Nous proposons également d'étendre cette approche afin de sélectionner les k meilleures règles en utilisant plusieurs mesures simultanément où k est fixé par l'utilisateur [BSYN14].

Le reste du chapitre est organisé comme suit : La section 4.2 commence par présenter certaines approches utilisées pour étudier le comportement des mesures et les répartit selon deux grandes catégories : la première regroupe les approches qui qualifient "une bonne mesure de qualité" et la deuxième regroupe les approches qui classifient les mesures de qualité. La section 4.3 présente une solution pour la sélection des règles d'association selon plusieurs mesures en introduisant l'ensemble des règles non dominées et présente un algorithme pour l'extraire. Cette section compare ensuite le nombre de règles non dominées par rapport au nombre total de règles et au nombre de règles qui peuvent être extraites par une approche utilisant les seuils. Enfin, la section 4.4 présente une approche permettant de sélectionner les k meilleures règles en utilisant la relation de dominance.

4.2 Travaux sur les mesures de qualité

L'idée principale d'une mesure de qualité est d'associer une valeur numérique à une règle permettant d'évaluer son intérêt. Plusieurs mesures de qualité ont été développées pour compléter le *support* et la $confiance$. Fondées essentiellement sur des bases statistiques et probabilistes, ces mesures ne dépendent que des propriétés intrinsèques aux règles d'association et ne nécessitent aucune connaissance sur les données.

Définition 26 ***(mesure de qualité)*** *:*
Une mesure de qualité m est une fonction de l'ensemble des règles $\mathcal{AR}$ dans l'ensemble des nombres réels $\mathbb{R}$. Formellement, $m : \begin{array}{ccc} \mathcal{AR} & \rightarrow & \mathbb{R} \\ (X \rightarrow Y) & \mapsto & m(X \rightarrow Y) \end{array}$

Les mesures de qualité ont été aussi utilisées dans différentes applications liées à la fouille de données dans le but d'évaluer la qualité des connaissances extraites. En effet, dans [Azé03], Azé utilise les mesures de qualité afin d'évaluer les règles extraites à partir des données numériques et textuelles. Dans [MM04], McGarry et Malone proposent d'utiliser les mesures de qualité pour extraire les règles intéressantes et utiles à partir des réseaux de neurones. Dans [RFIG04], Romao *et al.*, ont proposé un algorithme génétique utilisant les mesures de qualité afin d'évaluer l'intérêt des règles de prédiction floues.

De nombreux travaux ont été proposés pour étudier les comportements des différentes mesures de qualité. Ces travaux peuvent être classés en deux catégories. La première catégorie regroupe les travaux qui se sont intéressés à la recherche des propriétés que devrait satisfaire une mesure de qualité. La deuxième catégorie regroupe les travaux qui se sont intéressés à la classification des mesures en se basant sur ces propriétés.

4.2.1 Caractérisation d'une bonne mesure de qualité

Face à l'abondance des mesures de qualité, plusieurs travaux ont été développés dans le but de définir ce qu'est "une bonne mesure" en fonction des besoins de l'expert. Dans [PS91], Piatetsky-Shapiro propose un cadre permettant d'évaluer la qualité d'une mesure. Il introduit trois propriétés qu'une "bonne mesure" doit vérifier :

Définition 27 *Soit $r : X \rightarrow Y$ une règle d'association. Une bonne mesure m doit vérifier les trois propriétés suivantes :*

1. *La valeur de la mesure m doit être nulle dans le cas de l'indépendance, i.e., lorsque $support(r) = support(X) \times support(Y)$ ou lorsque $support(Y/X) = support(Y)$*

2. *Lorsque la taille de la prémisse et la taille de la conclusion restent constantes, la valeur de la mesure m doit être croissante en fonction du nombre d'exemples de r, i.e., support(r).*
3. *Lorsque le nombre d'exemples de r reste constant, la valeur de la mesure m doit être décroissante en fonction de :*
   - *la taille de la prémisse lorsque celle de la conclusion reste constante.*
   - *la taille de la conclusion lorsque celle de la prémisse reste constante.*

La première propriété indique que si la prémisse et la conclusion d'une règle sont indépendantes dans un contexte d'extraction, alors la règle est considérée non pertinente et la mesure m doit avoir la valeur nulle. La deuxième propriété indique qu'une "bonne mesure" doit favoriser les règles qui se produisent fréquemment. En d'autres termes, cette propriété exige la croissance de la mesure en fonction de la fréquence de la règle lorsque d'autres expressions intervenant dans le calcul de m restent constantes. La troisième propriété indique qu'une "bonne mesure" ne doit pas privilégier les règles dont la prémisse ou la conclusion est fréquente. En effet, lorsqu'un itemset est très fréquent, il existe une grande probabilité de conclure sur cet itemset et donc une conclusion très fréquente n'est pas réellement pertinente.

Les travaux de Piatetsky-Shapiro ont été le point de départ pour la conception d'un certain nombre de propriétés proposées dans la littérature. Dans ce qui suit, nous passons en revue les principales propriétés permettant de caractériser une "bonne mesure".

- **Non symétrie :** la mesure doit évaluer différemment les règles $X \rightarrow Y$ et $Y \rightarrow X$, *i.e.,* $m(X \rightarrow Y) \neq m(Y \rightarrow X)$, puisque la prémisse et la conclusion jouent des rôles différents [Fre99].
- **Non symétrie au sens de la négation de la conclusion :** la mesure doit permettre de choisir entre $X \rightarrow Y$ et $X \rightarrow \overline{Y}$ [Fre99].
- **Implication :** la mesure doit évaluer de la même façon $X \rightarrow Y$ et $\overline{Y} \rightarrow \overline{X}$ dans le cas de l'implication logique, *i.e.,* $m(X \rightarrow Y) = m(\overline{Y} \rightarrow \overline{X})$ [BMS97].
- **Intelligibilité :** la sémantique de la mesure doit être facilement compréhensible par l'expert afin de pouvoir communiquer et expliquer les résultats obtenus [LMVL08].
- **Facilité à fixer un seuil :** la mesure utilisée pour extraire les règles d'association doit être accompagnée d'un seuil d'élagage permettant d'éliminer les règles non pertinentes pour l'expert. Il est donc important de pouvoir facilement fixer ce seuil [LMPV03]

- **Discrimination :** si une mesure varie de façon croissante avec la taille de données et admet une valeur maximale, alors elle peut perdre son pouvoir discriminant quand la taille de données devient très grande. Ainsi, la mesure doit permettre de discerner les règles pertinentes même lorsque l'ensemble d'apprentissage est volumineux [LT04].
- **Déviation à l'équilibre :** la mesure doit avoir une valeur constante ou asymptotiquement constante en fonction de la taille de l'échantillon de données lorsque le nombre d'exemples et celui des contre-exemples de la règle sont identiques [BGBG05].

4.2.2 Classification des mesures de qualité

A l'opposé des travaux qui ont utilisé les propriétés pour caractériser une bonne mesure, les travaux de cette catégorie ont utilisé les propriétés pour analyser le comportement des mesures dans le but de les regrouper d'une manière homogène et de ressortir leurs ressemblances et leurs divergences.

Dans [JA99], Bayardo et Agrawal proposent d'exprimer les mesures de qualité en fonction du *support* et de la $confiance$ afin de déterminer des points communs entre plusieurs mesures. Tan *el al.,* proposent, dans [TKS02], d'évaluer 21 mesures à travers 5 propriétés, afin de déterminer les domaines d'application appropriés à ces mesures. Ils introduisent alors un algorithme permettant de choisir une mesure de qualité adaptée. Les auteurs ont aussi montré expérimentalement que des similarités dans le comportement des mesures peuvent être induites suite à un encadrement du support de la règle. Dans [BGBG05], Blanchard *et al.,* présentent une étude sur 19 mesures à travers neuf propriétés. Cette étude met en évidence des similarités au niveau des comportements en effectuant une catégorisation des mesures par rapport aux deux critères suivants : la déviation à l'équilibre ou à l'independence et la nature de la mesure qui peut être descriptive (*i.e.,* sa valeur reste constante en cas de dilatation des données) ou statistique (sa valeur varie en cas de dilatation des données). Selon ces deux critères, les auteurs ont classifié les mesures en quatre catégories : (1) mesures descriptives/déviation à l'équilibre, (2) mesures descriptives/déviation à l'indépendance, (3) mesures statistiques/déviation à l'équilibre, (4) mesures statistiques/déviation à l'indépendance.

Dans cette même catégorie, d'autres travaux se sont basés sur des méthodes de classification existantes pour classifier les mesures de qualité. Dans [LVML07], Lenca *et al.* ont mené une étude indépendante du domaine applicatif sur 20 mesures selon 9 propriétés. Cette étude a conduit à la construction d'une matrice d'évaluation où chaque ligne repré-

sente une mesure et chaque colonne représente une propriété. Cette matrice d'évaluation est remplie de la manière suivante : une mesure prend "1" lorsqu'elle vérifie une propriété et "0" dans le cas contraire. Les auteurs ont par la suite appliqué la méthode de classification hiérarchique (CAH) sur la matrice d'évaluation pour classifier les mesures. Cinq classes de mesures ayant des comportements similaires ont été obtenues. Dans [HZ10] Heravi et Zaiane ont présenté une étude de 53 mesures, appliquées sur les règles d'association de classe, sous forme d'une matrice d'évaluation traduisant la présence ou l'absence de 16 propriétés pour chaque mesure. Les auteurs ont par la suite appliqué la méthode de classification hiérarchique sur la matrice d'évaluation pour déterminer les mesures similaires. Guillaume *et al.,* [GGN11] ont proposé d'étudier pour 61 mesures de qualité selon 19 propriétés. En se basant sur cette étude, les auteurs ont effectué une catégorisation des mesures en utilisant deux méthodes de classification à savoir CAH et k-moyennes. Cette catégorisation a permis de dégager sept groupes disjoints de mesures au comportement similaire. En s'appuyant sur l'étude des mesures présentée dans [GGN11], Belohlávek *et al.,* ont effectué une catégorisation des mesures en utilisant une analyse factorielle booléenne basée sur la décomposition de matrices binaires en un nombre optimal de facteurs (ou catégories) [BGG$^+$11]. Cette catégorisation a permis d'identifer des groupes qui se chevauchent représentant chacun une collection de mesures ayant des propriétés communes.

D'autres travaux se sont intéressés à l'analyse des comportements des mesures en se basant sur des études empiriques. Vaillant *et al.* [VLL04] [Vai06] ont développé une plateforme appelée "Herbs" permettant d'expérimenter les mesures sur des bases de règles. Le but de ces expérimentations est d'étudier l'influence du choix d'une mesure sur l'ordonnancement des règles en comparant les préordres obtenus par les mesures. Ces comparaisons ont permis de dégager 5 classes de mesures dont chacune contient les mesures donnant des classements similaires de règles. Dans [HGB06], Huynh et al. ont mené une étude sur 34 mesures en utilisant les valeurs de ces mesures pour des règles extraites à partir de deux bases de données de nature différente : une base dense (*i.e.,* contenant des données fortement corrélées) et une autre éparse (*i.e.,* contenant des données faiblement corrélées). Cette étude a permis de regrouper les mesures fortement corrélées.

4.2.3 Limites et motivations

Les travaux sur les mesures de qualité évoqués portent essentiellement sur deux axes de recherche complémentaires à savoir :

1. Proposer des propriétés permettant de dire quelles mesures sont les "bonnes" ou quelles mesures devrait utiliser un expert.

2. Proposer une catégorisation de mesures en fonction de propriétés communes mises en évidence de manière théorique ou empirique.

Face à l'abondance des mesures, ces études peuvent aider l'expert d'une part à restreindre le nombre de mesures à choisir et d'autre part à orienter sa sélection en fonction des propriétés qu'il souhaiterait qu'une mesure vérifie. En effet, après que l'utilisateur ait exprimé ses préférences sur les propriétés, une ou plusieurs mesures lui sont recommandées. Néanmoins, ces études ne permettent pas de résoudre complètement le problème de sélection de règles pertinentes pour les raisons suivantes :

- La pertinence d'une mesure de qualité depend de l'idée que l'expert se fait d'une règle intéressante pour son application et sur les données. Par conséquent, le choix de la mesure de qualité à appliquer sur les règles doit être effectué par l'expert en fonction de ses préférences. Il se pose ainsi clairement le problème de la sélection des meilleures règles dans un environnement coopératif où plusieurs experts interviennent dans la décision ayant chacun une préférence pour une ou plusieurs mesures de qualité.

- L'utilisateur peut avoir des difficultés dans la spécification d'un seuil pour une mesure. En effet, fixer un seuil élevé peut entraîner la perte de certaines règles pertinentes et un seuil faible peut produire des règles redondantes (par exemple, des règles similaires). Ce problème devient plus complexe lorsque plusieurs mesures sont utilisées. En effet, l'utilisateur doit faire face à deux sous-problèmes : fixer un seuil fiable pour chaque mesure et prendre une décision concernant les règles qui ne vérifient que certains seuils.

Pour résoudre ces deux problèmes, nous proposons de sélectionner un ensemble réduit de règles d'association, nommé *règles non dominées* qui a pour objectif d'agréger les préférences de plusieurs experts simultanément. Cette sélection repose sur l'utilisation de la relation de dominance qui va permettre de retenir les règles exprimant le meilleur compromis entre les différentes évaluations de mesures choisies par les experts tout en évitant de fixer des seuils.

4.3 Sélection des règles non dominées

Dans cette section, nous définissons les règles non dominées et nous présentons une méthode pour les extraire.

4.3.1 Règles non dominées

Soit le contexte d'extraction $\mathcal{T}$, présenté par le tableau 4.1(a), contenant 10 transactions et 4 items. Les règles qui peuvent être extraites à partir de ce contexte sont présentées dans le tableau 4.1(b) (la première colonne). L'évaluation de ces règles par l'ensemble de mesures, présentée dans le tableau 4.1(c), permet d'obtenir une table relationnelle Ω (*c.f.*, le tableau 4.1(b)). Formellement, $\Omega = (\mathcal{AR},\mathcal{M})$ tel que l'ensemble des mesures $\mathcal{M} = \{m_1, \ldots, m_k\}$ sont les attributs et les règles $\mathcal{AR} = \{r_1, \ldots, r_n\}$ sont les objets. Nous désignons par $r[m]$ la valeur de la mesure m pour la règle r avec $r \in \mathcal{AR}$ et $m \in \mathcal{M}$.

Puisque l'évaluation des règles varie d'une mesure à l'autre, l'utilisation de plusieurs mesures pourrait conduire à des résultats différents. En effet, une règle peut être considérée pertinente selon une mesure et non pertinente selon une autre. Par exemple, r_1, r_2 et r_3 sont les trois meilleures règles selon la mesure *Confiance* alors que les règles r_4 et r_6 sont les meilleures selon la mesure *Pearl*. Ces différentes évaluations sont une source de confusion pour tout processus de sélection ou classement de règles d'association.

La définition des règles non dominées repose sur la notion de dominance entre les valeurs d'une mesure :

Définition 28 *(Dominance entre valeurs d'une mesure)*
Soient $r[m]$ et $r'[m]$ deux valeurs d'une mesure m associées à deux règles r et r'. La valeur $r[m]$ domine $r'[m]$, notée $r[m] \succeq r'[m]$, si et seulement si $r[m]$ est préférée à $r'[m]$.

Définition 29 *(Dominance entre règles)*
Étant données deux règles r, $r' \in \mathcal{AR}$ et un ensemble de mesures $\mathcal{M}$, nous distinguons trois relations entre r et r' :

- *r **domine** r', notée $r \succeq r'$, ssi $r[m] \succeq r'[m]$, $\forall\, m \in \mathcal{M}$.*
- *Si $r \succeq r'$ et $r' \succeq r$, i.e., $r[m] = r'[m]$, $\forall\, m \in \mathcal{M}$, alors r et r' sont dites **équivalentes**, notées $r \equiv r'$.*
- *Si $r \succeq r'$ et $\exists\, m \in \mathcal{M}$ tel que $r[m] \succ r'[m]$, alors r' est **strictement dominée** par r et on note $r \succ r'$.*

	a	b	c	d
t_1			×	×
t_2	×			
t_3	×			×
t_4			×	
t_5		×		×
t_6	×			×
t_7			×	
t_8				×
t_9		×	×	
t_{10}			×	×

(a) *Un contexte d'extraction* $\mathcal{T}$

Règle	**Freq**	**Conf**	**Pearl**
$r_1 : a \rightarrow d$	0.20	0.67	0.02
$r_2 : b \rightarrow c$	0.10	0.50	0.00
$r_3 : b \rightarrow d$	0.10	0.50	0.02
$r_4 : c \rightarrow d$	0.20	0.40	0.10
$r_5 : d \rightarrow a$	0.20	0.33	0.02
$r_6 : d \rightarrow c$	0.20	0.33	0.10
$r_7 : c \rightarrow b$	0.10	0.20	0.01
$r_8 : d \rightarrow b$	0.10	0.17	0.02

(b) *La table relationnelle* $\Omega = (\mathcal{AR}, \mathcal{M})$

Nom	**Définition**	**Domaine**
$Frequence$	$\frac{supp(X \cup Y)}{\lvert D \rvert}$	[0, 1]
$Confiance$	$\frac{supp(X \cup Y)}{supp(X)}$	[0, 1]
$Pearl$	$\frac{supp(X)}{\lvert D \rvert} \times \lvert \frac{supp(X \cup Y)}{supp(X)} - \frac{supp(Y)}{\lvert D \rvert} \rvert$	[0, 1]

(c) *Un ensemble de mesure* $\mathcal{M}$

TABLE 4.1 – Exemple de contexte d'extraction.

La relation de dominance stricte vérifie les propriétés suivantes :
- **non réfléxivité** : $r \not\succ r$, *i.e.*, $r \succ r$ est fausse pour tout $m \in \mathcal{M}$,
- **transitivité** : $\forall\ r$, r' et $r'' \in \mathcal{R}$, si $r \succ r'$ et $r' \succ r''$ alors $r \succ r''$.

Exemple 23 *Considérons la table relationnelle Ω donnée par le tableau 4.1(b), la règle r_3 domine strictement r_2 puisque $r_3[Freq] \succeq r_2[Freq]$, $r_3[Conf] \succeq r_2[Conf]$ et $r_3[Pearl] \succ r_2[Pearl]$.*

Si une règle r domine une règle r' selon $\mathcal{M}$, cela signifie que r est équivalente ou meilleure que la règle r' pour toutes les mesures de $\mathcal{M}$. La relation de dominance permet alors de comparer deux règles selon un ensemble de mesures, permettant ainsi, de contourner le problème de l'hétérogénéité des évaluations. Les règles dominées par d'autres (au moins une), selon $\mathcal{M}$, ne sont pas pertinentes et doivent être éliminées du résultat final. L'opérateur *skyline* pour les règles d'association formalise cette intuition.

Définition 30 *(Opérateur skyline)*
Soit Ω une table relationnelle. Le skyline de Ω = ($\mathcal{AR}$,$\mathcal{M}$) est défini par :

$$Sky(\Omega) = \{r \in \mathcal{AR} \mid \nexists\ r' \in \mathcal{AR},\ r' \succ r\}$$

En d'autres termes, $Sky(\Omega)$ est l'ensemble des règles qui ne sont dominées par aucune autre règle selon l'ensemble de mesures $\mathcal{M}$.

Exemple 24 *Par exemple, à partir de la table relationnelle Ω présentée par tableau 4.1(b), nous avons $Sky(\Omega) = \{r_1, r_4\}$ puisqu'il n'y a aucune règle dans $\mathcal{AR}$ qui domine strictement r_1 ou r_4.*

4.3.2 Formalisation pour la sélection des règles non dominées

Pour extraire les règles non dominées, nous adoptons le principe de l'approche orientée par la recherche "diviser pour régner" [KRR02] utilisée pour répondre aux requêtes dans les applications de base de données. Dans ce qui suit, nous introduisons la formalisation nécessaire pour l'extraction des règles non dominées. Sur la base de cette formalisation, nous proposons un algorithme, appelé SkyRule, qui met en œuvre l'opérateur skyline.

Afin d'extraire les règles d'association non dominées, une approche naïve consiste à comparer chaque règle r avec toutes les autres règles dans le but de vérifier si r est dominée. Une telle solution est cependant très coûteuse voire irréalisable à cause du nombre de règles qui est souvent très grand. Ainsi, afin de minimiser le coût de la vérification, en termes

de temps d'exécution, il est impératif de réduire le nombre de règles à comparer avec r. A cet effet, nous introduisons, tout d'abord, la notion de *règle référence*.

Définition 31 *(Règle référence)*
Une règle référence $r^{\perp}$ est une règle fictive qui domine toutes les règles de $\mathcal{AR}$ selon $\mathcal{M}$. Formellement : $\forall\ r \in \mathcal{AR},\ r^{\perp} \succeq r$.

Exemple 25 *Par exemple, la règle référence $r^{\perp}$ peut être une règle telle que pour chaque mesure $m \in \mathcal{M}$, $r^{\perp}[m]$ égale à la plus grande valeur parmi les valeurs correspondantes aux différentes règles. Par conséquent, il n'existe aucune règle dans $\mathcal{AR}$ qui domine $r^{\perp}$. À partir de la table relationnelle Ω donnée par le tableau 4.1, $r^{\perp} = \langle 0.2,\ 0.67,\ 0.10 \rangle$.*

Définition 32 *(Degré de similarité)*
Étant donnée deux règles $r,\ r' \in \mathcal{AR}$, le degré de similarité entre r et r' selon $\mathcal{M}$ est défini comme suit :

$$DegSim(r, r') = \frac{\sum_{i=1}^{k} \mid r[m_i] - r'[m_i] \mid}{k}$$

avec k égal au nombre de mesures dans l'ensemble $\mathcal{M}$.

Exemple 26 *Considérons la table relationnelle Ω illustrée par le tableau 4.1(b). On a : $DegSim(r^{\perp},r_1) = 0.02$, $DegSim(r^{\perp},r_2) = 0.12$, $DegSim(r^{\perp},r_3) = 0.11$, $DegSim(r^{\perp},r_4) = 0.09$, $DegSim(r^{\perp},r_5) = 0.14$, $DegSim(r^{\perp},r_6) = 0.11$, $DegSim(r^{\perp},r_7) = 0.22$, $DegSim(r^{\perp},r_8) = 0.23$.*

Le lemme suivant montre qu'en calculant le degré de similarité entre chaque règle d'association et la règle de référence, il est possible d'identifier un sous-ensemble de règles non dominées tout en évitant de comparer les valeurs de mesures qui leurs sont associées avec celles des autres règles.

Lemme 1 *Soient un ensemble $\mathcal{R} \subseteq \mathcal{AR}$ et une règle $r \in \mathcal{AR}$ tel que parmi toutes les règles de $\mathcal{R}$, la règle r possède le degré de similarité minimal avec $r^{\perp}$. Si $\nexists\ r' \in \mathcal{AR} \setminus \mathcal{R}$ tel que $r' \succ r$, alors r est une règle non dominée.*

Preuve 2 *Par l'absurde, on suppose que r soit dominée. Alors il existerait une règle $r'' \in \mathcal{R}$ qui domine strictement r. Ceci signifie que $\forall\ m \in \mathcal{M},\ r''[m] \succeq r[m]$ et $\exists\ m' \in \mathcal{M}$ tel que $r''[m'] \succ r[m']$. Par conséquent, $DegSim(r^{\perp},r'') < DegSim(r^{\perp},r)$. On obtient la contradiction que r ne possède pas le degré de similarité minimal avec $r^{\perp}$.*

Étant donnée une règle $r \in \mathcal{AR}$ telle que r a le degré de similarité minimal avec $r^{\perp}$. Les règles dominées par r peuvent être identifiées en comparant toutes les règles de $\mathcal{AR}$ à r. Toutefois, nous montrons, dans ce qui suit, qu'il est possible de les identifier en comparant r à un sous-ensemble de $\mathcal{AR}$. Cette réduction repose sur le fait que pour toute règle r' non dominée par r tel que $r \not\equiv r'$, il existe au moins une mesure $m \in \mathcal{M}$ tel que $r'[m] \succ r[m]$.

Définition 33 *(Espace non dominé d'une règle)*
Soient une règle $r \in \mathcal{AR}$ et $\mathcal{M} = \{m_1, \ldots, m_k\}$ un ensemble de mesures. L'espace non dominé de r selon une mesure $m_i \in \mathcal{M}$, noté s_i^r, est défini comme l'ensemble de toutes les règles $r' \in \mathcal{AR}$ non dominées par r tel que $r'[m_i] \succ r\ [m_i]$. Nous avons donc :

$$s_i^r = \{r' \in \mathcal{AR} \mid r'[m_i] \succ r\ [m_i]\}$$

Ainsi, un espace non dominé de r est associé à chaque mesure m_i. L'ensemble des espaces non dominé de r, noté $\mathcal{S}^r$, est défini par :

$$\mathcal{S}^r = \{s_i^r\},\ i=1,\ldots,k.$$

Exemple 27 *A partir de notre exemple présenté dans le tableau 4.1(b), nous obtenons pour la règle non dominée r_1 : $s_1^{r_1} = \emptyset$, $s_2^{r_1} = \emptyset$ et $s_3^{r_1} = \{r_4, r_6\}$. L'ensemble $s_1^{r_1}$ est vide parce qu'il n'y a aucune règle possédant une valeur supérieure à celle de r_1 pour la mesure m_1 (idem pour $s_2^{r_1}$). Cependant, $s_3^{r_1}$ contient r_4 et r_6 puisque $r_4[m_3] \succ r_1[m_3]$ et $r_6[m_3] \succ r_1[m_3]$. En suivant le même raisonnement pour la règle non dominée r_4, nous obtenons : $s_1^{r_4} = \emptyset$, $s_2^{r_4} = \{r_1, r_2, r_3\}$ et $s_3^{r_4} = \emptyset$.*

Lemme 2 *Soient $r, r' \in \mathcal{AR}$ deux règles d'association et $s_i^r \in \mathcal{S}^r$ un espace non dominé de r associé à une mesure $m_i \in \mathcal{M}$. Si $r' \notin s_i^r$, alors $\forall\ r'' \in s_i^r$, $r' \not\succ r''$.*

Preuve 3 *La règle r' n'appartient pas à s_i^r, cela signifie que $r'[m_i] \not\succ r[m_i]$. Ainsi, $r[m_i] \succeq r'[m_i]$ (1).*
Soit une règle r'' appartenant à s_i^r. Alors $r''[m_i] \succ r[m_i]$ (2). Selon la transitivité de la dominance, (1) et (2) donnent $r''[m_i] \succ r'[m_i]$. Par conséquent, $r' \not\succ r''$.

Corrolaire 1 *Soient deux règles d'association r, $r' \in \mathcal{AR}$ et $s_i^r \in \mathcal{S}^r$ un espace non dominé de r associé à une mesure $m_i \in \mathcal{M}$ tel que $r' \in s_i^r$. Si, parmi toutes les règles qui appartiennent à s_i^r, la règle r' a le degré de similarité minimal avec $r^{\perp}$, alors $r' \in Sky((\mathcal{AR}, \mathcal{M}))$.*

Preuve 4 *Supposons que $r' \notin Sky((\mathcal{AR}, \mathcal{M}))$, cela signifie qu'il existe une règle $r'' \in \mathcal{AR}$ tel que $r'' \succ r'$. Selon lemme 2, r'' doit appartenir à s_i^r puisque toute règle qui n'appartient pas à s_i^r ne peut dominer r'. Par ailleurs, $\forall\ m \in \mathcal{M}$, $r''[m] \succeq r'[m]$ et $\exists\ m' \in \mathcal{M}$, $r''[m'] \succ r'[m']$. Ainsi, $DegSim(r^\perp,r'') < DegSim(r^\perp,r')$. On obtient la contradiction r' ne possède pas le degré de similarité minimal avec $r^\perp$ parmi les règles appartenant à s_i^r.*

4.3.3 Algorithme SkyRule

En se basant sur la formalisation présentée dans la sous-section précédente, nous introduisons l'algorithme SkyRule qui permet d'extraire les règles non dominées à partir d'une table relationnelle. Dans SkyRule, nous utilisons les variables suivantes lors de l'exécution de l'algorithme :

- La variable *Sky* : est une variable initialisée à l'ensemble vide, elle est utilisée pour garder la trace des règles non dominées.
- La variable C : est une variable qui contient l'ensemble des règles candidates pour être non dominées, elle est initialisée à $\mathcal{AR}$.
- La variable $\mathcal{E}$: est une variable qui contient l'ensemble des règles couvrant l'espace non dominé de toutes les règles non dominées, elle est initialisée à $\mathcal{AR}$.

L'algorithme SkyRule est itératif. Dans chaque itération, il commence par vérifier l'état de l'ensemble des règles candidates C :

- Si C est vide, alors l'algorithme se termine et retourne toutes les règles non dominées à travers la variable *Sky*.
- Sinon, parmi les règles candidates, la règle r^* ayant le degré de similarité minimal avec la règle référence $r^\perp$ est déterminée. Cette règle est alors une règle non dominée et elle est insérée dans *Sky* et supprimée de C (*c.f.,* lemme 1). Par la suite, seul l'espace non dominé qui contient r^* sera exploré en comparant chaque règle r dans cet espace à r^*. Deux cas sont à distinguer :

  1. Si r est dominée par r^*, alors r n'est plus une règle candidate et elle est retirée de C.

  2. Sinon, r n'est pas dominée par r^*, *i.e.,* r est encore une règle candidate et elle est ajoutée à l'espace non dominé de r^* (selon la définition 33)

  L'espace non dominé contenant r^* est retiré de $\mathcal{E}$ et l'espace non dominé de r^* est ajouté à $\mathcal{E}$. Ce processus prend fin lorsque tous les candidats sont traités.

Données : $\Omega = (\mathcal{AR}, \mathcal{M})$: table relationnelle
Résultats : Sky : ensemble des règles non dominées de Ω.

début
  $Sky \leftarrow \emptyset$
  $C \leftarrow \mathcal{AR}$
  $\mathcal{E} \leftarrow \{\mathcal{AR}\}$
  tant que $C \neq \emptyset$ **faire**
    $r^* \leftarrow r \in C$ ayant $min(DegSim(r,r^{\perp}))$
    $C \leftarrow C \backslash \{r^*\}$
    pour *i=1 à k* **faire**
      $s_i^{r^*} \leftarrow \emptyset$
    $Sky \leftarrow Sky \cup \{r^*\}$
    pour chaque $e \in \mathcal{E}$ *tel que* $r^* \in e$ **faire**
      pour chaque $r \in e$ **faire**
        si $r^* \succ r$ **alors**
          $C \leftarrow C \setminus \{r\}$
        sinon
          pour *i=1 à k* **faire**
            si $r[m_i] \succ r^*[m_i]$ **alors**
              $s_i^{r^*} \leftarrow s_i^{r^*} \cup \{r\}$
      $\mathcal{E} \leftarrow \mathcal{E} \setminus \{e\}$
    $\mathcal{E} \leftarrow \mathcal{E} \cup \{s_1^{r^*}, \ldots, s_k^{r^*}\}$
  retourner Sky
fin

Algorithme 5 : SkyRule

4.3.3.1 Correction et Complétude de l'algorithme SkyRule

Le théorème suivant prouve l'exactitude et la complétude de l'algorithme SkyRule.

Théorème 4.3.1 *L'algorithme* SkyRule *est correct et complet.*

Preuve 5 *Pour prouver la correction de l'algorithme* SkyRule, *nous devons montrer qu'il génère uniquement des règles non dominées. Par l'absurde, supposons qu'à la fin de l'exécution de l'algorithme, l'ensemble Sky comprend une règle dominée r. Cela signifie que la règle r a été sélectionnée comme étant la règle ayant le degré de similarité minimal avec la règle référence $r^{\perp}$, parmi toutes les règles candidates dans C et qu'il n'existe aucune règle dans $\mathcal{AR} \setminus C$ qui la domine. Ainsi, il existe une règle $r' \in C$ qui domine r. On obtient la contradiction r ne possède pas le degré de similarité minimal avec la règle référence $r^{\perp}$, parmi toutes les règles candidates dans C.*

L'algorithme SkyRule *est complet puisqu'il génère toutes les règles non dominées. En effet, pour toute itération, chaque règle candidate r dans C est soit :*

1. *insérée dans l'ensemble des règles non dominées Sky, si son degré de similarité avec $r^{\perp}$ est minimal,*
2. *éliminée, si elle est dominée par la règle ayant le degré de similarité minimal,*
3. *retenue dans l'ensemble candidat C, si elle n'est pas dominée par la règle ayant le degré de similarité minimal.*

L'algorithme SkyRule *se termine lorsque l'ensemble des candidats C est vide. Ainsi,* SkyRule *génère toutes les règles non dominées.*

4.3.4 Expérimentations

Dans cette section, nous allons mener une série d'expérimentations afin d'évaluer, d'une part, le taux de compacité des règles non dominées par rapport au nombre total de règles et au nombre de règles qui peuvent être extraites par une approche utilisant des seuils pour les mesures. D'autre part, nous évaluons la performance de l'algorithme SkyRule.

Ces expérimentations ont été réalisées sur des bases de données de référence provenant de l'UCI Machine Learning Repository. Le tableau 4.2 présente les caractéristiques de ces bases de données. Toutes les expérimentations ont été effectuées sur un processeur Intel 1.73 GHz avec le système d'exploitation Linux en utilisant 2 Go de mémoire vive. Les différents programmes ont été implémentés en langage C++.

Bases	♯ *items*	♯ *transactions*	*TailleMoyenne des transactions*
Diabete	75	3196	37
Flare	39	1389	10
Iris	4	150	3
*Monks*1	19	124	7
*Monks*2	19	169	7
*Monks*3	19	122	7
Nursery	32	12960	9
Zoo	42	101	9

TABLE 4.2 – Caractéristiques des données de test.

4.3.4.1 Réduction du nombre des règles

Dans ce qui suit, nous rapportons les expérimentations menées pour montrer l'intérêt de notre approche d'extraction de règles d'association non dominées. Afin de montrer la concision apportée par notre méthode, nous comparons le nombre de règles non dominées extraites par SKYRULE avec, respectivement, le nombre total de règles et le nombre de règles qui peuvent être générées par une approche optimale d'extraction sous contraintes. Nous considérons qu'une approche d'extraction sous contraintes est optimale, si les experts sont capables de déterminer, pour chaque mesure utilisée, le seuil optimal qui permet de valider toutes les règles non dominées. Ainsi, pour mettre en œuvre cette approche, nous attribuons à chaque mesure $m \in \mathcal{M}$, un seuil ε_m tel que ε_m est la valeur minimale que pourrait avoir m pour une règle non dominée, *i.e.*, $\varepsilon_m = min\{r[m] \mid r \in Sky((\mathcal{AR}, \mathcal{M}))\}$. Cela garantit qu'aucune règle non dominée ne sera omise du résultat. A partir de notre exemple illustré par le Tableau 4.1(b)), $\varepsilon_{freq} = 0.20$, $\varepsilon_{conf} = 0.40$ et $\varepsilon_{pearl} = 0.02$. L'ensemble des règles extraites par l'approche optimale sous contraintes est appelé *règles optimales sous contraintes*, noté *ROSC*. Même si cette approche semble en pratique irréaliste (les experts doivent deviner les seuils optimaux), nous pensons que ces expérimentations ont l'avantage de quantifier la réduction des règles apportées par SKYRULE même pour un scénario idéal où les experts sont capables de gérer parfaitement les seuils de sélection dans un paradigme d'extraction sous contraintes.

Dans nos expérimentations, nous considérons un nombre de combinaisons des mesures suivantes : Confiance [AIS93b], Rappel [LFZ99], Pearl [Pea88], Loevinger [Loe47], Zhang [Zha00]. Les définitions des ces mesures sont présentées par le tableau 4.3.

Nom	Définition
$Confiance$	$\frac{supp(X \cup Y)}{supp(X)}$
$Rappel$	$\frac{supp(X \cup Y)}{supp(Y)}$
$Pearl$	$\frac{supp(X)}{\|D\|} \times \|\frac{supp(X \cup Y)}{supp(X)} - \frac{supp(Y)}{\|D\|}\|$
$Loevinger$	$1 - \frac{supp(X \cup \overline{Y})}{supp(X) \times supp(\overline{Y})}$
$Zhang$	$\frac{supp(X \cup Y) - supp(X) \times supp(Y)}{max\{supp(X \cup Y) \times supp(\overline{Y}), supp(X \cup \overline{Y}) \times supp(Y)\}}$

TABLE 4.3 – Définitions des mesures de qualité

Bases (min_{freq} %)		{Conf;Loev}	{Conf;Pearl}	{Conf;Rappel}	{Conf;Zhang}	{Conf;Pearl Rappel}	{Conf;Loev Zhang}	{Conf;Loev ;Pearl Rappel ;Zhang}
Diabetes (10,00)	Sky-R	3411	9	6651	2996	9	171	171
	ROSC	59314	58124	59206	59309	44813	44602	42126
	AR	62132	62132	62132	62132	62132	62132	62132
Flare (10,00)	Sky-R	4975	48	4978	4857	48	48	48
	ROSC	56163	57101	56451	54524	53197	53116	52819
	AR	57476	57476	57476	57476	57476	57476	57476
Iris (0,00)	Sky-R	246	246	246	246	246	246	246
	ROSC	440	440	440	440	440	440	440
	AR	440	440	440	440	440	440	440
Monks1 (1,00)	Sky-R	768	1	788	656	1	1	1
	ROSC	60417	60692	59418	59452	58904	58811	58327
	AR	62184	62184	62184	62184	62184	62184	62184
Monks2 (1,00)	Sky-R	279	3	215	202	3	3	3
	ROSC	59611	59702	59568	59544	59103	58917	58662
	AR	59976	59976	59976	59976	59976	59976	59976
Monks3 (1,00)	Sky-R	1028	2	713	781	4	2	2
	ROSC	58662	58369	57922	58436	57816	57734	56038
	AR	59304	59304	59304	59304	59304	59304	59304
Nursery (2,00)	Sky-R	497	2	304	342	8	2	2
	ROSC	23872	23901	23875	23417	23176	22806	22139
	AR	25062	25062	25062	25062	25062	25062	25062
Zoo (10,00)	Sky-R	9784	36	9415	9112	36	36	36
	ROSC	67991	67305	67872	66146	65328	65116	63926
	AR	71302	71302	71302	71302	71302	71302	71302

TABLE 4.4 – Variation des Règles non dominées *Sky-R vs* règles optimales sous contraintes ($ROSC$) et toutes les règles AR

A la lecture du tableau 4.4, nous constatons que :

- Pour toutes les bases et toutes les combinaisons de mesures utilisées, le nombre de règles non dominées est beaucoup plus petit que celui de toutes les règles d'association. Par exemple, le nombre de règles est 62184 fois plus grand que celui des règles non dominées pour la base Monks 1 et la combinaison {Confiance, Pearl}.
- Même en utilisant une approche optimale d'extraction sous contraintes, le nombre de règles générées reste tout de même important par rapport à celui des règles non dominées.

Le Tableau 4.5 résume ce résultat en montrant pour chaque combinaison de mesures, le nombre moyen des règles non dominées, le nombre moyen des $ROSC$ et le taux de gain

Mesures	Nombre moyen de Sky-R	Nombre moyen de ROSC	Gain moyen de Sky-R
{Conf ;Loev}	2623,50	48308,75	18,41
{Conf ;Pearl}	43,37	47908,12	943,23
{Conf ;Rappel}	2913,75	48094,00	16,50
{Conf ;Zhang}	2399,00	47658,50	19,86
{Conf ;Loev ;Rappel}	43,37	45347,12	1045,58
{Conf ;Pearl ;Zhang}	63,62	45192,75	710,35
{Conf ;Loev ;Pearl ;Rappel ;Zhang}	63,62	44309,62	696,47

TABLE 4.5 – Nombre moyen des règles non dominées, Nombre moyen des ROSC et Gain moyen des règles non dominées

moyen des règles non dominées par rapport à *ROSC*. Le taux moyen de gain est mesuré comme suit : $\frac{taille\ de\ ROSC}{taille\ de\ Sky\text{-}R}$.

4.3.4.2 Evolution du nombre de règles non dominées suite à la variation des mesures

Le tableau 4.4 montre également l'évolution du nombre des règles optimales sous contraintes et celui des règles non dominées lorsque nous faisons varier le nombre de mesures considérées.

A la lecture du tableau 4.4, nous remarquons que :

- Le nombre des règles optimales sous contraintes diminue suite à chaque ajout d'une mesure. Cette diminution s'explique par le fait que les règles doivent satisfaire une contrainte supplémentaire pour être retenues. Cette contrainte est, en effet, le seuil de la mesure rajoutée.
- Cependant, en augmentant le nombre de mesures, le nombre de règles non dominées n'évolue pas d'une manière linéaire. Il peut diminuer ou augmenter. La diminution s'explique par le fait qu'une règle peut être non dominée selon un ensemble de mesures M_1 et dominée selon M_2, tel que $M_1 \subset M_2$. Par exemple, si deux règles r et r' sont équivalentes par rapport à M_1, il se peut que l'une d'entre elles domine l'autre en considérant une mesure de plus. Cependant, l'augmentation peut être expliquée par le fait qu'une règle peut être dominée selon M_1 et non dominée selon M_2. Par exemple, considérons une règle r qui domine une autre r' selon M_1, en ajoutant une mesure m à M_1, tel que $r'[m] \succ r[m]$, la règle r ne domine plus r'.

Nombre de règles	{Conf ;Loev}	{Conf ;Loev ; Zhang}	{Conf ;Loev ;Pearl ; Rappel ;Zhang}
Diabete			
10000	0,188s	0,121s	0,142s
20000	0,462s	0,262s	0,382s
30000	1,284s	0,967s	1,178s
40000	2,885s	1,951s	2,294s
50000	4,177s	3,211s	3,861s
Flare			
10000	0,264s	0,273s	0,277s
20000	0,547s	0,831s	0,912s
30000	1,341s	1,692s	1,778s
40000	3,638s	3,944s	4,012s
50000	5,416s	5,916s	6,181s
Monks 1			
10000	0,033s	0,121s	0,142s
20000	0,157s	0,262s	0,382s
30000	0,854s	0,967s	1,178s
40000	1,613s	1,951s	2,294s
50000	2,087s	3,211s	3,861s
Monks 2			
10000	0,089s	0,093s	0,183s
20000	0,369s	0,346s	0,688s
30000	0,943s	0,862s	1,148s
40000	1,815s	1,628s	1,916s
50000	2,293s	2,206s	2,459s
Monks 3			
10000	0,075s	0,289s	0,305s
20000	0,248s	0,411s	0,633s
30000	0,883s	0,993s	1,288s
40000	1,692s	1,850s	1,992s
50000	2,192s	2,327s	2,691s
Nursery			
10000	0,081s	0,089s	0,094s
20000	0,169s	0,183s	0,296s
30000	0,690s	0,566s	0,835s
40000	1,027s	1,181s	1,204s
50000	1,723s	1,738s	1,933s
Zoo			
10000	1,182s	1,346s	1,428s
20000	2,447s	2,627s	2,396s
30000	3,262s	3,519s	4,104s
40000	4,035s	4,288s	5,262s
50000	4,924s	5,013s	6,141s

TABLE 4.6 – Temps d'extraction des règles non dominées.

4.3.4.3 Analyse de temps d'exécution

Dans cette sous-section, nous étudions la variation du temps d'exécution de SkyRule suite à la variation du nombre de règles. Pour cela, nous considérons des échantillons de règles dont la taille varie de 10000 à 50000 règles. Le tableau 4.6 présente la durée d'exécution de SkyRule sur les échantillons en utilisant un nombre de combinaisons des mesures : *Confiance*, *Rappel*, *Pearl*, *Loevinger*, *Zhang*.

Un résultat important est que le temps d'exécution de SkyRule augmente linéairement avec le nombre de règles en entrée ce qui démontre l'efficacité de la méthode. Toutefois, le temps d'exécution peut diminuer ou augmenter en faisant varier la cardinalité de l'ensemble des mesures. Ceci peut être expliqué par le fait que le temps d'exécution de SkyRule dépend du nombre de règles non dominées extraites qui peut à son tour diminuer ou augmenter suite à l'ajout d'une ou plusieurs mesures.

4.4 Sélection des k meilleures règles d'association

Dans la section précédente, nous avons proposé une approche permettant de sélectionner les règles d'association selon plusieurs mesures. Ces règles expriment, en effet, le meilleur compromis entre les préférences d'un ou plusieurs experts. Toutefois, ces experts peuvent demander de sélectionner un nombre bien précis k de règles pertinentes. Dans la littérature, plusieurs approches ont été proposées permettant de sélectionner les k meilleures règles selon une seule mesure [FVWT12, Web11]. Cependant, aucune de ces approches ne permet de résoudre le problème en présence de plusieurs experts ayant chacun une préférence pour une ou plusieurs mesures de qualité.

Dans cette section, nous proposons une approche permettant de sélectionner les k meilleures règles selon plusieurs mesures en se basant sur la relation de dominance. A cet effet, nous distinguons les deux cas suivants :

1. si k est inférieur au nombre de règles non dominées, cela signifie qu'il faut sélectionner un sous-ensemble de règles parmi l'ensemble des règles non dominées. Dans ce cas, la question qui se pose est : Quelles règles non dominées doit-on choisir pour les présenter aux experts ?
2. si k est supérieur au nombre de règles non dominées, cela signifie qu'il faut rajouter à l'ensemble des règles non dominées des règles dominées. Dans ce cas, la question qui se pose est : Quelles règles dominées doit-on choisir pour les rajouter aux règles non dominées ?

Déterminer un ordre total sur les règles selon la pertinence peut être une réponse à ces deux questions. En effet, si les règles sont ordonnées des plus pertinentes aux moins pertinentes, alors il serait facile de déterminer les k premières règles. Ainsi, nous proposons de compléter l'approche de sélection des règles non dominées par un processus de classement de règles selon plusieurs mesures, tout en satisfaisant les conditions suivantes :

- Toute règle non dominée doit être mieux classée qu'une règle dominée, même si elle ne la domine pas.
- Une règle r doit être mieux classée que toutes les règles qu'elle domine.
- Les règles non dominées doivent être ordonnées suivant le degré de similarité par rapport à la règle de référence (ou "optimale") $r^{\perp}$. Cette condition est inspirée de la méthode TOPSIS développée par Yoon et Hwang [HY81] qui consiste à sélectionner l'alternative ayant la distance minimale avec la solution idéale positive et la distance maximale avec la solution idéale négative. L'intuition se cachant derrière cette condition est qu'une règle non dominée r ayant un degré de similarité inférieur à celui d'une règle non dominée r' signifie que r est plus proche de la règle "optimale" que r'.

4.4.1 Ordonnancement des règles selon plusieurs mesures

Dans ce qui suit, nous proposons un processus d'ordonnancement de règles d'association selon plusieurs mesures qui vérifiant les conditions évoquées précédemment. Ce processus repose essentiellement sur la relation de succession définie comme suit :

Définition 34 *(Règle successeur)*
Soient deux règles r, $r' \in \mathcal{AR}$, on dit que r succède immédiatement à r', notée $r \lhd r'$ ssi $r' \succ r$ et $\nexists\, r''$ tel que $r' \succ r'' \succ r$.

Exemple 28 *Considérons la table de relation Ω dans le Tableau 4.1(b), on a $r_6 \lhd r_4$ mais $r_5 \not\lhd r_4$ puisque $r_4 \succ r_6 \succ r_5$.*

Ainsi, pour satisfaire les conditions évoquées précédemment, une règle r doit être mieux classée qu'une règle r' qui la succède immédiatement.

Remarque 1 *Une règle non dominée ne peut pas succéder à une règle dominée.*

Définition 35 *(Opérateur successeur)*
Soit un ensemble de règles $E \subseteq \mathcal{AR}$. L'ensemble successeur immédiat de E dans $\mathcal{AR}$ selon $\mathcal{M}$ est défini comme suit :

$$Succ_{\mathcal{M}}(E,\mathcal{AR}) = \{r \in \mathcal{AR} \setminus E \mid \exists\, r' \in E,\, r \lhd r' \wedge \nexists\, r'' \in E,\, (r'' \succ r \wedge r \not\lhd r'')\}.$$

En d'autres termes, l'ensemble successeur de E contient les règles successeurs immédiat des règles de E tel que pour toute paire (r, r') appartenant à l'ensemble successeur de E, r ne peut jamais dominer r'.

Exemple 29 *Si nous considérons la table relationnelle Ω illustrée par le tableau 4.1(b) et on suppose que $E = \{r_1, r_4\}$. Alors, on a $r_1 \succ r_3 \succ r_2$, $r_1 \succ r_5 \succ r_7$, $r_5 \succ r_8$ et $r_4 \succ r_6$ $\succ r_5$ alors $Succ_{\mathcal{M}}(E,\mathcal{AR}) = \{r_3, r_6\}$. Notons que $r_5 \lhd r_1$, $r_5 \notin Succ_{\mathcal{M}}(E, \mathcal{AR})$ puisque $r_5 \not\lhd r_4$.*

Lemme 3 *Étant donné un ensemble de règles $E \subseteq \mathcal{AR}$, la relation suivante est satisfaite :*

$$Succ_{\mathcal{M}}(Sky(E, \mathcal{M}), E) = Sky(E \backslash Sky(E, \mathcal{M}), \mathcal{M})$$

Preuve 6 *Soit E un ensemble de règles tel que $E \subseteq \mathcal{R}$:*
1. D'abord, nous devons montrer que $Succ_{\mathcal{M}}(Sky(E, \mathcal{M}), E) \subseteq Sky(E \backslash Sky(E, \mathcal{M}), \mathcal{M})$: étant donné, $r \in Succ_{\mathcal{M}}(Sky(E, \mathcal{M}), E)$ alors $r \in E \setminus Sky(E, \mathcal{M})$. Pour toute $r' \in Sky(E, \mathcal{M})$, nous distinguons deux cas :
- *Si $r' \succ r$, alors $r \lhd r'$ ce qui signifie que $\nexists\, r'' \in E \setminus Sky(E, \mathcal{M})$ tel que $r' \succ r'' \succ r$.*
- *Si $r' \not\succ r$, alors $\nexists\, r''$ dans $E \setminus Sky(E, \mathcal{M})$ tel que $r' \succ r''$ et $r'' \succ r$*

Par conséquent, aucune règle de $E \setminus Sky(E, \mathcal{M})$ ne peut dominer r i.e., $r \in Sky(E \setminus Sky(E, \mathcal{M}), \mathcal{M})$.
*2. Ensuite, nous devons montrer que $Succ_{\mathcal{M}}(Sky(E, \mathcal{M}), E) \supseteq Sky(E \backslash Sky(E, \mathcal{M}), \mathcal{M})$: étant donné $r \in Sky(E \backslash Sky(E, \mathcal{M}), \mathcal{M})$ alors $\nexists\, r' \in E \backslash Sky(E, \mathcal{M})$ tel que $r' \succ r$ **(a)**. En outre, puisque $r \in E \backslash Sky(E, \mathcal{M})$ alors $\exists\, r'' \in Sky(E, \mathcal{M})$ tel que $r'' \succ r$ **(b)**. Par conséquent, selon **(a)** et **(b)**, $r \lhd r''$ **(c)**.*
*Par ailleurs, on suppose que $\exists\, r' \in Sky(E, \mathcal{M})$ tel que $r_1 \succ r$ et $r \not\lhd r_1$, alors $\exists\, r_2 \in E \backslash Sky(E, \mathcal{M})$ tel que $r_1 \succ r_2 \succ r$ qui contredit notre hypothèse (voir **(a)**). Par conséquent, $\nexists\, r_2 \in E \backslash Sky(E, \mathcal{M})$ tel que $r_1 \succ r_2 \succ r$ **(d)**. Selon **(c)** et **(d)**, r appartient nécessairement à $Succ_{\mathcal{M}}(Sky(E, \mathcal{M}), E)$.*

Ce lemme montre qu'il est possible de déterminer les successeurs immédiats d'un ensemble de règles non dominées associées à E en appliquant l'opérateur Sky sur les règles dominées de E, *i.e.*, $E \backslash Sky(E, \mathcal{M})$.

L'idée principale du processus d'ordonnancement des règles, que nous proposons, consiste à :

- classer l'ensemble des règles non dominées en premier lieu, puisqu'il ne succède à aucun autre ensemble,
- ordonner les règles non dominées selon le degré de similarité par rapport à la règle de référence $r^{\perp}$.
- classer les règles d'un ensemble E juste avant les règles de l'ensemble successeur de E,
- ordonner les règles appartenant à un même ensemble successeur selon le degré de similarité par rapport à la règle de référence $r^{\perp}$.

4.4.2 Algorithme RankRule

Dans ce qui suit, nous proposons un algorithme, appelé RankRule, qui permet de sélectionner les k meilleures règles selon plusieurs mesures en se basant sur le processus d'ordonnancement décrit ci-dessus.

Données : $\Omega = (\mathcal{AR}, \mathcal{M})$: table relationnelle, k : nombre de règles
Résultats : Ensembles ordonnés de règles ordonnées

```
1 début
2     p ← 0
3     tant que k > 0 et AR ≠ ∅ faire
4         p ← p + 1
5         E_p ← SkyRule((AR, M), k)
6         AR ← AR \ E_p
7         k ← k - |E_p|
8     retourner (E_1, . . ., E_p)
9 fin
```

Algorithme 6 : RankRule

L'algorithme RankRule prend en entrée une table relationnelle Ω=($\mathcal{AR}$, $\mathcal{M}$) et un nombre de règles à sélectionner k. Il procède de la manière suivante : dans la première itération, l'algorithme commence par sélectionner les règles non dominées qui seront classées en premier lieu puisqu'elles ne succèdent à aucune autre règle. Ces règles sont déjà ordonnées puisque SkyRule [1] sélectionne les règles non dominées en ordre selon le degré de similarité par rapport à la règle de référence. Dans la deuxième itération, l'algorithme RankRule applique SkyRule sur les règles non encore classées afin de déterminer les

1. L'algorithme SkyRule prend en entrée le nombre k en plus de la table relationnelle, pour qu'il n'extrait pas des règles en excès.

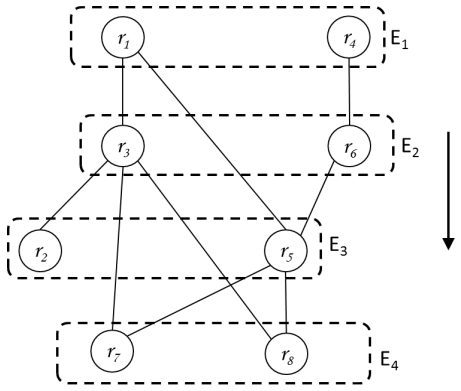

FIGURE 4.1 – Résultat de l'application de RANKRULE sur la table relationnelle Ω donnée par le Tableau 4.1(b).

règles qui seront classées en deuxième lieu, *i.e.,* les règles successeurs immédiats des règles classées en premier lieu. Le processus est réitéré jusqu'à ce qu'on obtienne k règles ou lorsque toutes les règles sont traitées.

Exemple 30 *Dans cet exemple, nous appliquons* RANKRULE *sur* Ω *illustré par le tableau 4.1(b) avec* $k = 6$*. Puisque,* r_1 *et* r_4 *sont des règles non dominées alors* $E_1 = \{r_1, r_4\}$*. Ensuite, en ignorant les règles* r_1 *et* r_4 *les règles* r_3 *et* r_6 *sont considérées comme non dominées. En effet,* r_3 *est uniquement dominée par* r_1 *et* r_6 *est uniquement dominée par* r_4*, alors* $E_2 = \{r_3, r_6\}$*. Ensuite, en ignorant aussi les règles* r_3 *et* r_6*, les règles* r_2 *et* r_5 *sont considérées comme non-dominées. En effet,* r_2 *est uniquement dominée par* r_3*, et* r_5 *est dominée par* r_1 *et* r_6*, alors* $E_3 = \{r_2, r_5\}$*. Cet exemple est illustré par la Figure 4.1. La flèche indique le sens du processus qui commence à partir des règles non dominées.* E_1 *contient les règles les mieux classées qui sont eux-mêmes ordonnées au sein de* E_1 *de gauche à droite selon le degré de similarité par rapport à la règle référence. Par conséquent, la règle* r_1 *est mieux classée que la règle* r_4*.*

4.4.3 Dualité

A l'opposé de la solution proposée pour ordonner les règles selon plusieurs mesures qui consiste à commencer par classer les règles non dominées en premier pour ensuite classer le reste des règles (des plus intéressantes aux moins intéressantes), une deuxième solution est envisageable, qui consiste à commencer par classer les règles qui ne dominent aucune autre règle en dernier lieu pour ensuite déterminer les plus intéressantes (des moins

intéressantes aux plus intéressantes). Nous expliquons le fonctionnement de cette solution à travers l'exemple suivant :

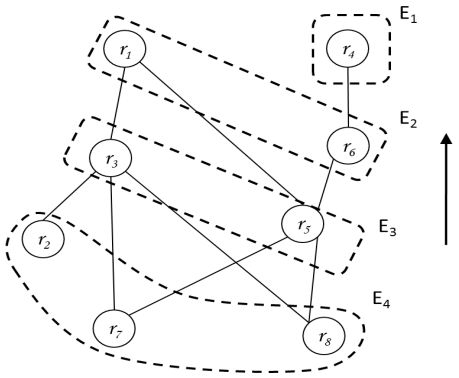

FIGURE 4.2 – Approche duale appliquée sur Ω donnée dans le tableau 4.1(b).

Exemple 31 *Considérons Ω du Tableau 4.1 (b). Tout d'abord, nous identifions l'ensemble des règles qui ne dominent aucune autre règle. Ces règles sont r_2, r_7 et r_8, alors on a $E_4 = \{r_2, r_7, r_8\}$. En ignorant r_2, r_7 et r_8. Les règles qui ne dominent aucune autre règle sont r_3 et r_5. En effet, r_3 domine uniquement r_2, r_7 et r_8, et r_5 domine uniquement r_7 et r_8, alors $E_3 = \{r_3, r_5\}$. En ignorant r_3 et r_5, les règles qui ne dominent aucune autre règle sont r_1 et r_6 puisqu'elles dominent r_3 et r_5 respectivement, alors $E_2 = \{r_1, r_6\}$. Finalement, $E_1 = \{r_4\}$. Cet exemple est illustré par la Figure 4.2.*

Toutefois, cette solution est très difficile, voire impossible à appliquer dans la sélection des k meilleures règles. En effet, toutes les règles dont le nombre est souvent très grand, doivent être ordonnées pour pouvoir ensuite déterminer celles qui sont classées dans les k premières positions.

4.4.4 Expérimentations

Ces expérimentations poursuivent deux objectifs. En premier lieu, nous souhaitons étudier le comportement de l'algorithme RANKRULE. En second lieu, nous voulons analyser empiriquement certaines mesures de qualité en utilisant la sélection des k meilleures règles. Ces expérimentations sont menées sur des bases de données de UCI Machine Learning Repository utilisées dans la section précédente.

4.4.4.1 Étude du comportement de l'algorithme RankRule

Dans ce qui suit, nous allons étudier le comportement de l'algorithme RankRule suite à la variation du :

- Nombre de règles d'association ;
- Nombre des k meilleures règles d'association.

Effet de la variation du nombre de règles

Dans ce qui suit, nous étudions l'effet de la variation du nombre de règles sur le comportement de RankRule. Pour ce faire, nous avons pour chaque couple de mesures suivantes : {Conf;Loev}, {Conf;Pearl}, {Conf;Rappel} et {Conf;Zhang}, exécuté RankRule sur 10000, 20000, 30000, 40000, 50000 règles. Ces règles sont sélectionnées d'une manière arbitraire parmi les règles extraites à partir des bases de données utilisées. La figure 4.3 présente les temps d'exécution enregistrés pour $k = 100$, un nombre de règles qui semble raisonnable pouvant être exploré par un utilisateur. A la lecture des courbes de la figure 4.3, nous pouvons constater qu'en augmentant le nombre de règles, le temps d'exécution de RankRule augmente légèrement pour les couples {Conf;Loev}, {Conf;Rappel} et {Conf;Zhang}. Par exemple, en passant de 10000 à 50000 règles, l'écart maximal entre les temps d'exécution enregistrés pour le couple {Conf;Loev} est de 4 secondes, pour {Conf;Pearl} est de 6 secondes et pour {Conf;Zhang} est de 8 secondes. Cependant, pour le couple {Conf;Pearl}, le temps d'exécution augmente considérablement, il dépasse les 20 secondes pour la base Monsk 1. Ceci peut être expliqué par le fait que le nombre de règles non dominées est souvent très petit pour le couple {Conf;Pearl} (*c.f.,* Tableau 4.4). Par conséquent, l'algorithme RankRule aura besoin d'effectuer plusieurs appels à SkyRule pour arriver à sélectionner les 100 meilleures règles. Au contraire, pour les autres couples, le nombre des règles non dominées dépasse 100. Ainsi, un seul appel à SkyRule est suffisant pour déterminer les 100 meilleures règles.

Effet de la variation de k

Afin d'étudier l'effet de la variation de k, nous présentons dans la figure 4.4 les temps d'exécution de l'algorithme RankRule en faisant varier k de 50 à 500. A la lecture des courbes de la figure 4.4, nous pouvons constater que :

- L'exécution de RankRule en utilisant le couple {Conf;Pearl} prend plus de temps au fur et à mesure qu'on augmente la valeur de k. Nous expliquons cette sensibilité comme suit. En regardant le tableau 4.4, nous remarquons que le nombre de règles

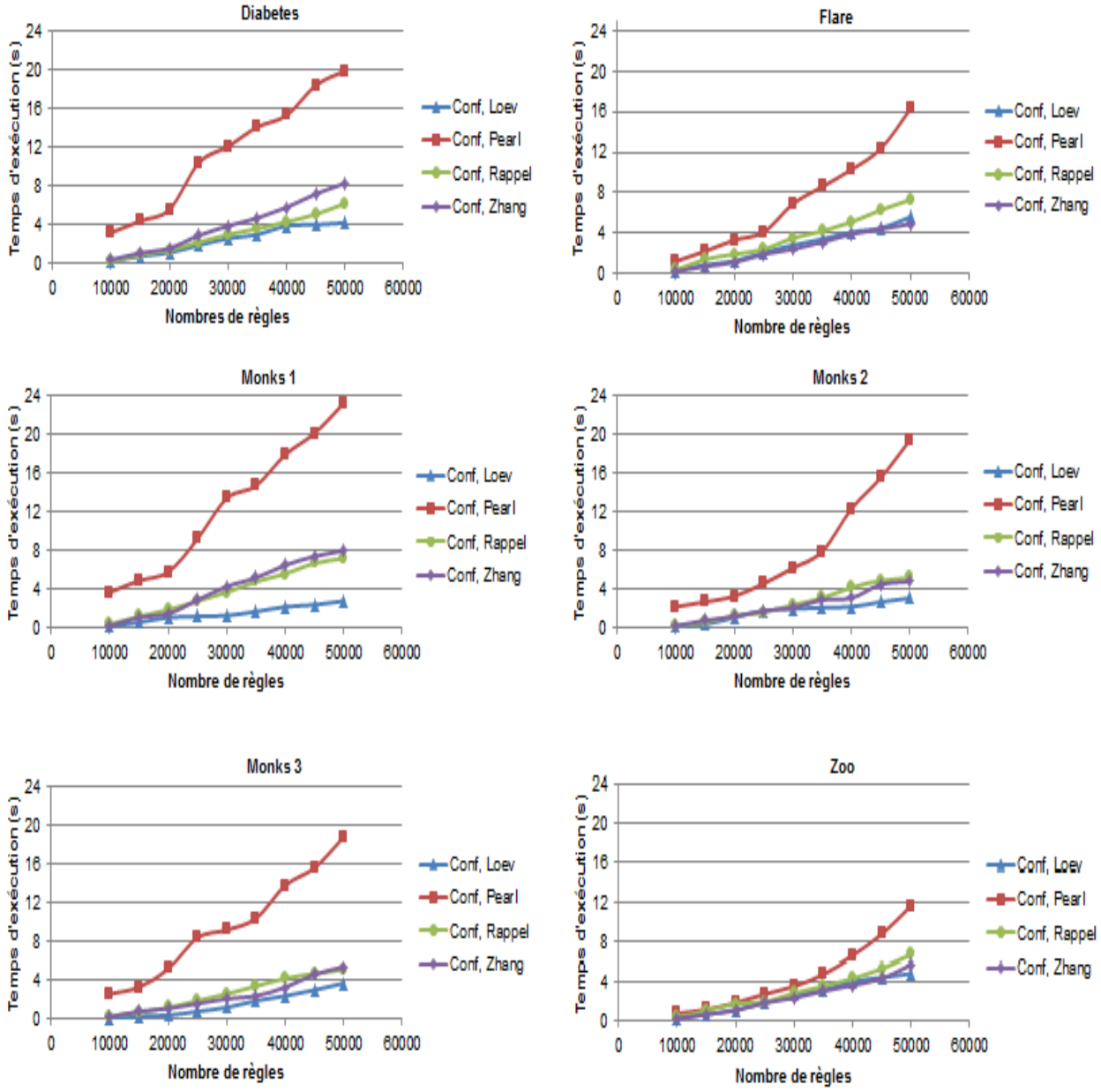

FIGURE 4.3 – Temps d'exécution de RANKRULE suite à la variation du nombre de règles.

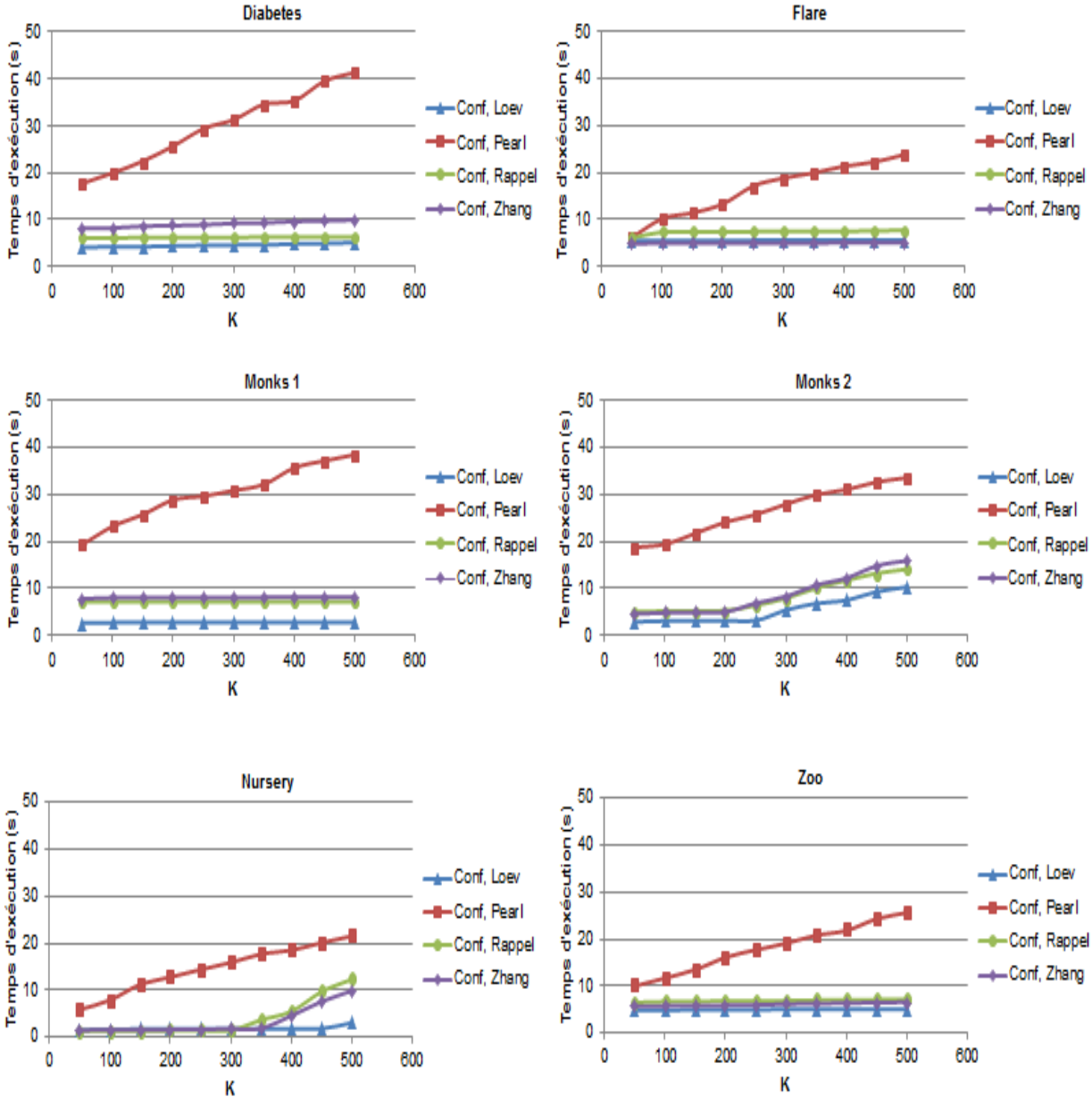

FIGURE 4.4 – Temps d'exécution de RANKRULE suite à la variation de k

non dominées est souvent très petit pour le couple {Conf;Pearl}, il varie de 2 à 36. Ainsi, l'algorithme RANKRULE fera plus d'appels à SKYRULE à chaque fois que la valeur de k augmente.

- En utilisant les couples {Conf;Loev}, {Conf;Rappel} et {Conf;Zhang}, le temps d'exécution de RANKRULE reste quasiment constant pour les bases Diabetes, Flare, Monks1 et Zoo lorsque k augmente. Ceci est dû au fait que le nombre des règles non dominées est supérieur à 500 pour toutes les bases (*c.f.,* figure 4.4). Par conséquent, l'algorithme RANKRULE fera un seul appel à SKYRULE et un simple calcul de degré de similarité pour déterminer les k meilleures règles.
- Les temps d'exécution de RANKRULE en utilisant le couple {Conf;Rappel} pour la base Monks 2, commence à augmenter à partir de k=250. Ceci peut être expliqué par le fait que le nombre de règles non dominées est compris entre 200 et 250 pour la base Monks 2. Ainsi, pour toute valeur de k inférieure à 250, l'algorithme RANKRULE fera un seul appel à SKYRULE. Tandis qu'à partir de k=250 RANKRULE fera plus d'appels à SKYRULE au fur et à mesure qu'on augmente la valeur de k. Pour la même raison, le temps d'exécution de RANKRULE commence à augmenter à partir de k = 350 pour la base Nursery en utilisant le couple {Conf;Zhang}.

4.4.4.2 Vers une analyse empirique de mesures en utilisant la sélection des k meilleures règles

Dans la littérature, plusieurs travaux de synthèse ont été réalisés afin d'analyser le comportement des mesures de qualité dans le but de faire ressortir leurs ressemblances et leurs divergences. Dans la sous section 4.2.2, nous avons classé ces travaux en deux catégories.

1. La première regroupe les travaux qui se sont basés sur des études formelles dont l'objectif est de comparer les mesures selon un ensemble de propriétés. Les mesures qui possèdent des propriétés similaires sont regroupées dans une même classe.
2. La deuxième regroupe les travaux qui se sont basés sur des études expérimentales dont l'objectif est de comparer les différentes évaluations des mesures sur des bases de règles. Les mesures ayant des évaluations similaires sont regroupées dans une même classe.

Dans le cadre de cette deuxième catégorie, nous proposons une démarche expérimentale dont l'objectif est de valider les résultats de la classification formelle. Cette démarche consiste à vérifier en pratique si une mesure m appartient bien à une classe de mesures C

Classes	{Conf, Loev}	{Conf, Zhang}	{Conf, Loev, Zhang}
Diabetes			
Confiance	89%	84%	78%
Loevinger	64%	53%	51%
Zhang	57%	66%	32%
Flare			
Confiance	86%	63%	44%
Loevinger	71%	52%	33%
Zhang	44%	61%	28%
Monks 1			
Confiance	91%	74%	66%
Loevinger	77%	48%	29%
Zhang	32%	59%	16%
Monks 2			
Confiance	93%	68%	54%
Loevinger	71%	46%	22%
Zhang	36%	61%	21%
Monks 3			
Confiance	88%	61%	56%
Loevinger	63%	38%	19%
Zhang	31%	53%	24%
Nursery			
Confiance	91%	93%	67%
Loevinger	77%	45%	37%
Zhang	66%	79%	49%
Zoo			
Confiance	83%	81%	71%
Loevinger	69%	47%	34%
Zhang	48%	73%	29%
MOYENNE			
Confiance	89%	75%	62%
Loevinger	70%	47%	32%
Zhang	45%	65%	28%

TABLE 4.7 – Valeurs de degré de ressemblance entre les mesures *Confiance*, *Loevinger*, *Zhang* et les classes de mesures {Conf, Loev}, {Conf, Zhang}, {Conf, Loev, Zhang}

en se basant sur la sélection des k meilleures règles selon plusieurs mesures. Pour ce faire, nous proposons, tout d'abord, d'extraire l'ensemble des k meilleures règles pour la mesure m et de sélectionner les k meilleures règles pour la classe C en appliquant l'algorithme RANKRULE. Ensuite, nous comparons les deux ensembles afin de calculer le taux de règles communes à ces deux ensembles permettant ainsi d'obtenir le degré d'appartenance de la mesure m à la classe C. Le degré d'appartenance de m à C est calculé comme suit :

$$DegApp_k(m, C) = \frac{|E_{mk} \cap E_{Ck}|}{k}$$

avec $|E_{mk} \cap E_{Ck}|$ désigne la cardinalité de l'intersection de E_{mk} et E_{Ck}, les ensembles des k meilleures règles pour m et C respectivement. Notons que lors de la comparaison, nous nous intéressons uniquement à la présence des règles dans les deux ensembles et non pas à l'ordre avec lequel m et C ordonnent ces règles.

Le tableau 4.7 montre respectivement, pour une valeur de k égale à 100, les degrés d'appartenance des mesures *Confiance*, *Loevinger* et *Zhang* aux classes {Conf, Loev}, {Conf, Zhang} et {Conf, Loev, Zhang}.

A partir des résultats présentés dans le tableau 4.7, nous pouvons constater que :

- Pour les classes {Conf, Loev}, {Conf, Zhang} et {Conf, Loev, Zhang}, c'est la mesure *confiance* qui possède toujours le plus grand degré d'appartenance. Cela signifie que cette mesure influe considérablement la sélection des k meilleures règles pour les trois classes. Elle pourrait être ainsi considérée comme le noyau de chacune de ces trois classes.
- A l'opposé, pour les différentes bases de test, les deux mesures *Loevinger* et *Zhang* possèdent les plus petits degrés d'appartenance aux classes {Conf, Loev}, {Conf, Zhang} et {Conf, Loev, Zhang}. Par exemple, les degrés d'appartenance de Loevinger et Zhang à la classe {Conf, Loev, Zhang} atteignent seulement 23% et 16% respectivement pour la base Monks 1. Ce résultat montre que les mesures Confiance, Loevinger et Zhang présentent des comportements différents et que réellement elles ne peuvent être regroupées en une même classe.

4.5 Conclusion

Dans ce chapitre, nous avons introduit une nouvelle approche permettant la sélection des règles d'association selon plusieurs mesures en tirant profit de la relation de dominance. Dans le cadre de cette approche, nous avons proposé un algorithme, appelé SKYRULE,

dont l'objectif est d'éliminer les règles dominées, *i.e,* celles qui ne sont pas pertinentes selon la combinaison de toutes les mesures, et de retenir seulement les règles non dominées, *i.e,* celles qui représentent le meilleur compromis entre les différentes évaluations de mesures choisies. Afin de mettre en évidence les gains qu'apporte notre approche, nous avons mené des expérimentations qui comparent le nombre des règles non dominées au nombre de règles qui peuvent être générées par une approche optimale d'extraction sous contraintes. Les résultats des expérimentations ont montré que notre approche apporte un gain important en terme de compacité par rapport à l'approche d'extraction sous contraintes.

Par ailleurs, nous avons proposé une approche permettant de sélectionner les k meilleures règles selon plusieurs mesures. Cette approche a permis ainsi de répondre à une requête personnalisée de l'utilisateur qui a souvent besoin d'un certain nombre de règles qui peut être supérieur ou inférieur à celui des règles non dominées.

Points clés

- Nous avons exposé le problème de l'hétérogénéité des mesures dont l'objectif est d'associer une valeur numérique à une règle permettant de quantifier son intérêt et avons présenté les approches qui permettent d'étudier leurs comportements.
- Nous avons présenté les problèmes liés à la sélection des règles d'association en utilisant différentes mesures.
- Nous avons proposé une première solution à ces problèmes qui consiste à sélectionner les *règles non dominées* qui expriment le meilleur compromis entre les différentes évaluations des mesures utilisées.
- Nous avons également proposé une deuxième solution qui consiste à sélectionner les k meilleures règles selon plusieurs mesures en se basant sur la relation de dominance.

Publications

- S. Bouker, R.Saidi, S.Ben Yahia, E.Mephu Nguifo. Ranking and Selecting Association Rules Based on Dominance Relationship. Proceedings of the 2012 IEEE 24th International Conference on Tools with Artificial Intelligence - Volume 01 ; 11/2012 (**Award : Best Student Paper**).
- S. Bouker, R.Saidi, S.Ben Yahia, E.Mephu Nguifo. Mining Undominated Association Rules Through Interestingness Measures. International Journal of Artificial Intelligence Tools 08/2014 ; 23(04-4).
- Présentation orale à la Conférence Francophone sur l'Apprentissage Automatique (CAP'2013)

Chapitre 5

Le modèle des règles d'association représentatives

Sommaire

5.1 Introduction 101
5.2 Famille des règles d'association représentatives 103
5.2.1 Motivation 103
5.2.2 Règles d'association représentatives 104
5.3 Sélection des règles d'association représentatives 106
5.3.1 Cas de comparabilité transitive 107
5.3.2 Cas de comparabilité non-transitive 109
5.3.3 Compacité des règles représentatives 119
5.4 Conclusion 121

Le présent chapitre propose le modèle dit *des règles représentatives* dans le but de sélectionner un ensemble réduit de règles d'association selon plusieurs mesures de qualité tout en tenant compte des similarités structurelles et sémantiques qui peuvent exister entre les règles.

5.1 Introduction

Dans le chapitre précédent, nous avons présenté une approche permettant de sélectionner les règles d'association selon plusieurs mesures de qualité en se basant sur la relation de dominance. Toutefois, en raison de sa nature statistique, cette relation n'accorde aucune importance aux similarités structurelles et sémantiques qui peuvent exister entre

les règles. En effet, pour déterminer si une règle domine une autre, seules les valeurs des mesures associées à ces règles sont comparées. Cependant, une telle comparaison pourrait présenter un problème de perte d'informations lorsqu'une une règle non dominée r élimine une autre règle r' légèrement moins intéressante alors qu'elles véhiculent des informations totalement différentes. Dans ce cas, l'information contenue dans la règle r' ne sera pas présentée à l'expert. Par exemple, la règle $r_1 : a \rightarrow d$ extraite à partir de la table relationnelle Ω (*c.f.,* Table 5.1) est dominée par la règle $r_2 : b \rightarrow c$ alors que ces deux règles ne véhiculent pas la même information puisqu'elles ne sont pas structurellement comparables.

	a	b	c	d
t_1			×	×
t_2	×			
t_3	×			×
t_4			×	
t_5		×		×
t_6	×			×
t_7			×	
t_8				×
t_9		×	×	×
t_{10}		×	×	

(a) *Contexte d'extraction* $\mathcal{T}$

Regles	*Freq*	*Conf*	*Pearl*
$r_1 : a \rightarrow d$	0.20	0.66	0.02
$r_2 : b \rightarrow c$	0.20	0.66	0.05
$r_3 : b \rightarrow d$	0.20	0.66	0.02
$r_4 : c \rightarrow b$	0.20	0.40	0.05
$r_5 : c \rightarrow d$	0.20	0.40	0.10
$r_6 : d \rightarrow a$	0.20	0.33	0.02
$r_7 : d \rightarrow b$	0.20	0.33	0.01
$r_8 : d \rightarrow c$	0.20	0.33	0.10
$r_9 : b \rightarrow cd$	0.10	0.33	0.03
$r_{10} : c \rightarrow bd$	0.10	0.20	0.00
$r_{11} : d \rightarrow bc$	0.10	0.16	0.02
$r_{12} : bc \rightarrow d$	0.10	0.50	0.02
$r_{13} : bd \rightarrow c$	0.10	0.50	0.00
$r_{14} : cd \rightarrow b$	0.10	0.50	0.04

(b) *Table relationnelle* $\Omega = (\mathcal{AR}, \mathcal{M})$

TABLE 5.1 – Exemples d'un contexte d'extraction $\mathcal{T}$ et d'une table relationnelle $\Omega = (\mathcal{AR}, \mathcal{M})$

Dans ce chapitre, nous introduisons le *modèle de règles d'association représentatives*, qui propose une méthode de sélection des règles d'association selon plusieurs mesures tout en remédiant au problème de perte d'information. L'objectif de ce modèle est en fait de fournir un ensemble réduit de règles qui, à la fois, véhiculent le maximum d'informations utiles et expriment le meilleur compromis entre les différentes évaluations des mesures utilisées. Intuitivement, une règle r est *représentative* d'une autre règle r' si elle la domine et lui est structurellement comparable.

Ce chapitre est organisé en deux sections. La première section, introduit le *modèle de règles d'association représentatives* et présente une définition formelle de ce modèle. La deuxième section introduit un algorithme pour sélectionner les règles représentatives lorsque la similarité structurelle entre les règles est transitive, puis deux algorithmes dans le cas contraire (*i.e.,* lorsque la similarité structurelle entre les règles est non-transitive).

5.2 Famille des règles d'association représentatives

5.2.1 Motivation

La limite de l'approche de sélection des règles non dominées résulte du caractère uniquement statistique de la relation de dominance qui ne tient pas compte du caractère sémantique des règles. Cette limite nous a motivé à introduire le *modèle des règles d'association représentatives* développé ci-après dont l'objectif est de produire un nombre réduit de règles vérifiant les conditions suivantes :

- **"Consensus"** : les règles représentatives doivent exprimer le meilleur compromis entre les différentes évaluations des mesures choisies.
- **"Informativité"** : le *modèle des règles représentatives* ne doit pas perdre trop d'information par rapport à la donnée de l'ensemble des règles d'association. A cet effet, nous considérons qu'une règle r ne peut éliminer une règle r' quand elles ne sont pas comparables, même si r domine r'.
- **"Non-redondance"** : la sélection d'un nombre raisonnable de règles d'association très pertinentes ne suffit pas toujours à en donner une solution sans défaut à l'expert. Il se peut en effet que les informations véhiculées par les différentes règles soient certes pertinentes mais se recoupent, ou soient identiques. Dans ce cas, les informations sont qualifiées de redondantes. Il est alors nécessaire d'éliminer à partir d'un ensemble de règles structurellement comparables celles qui sont les moins *intéressantes*. Par exemple, considérons les règles suivantes : $r_2 : b \rightarrow c$, $r_9 : b \rightarrow cd$ et $r_{13} : bd \rightarrow c$ extraites à partir du contexte $\mathcal{T}$ (*c.f.,* Table 5.1 (a)). Les deux dernières règles peuvent être éliminées puisqu'elles sont structurellement comparables à r_2 et pour toutes les mesures, les valeurs qui leur correspondent sont inférieures à celles de r_2 (*i.e.,* r_2 domine r_{10} et r_{13}). En fait, nous pouvons déduire que l'ajout de l'item d soit à l'antécédent soit à la conclusion de r_2 a un "effet négatif" sur la qualité de la règle (les valeurs des mesures associées à la règle ont diminué). Par conséquent, nous

appelons une règle r redondante s'il existe une règle r' qui lui est comparable et tel que $r' \succ r$. Le *modèle des règles représentatives* ne doit pas ainsi contenir des règles redondantes.

Ainsi, la sélection des règles d'association représentatives est nécessairement un problème d'optimisation qui détermine les règles exprimant le meilleur compromis entre leurs évaluations par l'ensemble des mesures choisies et l'information contenue dans leur description qui doit être simultanément informative et peu redondante. Elle retient donc l'idée de l'utilisation de la relation de dominance adoptée par la sélection des règles non dominées pour exprimer ce compromis. La principale différence est de sélectionner en plus, les règles ayant une plus-value dans l'information présentée à l'expert.

5.2.2 Règles d'association représentatives

Dans ce qui suit, nous introduisons l'ensemble des règles représentatives qui satisfait au plus près les critères évoqués précédemment. La définition de cet ensemble est basée sur les notions de dominance et de similarité structurelle (ou comparabilité) entre les règles :

5.2.2.1 Comparabilité entre règles d'association

Afin d'exprimer une relation de dépendance sémantique entre deux règles (*i.e.,* qui véhiculent une même information), plusieurs définitions ont été proposées [CVMC08], [RF07]. Ces définitions reposent essentiellement sur une comparaison au niveau des structures des règles. Par exemple, des règles liées sémantiquement peuvent avoir des items communs, ou des antécédents communs, ou des conclusions communes, etc. Formellement, la dépendance sémantique ou la comparabilité entre deux règles est définie comme suit :

Définition 36 ***(comparabilité) :***
Une règle r est comparable à une règle r', noté $\boldsymbol{comp(r,\ r') = vrai}$*, si et seulement si r est structurellement similaire à r'.*

Évidemment, toute relation de comparabilité est *reflexive*, *i.e.,* $\forall\ r \in \mathcal{AR}$, $comp(r, r) = vrai$. Toutefois, une relation de comparabilité peut être soit :

- ***Transitive*** : $\forall\ r,\ r',\ r'' \in \mathcal{AR}$ tels que $comp(r,\ r') = vrai$ et $comp(r', r'') = vrai$, alors $comp(r,\ r'') = vrai$
- ***Non-Transitive*** : $\exists\ r,\ r',\ r'' \in \mathcal{AR}$ tels que $comp(r,\ r') = vrai$ et $comp(r',\ r'') = vrai$, mais $comp(r,\ r'') = faux$

Cette distinction est très importante dans la mesure où la procédure de sélection des règles représentatives basée sur une comparabilité transitive diffère de celle d'une sélection des règles représentatives basée sur une comparabilité non-transitive. Cette différence fera l'objet d'une discussion dans la section 5.3.

5.2.2.2 Règles d'association représentatives

Dans ce qui suit, nous montrons que l'utilisation conjointe de la relation de dominance et la relation de comparabilité entre les règles réalise un compromis acceptable entre "consensus" et "informativité" des règles.

Définition 37 *(règle représentative)*
Soit une table relationnelle $\Omega = (\mathcal{AR}, \mathcal{M})$. *Une règle* $r \in \mathcal{AR}$ *est dite représentative d'une règle* $r' \in \mathcal{AR}$, *noté* ***repres***(r, r')= *vrai si et seulement si :*

- r *et* r' *sont structurellement comparable (i.e., comp(*r, r'*)= vrai),*
- r *domine* r' *(i.e.,* $r \succ r'$*).*

A partir de la définition précédente, nous pouvons déduire que :

1. aucune règle ne peut être représentative d'une règle non dominée.
2. une règle ne peut être représentative d'une autre règle qui ne lui est pas comparable, même si elle la domine. Il serait, alors, intéressant de sélectionner toutes les règles - qu'elles soient dominées ou non dominées - qui n'ont aucune règle représentative, augmentant ainsi les connaissances présentées à l'expert. Bien évidemment, ces règles sont représentatives de toutes les règles qu'elles dominent et qui leur sont comparables.

Nous définissons l'ensemble des règles d'association représentatives comme suit :

Définition 38 *(Ensemble de règles d'association représentatives)*
Soit la table relationnelle $\Omega = (\mathcal{AR}, \mathcal{M})$. *L'ensemble* $\mathcal{RR} \subseteq \mathcal{AR}$ *représente les règles représentatives selon* $\mathcal{M}$ *si et seulement si :*

1. *toute règle* $r \in \mathcal{AR} \backslash \mathcal{RR}$ *a au moins une règle représentative* $r' \in \mathcal{RR}$, *i.e., repres(*r', r*) = vrai,*
2. *toute règle* $r \in \mathcal{RR}$ *ne peut être représentative d'une autre règle* $r' \in \mathcal{RR}$, *i.e.,* $\forall$ r, $r' \in \mathcal{RR}$, *repres(*r, r'*) = faux et repres(*r', r*) = faux.*

Remarque 2 *La première condition de la définition 38 garantit une "informativité" maximale des règles présentées à l'expert (i.e., elles véhiculent le maximum de connaissances utiles). En effet, toutes les règles de l'ensemble $\mathcal{RR}$ sont représentatives des règles qui n'appartiennent pas à $\mathcal{RR}$, la perte d'information étant alors nulle.*

Remarque 3 *La deuxième condition de la définition 38 garantit une "concision" optimale de l'ensemble des règles représentatives. En effet, pour toute paire de règles (r, r') $\in \mathcal{RR}$, r ne peut être représentative de r' et vice versa. Cette condition permet de réduire au maximum le nombre des règles présentées à l'expert et d'éliminer une grande partie de la redondance entre règles.*

Proposition 1 *Soit la relation $\Omega = (\mathcal{AR}, \mathcal{M})$ et $\mathcal{RR} \subseteq \mathcal{AR}$ l'ensemble des règles représentatives selon $\mathcal{M}$. Alors :*

$$Sky\ (\Omega) \subseteq \mathcal{RR}$$

Preuve 7 *Par l'absurde s'il existe une règle non dominée r qui n'appartient pas à $\mathcal{RR}$, alors il existe une règle $r' \in \mathcal{RR}$ qui lui est représentative. On obtient la contradiction r' domine r.*

5.3 Sélection des règles d'association représentatives

Supposons qu'avant de commencer la sélection des règles représentatives à partir d'un ensemble de règles $\mathcal{AR}$ évaluées par un ensemble de mesures $\mathcal{M}$, nous voulons regrouper les règles structurellement comparables en différents groupes. Pouvons-nous affirmer que chaque règle appartiendra à un seul groupe ? Une telle affirmation serait fort attrayante puisqu'elle pourrait considérablement simplifier la sélection des règles représentatives en vérifiant, seulement, si une règle est représentative de son groupe. Malheureusement elle n'est pas vérifiée lorsque nous utilisons une relation de comparabilité non transitive pour le regroupement. En effet, une règle peut être, dans ce cas, comparable à deux règles qui ne sont pas comparables et par conséquent, elle appartiendra à deux groupes différents.

Ainsi, dans cette section, nous proposons trois algorithmes pour sélectionner les règles représentatives à partir d'une table relationnelle $\Omega = (\mathcal{AR}, \mathcal{M})$. Le premier permet de sélectionner les règles représentatives dans le cas d'une relation de comparabilité transitive, alors que le deux autres algorithmes traitent le cas d'une relation de comparabilité non transitive.

5.3.1 Cas de comparabilité transitive

Avant d'entamer la présentation de notre algorithme permettant de sélectionner les règles représentatives lorsque la relation de comparabilité est transitive, nous commençons par définir l'ensemble des règles non représentées.

5.3.1.1 Règles non représentées

Lorsqu'une règle r est représentative d'une règle r', cela signifie d'une part, que r est meilleure ou équivalente à la règle r' pour toutes les mesures de $\mathcal{M}$ et d'autre part, que r est structurellement comparable à r'. Ainsi, l'ensemble des règles non représentées, noté $\mathcal{RNR}$, est défini comme suit :

Définition 39 *(Règles non représentées)*
Soit $\Omega = (\mathcal{AR}, \mathcal{M})$ *une table relationnelle. L'ensemble des règles non représentées selon* $\mathcal{M}$ *est défini par :*

$$\mathcal{RNR}(\Omega) = \{r \in \mathcal{AR} \mid \nexists\, r' \in \mathcal{AR},\ comp(r, r') = vrai \text{ et } r' \succ r\}$$

Proposition 2 *L'ensemble des règles non représentées* $\mathcal{RNR}$ *selon* $\mathcal{M}$ *est égal à l'ensemble des règles représentatives* $\mathcal{RR}$ *lorsque la relation de comparabilité est transitive.*

Preuve 8 *Étant donnée une relation de comparabilité transitive et* r *une règle quelconque de* $\mathcal{AR}$*. Cette règle est soit :*

- *non représentée, i.e.,* $r \in \mathcal{RNR}$*. Alors, elle ne peut être représentative d'une autre règle* $r' \in \mathcal{RNR}$*.*
- *représenté par au moins une règle, alors* $r \in \mathcal{AR} \backslash \mathcal{RNR}$*. Puisque la dominance et la comparabilité sont toutes les deux transitives, alors la règle* r *est nécessairement représentée par au moins une règle dans* $\mathcal{RNR}$*.*

Suivant la définition 38, nous pouvons déduire que l'ensemble $\mathcal{RNR}$ *est égal à l'ensemble des règles représentatives.*

Ainsi, toutes les règles représentatives sont des règles non représentées. Par conséquent, sélectionner les règles représentatives selon une relation de comparabilité transitive revient à sélectionner les règles non représentées.

Remarque 4 *Il est à noter qu'une règle non représentée ne peut être représentative que d'une règle qui lui est comparable.*

5.3.1.2 Algorithme

Dans ce qui suit, nous proposons un algorithme, appelé TransRule, qui permet de sélectionner les règles non représentées à partir d'une table relationnelle selon une relation de comparabilité transitive.

L'algorithme TransRule commence par subdiviser l'ensemble des règles d'association sur des sous-ensembles de règles comparables, *i.e.,* chaque règle va appartenir au sous-ensemble des règles qui lui est comparable. Puisque la relation de comparabilité est transitive, toute règle ne peut appartenir qu'à un seul sous-ensemble. Ainsi, une règle ne peut représenter une autre règle que si elle appartient à son "cluster"(*c.f,* Remarque 4). L'algorithme TransRule détermine ensuite, les règles non représentées en appliquant l'algorithme SkyRule sur chaque sous-ensemble.

Données : $\Omega = (\mathcal{AR}, \mathcal{M})$: table relationnelle, $\mathcal{C}_t$: relation de comparabilité transitive
Résultats : $\mathcal{RNR}$: ensemble des règles non représentées de Ω.

```
1 début
2     {AR} ← {R ← {r ∈ AR | ∀(r', r'') ∈ R, C_t(r', r'') = vrai) }}
3     RNR ← ∅
4     pour chaque R ⊂ {AR} faire
5         S_R ← SkyRule((R, M))
6         RNR ← RNR ∪ S_R
7     retourner RNR
8 fin
```

Algorithme 7 : TransRule

Théorème 5.3.1 *L'algorithme* TransRule *est correct et complet.*

Preuve 9 *L'algorithme est complet puisque si une règle r parmi les non représentées venait à manquer dans le résultat final, c'est que soit elle n'a pas été traitée par l'algorithme* SkyRule*, soit qu'elle n'a pas été retenue suite à l'application de* SkyRule *sur le sous-ensemble de règles à qui elle appartient. Comme l'algorithme garantit d'appliquer* SkyRule *à tous les sous-ensembles qui contiennent toutes les règles, c'est donc que r n'a pas été retenue. Cela ne peut se produire que si une autre règle de son sous-ensemble la domine. Ceci entraîne une contradiction avec le fait que r est non représentée puisqu'elle ne peut être dominée par une règle qui lui est comparable.*

L'algorithme est correct parce qu'une règle ne peut être retenue dans le résultat final que si elle est non représentée. En effet, pour tout sous-ensemble, l'algorithme SkyRule *élimine toutes les règles dominées et garde seulement les règles non représentées. Puisqu'une règle non représentée dans son sous-ensemble ne peut être comparable à une règle appartenant à un sous-ensemble différent, alors elle est une règle non représentée dans l'ensemble de toutes les règles.*

5.3.2 Cas de comparabilité non-transitive

Contrairement à la sélection des règles représentatives selon une comparabilité transitive, la solution qui consiste à prendre les règles non représentées ne fonctionne pas pour la sélection des règles représentatives selon une comparabilité non-transitive. En effet, les non représentées ne sont pas nécessairement suffisantes pour garantir que toute règle ait une règle représentative. En effet, si une règle r non représentée domine une règle r' qui lui est comparable, et que r' à son tour domine une règle r'' qui lui est comparable, alors r'' risque de ne pas avoir une règle représentative dans $\mathcal{RNR}$. En effet, l'ensemble $\mathcal{RNR}$ va contenir seulement r qui ne peut être représentative de r'' lorsque les deux règles ne sont pas comparables.

Rule	*Freq*	*Conf*	*Pearl*
$r_3 : b \rightarrow d$	0.20	0.66	0.02
$r_{10} : c \rightarrow bd$	0.10	0.20	0.00
$r_{12} : bc \rightarrow d$	0.10	0.50	0.02

Table 5.2 – Table relationnelle $\Omega' \subset \Omega$

Exemple 32 *Prenons l'exemple d'une table relationnelle* Ω*'=* $(\{r_3, r_{10}, r_{12}\}, \mathcal{M})$ *extraite à partir de table relationnelle* Ω *(*c.f*, Table 5.1), et d'une relation de comparabilité* $\mathcal{C}$ *non-transitive suivante : Deux règles* $r : X \rightarrow Y$ *et* $r' : X' \rightarrow Y'$ *sont comparables, si et seulement si* $(X \subseteq X'$ *et* $Y \subseteq Y')$ *ou* $(X' \subseteq X$ *et* $Y' \subseteq Y)$*. L'intuition se cachant derrière cette relation de comparabilité est qu'en ajoutant ou en supprimant un (ou plusieurs) items de l'antécédent et/ou la conséquence d'une règle donnée, une nouvelle règle est obtenue qui pourrait être plus ou moins pertinente.*

Puisque r_3 *est représentative de* r_{12}*, et* r_{12} *est représentative de* r_{10} *alors, l'ensemble des règles non représentées de* Ω*' contient seulement* r_3*,* $\mathcal{RNR} = \{r_3\}$*. Toutefois,* r_3 *n'est pas représentative de* r_{10}*. Ainsi, il faut rajouter* r_{10} *à* $\mathcal{RNR}$ *pour obtenir l'ensemble des règles représentatives de* Ω*'.*

Ainsi, la méthode utilisée pour sélectionner les règles représentatives lorsque la comparabilité est transitive n'est pas applicable dans le cas d'une comparabilité non-transitive. Deux solutions sont envisageables pour sélectionner les règles représentatives lorsque la comparabilité est non transitive.

La première consiste à parcourir l'espace de recherche par niveau selon un ordre partiel sur les règles fondé sur la relation de dominance. A chaque niveau, un ensemble de règles dominées candidates est déterminé pour être filtré selon un rapprochement avec l'ensemble des règles représentatives appartenant à tous les niveaux inférieurs. En adoptant la technique "diviser pour régner", la deuxième méthode consiste à subdiviser l'ensemble des règles en des sous-ensembles en se basant sur les relations de dominance et de comparabilité, et à déterminer les règles représentatives à partir de ces sous-ensembles. Ces deux méthodes sont décrites dans les deux paragraphes suivants.

5.3.2.1 Une première solution fondée sur un parcours nivelé des règles

Dans ce paragraphe, nous proposons une nouvelle structure, le treillis de dominance, sur laquelle s'appuie notre première méthode pour sélectionner les règles représentatives.

Définition 40 *(Treillis de dominance et diagramme de dominance)*
Le treillis de dominance associé à $\Omega = (\mathcal{AR}, \mathcal{M})$*, noté* $\mathcal{L}_\Omega = (\mathcal{AR}, <)$ *est l'ensemble de toutes les règles* $\mathcal{AR}$*, suivant un ordre partiel :* $\forall\ r, r' \in \mathcal{AR}$*,* $r < r'$ *si et seulement si* $r' \succ r$*. Nous pouvons utiliser l'ordre partiel entre les règles pour générer le graphe du treillis, appelé diagramme de dominance, de la manière suivante : Il existe un arc orienté* $r' \rightarrow r$*, si* $r < r'$*,* $\nexists\ r'' \in \mathcal{AR}$*, tel que* $r < r'' < r'$*.*

La figure 5.1 présente le diagramme de dominance associé à $\Omega = (\mathcal{AR},\mathcal{M})$ illustrée par la table 5.1(b). Nous convenons que les règles non dominées sont représentées en haut de diagramme. Ainsi, r_2 et r_5 forment le premier niveau, r_1, r_3, r_4, r_8 et r_{14} forment le deuxième niveau, r_6, r_9 et r_{12} forment le troisième niveau, r_7 et r_{13} forment le quatrième niveau et r_{10} et r_{11} forment le cinquième niveau.

Une solution pour extraire les règles représentatives consiste à parcourir les arcs du diagramme dans l'ordre des règles $(\mathcal{AR}, <)$ selon un parcours en profondeur partant de l'ensemble des règles non dominées. Pour déterminer les règles représentatives dans un

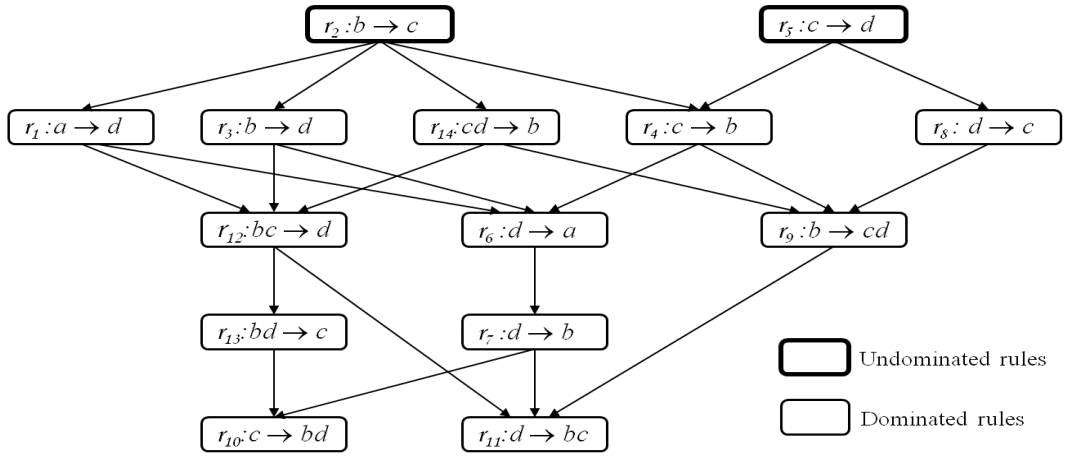

FIGURE 5.1 – Diagramme de dominance associé à la table relationnelle Ω illustrée par la table 5.1(b).

niveau n ($n \neq 1$), il suffit de comparer (statistiquement [1] et structurellement [2]) les règles de ce niveau avec seulement les règles représentatives qui appartiennent à des niveaux inférieurs à n. En effet, une règle r appartenant à un niveau n ne peut être représentée par une règle r' si cette dernière appartient à un niveau supérieur à n puisque r' ne peut pas dominer r. Ainsi, il est inutile de comparer r avec des règles appartenant à des niveaux supérieurs à n pour vérifier si elle est représentative. Cependant, il nécessaire de la comparer aux règles représentatives appartenant à des niveaux inférieurs à n. En effet, s'il existe au moins une règle représentative qui lui est comparable et qui la domine, alors r n'est pas représentative. Dans, le cas contraire, *i.e.,* s'il n'existe aucune règle représentative de r dans les niveaux inférieurs à n, alors r est une règle représentative.

Algorithme

Le processus d'extraction des règles représentatives basé sur le parcours par niveau du diagramme dans l'ordre des règles $(\mathcal{AR}, <)$ est donné par l'algorithme RPRRULE_1 (*c.f.,* algorithme 8). Cet algorithme prend en entrée une table relationnelle $\Omega = (\mathcal{AR}, \mathcal{M})$ et une relation de comparabilité non-transitive $\mathcal{C}_{nt}$. Il donne en sortie l'ensemble des règles représentative $\mathcal{RR}$.

1. Comparer statistiquement deux règles reviendrait à comparer les différentes valeurs de mesures qui leurs sont associées.
2. Comparer structurellement deux règles reviendrait souvent à réaliser des tests d'inclusion entre leurs prémisses et leurs conclusions.

L'algorithme opère de la manière suivante : puisque les règles non dominées font partie de l'ensemble des règles représentatives (d'après proposition 1), l'algorithme commence, tout d'abord, par initialiser $\mathcal{RR}$ à l'ensemble des règles non dominées. Ainsi, on obtient les règles représentatives qui appartiennent au premier niveau du diagramme de l'ordre des règles ($\mathcal{AR}$, $<$). Ensuite, pour déterminer les règles candidates du niveau suivant, l'algorithme applique *SkyRule* sur l'ensemble des règles dominées. En effet, les règles qui appartiennent au deuxième niveau sont, en fait, les non dominées de l'ensemble des règles dominées par les règles du premier niveau. Ceci peut être généralisé comme suit : les règles candidates d'un niveau courant sont les règles non dominées parmi les règles dominées par le niveau précédent. Une fois l'ensemble des règles candidates appartenant à un niveau courant déterminé, l'algorithme vérifie, pour chaque règle candidate r, s'il existe au moins une règle qui lui est représentative dans l'ensemble $\mathcal{RR}$. Si c'est le cas, alors r est éliminée. A l'issue de cet élagage, on obtient l'ensemble des règles représentatives du niveau courant. Le processus est réitéré pour tous les niveaux possibles.

Données : $\Omega = (\mathcal{AR}, \mathcal{M})$: table relationnelle, $\mathcal{C}_{nt}$: relation de comparabilité non-transitive
Résultats : $\mathcal{RR}$: ensemble des règles représentatives de Ω.

```
1  début
2      RR ← SkyRule(Ω)
3      R ← AR \ RR
4      tant que R ≠ ∅ faire
5          C ← SkyRule(R)
6          R ← R \ C
7          pour chaque r ∈ C faire
8              pour chaque r' ∈ RR | comp(r, r')= vrai faire
9                  si r' ≻ r alors
10                     éliminer r de C
11         RR ← RR ∪ C
12 fin
```

Algorithme 8 : RprRule_1

Exemple 33 *Prenons notre exemple illustré par la table 5.1, nous avons :*

- *Le premier niveau contient les règles non dominées* r_2 *et* r_5*, alors* $\mathcal{RR} = \{r_2, r_5\}$,
- *Le deuxième niveau contient les règles* r_1, r_3, r_{14}, r_4, r_8 *:*
  Puisque r_2 *et* r_5 *ne sont comparables à aucune règle appartenant au niveau 2, alors* $\mathcal{RR} = \{r_2, r_5, r_1, r_3, r_{14}, r_4, r_8\}$,
- *Le troisième niveau contient les règles* r_6, r_9, r_{12} *:*
  Alors que r_9 *et* r_{12} *sont comparables et dominées respectivement par* r_2 *et* r_3*, la règle* r_6 *n'est comparable à aucune règle appartenant à* $\mathcal{RR}$*. Ainsi,* r_6 *est une règle représentative. Par conséquent,* $\mathcal{RR} = \{r_2, r_5, r_1, r_3, r_{14}, r_4, r_8, r_6\}$.
- *Le quatrième niveau contient les règles* r_7 *et* r_{13} *:*
  A l'exception de r_{14}*, toutes les règles dans* $\mathcal{RR}$ *ne sont pas comparables à* r_7*, alors la règle* r_7 *est représentative puisqu'elle n'est pas dominée par* r_{14}*. Cependant, la règle* r_2 *est comparable à* r_{13} *et elle la domine. Ainsi, l'ensemble des règles représentatives devient égal à* $\mathcal{RR} = \{r_2, r_5, r_1, r_3, r_{14}, r_4, r_8, r_6, r_7\}$,
- *Le cinquième et dernier niveau contient les règles* r_{10} *et* r_{11} *:*
  Ces règles sont dominées respectivement par les règles r_4 *et* r_7 *qui leurs sont comparables, alors* r_{10} *et* r_{11} *ne peuvent être représentatives. Finalement, l'ensemble des règles représentatives reste égal à* $\mathcal{RR} = \{r_2, r_5, r_1, r_3, r_{14}, r_4, r_8, r_6, r_7\}$.

Ce processus est illustré par la figure 5.3.2.1. Les règles représentatives non dominées, les règles représentative dominées et les règles non-représentatives sont représentées respectivement par, un rectangle en gras, rectangle et rectangle pointillé.

Théorème 5.3.2 *L'algorithme* RprRule_*1 est correct et complet.*

Preuve 10 *L'algorithme est complet. En effet, l'algorithme passe en revue toutes les règles de tous les niveaux du diagramme de l'ordre des règles* $(\mathcal{AR}, <)$*. Les règles du premier niveau sont retenues dans le résultat final puisqu'elles sont non dominées. Par conséquent, si une règle représentative* r *appartenant à niveau* n *supérieur à 1 venait à manquer dans le résultat* $\mathcal{RR}$*, c'est qu'elle est dominée par une autre règle représentative qui lui est comparable et qui appartient à un niveau inférieur à* n*. Ceci entraîne une contradiction avec le fait que* r *soit représentative.*

L'algorithme est correct puisqu'une règle ne peut être retenue dans le résultat final que si elle est représentative. En effet :

- *pour le premier niveau, l'algorithme retient toutes les règles qui sont nécessairement représentatives puisqu'elles sont non dominées.*

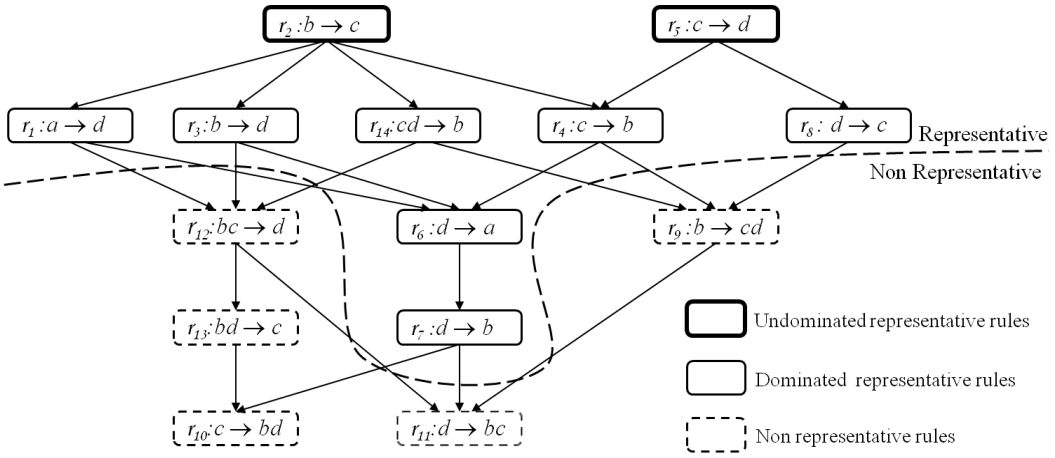

FIGURE 5.2 – Règles représentatives dans le diagramme de dominance associé à Ω illustré par la table 5.1(b).

– *pour n'importe quel autre niveau n ($n \neq 1$), l'algorithme élimine toute règle qui possède au moins une règle qui lui est représentative et qui appartient à un niveau inférieur à n. Puisqu'une règle appartenant au niveau n ne peut être représentée par une autre règle appartenant à un niveau supérieur à n, alors les règles retenues sont nécessairement des règles représentatives.*

Par conséquent, toutes les règles retenues sont des règles représentatives.

5.3.2.2 Une deuxième solution adoptant la technique "diviser pour régner"

Une deuxième méthode pour sélectionner les règles représentatives consiste à subdiviser l'espace de recherche, *i.e.,* l'ensemble des règles en des sous-ensembles sur la base des notions de dominance et de comparabilité. L'idée est donc d'obtenir des sous-ensembles quasiment indépendants dans le sens où une règle r qui n'appartient pas à un sous-ensemble s, ne peut jamais être représentative d'une règle appartenant à s. Une telle subdivision permettra de minimiser le nombre de comparaisons (de nature statistiques et structurelles) à réaliser entre les règles, réduisant d'une manière drastique le coût de sélection des règles représentatives en terme de temps d'exécution. Cette subdivision repose sur la notion d'*incomparabilité* entre les règles définie comme suit :

Définition 41 *Soient r et r' deux règles. La règle r est incomparable avec r' si et seulement si $r \succ r'$ et comp(r,r')= faux*
L'ensemble de toutes les règles incomparables avec r est noté par Incomp(r) ;

$$Incomp(r) = \{\ r' \in \mathcal{AR} \mid r \succ r' \wedge comp(r,r') = faux\}$$

Lemme 4 *Soient r, r', $r'' \in \mathcal{AR}$ avec $r' \in Incomp(r)$.*
Si $r \not\succ r''$ alors $r' \not\succ r''$

Preuve 11 *$r' \in Incomp(r)$ implique que $r \succ r'$. Si $r \not\succ r''$ alors ils existent deux cas à distinguer :*

1. *Si $r \equiv r''$, alors $r \equiv r'' \succ r'$.*
2. *Si $r \not\equiv r''$, alors $\exists\ m \in \mathcal{M}$ tel que $r''[m] \succ r[m] \succeq r'[m]$*

Ainsi, dans les deux cas r' ne peut pas dominer r''.

Le lemme 4 montre que toute règle r' appartenant à l'ensemble $Incomp(r)$ ne peut dominer un règle n'appartenant pas à $Incomp(r)$. Par conséquence, si on veut éliminer les règles dont r' leur est représentative, il serait inutile de la comparer avec des règles qui ne sont pas dominées par r. Le lemme suivant permet de caractériser l'ensemble des règles candidates à être éliminées par r'.

Lemme 5 *Soient r, r', $r'' \in \mathcal{AR}$ avec $r' \in Incomp(r)$.*
Si $r' \succ r''$ et comp(r, r'') = $faux$ alors $r'' \in Incomp(r)$

Preuve 12 *$r' \in Incomp(r)$ implique que $r \succ r'$. Si $r' \succ r''$ alors en utilisant la transitivité de la dominance, on a $r \succ r''$. De plus, si comp(r, r'') = $false$, alors suivant la définition 41, on a $r'' \in Incomp(r)$.*

Lemme 6 *Soient un ensemble $\mathcal{R} \subseteq \mathcal{AR}$ et une règle $r \in \mathcal{R}$ tel que parmi toutes les règles de $\mathcal{R}$, la règle r possède le degré de similarité minimal avec $r^{\perp}$. Si $\nexists\ r' \in \mathcal{AR} \setminus \mathcal{R}$ tel que repres(r', r) = vrai, alors la règle r est une règle représentative.*

Preuve 13 *D'après le lemme 1 du chapitre précédent (c.f., section 4.3.2), la règle ne peut être dominée par une règle appartenant à $\mathcal{R}$. Ainsi, aucune règle ne peut être représentative de r. Par conséquent r est une règle représentative.*

Algorithme

Le processus d'extraction des règles représentatives utilisant la technique "diviser pour régner" est donné par l'algorithme 9. Cet algorithme est similaire à l'algorithme SkyRule du chapitre 4 (*c.f., section 4.3.3*). Les sous-espaces non-dominés sont déterminés de la même manière que dans SkyRule. L'algorithme utilise les variables suivantes lors de son exécution :

- La variable $\mathcal{RR}$: est une variable initialisée à l'ensemble vide, elle est utilisée pour garder la trace des règles représentatives.
- La variable $Incomp$: est une variable qui contient l'ensemble des règles candidates pour être représentatives. Cet ensemble contient seulement les règles incomparables avec les règles de l'ensemble $\mathcal{RR}$.
- La variable S : est une variable qui contient l'ensemble des règles couvrant l'espace non dominé de toutes les règles non dominées, elle est initialisée à $\mathcal{AR}$.

De manière informelle, l'algorithme RprRule_2 fonctionne comme suit :

- Si l'ensemble des candidates $S \cup Incomp$ est vide, alors l'algorithme se termine et retourne toutes les règles représentatives à travers la variable $\mathcal{RR}$.
- Sinon, chaque règle r dans $\{S \cup Incomp\}$ pourrait être parmi les représentatives. Si r a le degré minimal de similarité avec la règle de référence $r^{\perp}$, alors r est une règle représentative et elle est ajoutée à $\mathcal{RR}$. Deux cas sont à distinguer :

  1. Si r appartient à l'ensemble des incomparables, alors r n'est plus une candidate et elle est retirée de $Incomp$. Ainsi, seul l'ensemble des incomparables est exploré dans le but d'éliminer les règles qui sont représentées par r.

  2. Sinon (*i.e.,* r appartient à S), l'ensemble des incomparables et l'espace des non dominés contenant r sont tous les deux explorés. Toute règle de l'ensemble des incomparables représentée par r est éliminée. Alors que pour toute règle r' appartenant à l'espace des non-dominées contenant r, nous distinguons trois cas :

     (a) si r et r' sont comparables et r domine r', alors r' n'est plus une règle candidate et elle est retirée de S.

     (b) si r et r' sont incomparables, alors r' est encore une règle candidate et elle est ajoutée à l'ensemble $Incomp$.

     (c) sinon, r' n'est pas dominée par r, alors r' est encore une règle candidate et elle est ajoutée à l'espace non dominé de r.

Données : $\Omega = (\mathcal{AR}, \mathcal{M})$
Résultats : $\mathcal{RR}$: Règles représentatives

```
1  début
2    RR ← ∅, Incomp ← ∅, S ← AR
3    tant que S ≠ ∅ ou Incomp ≠ ∅ faire
4      r* une règle appartenant à S ∪ Incomp ayant min(DegSim(r,r^⊥))
5      ajouter r* à RR
6      pour chaque r ∈ Incomp faire
7        si r* ≻ r et comp(r*,r) = vrai alors
8          retirer r de Incomp
9      si r* ∈ Incomp alors
10       retirer r* de Incomp
11     sinon
12       pour chaque sous-espace s ∈ S tel que r* ∈ s faire
13         pour chaque r ∈ s faire
14           si r* ≻ r alors
15             si comp(r*, r) = faux alors
16               ajouter r à Incomp
17             retirer r de S
18           sinon
19             pour tous les i tel que r[m_i] > r*[m_i] faire
20               ajouter r au sous-espace s_i^{r*}
21           retirer s de S
22       ajouter ∪_i s_i^{r*} à S
23   retourner RR
24 fin
```

Algorithme 9 : RprRule_2

L'espace non dominé contenant r est retiré de S et l'espace non dominé de r est ajouté à S. Ce processus prend fin lorsque toutes les règles candidates sont traitées.

Théorème 5.3.3 *L'algorithme* RprRule_2 *est correct et complet.*

Preuve 14 *Pour prouver la correction de l'algorithme* RprRule_2*, nous devons montrer qu'il génère uniquement des règles représentatives. Par l'absurde, supposons qu'à la fin de l'execution de l'algorithme, l'ensemble $\mathcal{RR}$ comprend une règle non représentative r. Cela signifie que la règle r a été sélectionnée comme étant la règle ayant le degré de similarité minimal avec la règle référence $r^{\perp}$, parmi toutes les règles candidates dans $\{S \cup Incomp\}$ et qu'il n'existe aucune règle dans $\mathcal{AR} \backslash \{S \cup Incomp\}$ qui la représente. Ainsi, il existe une règle $r' \in \{S \cup Incomp\}$ représentative de r. On obtient la contradiction r ne possède pas le degré de similarité minimal avec la règle référence $r^{\perp}$, parmi toutes les règles candidates dans $\{S \cup Incomp\}$.*

L'algorithme RprRule_2 *est complet puisqu'qu'il génère toutes les règles représentatives. En effet, pour toute itération, chaque règle candidate r dans $\{S \cup Incomp\}$ est soit :*

1. *insérée dans l'ensemble des règles représentatives RR, si son degré de similarité avec $r^{\perp}$ est minimal,*
2. *éliminée, si elle est à la fois comparable et dominée par la règle ayant le degré de similarité minimal,*
3. *retenue dans l'ensemble candidat Incomp, si elle n'est pas comparable et elle est dominée à la règle ayant le degré de similarité minimal.*
4. *retenue dans l'ensemble candidat S, si elle n'est pas dominée par la règle ayant le degré de similarité minimal.*

L'algorithme RprRule_2 *se termine lorsque les deux ensembles Incomp et S sont vides. Ainsi,* RprRule_2 *génère toutes les règles non dominées.*

5.3.2.3 Analyse comparative des performances

Si les algorithmes RprRule_1 et RprRule_2 sont équivalents en termes de résultat, puisqu'ils sont tous deux corrects et complets, ces algorithmes se différencient par leurs performances en termes de temps d'exécution. Dans ce qui suit, nous présentons les tests qui ont été réalisés afin de comparer les performances respectives des deux algorithmes. Ces tests ont tous été réalisés sur les mêmes bases de données utilisées dans le chapitre précédent. Les combinaisons de mesures utilisées sont : {Confiance, Loevinger}, {Confiance, Loevinger, Zhang} et {Confiance, Loevinger, Pearl, Rappel, Zhang}. Nous considérons que deux règles r : X→Y et r' : X'→Y' sont comparables, si et seulement si ($X \subseteq X'$ et $Y \subseteq Y'$) ou ($X' \subseteq X$ et $Y' \subseteq Y$).

Le tableau 5.3 présente les temps d'exécution de chacun des deux algorithmes. Le tableau montre que quelque soit la base de données considérée et la combinaison de mesures utilisée, l'algorithme RprRule_1 apparaît toujours plus lent que l'algorithme RprRule_2. Ceci peut s'expliquer par le fait que RprRule_1 parcourt totalement le treillis de dominance en faisant appel, pour chaque niveau i, à SkyRule sans pour autant éliminer les règles représentées par les règles du niveau i. Cependant, une fois que l'algorithme RprRule_2 détermine une règle représentative r, il procède à l'élimination de toutes les règles représentées par r, réduisant ainsi l'espace de recherche.

Algorithme	{Conf ;Loev}	{Conf ;Loev ; Zhang}	{Conf ;Loev ;Pearl ; Rappel ;Zhang}
Diabetes			
RprRule_1	13,34s	15,66s	17,14s
RprRule_2	8,27s	7,21s	8,61s
Flare			
RprRule_1	15,88s	18,35s	20,92s
RprRule_2	11,43s	12,11s	13,61s
Monks 1			
RprRule_1	9,88s	14,83s	15,42s
RprRule_2	6,57s	8,56s	8,67s
Monks 2			
RprRule_1	9,13s	13,16s	14,07s
RprRule_2	6,33s	7,51s	7,88s
Monks 3			
RprRule_1	8,44s	12,09s	13,16s
RprRule_2	6,17s	7,26s	7,93s
Nursery			
RprRule_1	9,88s	12,07s	13,49s
RprRule_2	4,88s	6,33s	6,86s
Zoo			
RprRule_1	20,34s	22,93s	26,76s
RprRule_2	9,43s	10,19s	11,83s

Table 5.3 – Temps d'extraction des règles représentatives.

5.3.3 Compacité des règles représentatives

Dans ce qui suit, nous allons mener une série d'expérimentations afin de montrer la compacité des règles représentatives par rapport au nombre total de règles et au nombre de règles qui peuvent être extraites par l'approche optimale d'extraction sous contraintes ($ROSC$). En effet, d'après le tableau 5.4, nous pouvons constater que :

Bases (min_{freq} %)		{Conf;Loev}	{Conf;Pearl}	{Conf;Rappel}	{Conf;Zhang}	{Conf;Pearl Rappel}	{Conf;Loev Zhang}	{Conf;Loev ;Pearl Rappel ;Zhang}
Diabetes	$\mathcal{RR}$	5084	619	8512	4931	481	1315	1012
	$\mathcal{S}ky\text{-}\mathcal{AR}$	3411	9	6651	2996	9	171	171
	$\mathcal{ROSC}$	59314	58124	59206	59309	44813	44602	42126
(10,00)	$\mathcal{AR}$	62132	62132	62132	62132	62132	62132	62132
Flare	$\mathcal{RR}$	6883	502	6993	6817	443	269	291
	$\mathcal{S}ky\text{-}\mathcal{AR}$	4975	48	4978	4857	48	48	48
	$\mathcal{ROSC}$	56163	57101	56451	54524	53197	53116	52819
(10,00)	$\mathcal{AR}$	57476	57476	57476	57476	57476	57476	57476
Iris	$\mathcal{RR}$	302	265	271	262	261	264	253
	$\mathcal{S}ky\text{-}\mathcal{AR}$	246	246	246	246	246	246	246
	$\mathcal{ROSC}$	440	440	440	440	440	440	440
(0,00)	$\mathcal{AR}$	440	440	440	440	440	440	440
Monks1	$\mathcal{RR}$	3883	2106	2891	2797	1003	816	694
	$\mathcal{S}ky\text{-}\mathcal{AR}$	768	1	788	656	1	1	1
	$\mathcal{ROSC}$	60417	60692	59418	59452	58904	58811	58327
(1,00)	$\mathcal{AR}$	62184	62184	62184	62184	62184	62184	62184
Monks2	$\mathcal{RR}$	414	287	503	471	215	227	223
	$\mathcal{S}ky\text{-}\mathcal{AR}$	279	3	215	202	3	3	3
	$\mathcal{ROSC}$	59611	59702	59568	59544	59103	58917	58662
(1,00)	$\mathcal{AR}$	59976	59976	59976	59976	59976	59976	59976
Monks3	$\mathcal{RR}$	3107	773	2094	2362	1266	814	458
	$\mathcal{S}ky\text{-}\mathcal{AR}$	1028	2	713	781	4	2	2
	$\mathcal{ROSC}$	58662	58369	57922	58436	57816	57734	56038
(1,00)	$\mathcal{AR}$	59304	59304	59304	59304	59304	59304	59304
Nursery	$\mathcal{RR}$	2883	658	1738	1846	573	612	554
	$\mathcal{S}ky\text{-}\mathcal{AR}$	497	2	304	342	8	2	2
	$\mathcal{ROSC}$	23872	23901	23875	23417	23176	22806	22139
(2,00)	$\mathcal{AR}$	25062	25062	25062	25062	25062	25062	25062
Zoo	$\mathcal{RR}$	11216	493	11161	11124	477	462	446
	$\mathcal{S}ky\text{-}\mathcal{AR}$	9784	36	9415	9112	36	36	36
	$\mathcal{ROSC}$	67991	67305	67872	66146	65328	65116	63926
(10,00)	$\mathcal{AR}$	71302	71302	71302	71302	71302	71302	71302

TABLE 5.4 – Règles représentatives *vs* règles non dominées, $\mathcal{ROSC}$ et $\mathcal{AR}$

- Pour toutes les bases et les combinaisons de mesures utilisées, le nombre de règles représentatives est beaucoup plus petit que celui de toutes les règles d'association. En effet, le nombre des règles représentatives ne dépasse pas 16% de l'ensemble des règles dans le pire des cas avec la base "Zoo" et la combinaison {Confiance, Loevinger} (11216÷71302 = 0,157). Ce rapport atteint 0,3% pour la base "Monks2" et la combinaison {Confiance, Pearl, Rappel} (215÷59976 = 0,003).
- Même en utilisant une approche optimale d'extraction sous contraintes, le nombre de règles générées reste tout de même important par rapport à celui des règles représentatives. En effet, le rapport entre nombre des règles représentatives et celui des $\mathcal{ROSC}$ atteint 0,3% pour la base "Monks2"et la combinaison {Confiance, Pearl, Rappel} (215÷59568 = 0,003).
- Le nombre de règles non dominées est toujours inférieur à celui des règles représentatives. Ceci confirme la proposition 1 : l'ensemble des règles non dominées est inclus dans l'ensemble des règles représentatives.

5.4 Conclusion

Dans ce chapitre, nous avons introduit un nouveau modèle de règles d'association permettant de sélectionner les règles les plus pertinentes selon plusieurs mesures de qualité tout en tenant compte des similarités structurelles et sémantiques qui peuvent exister entre les règles. Ce modèle permet de retenir un nombre réduit de règles qui sont informatives, non redondantes et représentent le meilleur compromis entre les différentes évaluations des mesures choisies par l'expert. Nous avons également introduit des méthodes permettant d'extraire les règles de ce modèle, appelées *règles représentatives*, en tenant compte de la nature des similarités structurelles des règles qui peuvent être transitives ou non transitives.

Points clés

- Nous avons montré que la sélection des règles d'association selon plusieurs mesures basée uniquement sur la relation de dominance peut entraîner une perte d'informations dans le résultat présenté à l'expert.
- Nous avons introduit le modèle des *règles d'association représentatives* dont l'objectif est de fournir un ensemble réduit de règles qui, à la fois, véhiculent le maximum d'informations utiles et expriment le meilleur compromis entre les différentes évaluations des mesures utilisées.
- Nous avons introduit différentes méthodes permettant de sélectionner les règles représentatives en tenant compte de la nature des similarités structurelles qui peuvent être transitives ou non transitives.

Chapitre 6

Conclusion et perspectives

6.1 Conclusion

Durant ces dernières années, les bases de données ont pris une place de plus en plus importante dans tous les secteurs d'activité. Il n'est plus aujourd'hui imaginable de développer une application commerciale ou industrielle, qui ne s'appuie pas sur une base de données et/ou de connaissances. Suite à la prolifération de ces bases, il devient nécessaire de pouvoir en exploiter le contenu de façon intelligente. En effet, le besoin d'interpréter et de trouver de nouvelles relations entre les éléments stockés dans ces bases a suscité beaucoup d'intérêt. Dans ce cadre, le processus d'extraction des connaissances à partir des données (ECD) se propose de fournir un certain nombre de techniques pour l'extraction de connaissances nouvelles, potentiellement utiles et compréhensibles à partir d'une grande collection de données.

Dans cette thèse, nous nous sommes intéressés particulièrement au problème d'extraction de règles d'association qui a été intensivement étudié depuis sa définition par Agrawal et al [AIS93b]. Plusieurs algorithmes ont été proposés permettant de résoudre ce problème. Cependant, ces algorithmes produisent un nombre difficilement exploitable de règles d'association. Par conséquent, l'expert noyé dans cette masse de connaissances ne peut tirer profit de cette connaissance. Il s'avère alors indispensable d'aider l'expert dans sa recherche des règles pertinentes à l'aide des mesures de qualité dont l'objectif est d'évaluer les règles. Dans les deux dernières décennies, une panoplie de mesures de qualité obéissant à différentes sémantiques ont été proposées dans la littérature (environ une soixantaine). Néanmoins, le problème est loin d'être résolu car les mesures proposées sont très hétérogènes. En effet, une règle peut être considérée pertinente selon une

mesure et non pertinente selon une autre. Ainsi, dans un cadre coopératif où plusieurs experts interviennent dans la décision ayant chacun une préférence pour une mesure de qualité, le problème de recherche des règles se posent dans d'autres termes : "Comment déterminer les règles d'association pertinentes selon plusieurs mesures ?" Afin d'apporter une réponse à cette question, nous avons, tout d'abord, passé en revue les différentes approches d'agrégation des préférences et nous avons montré leur connexion avec l'approche de recherche de règles d'association. Ensuite, nous avons proposé notre première contribution qui consiste à proposer une approche permettant de sélectionner les règles pertinentes selon plusieurs mesures en se basant sur la notion de dominance de Pareto. Dans le cadre de cette approche, nous avons proposé un algorithme, appelé SkyRule, qui permet d'éliminer les règles dominées, *i.e,* celles qui ne sont pas pertinentes selon la combinaison de toutes les mesures. Ainsi, seules les règles non dominées sont retenues.

En fouille de données, la recherche des k meilleures règles selon une mesure de qualité se révèle très utile pour trouver les règles d'association les plus significatives par rapport aux préférences de l'expert. Ainsi, comme deuxième contribution, nous avons proposé une approche *top-k* permettant de sélectionner les k meilleures règles selon plusieurs mesures en se basant sur la relation de dominance. Nous avons alors traité les deux cas possibles suivants :

1. Si k est inférieur au nombre de règles non dominées, alors une sélection parmi les règles non dominées est effectuée.

2. Si k est supérieur au nombre de règles non dominées, alors une sélection parmi les règles dominées est effectuée afin de rajouter celles qui sont pertinentes au résultat final.

Dans le but d'améliorer la qualité des connaissances présentées à l'expert, nous avons aussi proposé d'intégrer la similarité sémantique qui peut exister entre les règles comme contraintes dans le processus de sélection des règles d'association selon plusieurs mesures. L'objectif de cette contrainte est de ne pas permettre à une règle r d'éliminer une autre règle r' lorsque les deux règles véhiculent des informations totalement différentes, même si r domine r'. Dans le cadre de cette approche, nous avons introduit le modèle de règles représentatives dont l'objectif est de sélectionner un ensemble réduit de règles, qui sont à la fois les plus représentatives des données (*i.e.,* véhiculant le maximum d'informations utiles) et non redondantes (*i.e.,* ne véhiculant pas la même information).

6.2 Perspectives

Les travaux réalisés dans le cadre de cette thèse ansi que les résultats obtenus nous permettent d'envisager diverses perspectives de prolongement de nos travaux, selon les orientation suivantes :

- **Priorité dans les préférences :** dans notre travail, la dominance de Pareto est utilisée dans le but de sélectionner les règles d'association selon plusieurs mesures présentant les préférences d'un ou plusieurs experts. Cette sélection est faite en accordant la même importance pour toutes les mesures choisies. Toutefois, l'expert pourrait exprimer ses préférences sous une autre forme tel que "je préfère la mesure m_1 à la mesure m_2" qui traduit la priorité de m_1 par rapport à m_2. Par exemple un expert peut choisir d'utiliser les deux mesures $confiance$ et $lift$, mais pour lui, la mesure $confiance$ est plus importante que la mesure $lift$. Ainsi, dans ce cas, la mesure $lift$ est considérée seulement lorsque deux règles possèdent la même valeur de $confiance$. Cette situation peut également se produire dans un environnement multi-experts où les préférences d'un expert peuvent être prioritaires à celles d'un autre expert. À cet effet, nous proposons d'étudier la sélection des règles d'association en considérant la notion de priorité entre les mesures. Cette notion va permettre une gestion plus flexible des préférences des experts et une meilleure exploitation des connaissances.
- **Outil de visualisation :** les approches que nous avons proposées dans cette thèse fournissent un ensemble de règles sélectionnées selon les préférences d'un ou plusieurs experts. En revanche, les experts pourront avoir du mal à comprendre pourquoi une règle est sélectionnée ou encore quelles sont les mesures qui ont permis à une telle règle de faire partie des non dominées ou des représentatives. De telles informations permettent aux experts d'approfondir davantage leur analyse. Une solution possible consiste à exploiter la visualisation d'information [CMS99] afin de présenter ces informations d'une manière graphique. En effet, plusieurs outils de visualisation ont été proposées permettant de visualiser les règles d'association. Parmi ces outils, nous citons IRSetNav [FR04] (basé sur la représentation textuelle), GerVis [YN06] (basé sur la représentation en deux dimensions), l'outil Minset Tree Viewer [Leh00] (basé sur la représentation en trois dimensions) et l'outil ARvis[BGRB03] (basé sur la réalité virtuelle). Par conséquent, il serait intéressant de proposer un environnement interactif intégrant nos approches et permettant la visualisation des règles sélectionnées avec leurs valeurs associées aux différentes mesures.
- **Intégrer la sélection dans le processus de génération des règles :** afin d'amélio-

rer la performance du processus de sélection de règles représentatives, une solution consisterait à intégrer la sélection dans le processus de la génération des règles d'association. Toutefois, la génération de règles dans un ordre arbitraire peut entraîner la perte de certaines règles représentatives. Supposons qu'à partir d'un contexte d'extraction, seules les trois règles $r_1 : a \rightarrow b$, $r_2 : ac \rightarrow b$ et $r_3 : c \rightarrow b$ sont générées et que $r_1 \succ r_2 \succ r_3$. Il est facile de voir que r_1 et r_3 sont des règles représentatives. Cependant, si r_2 et r_3 sont générées avant r_1, alors r_2 va éliminer la règle représentative r_3 puisqu'elles sont comparables et $r_2 \succ r_3$. Un ordre de comparaison entre les règles peut aussi entraîner la perte de certaines règles représentatives. Par exemple, si r_1 et r_3 sont générées avant r_2, alors deux cas peuvent être distingués après la génération de r_2 :

1er cas : Si r_2 est comparée à r_3 en premier lieu, alors r_2 va éliminer la règle représentative r_3. Par conséquent, seule la règle représentative r_1 sera retenue puisqu'elle va éliminer r_2.

2ème cas : Si r_2 est comparée à r_1 en premier lieu, alors r_1 va éliminer la règle r_2. Par conséquent, les règles représentatives r_1 et r_3 seront retenues.

Ainsi, il serait intéressant d'étudier l'ordre de la génération des règles d'association afin de trouver l'ordre adéquat permettant l'intégration de la sélection dans le processus de la génération des règles d'association.

- **Parallélisme :** Une autre solution permettra d'améliorer les performances des processus de sélection de règles consisterait à intégrer les récentes avancées des architectures multi-processeurs dans le but de paralléliser les algorithmes SkyRules, TransRule, RprRule_1 et RprRule_2. Dans ce contexte, nous suggérons de sélectionner les règles non dominées ainsi que les règles représentatives en utilisant plusieurs processeurs en exploitant les propriétés suivantes :

  Propriété 4 *Soit $\Omega = (\mathcal{AR}, \mathcal{M})$ une table de relation avec $\mathcal{AR}$ est l'ensemble des règles d'association et $\mathcal{M}$ est l'ensemble des mesures. Soit $\Omega' = (\mathcal{AR}', \mathcal{M})$ une sous-table de Ω tel que $\mathcal{AR}' \subseteq \mathcal{AR}$.*

  – *Si r est dominée dans Ω' alors r est aussi dominée dans Ω.*

  – *Si r est non représentative dans Ω' alors r est aussi non représentative dans Ω.*

  Ainsi, la table de relation $\Omega = (\mathcal{AR}, \mathcal{M})$ peut être décomposée horizontalement en des sous-tables dont chacune sera traitée par un processeur. Chaque processeur appliquera l'algorithme SkyRules (respectivement TransRule, RprRule_1 et

RprRule_2) sur une sous-table. Les règles élaguées sont des règles dominées (respectivement non représentatives) dans Ω, tandis que les règles retenues sont des candidats à être non dominées (respectivement représentatives). L'espace de recherche étant réduit, il est alors possible de sélectionner les règles non dominées (respectivement représentatives) en appliquant SkyRules (respectivement TransRule, RprRule_1 et RprRule_2) sur les règles candidats. Il serait aussi important d'étudier le comportement et la scalabilité des algorithmes SkyRules, TransRule, RprRule_1 et RprRule_2 sur un nombre important de processeurs.

- **Règles complexes :** nous proposons d'étendre la sélection des règles représentatives à des règles traduisant des corrélations entre des motifs plus complexes tels que les graphes. Ce problème n'a pas été traité en pratique pourtant la notion de règle d'association est théoriquement généralisable à tout ensemble ordonné. En effet, une règle d'association entre graphes peut être définie comme une règle $G_1 \rightarrow G_2$ où G_1 sont des graphes connexes tel que G_1 est isomorphe à G_2. Cependant, dans ce cas, la mise en œuvre de la comparabilité entre les règles nécessite des calculs plus lourds que dans le cas des motifs d'items. En effet, la comparaison entre les règles de type graphes consiste à vérifier l'isomorphisme entre les graphes, tandis que la comparaison entre les motifs d'items est triviale.
- **Règles négatives :** les règles d'association ont été introduites pour exprimer des corrélations entre les occurrences des items. Toutefois, l'expert a besoin aussi d'une technique qui exprime la corrélation entre la non occurrence des items. À cet effet, il serait intéressant de prendre en considération les règles d'association négatives de la forme $\overline{X} \rightarrow Y$, $X \rightarrow \overline{Y}$ et $\overline{X} \rightarrow \overline{Y}$. Nous pouvons envisager d'appliquer la sélection des règles non dominées (respectivement règles représentatives) sur les règles positives et négatives. Cependant, présenter simultanément les deux règles $X \rightarrow Y$ et $X \rightarrow \overline{Y}$ peut entraîner une exploitation et une interprétation inefficaces des connaissances de la part de l'expert. Ainsi, nous envisageons d'étendre les notions de dominance et de comparabilité afin de pouvoir choisir entre $X \rightarrow Y$ et $X \rightarrow \overline{Y}$.

Bibliographie

[AIS93a] R. Agrawal, T. Imielinski, and A. Swami. Database mining :a performance perspective. *IEEE Transactions on Knowledge and Data Engineering*, 5(6) :914–925, 1993.

[AIS93b] R. Agrawal, T. Imielinski, and A. Swami. Mining association rules between sets of items in large databases. In *Proceedings of the ACM SIGMOD Intl. Conference on Management of Data, Washington, USA*, pages 207–216, June 1993.

[Arm74] W.W. Armstrong. Dependency structures of database relationships. In *IFIP Congress*, pages 580–583, September 1974.

[Arr51] K-J. Arrow. Alternative Approaches to the Theory of Choice in Risk-Taking Situations. *Econometrica*, 19(4), 1951.

[AS94] R. Agrawal and R. Skirant. Fast algorithms for mining association rules. In *Proceedings of the 20th Intl. Conference on Very Large Databases, Santiago, Chile*, pages 478–499, June 1994.

[Azé03] J. Azé. *Extraction de connaissances à partir de données numériques et textuelles*. PhD thesis, Paris 11, 2003. Thèse doctorat : Informatique.

[Azé03] J. Azé. Une nouvelle mesure de qualité pour l'extraction de pépites de connaissances. In *EGC*, pages 171–182, 2003.

[BF07] S-J. Brams and P-C. Fishburn. *Approval voting (2. ed.)*. Springer, 2007.

[BGBG05] J. Blanchard, F. Guillet, H. Briand, and R. Gras. Assessing rule interestingness with a probabilistic measure of deviation from equilibrium. In *Proceedings of the 11th international symposium on Applied Stochastic Models and Data Analysis ASMDA-2005*, pages 191–200. ENST, 2005.

[BGG+11] R. Belohlávek, D. Grissa, S. Guillaume, E. Mephu Nguifo, and J. Outrata. Boolean factors as a means of clustering of interestingness measures of association rules. In *CLA*, pages 207–222, 2011.

[BGGB05] J. Blanchard, F. Guillet, R. Gras, and H. Briand. Using information-theoretic measures to assess association rule interestingness. In *ICDM*, pages 66–73, 2005.

[BGRB03] J. Blanchard, F. Guillet, F. Rantière, and H. Briand. Vers une représentation graphique en réalité virtuelle pour la fouille interactive de règles d'association. In *EGC*, pages 105–117, 2003.

[BJ10] J-F. Boulicaut and B. Jeudy. Constraint-based data mining. In *Data Mining and Knowledge Discovery Handbook*, pages 339–354. 2010.

[BKK01] S. Börzsönyi, D. Kossmann, and K.Stocker. The skyline operator. In *Proceedings of the 17th International Conference on Data Engineering*, pages 421–430, Washington, DC, USA, 2001. IEEE Computer Society.

[BLLV06] J-P. Barthélemy, A. Legrain, P. Lenca, and B. Vaillant. Aggregation of valued relations applied to association rule interestingness measures. In *Modeling Decisions for Artificial Intelligence, Third International Conference, MDAI 2006, Tarragona, Spain, April 3-5, 2006, Proceedings*, pages 203–214, 2006.

[BM70] M. Barbut and B. Monjardet. *Ordre et classification. Algèbre et Combinatoire.* Hachette, Tome II, 1970.

[BMS97] S. Brin, R. Motawni, and C. Silverstein. Beyond market baskets :generalizing association rules to correlation. In *Proceedings of the SIGMOD, Tucson, Arizona (USA)*, pages 265–276, May 1997.

[BMU97] S. Brin, R. Motawni, and J. D. Ullman. Dynamic itemset counting and implication rules for market basket data. In *Proceedings of the ACM SIGMOD, Tucson, Arizona (USA)*, pages 255–264, May 1997.

[Bor] J-C. Borda. Mémoire sur les élections au scrutin.

[BPT+00] Y. Bastide, N. Pasquier, R. Taouil, L. Lakhal, and G. Stumme. Mining minimal non-redundant association rules using frequent closed itemsets. In *Proceedings of the Intl. Conference DOOD'2000, LNCS, Springer-Verlag*, pages 972–986, July 2000.

[Bra07] S-J. Brams. Mathematics and democracy - designing better voting and fair-division procedures. pages I–XVI, 1–373. Princeton University Press, 2007.

[BSYN12] S. Bouker, R. Saidi, S. Ben Yahia, and E. Mephu Nguifo. Ranking and selecting association rules based on dominance relationship. In *ICTAI*, pages 658–665, 2012.

[BSYN14] S. Bouker, R. Saidi, S. Ben Yahia, and E. Mephu Nguifo. Mining undominated association rules through interestingness measures. *International Journal on Artificial Intelligence Tools*, 23(4), 2014.

[BTP+00] Y. Bastide, R. Taouil, N. Pasquier, G. Stumme, and L. Lakhal. Mining frequent patterns with counting inference. *SIGKDD Explorations*, 2(2) :66–75, 2000.

[BTP+02] Y. Bastide, R. Taouil, N. Pasquier, G. Stumme, and L. Lakhal. PASCAL : un algorithme d'extraction des motifs fréquents. *Techniques et Science Informatiques*, 21(1) :65–95, 2002.

[CMS99] S-K. Card, J-D. Mackinlay, and B. Shneiderman. *Readings in information visualization - using vision to think*. Academic Press, 1999.

[Con85] N. Condorcet. Essai sur l'application de l'analyse à la probabilité des décisions rendues à la pluralité des voix. *Imprimerie royale, Paris*, 1785.

[CVMC08] P. Chen, R-M. Verma, J-C. Meininger, and W. Chan. Semantic analysis of association rules. In *FLAIRS Conference*, pages 270–275, 2008.

[Fay96] U. M. Fayyad. Data mining and knowledge discovery :making sense out of data. *IEEE Expert*, 11(5) :20–25, 1996.

[FC07] F.Bonchi and C.Lucchese. Extending the state-of-the-art of constraint-based pattern discovery. *Data Knowl. Eng.*, (2) :377–399, 2007.

[FG82] P.C. Fishburn and W.V. Gehrlein. Majority efficiency for simple voting procedures :summary and interpretation. In *Theory Decision, 4*, pages 141–153, 1982.

[FPSM92] W-J. Frawley, G. Piatetsky-Shapiro, and C-J. Matheus. Knowledge discovery in databases :an overview. *AI Magazine*, 13(3) :57–70, 1992.

[FR04] P. Fule and J-F. Roddick. Experiences in building a tool for navigating association rule result sets. In James M. Hogan, Paul Montague, Martin K. Purvis, and Chris Steketee, editors, *ACSW Frontiers*, volume 32 of *CRPIT*, pages 103–108. Australian Computer Society, 2004.

[Fre99] A-A. Freitas. On rule interestingness measures. *Knowl.-Based Syst.*, 12(5-6) :309–315, 1999.

[FVWT12] P. Fournier-Viger, C-W. Wu, and V-S. Tseng. Mining top-k association rules. In *Canadian Conference on AI*, pages 61–73, 2012.

[GD86] J.L. Guigues and V. Duquenne. Familles minimales d'implications informatives résultant d'un tableau de données binaires. *Mathématiques et Sciences Humaines*, (95) :5–18, 1986.

[GGN11] S. Guillaume, D. Grissa, and E. Mephu Nguifo. Catégorisation des mesures d'intérêt pour l'extraction des connaissances. In *EGC*, pages 551–562, 2011.

[GH06] L. Geng and H-J. Hamilton. Interestingness measures for data mining :a survey. *ACM Comput. Surv.*, 38(3), 2006.

[GPW98] G. Gardarin, P. Pucheral, and F. Wu. Bitmap based algorithms for mining association rules. In M. Bouzeghoub, editor, *Proceedings of 14th Intl. Conférence Bases de Données Avancées, Hammamet, Tunisia*, pages 157–175, 26–30 October 1998.

[GW99] B. Ganter and R. Wille. Formal concept analysis - mathematical foundations. pages I–X, 1–284. Springer, 1999.

[HGB06] H. Xuan Huynh, F. Guillet, and H. Briand. Discovering the stable clusters between interestingness measures. In *ICEIS (2)*, pages 196–201, 2006.

[HK00] J. Han and M. Kamber. Data mining :concepts and techniques. Morgan Kaufmann, 2000.

[HPY00] J. Han, J. Pei, and Y. Yin. Mining frequent patterns without candidate generation. In *Proceedings of the ACM-SIGMOD Intl. Conference on Management of Data, Dallas, Texas*, pages 1–12, May 2000.

[HS00] E. Hartuv and R. Shamir. A clustering algorithm based on graph connectivity. *Inf. Process. Lett.*, 76(4-6) :175–181, 2000.

[H.T96] H.Toivonen. Discovery of frequent pattern in large data collections. In *Ph.D. Thesis, Report A-1996-5, University of Helsinki*, 1996.

[HY81] C-L. Hwang and K. Yoon. Multiple attribute decision making :methods and applications. 1981.

[HYH+05] H. Hu, X. Yan, Y. Huang, J. Han, and X. Zhou. Mining coherent dense subgraphs across massive biological networks for functional discovery. *Bioinformatics*, 21(1) :213–221, January 2005.

[HZ10] M-J. Heravi and O. Zaïane. A study on interestingness measures for associative classifiers. In *SAC*, pages 1039–1046, 2010.

[JA99] Roberto J. Bayardo Jr. and Rakesh Agrawal. Mining the most interesting rules. In *KDD*, pages 145–154, 1999.

[JAG99] Roberto J. Bayardo Jr., Rakesh Agrawal, and Dimitrios Gunopulos. Constraint-based rule mining in large, dense databases. In *ICDE*, pages 188–197, 1999.

[JB02] B. Jeudy and J-F. Boulicaut. Constraint-based discovery and inductive queries :application to association rule mining. In *Pattern Detection and Discovery*, pages 110–124, 2002.

[Kac11] S. Kaci. *Working with Preferences :Less Is More.* Cognitive Technologies. Springer, 2011.

[KG02] M. Kryszkiewicz and M. Gajek. Concise representation of frequent patterns based on generalized disjunction-free generators. In *PAKDD*, pages 159–171, 2002.

[KLP75] H. T. Kung, Fabrizio Luccio, and Franco P. Preparata. On finding the maxima of a set of vectors. *J. ACM*, 22(4) :469–476, 1975.

[KRR02] D. Kossmann, F. Ramsak, and S. Rost. Shooting stars in the sky :an online algorithm for skyline queries. In *VLDB*, pages 275–286, 2002.

[Kry98] M. Kryszkiewicz. Representative association rules. In *Research and Development in Knowledge Discovery and Data Mining. Proc. of Second Pacific-Asia Conference (PAKDD). Melbourne, Australia*, 1998.

[Kry02a] M. Kryszkiewicz. Concise representations of association rules. In D. J. Hand, N.M. Adams, and R.J. Bolton, editors, *Proceedings of Pattern Detection and Discovery, ESF Exploratory Workshop, London, UK*, volume 2447 of *Lecture Notes in Computer Science*, pages 92–109. Springer-Verlag, September 2002.

[Kry02b] M. Kryszkiewicz. *Concise Representations of frequent patterns and association rules.* PhD thesis, Institute of Computer Science Warsaw University of Technology, 2002.

[Leh00] R. Lehn. *Un système interactif de visualisation et de fouille de règles pour l'extraction de connaissances dans les bases de données.* Doctorat d'université, Université de Nantes, France, 2000.

[LFZ99] N. Lavrac, P-A. Flach, and B. Zupan. Rule evaluation measures :a unifying view. In *ILP*, pages 174–185, 1999.

[LHCM00] B. Liu, W. Hsu, S. Chen, and Y. Ma. Analyzing the subjective interestingness of association rules. *IEEE Intelligent Systems*, 15(5) :47–55, 2000.

[LJZ11] H. Lu, C-S. Jensen, and Z. Zhang. Flexible and efficient resolution of skyline query size constraints. *IEEE Trans. Knowl. Data Eng.*, 23(7) :991–1005, 2011.

[LMPV03] P. Lenca, P. Meyer, P. Picouet, and B. Vaillant. Aide multicritère à la décision pour évaluer les indices de qualité des connaissances. In *EGC*, pages 271–282, 2003.

[LMVL08] P. Lenca, P. Meyer, B. Vaillant, and S. Lallich. On selecting interestingness measures for association rules : User oriented description and multiple criteria decision aid. *European Journal of Operational Research*, 184(2) :610–626, 2008.

[LN90] M. Liquière and E. Mephu Nguifo. Legal (learning with galois lattice) : Un système d'apprentisage de concepts à partir d'exemples. In *Proceedings of the Intl. 5th Journées Francaises de l'apprentissage, Lannion, France*, pages 93–114, 1990.

[Loe47] J. Loevinger. *A systemic approach to the construction and evaluation of tests of ability Application.* Psychological monographs, 1947.

[LT04] S. Lallich and O. Teytaud. Evaluation et validation de mesures d'interêt des règles d'association. In *RNTI-E-1, spécial 193-217 2004*, 2004.

[Lux91] M. Luxenburger. Implication partielles dans un contexte. *Mathématiques et Sciences Humaines*, 29(113) :35–55, 1991.

[LVML07] P. Lenca, B. Vaillant, P. Meyer, and S. Lallich. Association rule interestingness measures :experimental and theoretical studies. In *Quality Measures in Data Mining*, pages 51–76. 2007.

[Mat91] J. Matousek. Computing dominances in E^n. *Inf. Process. Lett.*, 38(5) :277–278, 1991.

[MM04] K. McGarry, , and J. Malone. The analysis of rules discovered by the data mining process. *Applications and Science in Soft Computing Series :Advances in Soft Computing*, pages 219–224, 2004.

[MS] V. Merlin and D-G. Sarri. Copeland method. manipulation, monotonicity, and paradoxes.

[Ngu94] E. Mephu Nguifo. Galois lattice :A framework for concept learning-design, evaluation and refinement. In *Sixth International Conference on Tools with Artificial Intelligence, ICTAI '94, New Orleans, Louisiana, USA, November 6-9, 1994*, pages 461–467, 1994.

[NLHP98] Raymond T. Ng, Laks V. S. Lakshmanan, J. Han, and A. Pang. Exploratory mining and pruning optimizations of constrained association rules. In *SIGMOD Conference*, pages 13–24, 1998.

[Pas00] N. Pasquier. *Data Mining :algorithmes d'extraction et de réduction des règles d'association dans les bases de données*. Doctorat d'université, Université de Clermont-Ferrand II, France, 2000.

[PBTL98] N. Pasquier, Y. Bastide, R. Touil, and L. Lakhal. Pruning closed itemset lattices for association rules. In M. Bouzeghoub, editor, *Actes des 14éme journées de Données Avancées, Hammamet, Tunisia*, pages 177–196, 26–30 October 1998.

[PBTL99a] N. Pasquier, Y. Bastide, R. Taouil, and L. Lakhal. Efficient Mining of Association Rules Using Closed Itemset Lattices. *Information Systems Journal*, 24(1) :25–46, 1999.

[PBTL99b] N. Pasquier, Y. Bastide, R. Touil, and L. Lakhal. Discovering frequent closed itemsets for association rules. In *Proceedings of 7th International Conference on Database Theory (ICDT'99), LNCS, Vol. 1540, Springer Verlag*, pages 398–416, January 1999.

[PCY95] J.S. Park, M. Chen, and P.S. Yu. An effective hash based algorithm for mining association rules. In *Proceedings of the ACM SIGMOD, San Jose, California*, pages 175–186, May 1995.

[Pea88] J. Pearl. On logic and probability. *Computational Intelligence*, 4 :99–103, 1988.

[PLM08] A-N. Papadopoulos, A. Lyritsis, and Y. Manolopoulos. Skygraph :an algorithm for important subgraph discovery in relational graphs. *Data Min. Knowl. Discov.*, 17(1) :57–76, 2008.

[PN09] F. Pennerath and A. Napoli. The model of most informative patterns and its application to knowledge extraction from graph databases. In *ECML/PKDD (2)*, pages 205–220, 2009.

[PS91] G. Piatetsky-Shapiro. Discovery, analysis, and presentation of strong rules. In *Knowledge Discovery in Databases*, pages 229–248. AAAI/MIT Press, 1991.

[RF07] John F. Roddick and Peter Fule. Semgram - integrating semantic graphs into association rule mining. In Peter Christen, Paul J. Kennedy, Jiuyong Li, Inna Kolyshkina, and Graham J. Williams, editors, *Sixth Australasian Data*

Mining Conference (AusDM 2007), volume 70 of *CRPIT*, pages 129–137, Gold Coast, Australia, 2007. ACS.

[RFIG04] W. Romao, A. Freitas, M. Itana, and S. Gimenes. Discovering interesting knowledge from a science and technology database with a genetic algorithm. In *In Applied Soft Computing 4*, pages 121–137, 2004.

[RLM13] C. Rudin, B. Letham, and D. Madigan. Learning theory analysis for association rules and sequential event prediction. *Journal of Machine Learning Research*, 14(1) :3441–3492, 2013.

[Rol13] A. Rolland. Reference-based preferences aggregation procedures in multi-criteria decision making. *European Journal of Operational Research*, 225(3) :479–486, 2013.

[Roy72] B. Roy. Décision avec critères multiples : Problèmes et méthodes. 6(1) :121–151, 1972.

[SC07] A. Soulet and B. Crémilleux. Extraction des top-k motifs par approximer-et-pousser. In *EGC*, pages 271–282, 2007.

[SM02] J. Sese and S. Morishita. Answering the most correlated n association rules efficiently. In *PKDD*, pages 410–422, 2002.

[SON95] A. Savasere, E. Omiecinski, and S. Navathe. An efficient algorithm for mining association rules in large databases. In *Proceedings of the 21th VLDB Conference, Zurich, Switzerland*, pages 432–444, September 1995.

[SRPC11] A. Soulet, C. Raïssi, M. Plantevit, and B. Crémilleux. Mining dominant patterns in the sky. In *ICDM*, pages 655–664, 2011.

[ST95] A. Silberschatz and A. Tuzhilin. On subjective measures of interestingness in knowledge discovery. In *KDD*, pages 275–281, 1995.

[STB+02] G. Stumme, R. Taouil, Y. Bastide, N. Pasquier, and L. Lakhal. Computing iceberg concept lattices with TITANIC. *J. on Knowledge and Data Engineering (KDE)*, 2(42) :189–222, 2002.

[SVA97] R. Srikant, Q. Vu, and R. Agrawal. Mining Association Rules with Item Constraints. In David Heckerman, Heikki Mannila, Daryl Pregibon, and Ramasamy Uthurusamy, editors, *Proc. 3rd Int. Conf. Knowledge Discovery and Data Mining, KDD*, pages 67–73. AAAI Press, 1997.

[Tid06] N. Tideman. *Collective Decisions and Voting :The Potential for Public Choice*. Ashgate Publishing, 2006.

[TKS02] P. Tan, V. Kumar, and J. Srivastava. Selecting the right interestingness measure for association patterns. In *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (ICDM'02), ACM Press*, pages 32–41, 2002.

[Vai06] B. Vaillant. *Mesurer la qualité des règles d'association : études formelles et expérimentales.* PhD thesis, Université de Bretagne sud, 2006. Thèse doctorat : Informatique.

[VLL04] B. Vaillant, P. Lenca, and S. Lallich. A clustering of interestingness measures. In *Discovery Science*, pages 290–297, 2004.

[Web11] G-I. Webb. Filtered-top-k association discovery. *Wiley Interdisc. Rew. :Data Mining and Knowledge Discovery*, 1(3) :183–192, 2011.

[Wil82] R. Wille. Restructuring Lattice Theory :An approach based on hierarchies of concepts. pages 445–470. Reidel Edition, 1982.

[Wil89] R. Wille. Knowledge acquisition by methods of formal concept analysis. In E. Diday, editor, *Data analysis, learning symbolic and numeric knowledge.* Nova Science, New York, 1989.

[YGN09] S. Ben Yahia, G. Gasmi, and E. Mephu Nguifo. A new generic basis of "factual" and "implicative" association rules. *Intell. Data Anal.*, 13(4) :633–656, 2009.

[YHC06] C. Hun You, L-B. Holder, and D-J. Cook. Application of graph-based data mining to metabolic pathways. In *ICDM Workshops*, pages 169–173, 2006.

[YMH+07] X. Yan, M. Mehan, Y. Huang, M. Waterman, P. Yu, and X. Zhou. A graph-based approach to systematically reconstruct human transcriptional regulatory modules. In *ISMB/ECCB (Supplement of Bioinformatics)*, pages 577–586, 2007.

[YN06] S. Ben Yahia and E. Mephu Nguifo. Visualisation des règles associatives : vers une approche méta-cognitive. In *INFORSID*, pages 735–750, 2006.

[Zak00] M. J. Zaki. Generating non-redundant association rules. In *Proceedings of the 6th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining,Boston, MA*, pages 34–43, August 2000.

[Zha00] T. Zhang. Association rules. In *PAKDD*, pages 245–256, 2000.

[ZP03] M. J. Zaki and B. Phoophakdee. Mirage :a framework for mining, exploring and visualizing minimal association rules. Technical report, Computer Science Dept, Rensselaer Polytechnic Institute., 2003.

[ZPOL97] M.J. Zaki, S. Pathasarathy, M. Ogihara, and W. Li. New algorithms for fast discovering association rules. In *Proceedings of the third international conference on Knowledge Discovery and Datamining (KDD'97)*, pages 283–286, August 1997.

Résumé

La fouille de données est un processus qui vise à extraire un ensemble réduit de connaissances à fortes valeurs ajoutées à partir d'un grand volume de données. Parmi les techniques de fouille de données, nous trouvons l'extraction des règles associations qui permettent d'identifier les liens entre attributs décrivant les objets dans une base de données. Les règles d'association ont montré leur utilité dans plusieurs domaines d'application tel que la gestion de la relation client dans la grande distribution (déterminer les produits souvent achetés simultanément, organiser les rayons et proposer des promotions en conséquence), la biologie moléculaires (déterminer les associations entre les gènes), etc. Toutefois, les algorithmes d'extraction de règles d'association présentent l'inconvénient de générer un nombre important de règles, dont beaucoup se révèlent sans aucun intérêt pour l'expert. Ainsi, dans le but de sélectionner les règles pertinentes, plusieurs mesures de qualité ont été proposées dans la littérature dont l'objectif est d'associer une valeur numérique à une règle permettant de quantifier son intérêt. Néanmoins, le problème est loin d'être résolu car les mesures proposées sont très hétérogènes puisqu'une règle peut être considérée pertinente selon une mesure et non pertinente selon une autre. Ce genre d'observations permet de soulever de nouvelles questions : Comment déterminer les meilleures règles en présence de plusieurs experts simultanément, ayant chacun une préférence pour une mesure de qualité ? Comment déterminer les règles les plus pertinentes lorsqu'un expert a des préférences pour différentes mesures ?

Dans cette thèse, nous avons essayé de répondre à ces questions en proposant trois approches permettant de sélectionner les règles d'association pertinentes selon différentes mesures en se basant sur la relation de dominance. La première approche consiste à éliminer les règles dont les évaluations selon toutes les mesures utilisées sont inférieures à celles d'autres règles, elles sont appelées : règles dominées. Ainsi, seules les *règles non dominées* sont retenues. La deuxième approche consiste à ordonner les règles d'association selon plusieurs mesures pour en sélectionner les k meilleures règles, où k est fixé par l'expert. La troisième approche consiste à sélectionner un ensemble réduit de règles selon plusieurs mesures, appelées *règles représentatives*, tout en tenant compte de l'aspect sémantique des règles. L'objectif de cette approche est en effet de fournir un ensemble réduit de règles qui, à la fois, véhiculent le maximum d'informations utiles et expriment le meilleur compromis entre les différentes évaluations des mesures utilisées.

Mots clés : Fouille de données, extraction des règles d'association, mesures de qualité, préférences des experts, relation de dominance.

Abstract

Data mining in Databases offers a process for the non trivial extraction of previously unknown and valuable knowledge from a huge amount of data. Much research in data mining has focuser on extraction of associations rules which are user to identify relationships between attributes in a database. The extraction of association rules can be user in various applications areas such as the customer relationship management in a wide distribution (determine the products often purchased together, organize the shelves and offer promotions), molecular biology (analyze associations between genes), etc. However, association rule mining algorithms have the disadvantage to generate large amounts of rules, many of which do not even have any interest for the expert.Hence, in order to select relevant rules, many interestingness measures have been proposed in the literature which aims to associate a numerical value to a rule to quantify its interest. Nevertheless, the problem is far from resolved because the proposed measures are very heterogenous since rule may be considered relevant by a measure and irrelevant by another one. Such observations allows to rise new questions : How to determine the best rules in a cooperative area where several experts work simultaneously, each with a preference for a measure ? How to determine the most relevant rules when an expert has preferences for different measures ?

In this thesis, we have tried to answer these questions by proposing three approaches, based on dominance relationship, for selecting the relevant association rules when different measures are used. The first approach consists in eliminating rules that evaluations by all measures are lower than others rules, they are called dominated rules. Thus, only *undominated rules* are retained. The second approach consists in ordering association rules according several measures to select the *top-k rules*, where k is set by the expert. The third approach consists in selecting a small set of rules according to several measures, called *representative rules*, taking into account the semantic relationship between rules. The objective of this approach is indeed to provide a small set of rules that convey useful information and express the best compromise between different assessments of used measures.

Keywords : Data mining, extraction of association rules, interestingness measures, experts preferences, dominance relationship.