Extraction de bases pour les règles d'association à partir des itemsets fermés fréquents

Nicolas Pasquier

Extraction de Bases pour les Règles d'Association à partir des Itemsets Fermés Fréquents

Nicolas Pasquier

Laboratoire d'Informatique (LIMOS) - Université Clermont-Ferrand II
Complexe scientifique des Cézeaux, 24 avenue des Landais, 63177 Aubière cedex France
pasquier@libd2.univ-bpclermont.fr, Tel : 04 73 40 77 75, Fax : 04 73 40 74 40
Catégorie : Jeune Chercheur[1]

Résumé :

Le problème de l'utilité et de la pertinence des règles d'association extraites est primordial car, dans la plupart des cas, les jeux de données réels conduisent à plusieurs milliers voire plusieurs millions de règles d'association dont la mesure de confiance est élevée, et parmi lesquelles se trouvent de nombreuses règles redondantes. Utilisant la sémantique basée sur la fermeture de la connexion de Galois, des bases pour les règles d'association, qui sont des ensembles générateurs pour toutes les règles d'association ainsi que leurs supports et leurs confiances, sont définies. Ces bases sont constituées des règles d'association non redondantes d'antécédents minimaux et de conséquences maximales et sont définies à partir des itemsets fermés fréquents et leurs générateurs extraits par les algorithmes Close et A-Close. Des algorithmes efficaces de génération des bases à partir de ces derniers sont présentés et les résultats des expérimentations menées sur des bases de données réelles montrent l'efficacité des algorithmes et l'utilité des bases proposées.

Mots-clés :

Extraction de connaissances dans les bases de données, data mining, fermeture de la connexion de Galois, itemsets fermés fréquents, règles d'association minimales non-redondantes, bases pour les règles d'association, algorithmes.

Abstract :

The problem of the relevance and the usefulness of extracted association rules is of primary importance because, in the majority of cases, real-life databases lead to several thousands and even millions of association rules whose confidence measure is high, and among which are many redundant rules. Using the semantic based on the closure of the Galois connection, we define two new bases for association rules, which are generating sets for all valid association rules, their supports and their confidences. These bases, characterized using the frequent closed itemsets and their generators, consist of the non-redundant association rules of minimal antecedents and maximal consequents, i.e. the most relevant association rules. Algorithms for extracting these bases are presented and results of experiments carried out on real-life databases show the usefulness of the bases proposed.

Keywords :

Knowledge discovery in databases, data mining, closure operator of the Galois connection, frequent closed itemsets, minimal non-redundant association rules, bases for association rules, algorithms.

[1] Les travaux présentés dans cet article ont été réalisés dans le cadre d'une thèse de doctorat sous la direction du professeur Lotfi Lakhal.

1 Introduction

L'extraction dcie règles d'association a pour but de découvrir des relations significatives entre attributs binaires extraits des bases de données. Un exemple de règle d'association extraite d'une base de données de ventes de supermarché est : « céréales $\wedge$ sucre $\rightarrow$ lait (support 7%, confiance 50%) ». Cette règle indique que les clients qui achètent des céréales et du sucre ont également tendance à acheter du lait. La mesure de *support* définit la portée de la règle, c'est à dire la proportion de clients qui ont acheté les trois articles, et la mesure de *confiance* définit la précision de la règle, c'est à dire la proportion de clients qui ont acheté du lait parmi ceux qui ont acheté des céréales et du sucre. L'extraction de règles d'association consiste à extraire les règles dont le support et la confiance sont au moins égaux à des seuils minimaux de support et de confiance définis par l'utilisateur. Les règles d'association ont été utilisées avec succès dans de nombreux domaines, parmi lesquels l'aide à la planification commerciale, l'aide au diagnostic et en recherche médicale, l'amélioration des processus de télécommunications, l'organisation et l'accès aux sites Internet, et l'analyse d'images, de données spatiales, de données géographiques et de données statistiques.

L'extraction de règles d'association est un processus itératif et interactif constitué de plusieurs phases allant de la sélection et la préparation des données jusqu'a l'interprétation des résultats, en passant par la phase de recherche des connaissances : le *data mining* [FPSSU96]. La plupart des approches proposées pour l'extraction des itemsets fréquents reposent sur les quatre phases suivantes :

Préparation des données Cette phase consiste à sélectionner les données (attributs et objets) de la base de données utiles à l'extraction des règles d'association et transformer ces données en un *contexte d'extraction*. Ce contexte, ou jeu de données, est un triplet $B=(O, A, R)$ dans lequel O est un ensemble d'objets, A est un ensemble d'attributs, également appelés *items*, et R est une relation binaire entre O et A. Un contexte d'extraction de règles d'association D constitué de six objets, chacun identifié par son *OID*, et cinq items est représenté dans la Table 1. Ce contexte est utilisée comme support pour les exemples dans la suite de l'article. Cette phase est nécessaire afin qu'il soit possible d'appliquer les algorithmes d'extraction des règles d'association sur des données de natures différentes provenant de sources différentes, de concentrer la recherche sur les données utiles pour l'application et de minimiser les temps d'extraction.

OID	items
1	A C D
2	B C E
3	A B C E
4	B E
5	A B C E
6	B C E

Table 1: Contexte d'extraction de règles d'association D.

Extraction des ensembles fréquents d'attributs Cette phase consiste à extraire du contexte tous les ensembles d'attributs binaires $l \subseteq A$, appelés *itemsets*, qui sont fréquents dans le contexte B. Un itemset l est fréquent si son support, qui correspond au nombre d'objets du contexte qui « contiennent » l, est supérieur ou égal au seuil minimal de support *minsupport* défini par

l'utilisateur. L'ensemble des itemsets fréquents dans le contexte est noté F. Le problème de l'extraction des itemsets fréquents est de complexité exponentielle dans la taille m de l'ensemble d'items puisque le nombre d'itemsets fréquents potentiels est 2^m. Ces itemsets forment un treillis dont la représentation sous forme de diagramme de Hasse pour le contexte D est présentée dans la Figure 1. De plus, des balayages du contexte doivent être réalisés lors de cette phase, et il est donc nécessaire de développer des méthodes efficaces d'exploration de cet espace de recherche exponentiel.

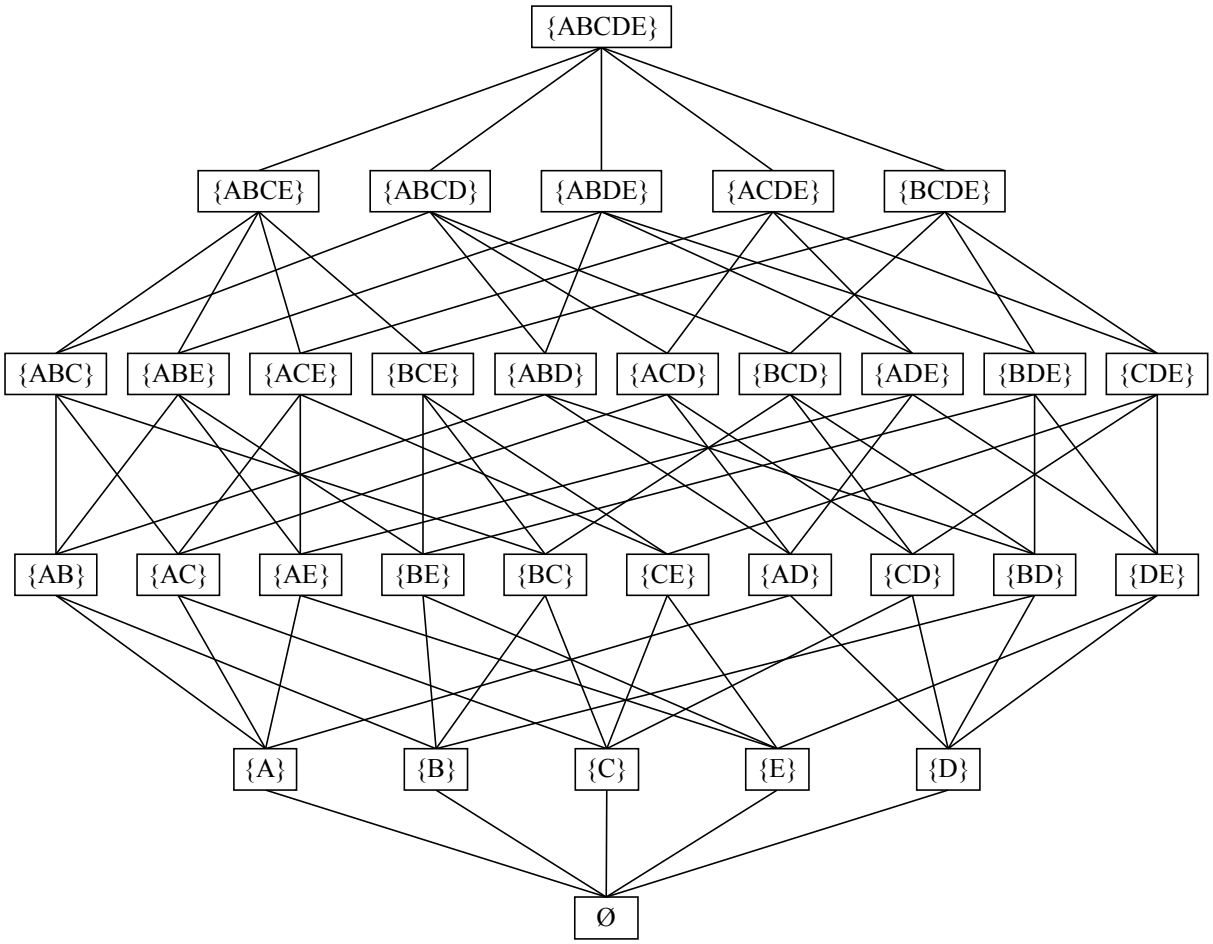

Figure 1 : Diagramme de Hasse représentant le treillis des itemsets.

Génération des règles d'association Durant cette phase, les itemsets fréquents extraits durant la phase précédente sont utilisés afin de générer les règles d'association qui sont des implications entre deux itemsets fréquents $l_1, l_2 \in F$ tels que $l_1 \subset l_2$, de la forme $r : l_1 \rightarrow (l_2 \setminus l_1)$. Afin de limiter l'extraction aux règles d'association les plus informatives, seules celles qui possèdent une confiance supérieure ou égale au seuil minimal *minconfiance* défini par l'utilisateur sont générées. La confiance d'une règle $r : l_1 \rightarrow (l_2 \setminus l_1)$ est définie comme la proportion d'objets contenant la conséquence $(l_2 \setminus l_1)$ de r parmi ceux qui contiennent l'antécédent l_1 de r. Cette valeur est égale au rapport entre le support de l'itemset l_2 et le support de l'itemset l_1.

Interprétation des résultats Cette phase consiste en la visualisation par l'utilisateur des règles d'association extraites du contexte et leur interprétation afin d'en déduire des connaissances utiles pour l'amélioration de l'activité concernée. Le nombre important de règles d'association extraites en général impose le développement d'outils de classification des règles, de sélection par l'utilisateur de sous-ensembles de règles, et de leur visualisation sous une forme intelligible. De

tels outils ont été proposés dans le système Rule Visualizer [KMR+94], qui utilise des *templates* de sélection des règles et permet de les visualiser sous forme textuelle ou bien sous forme de graphes dirigés.

Les connaissances de l'utilisateur concernant le domaine d'application sont nécessaires lors des phases de pré-traitement, afin d'assister la sélection et la préparation des données, et de post-traitement, pour l'interprétation et l'évaluation des règles extraites. En fonction de l'évaluation des règles extraites, les paramètres utilisés lors des précédentes phases (critères de sélection et préparation des données et seuils minimaux de support et de confiance) peuvent être modifiés avant d'effectuer à nouveau l'extraction des règles d'association, ceci afin d'améliorer la qualité du résultat.

Deux problèmes majeurs pour l'utilisation de l'extraction des règles d'association ont donné lieu à de nombreuses recherches : le problème des temps d'extraction des règles d'association à partir du jeu de données et le problème de la pertinence et de l'utilité des règles d'association extraites. Ces deux problèmes et les solutions proposées dans la littérature sont présentés dans les Paragraphes 2 et 3. Les nouvelles bases pour les règles d'association sont définies dans le Paragraphe 4. Les algorithmes permettant l'extraction efficace de ces bases sont présentés dans le Paragraphe 5. Les résultats des expérimentations, qui montrent l'efficacité des algorithmes et l'utilité de la génération de ces bases, sont présentés dans le Paragraphe 6. La conclusion et les perspectives de travaux ultérieurs font l'objet du Paragraphe 7.

2 Problème du temps d'extraction des règles d'association

Les temps de réponse de l'extraction des règles d'association dépendent principalement des temps d'extraction des itemsets fréquents car plusieurs balayages du contexte doivent être réalisés en comptant pour chaque itemset fréquent potentiel le nombre d'objets du contexte dans lesquels il est contenu. Le nombre d'itemsets à considérer ($2^{|A|}$) et la taille des jeux de données (contexte d'extraction) étant importants (de plusieurs dizaines de milliers à plusieurs millions d'objets et de plusieurs centaines à plusieurs milliers d'items) des algorithmes permettant de minimiser le nombre d'itemsets candidats (itemsets potentiellement fréquents) considérés et le nombre de balayages du contexte ont été proposés.

2.1 Algorithmes d'extraction des itemsets fréquents

Les algorithmes d'extraction des itemsets fréquents par niveaux considèrent un ensemble d'itemsets d'une taille donnée lors de chaque itération, c'est à dire un ensemble d'itemsets d'un « niveau » du treillis des itemsets. Ces algorithmes se basent sur les propriétés suivantes afin de limiter le nombre d'itemsets candidats considérés, en les générant à partir des itemsets fréquents de l'itération précédente : tous les sur-ensembles d'un itemset infréquents sont infréquents et tous les sous-ensembles d'un itemset fréquent sont fréquents [AS94, MTV94]. Parmi ceux-ci nous pouvons citer les algorithmes Apriori [AS94] et OCD [MTV94] qui réalisent un nombre de balayages du contexte égal à la taille des plus longs itemsets fréquents, l'algorithme Partition [SON95] qui autorise la parallélisation du processus d'extraction, et l'algorithme DIC [BMUT97] qui réduit le nombre de balayages du contexte en considérant les itemsets de plusieurs tailles différentes lors de chaque itération. Les algorithmes Partition et DIC entraînent un coût supplémentaire en temps CPU par rapport aux algorithmes Apriori et OCD dû à l'augmentation du nombre d'itemsets candidats testés.

2.2 Algorithmes d'extraction des itemsets fréquents maximaux

Ces algorithmes sont basés sur la propriété que les itemsets fréquents maximaux, c'est à dire les itemsets dont tous les sur-ensembles sont infréquents, forment une bordure au dessous de laquelle tous les itemsets sont fréquents. L'extraction des itemsets fréquents maximaux est réalisée par une exploration itérative du treillis des itemsets fréquents, en « avancant » de un niveaux du bas vers le haut et de un ou plusieurs niveaux du haut vers le bas lors de chaque itération. À partir des itemsets fréquents maximaux, tous les itemsets fréquents sont dérivés et leurs supports sont déterminés en réalisant un balayage du contexte. Quatre algorithmes basés sur cette approche ont été proposés, ce sont les algorithmes Pincer-Search [LK98], MaxClique et MaxEclat [ZPOL97], et Max-Miner [Bay98]. Ces algorithmes permettent de réduire le nombre d'itérations, et donc de diminuer le nombre de balayages du contexte et d'opérations CPU, réalisés.

2.3 Algorithmes d'extraction des itemsets fermés fréquents

Les itemsets fermés fréquents [Pas00, PBTL98] sont définis en utilisant la fermeture de la connexion de Galois d'une relation binaire finie [DP94, GW99]. Ces itemsets sont les itemsets fréquents qui sont fermés selon l'opérateur de fermeture γ de la connexion de Galois qui est la composition de l'application φ, qui associe à un ensemble $O \subseteq$ O les items communs à tous les objets $o \in O$, et de l'application ψ, qui associe à un itemset $l \subseteq$ I les objets en relation avec tous les items $i \in l$. L'opérateur $\gamma = \varphi \circ \psi$ associe à un itemset l l'ensemble maximal d'items communs à tous les objets contenant l, c'est-à-dire l'intersection de ces objets. Par exemple, dans le contexte D, l'itemset {BCE} est un itemset fermé car il est l'ensemble maximal d'items communs aux objets {2, 3, 5, 6}. L'itemset {BC} n'est pas un itemset fermé car il n'est pas un ensemble maximal d'items communs à certains objets : tous les objets contenant les items B and C (les objets 2, 3, 5 et 6) contiennent également l'item E. Dans la cas d'une base de données de ventes, cela signifie que les clients achètent *au plus* les articles B, C et E, et que tous les clients qui achètent les articles B et C achètent également l'article E. Les itemsets fermés fréquents, selon cet opérateur de fermeture, constituent un ensemble générateur non redondant minimal pour tous les itemsets fréquents et leurs supports. Tous les itemsets fréquents et leurs supports, et donc toutes les règles d'association ainsi que leurs supports et leurs confiances, peuvent donc être déduits efficacement, sans accéder au jeu de données, à partir des itemsets fermés fréquents et leurs supports. Cette propriété découle du fait que le support d'un itemset fréquent est égal au support de sa fermeture et que les itemsets fréquents maximaux sont des itemsets fermés fréquents maximaux [Pas00, PBTL98]. Les itemsets fermés fréquents forment un treillis dont la taille est bornée par la taille du treillis des itemsets fréquents $2^{|A|}$. Toutefois, en pratique, la taille de ce treillis est en moyenne bien inférieure à la taille du treillis des itemsets [GM94]. Le diagramme de Hasse représentant le treillis des itemsets fermés associé au contexte D est représenté dans la Figure 2.

Dans [Pas00, PBTL99b], les *générateurs* des itemsets fermés sont définis : les générateurs d'un itemset fermé f sont les itemsets minimaux g dont la fermeture est égale à f : $\gamma(g) = f$. Les générateurs de l'itemset fermé {BCE} sont les itemsets {BC} et {CE} car les itemsets {BE} et {C} sont fermés et la fermeture des itemsets {B} et {E} est {BE}. Les algorithmes Close [Pas00, PBTL99a] et A-Close [Pas00, PBTL99b] sont des algorithmes d'extraction des itemsets fermés fréquents par niveaux : ils considèrent un ensemble de générateurs candidats d'une taille donnée, et déterminent leurs supports et leurs fermetures en réalisant un balayage du contexte lors de chaque itération. Les fermetures (fréquentes) des générateurs fréquents sont les itemsets fermés fréquents extraits lors de l'itération. Les générateurs candidats sont construits en combinants les

générateurs fréquents extraits durant l'itération précédente. L'algorithme Close$^+$ [Pas00] permet d'identifier les itemsets fermés fréquents et leurs générateurs parmi les itemsets fréquents, sans accéder au jeu de données.

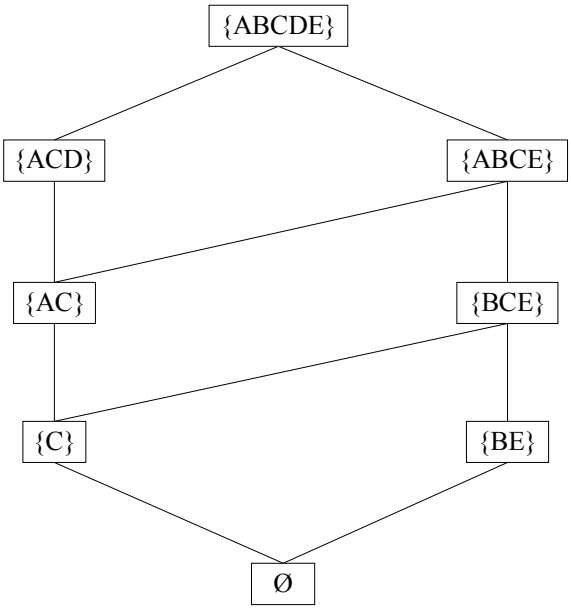

Figure 2 : Diagramme de Hasse représentant le treillis des itemsets fermés.

Algorithme Close d'extraction des itemsets fermés fréquents L'algorithme Close, proposé en 1998, est un algorithme itératif d'extraction des itemsets fermés fréquents qui parcourt l'ensemble des générateurs des itemsets fermés fréquents par niveaux. Durant chaque itération k de l'algorithme, un ensemble FFC_k de k-générateurs candidats est considéré. Chaque élément de cet ensemble est constitué de trois éléments : le k-générateur

candidat, sa fermeture, qui est un itemset fermé candidat, et leur support. À la fin de l'itération k, l'algorithme stocke un ensemble FF_k contenant les k-générateurs fréquents, leurs fermetures, qui sont des itemsets fermés fréquents, et leurs supports.

L'algorithme commence par initialiser l'ensemble FFC_1 des 1-générateurs avec la liste des 1-itemsets du contexte et exécute ensuite un ensemble d'itérations. Durant chaque itération k :

- La fermeture de tous les k-générateurs ainsi que leur support sont calculés. La détermination des fermetures des générateurs est basée sur la propriété que la fermeture d'un itemset l est égale à l'intersection de tous les objets du contexte contenant l dont le décompte fournit le support du générateur qui est identique au support de sa fermeture. Un seul balayage du contexte est donc nécessaire pour déterminer les fermetures et les supports de tous les k-générateurs.
- Tous les k-générateurs fréquents, dont le support est supérieur ou égal au seuil minimal de support *minsupport*, ainsi que leur fermeture et leur support sont insérés dans l'ensemble FF_k des itemsets fermés fréquents identifiés durant l'itération k.
- L'ensemble des (k+1)-générateurs candidats (utilisés durant l'itération suivante) est construit, en joignant les k-générateurs fréquents de l'ensemble FF_k comme suit (procédure Gen-Generator) :

1. Les (k+1)-générateurs candidats sont créés en joignant les k-générateurs de FF_k qui possèdent les mêmes k-1 premiers items. Les 3-générateurs {ABC} et {ABD} par exemple seront joints afin de créer le 4-générateur candidat {ABCD}.
2. Les (k+1)-générateurs candidats dont on sait qu'ils sont soit infréquents, soit non minimaux sont ensuite supprimés. Ces générateurs sont identifiés par l'absence d'un de leurs sous-ensembles de taille k parmi les k-générateurs fréquents de FF_k.
3. La troisième phase permet de supprimer parmi ces générateurs ceux dont la fermeture a déjà été calculée. Un tel générateur est identifié car il est inclus dans la fermeture d'un k-générateur fréquent de FF_k dont il est un sur-ensemble.

Les itérations cessent lorsque aucun nouveau générateur candidat ne peut être créé et l'algorihtme s'arrête alors. L'algorithme A-Close, développé afin d'améliorer l'efficacité de l'extraction dans le cas de données faiblement corrélées, ne calcule pas les fermetures des générateurs candidats durant les itérations, mais lors d'un ultime balayage réalisé après la fin de ces itérations.

Exemple 4 La Figure 3 représente l'exécution de l'algorithme Close au contexte d'extraction D pour un seuil minimal de support de 2/6.

	FFC_1				FF_1		
	Générateur	Fermé	Support		Générateur	Fermé	Support
Balayage	{A}	{AC}	3/6	Suppression	{A}	{AC}	3/6
de D	{B}	{BE}	5/6	des itemsets	{B}	{BE}	5/6
→	{C}	{C}	5/6	infréquents	{C}	{C}	5/6
	{D}	{ACD}	1/6	→			
	{E}	{BE}	5/6		{E}	{BE}	5/6

	FFC_2				FF_2		
	Générateur	Fermé	Support		Générateur	Fermé	Support
Balayage	{AB}	{ABCE}	2/6	Suppression	{AB}	{ABCE}	2/6
de D	{AE}	{ABCE}	2/6	des itemsets	{AE}	{ABCE}	2/6
→	{BC}	{BCE}	4/6	infréquents	{BC}	{BCE}	4/6
	{CE}	{BCE}	4/6	→	{CE}	{BCE}	4/6

Figure 3 : Extraction des itemsets fermés fréquents dans le contexte D *avec Close pour* minsupport = *2/6.*

L'ensemble FFC_1 est initialisé avec la liste des 1-itemsets du contexte D. La procédure Gen-Closure génère les fermetures des 1-générateurs, qui sont les itemsets fermés fréquents potentiels, et leurs supports dans FFC_1. Les groupes candidats de FFC_1 qui sont fréquents sont insérés dans l'ensemble FF_1. La première phase de la procédure Gen-Generator appliquée à l'ensemble FF_1 génère six nouveaux 2-générateurs candidats : {AB}, {AC}, {AE}, {BC}, {BE} et {CE} dans FFC_2. Les 2-générateurs {AC} et {BE} sont supprimés de FFC_2 par la troisième phase de la procédure Gen-Generator car nous avons {AC} $\subseteq \gamma(\{A\})$ et {BE} $\subseteq \gamma(\{B\})$. La procédure Gen-Closure calcule ensuite les fermetures et les supports des 2-générateurs restant dans FFC_2 et les ensembles FF_2 et FFC_2 sont identiques car tous les itemsets fermés de FFC_2 sont fréquents. L'application de la procédure Gen-Generator à l'ensemble FF_2 génère le 3-générateur {ABE} qui est supprimé car le 2-générateur {BE} n'appartient pas à FF_2 et l'algorithme s'arrête.

3 Problème de la pertinence et de l'utilité des règles d'association extraites

Le problème de la pertinence et de l'utilité est lié au nombre de règles d'association extraites qui est en général très important et à la présence d'une forte proportion de règles redondantes, c'est à dire de règles convoyant la même information, parmi celles-ci. Si le problème de la visualisation d'un nombre relativement important de règles peut être simplifié par l'utilisation de systèmes de visualisation tels que le système Rule Visualizer proposé par Klemettinen et al. [KMR+94], le problème de la suppression des règles d'association redondantes nécessite d'autres solutions. De plus, les règles d'association redondantes représentant pour certains type de données la majorité des règles extraites, leur suppression permet de réduire considérablement le nombre de règles à gérer lors de la visualisation.

Exemple 1 Afin d'illustrer le problème des règles d'association redondantes, neuf règles d'association extraites du jeu de données Mushrooms décrivant les caractéristiques de 8 416 champignons sont présentées ci-dessous. Ces neuf règles possèdent un support et une confiance identiques de 51% et 54% respectivement :

1) lamelles libres $\rightarrow$ comestible
2) lamelles libres $\rightarrow$ comestible, voile partiel
3) lamelles libres $\rightarrow$ comestible, voile blanc
4) lamelles libres $\rightarrow$ comestible, voile partiel, voile blanc
5) lamelles libres, voile partiel $\rightarrow$ comestible
6) lamelles libres, voile partiel $\rightarrow$ comestible, voile blanc
7) lamelles libres, voile blanc $\rightarrow$ comestible
8) lamelles libres, voile blanc $\rightarrow$ comestible, voile partiel
9) lamelles libres, voile partiel, voile blanc $\rightarrow$ comestible

Il est évident que les règles 1 à 3 et 5 à 9 sont redondantes par rapport à la règle 4 puisque, du point de vue de l'utilisateur, ces 8 règles n'apportent aucune information supplémentaire par rapport à la règle 4 qui est la plus générale. Afin d'améliorer la pertinence et l'utilité des règles extraites, il est souhaitable que seule cette dernière règle soit extraite et présentée à l'utilisateur.

Dans la suite, deux types de règles d'association sont distingués :

- Les règles d'association exactes dont la confiance est égale à 1 (100\%) qui sont vérifiées dans tous les objets du contexte.
- Les règles d'association approximatives dont la confiance est inférieure à 1 qui sont vérifiées dans une proportion d'objets du contexte égale à leur confiance.

La solution proposée dans cet article consiste à générer des bases, également appelées couvertures réduites, pour les règles d'association qui sont des ensembles de taille réduite ne contenant aucune règle redondante. Le but est de limiter l'extraction aux règles d'association les plus informatives, c'est-à-dire les plus générales et, éventuellement, dont les mesures de précision sont les plus élevées parmi toutes les règles valides, du point de vue de l'utilisateur.

3.1 État de l'art

Plusieurs méthodes permettant de réduire le nombre de règles d'association extraites, ou bien de sélectionner un sous-ensembles de règles, ont été proposées. Nous présentons un état de l'art non exhaustif de ces méthodes dans la suite.

Règles d'association généralisées Les règles d'association généralisées sont définies en utilisant une taxonomie des items du contexte : ce sont des règles d'association entre ensembles d'items pouvant appartenir à différents niveaux de la taxonomie. Par exemple, la règle généralisée « r_1 : lait $\rightarrow$ sucre » est appelée sur-règle des deux règles « r_2 : lait entier $\rightarrow$ sucre » et « r_3 : lait écrémé $\rightarrow$ sucre » car les items « lait entier » et « lait écrémé » sont des descendants de l'item « lait ». Dans [HF95, SA95], les auteurs proposent de générer seulement la sur-règle r_1 si son support est supérieur à la somme des supports de r_2 et r_3 et sa confiance est supérieure aux confiances de r_2 et r_3. Cette méthode nécessite la création d'une taxonomie des items et ne permet pas de supprimer les règles d'association redondantes. De plus, elle ne permet pas de réduire significativement le nombre de règles extraites dans le cas de données denses ou corrélées.

Utilisation d'autres mesures statistiques L'utilisation d'autres mesures statistiques que la confiance pour déterminer la précision des règles d'association a fait l'objet de nombreuses études [BMS97, PS91, SBM98]. Parmi ces mesures, nous retenons les deux mesures suivantes qui fournissent les résultats les plus intéressants. La mesure de *conviction* permet de mesurer pour chaque règle la déviation de la dépendance entre la probabilité d'occurrence de l'antécédent et la probabilité de non occurrence de la conséquence dans les objets. La mesure du χ^2 spécifie le degré de dépendance entre les items d'un itemset en comparant la distribution réelle de leur occurrence avec la distribution attendue de leur occurrence sous l'assomption d'une complète indépendance et d'une distribution normale. L'utilisation de ces mesures entraîne des problèmes de performances car le calcul de leurs valeurs nécessite des temps d'exécution importants.

Mesures de déviation Les mesures de déviation sont des mesures de distance entre règles d'association définies en fonction de leurs supports et leurs confiances. Ces mesures peuvent être utilisées afin d'identifier les règles d'association fortement semblables, caractérisées par une faible distance entre elles, et ensuite regrouper ou supprimer certaines de ces règles [BAG99, TKR+95]. Cette méthode entraîne une perte d'information et nécessite des traitements supplémentaires de comparaison des règles extraites deux à deux. Une autre utilisation consiste à identifier les règles d'association inattendues pour l'utilisateur qui apportent donc une connaissance importante car nouvelle [Hec96, PSM94, ST96]. Les connaissances de l'utilisateur sont représentées en utilisant des modèles probabilistes auxquels sont confrontées les règles d'association extraites. La déviation d'une règle correspond à la différence entre la valeur attendue pour la règle dans le modèle probabiliste et la valeur réelle pour la règle dans le contexte. Cette méthode nécessite la définition par l'utilisateur de ces connaissances, ce qui dans de nombreux cas se révèle très complexe, et les temps des calculs des mesures de déviation sont très importants car ils requièrent de très nombreuses opérations.

Templates Les templates [KMR+94] sont des expressions booléennes permettant de sélectionner un sous-ensemble de l'ensemble des règles d'assocation valides. Ce sous-ensemble est construit en conservant les règles d'association valides qui vérifient les critères spécifiés par les templates parmi cet ensemble. Un template spécifie des contraintes d'occurrence ou de non occurrence des items dans l'antécédent et la conséquence des règles. Cette méthode de traitement a posteriori des règles extraites ne permet pas de supprimer les règles redondantes, mais peut faciliter la visualisation de l'ensemble des règles extraites en visualisant les règles par groupes, chaque groupe correspondant à un ensemble de templates.

Contraintes sur les items Les contraintes sur les items [BAG99, NLHP98, SVA97] sont des expressions portant sur l'antécédent et la conséquence des règles, définies par l'utilisateur, qui spécifient la forme des règles d'association à extraire. Ces contraintes sont utilisées lors de la phase d'extraction des itemsets fréquents afin de limiter l'espace de recherche de cette phase aux itemsets permettant de générer les règles vérifiant les contraintes. Elles sont prises en compte

lors de la génération des itemsets candidats afin de considérer seulement les candidats permettant de générer des règles satisfaisant les contraintes. Toutefois, cette approche ne permet pas d'éliminer les règles redondantes et ne fournit qu'un résultat partiel, tous les items du contexte n'étant pas considérés.

Bases pour les règles d'implication La définition de bases pour les règles d'implication entre deux ensembles d'attributs binaires a été étudiée essentiellement dans les domaines de l'analyse de données et de l'analyse formelle de concepts. L'adaptation de la base de Duquenne-Guigues [DG86, GW99] pour les implications (globales), et la base de Luxenburger [Lux91] pour les implications partielles[2] dans le cadre de l'extraction de règles d'association exactes et approximatives est présentée dans [Pas00,PBTL99c]. Les bases obtenues sont des réductions de l'ensemble des règles d'association qui minimisent autant que faire se peut le nombre de règles générées, sans tenir compte du support des règles. Cela signifie que les antécédents et les conséquences de toutes les règles d'association peuvent être déduits de l'union de ces bases, mais pas leurs supports.

3.2 Contribution

Utilisant la sémantique pour le problème de l'extraction de règles d'association basée sur la fermeture de la connexion de Galois [Pas00], des bases pour les règles d'association sont caractérisées. Ces bases, qui sont la base générique pour les règles d'association exactes et la base informative pour les règles d'association approximatives, sont définies à partir des itemsets fermés fréquents et leurs générateurs. Ce sont des ensembles de taille réduite qui minimisent le nombre de règles d'association générées tout en maximisant la quantité et la qualité des informations convoyées. Elles permettent :

- La génération des règles d'association non redondantes les plus informatives seulement, c'est à dire des règles les plus utiles et pertinentes : celles qui ont un antécédent (partie gauche) minimal et une conséquence (partie droite) maximale. Les règles redondantes représentent pour certains jeux de données la majorité des règles extraites, plus particulièrement dans le cas de jeux de données denses ou corrélées pour lesquels le nombre total de règles valides est très important.
- La présentation à l'utilisateur d'un ensemble de règles couvrant tous les attributs de la base de données, c'est-à-dire contenant des règles dont l'union des conséquences est égale à l'union des conséquences de toutes les règles d'association valides dans le contexte. En effet, il est nécessaire de ne pas limiter la recherche à un seul sous-ensemble des attributs de la base de données car les règles « surprenantes » pour l'utilisateur constituent des informations utiles qu'il est nécessaire de considérer [Hec96, PSM94, ST96].
- L'extraction d'un ensemble de règles ne représentant aucune perte d'information, c'est à dire véhiculant toutes les informations convoyées par l'ensemble des règles d'association valides. Il est possible de déduire de manière efficace, sans accès au jeu de données, toutes les règles d'association valides ainsi que leurs supports et leurs confiances à partir des règles d'association de ces bases.

L'union de ces deux bases constitue donc un ensemble générateur minimal non redondant pour toutes les règles d'association valides, leurs supports et leurs confiances.

[2] Les règles d'implication partielles sont également appelées règles d'implication avec quelques contre-exemples ou règles d'implication valides dans un sous-contexte. Les règles d'implication globales sont des règles d'association exactes, et les règles d'implication partielles sont des règles d'association approximatives, avec une mesure de support associée.

4 Bases pour les règles d'association

Une règle d'association est une règle d'implication entre deux itemsets à laquelle sont associées la mesure de support, qui définit la portée de la règle, et la mesure de confiance, qui définit la précision de la règle dans le contexte d'extraction. Le support et la confiance indiquent l'utilité et la pertinence de la règle et doivent donc être pris en considération lors de la définition des règles d'association redondantes. Une règle d'association $r : l_1 \rightarrow l_2$ de support s et de confiance c est notée $r : l_1 \xrightarrow{s,c} l_2$. Une règle d'association $r \in E$ est redondante si la règle r peut être déduite ainsi que son support s et sa confiance c de l'ensemble $E \setminus r$.

Définition 1 (Règles d'association redondantes [Pas00])
Soit un ensemble E de règles d'association. Une règle $r : l_1 \xrightarrow{s,c} l_2$ \in E est redondante si et seulement si $E \setminus \{r : l_1 \xrightarrow{s,c} l_2\}$ \models $r : l_1 \xrightarrow{s,c} l_2$.

Comme nous l'avons vu dans l'Exemple 1, il est souhaitable que seules les *règles d'association non redondantes minimales*, qui sont les règles les plus utiles et les plus pertinentes, soient extraites et présentées à l'utilisateur. Une règle d'association est redondante si elle convoie la même information ou une information moins générale que l'information convoyée par une autre règle de même utilité et de même pertinence. Une règle d'association r est non redondante minimale s'il n'existe pas une autre règle d'association r' possédant le même support et la même confiance, dont l'antécédent est un sous-ensemble de l'antécédent de r et la conséquence est un sur-ensemble de la conséquence de r.

Définition 2 (Règles d'association non redondantes minimales [Pas00])
Soit l'ensemble AR des règles d'association extraites du contexte. Une règle d'association $r : l_1 \rightarrow l_2 \in AR$ est non redondante minimale s'il n'existe pas de règle d'association $r' : l'_1 \rightarrow l'_2 \in AR$ telle que $support(r) = support(r')$, $confiance(r) = confiance(r')$, $l'_1 \subseteq l_1$ et $l_2 \subseteq l'_2$.

À partir de cette définition, les règles d'association exactes non redondantes minimales sont définies dans le Paragraphe 4.1 et les règles d'association approximatives non redondantes minimales sont définies dans le Paragraphe 4.2. Ces règles constituent la *base générique* pour les règles d'association exactes et la *base informative* pour les règles d'association approximatives respectivement, et sont générées à partir des itemsets fermés fréquents et leurs générateurs.

4.1 Base générique pour les règles d'association exactes

Les règles d'association exactes, de la forme $r : l_1 \rightarrow (l_2 \setminus l_1)$, sont des règles entre deux itemsets fréquents l_1 et l_2 dont les fermetures sont identiques : $\gamma(l_1) = \gamma(l_2)$ [Pas00]. En effet, de $\gamma(l_1) = \gamma(l_2)$ nous déduisons que $l_1 \subset l_2$ et $support(l_1) = support(l_2)$, et donc $confiance(r) = 1$. Puisque l'itemset maximal parmi ces itemsets (qui possèdent le même support) est l'itemset $\gamma(l_2)$ (évident) , tous les sur-ensembles stricts de l_1 qui sont des sous-ensembles de $\gamma(l_2)$ possèdent le même support, et les règles entre deux de ces itemsets sont des règles exactes.

Soit l'ensemble $G_{\gamma(l2)}$ des générateurs de l'itemset fermé fréquent $\gamma(l_2)$. Par définition, les itemsets minimaux qui sont des sur-ensembles stricts de l_1 et des sous-ensembles de $\gamma(l_2)$ sont les générateurs $g \in G_{\gamma(l2)}$. Nous en concluons que les règles de la forme $g \rightarrow (\gamma(l_2) \setminus g)$ entre les générateurs $g \in G_{\gamma(l2)}$ et l'itemset fermé fréquent $\gamma(l2)$ sont les règles d'antécédents minimaux et de conséquences maximales parmi les règles entre les sur-ensembles stricts de l_1 et les sous-ensemble de $\gamma(l_2)$. La généralisation de cette propriété à l'ensemble des itemsets fermés fréquents définit la base générique constituée de toutes les règles d'association exactes non redondantes, selon la Définition 1, d'antécédents minimaux et de conséquences maximales.

Définition 3 (Base générique pour les règles d'association exactes [Pas00])
Soit l'ensemble *FF* des itemsets fermés fréquents extraits du contexte et pour chaque itemset fermé fréquent f l'ensemble G_f des générateurs de f. La base générique pour les règles d'association exactes est :

$$BG = \{r : g \rightarrow (f \setminus g) \mid f \in FF \wedge g \in G_f \wedge g \neq f\}.$$

La condition $g \neq f$ est nécessaire car les règles entre un générateur g d'un itemset fermé fréquent f tel que $g = f$ sont de la forme $g \rightarrow \varnothing$ et n'appartiennent pas à l'ensemble des règles d'association valides (règles non informatives).

Exemple 2 La base générique pour les règles d'association exactes extraite du contexte D pour *minsupport* = 2/6 est présentée dans la Table 2. Cette base ne représente aucune perte d'information : toutes les règles d'association exactes valides dans le contexte peuvent être déduites ainsi que leurs supports (et leurs confiances qui sont égales à 1) à partir des règles de la base générique [Pas00].

Générateur	Fermeture	Règle exacte	Support
{A}	{AC}	A → C	3/6
{B}	{BE}	B → E	5/6
{C}	{C}		
{E}	{BE}	E → B	5/6
{AB}	{ABCE}	AB → CE	2/6
{AE}	{ABCE}	AE → BC	2/6
{BC}	{BCE}	BC → E	4/6
{CE}	{BCE}	CE → B	4/6

Table 2 : Base générique extraite du contexte D.

4.2 Base informative pour les règles d'association approximatives

Les règles d'association approximatives, de la forme $l_1 \rightarrow (l_2 \setminus l_1)$, sont des règles entre deux itemsets fréquents l_1 et l_2 tel que la fermeture de l_1 est un sous-ensemble de la fermeture de l_2 : $\gamma(l_1) \subset \gamma(l_2)$ [Pas00]. Les règles d'association approximatives non redondantes d'antécédent l_1 minimal et de conséquence $(l_2 \setminus l_1)$ maximale sont déduites de cette caractérisation.

Soit l'itemset fermé fréquent f_1 qui est la fermeture de l_1 et le générateur unique g_1 de f_1 tels que $g_1 \subseteq l_1 \subseteq f_1$. Soit l'itemset fermé fréquent f_2 qui est la fermeture de l_2 et le générateur unique g_2 de f_2 tels que $g_2 \subseteq l_2 \subseteq f_2$. La règle $g_1 \rightarrow (f_2 \setminus g_1)$ entre le générateur g_1 et l'itemset fermé fréquent f_2 est la règle non redondante d'antécédent minimal et de conséquence maximale parmi les règles entre un itemset de l'intervalle $[g_1, f_1]$[3] et un itemset de l'intervalle $[g_2, f_2]$. En effet, le générateur g_1 est l'itemset minimal dont la fermeture est f_1, ce qui signifie que l'antécédent g_1 de cette règle est minimal, et la conséquence $(f_2 \setminus g_1)$ est maximale puisque f_2 est l'itemset maximal de l'intervalle $[g_2, f_2]$. La généralisation de cette propriété à l'ensemble des règles entre deux itemsets l_1 et l_2 définit la base informative qui est donc constituée de toutes les règles d'association approximatives non redondantes selon la Définition 1 d'antécédents minimaux et de conséquences maximales.

[3] L'intervalle $[g_1, f_1]$ contient tous les sur-ensembles de g_1 qui sont des sous-ensembles de f_1.

Définition 3 (Base informative pour les règles approximatives [Pas00])
Soit l'ensemble *FF* des itemsets fermés fréquents et l'ensemble *G* de leurs générateurs extraits du contexte. La base informative pour les règles d'association approximatives est :

$$BI = \{r : g \rightarrow (f \setminus g) \mid f \in FF \wedge g \in G \wedge \gamma(g) \subset f\}.$$

La base informative ne représente aucune perte d'information car toutes les règles d'association approximatives valides dans le contexte peuvent être déduites ainsi que leurs supports et leurs confiances à partir des règles qui la constituent [Pas00].

De la définition de la base informative il est possible de déduire la réduction transitive de celle-ci qui constitue elle-même une base pour les règles d'association approximatives. Les règles transitives de la base informative sont de la forme $r : g \rightarrow (f \setminus g)$ pour un itemset fermé fréquent f et un générateur fréquent g tels que $\gamma(g) \subset f$ et $\gamma(g)$ n'est pas un prédécesseur immédiat de f : $\exists f' \in FF$ tel que $\gamma(g) \subset f' \subset f$, noté $\gamma(g) \not\prec f$.

Définition 4 (Réduction transitive de la base informative [Pas00])
Soit l'ensemble *FF* des itemsets fermés fréquents et l'ensemble *G* de leurs générateurs extraits du contexte. La réduction transitive de la base informative pour les règles d'association approximatives est :

$$RI = \{r : g \rightarrow (f \setminus g) \mid f \in FF \wedge g \in G \wedge \gamma(g) \neq f\}.$$

Il est possible de déduire toutes les règles d'association de la base informative ainsi que leurs supports et leurs confiances, et donc toutes les règles approximatives valides, à partir des règles de la réduction transitive [Pas00]. Cette réduction permet de diminuer le nombre de règles extraites en conservant les règles dont la confiance est la plus élevée puisque les règles transitives possèdent par construction des confiances inférieures aux règles non transitives.

Exemple 3 La réduction transitive de la base informative extraite du contexte D pour *minsupport* = 2/6 et *minconfiance* = 2/6 est présentée dans la Table 3.

Générateur	Fermeture	Sur-ensemble fermé	Règle approximative	Support	Confiance
{A}	{AC}	{ABCE}	A → BCE	2/6	2/3
{B}	{BE}	{BCE}	B → CE	4/6	4/5
{B}	{BE}	{ABCE}			
{C}	{C}	{AC}	C → A	3/6	3/5
{C}	{C}	{BCE}	C → BE	4/6	4/5
{C}	{C}	{ABCE}			
{E}	{BE}	{BCE}	E → BC	4/6	4/5
{E}	{BE}	{ABCE}			
{AB}	{ABCE}				
{AE}	{ABCE}				
{BC}	{BCE}	{ABCE}	BC → AE	2/6	2/4
{CE}	{BCE}	{ABCE}	CE → AB	2/6	2/4

Table 3 : Réduction transitive de la base informative extraite du contexte D.

5 Algorithmes de génération des bases

5.1 Base générique pour les règles d'association exactes

Le pseudo-code de l'algorithme Gen-BG [Pas00] de construction de la base générique pour les règles d'association exactes à partir de l'ensemble des itemsets fermés fréquents et de leurs générateurs est présenté dans l'Algorithme 1. Les notations utilisées sont présentées dans la Table 4.

L'algorithme commence par initialiser l'ensemble *BG* avec l'ensemble vide (ligne 1). Chaque ensemble FF_k de *k*-groupes fréquents est ensuite examiné successivement (lignes 2 à 6). Pour chaque *k*-générateur $g \in FF_k$ de l'itemset fermé fréquent $\gamma(g)$ pour lequel g est différent de se fermeture $\gamma(g)$ (lignes 3 à 5), la règle $r : g \rightarrow (\gamma(g) \setminus g)$, dont le support est égal au support de g et $\gamma(g)$, est insérée dans *BG* (ligne 4). L'algorithme retourne finalement l'ensemble *BG* qui contient toutes les règles exactes informatives entre un générateur et sa fermeture (ligne 7).

FF_k	Ensemble de *k*-groupes fréquents des *k*-générateurs. Chaque élément de cet ensemble possède trois champs : *générateur*, *fermé* et *support*.
BG	Ensemble des règles d'association exactes de la base générique.

Table 4 : Notations utilisées dans l'algorithme Gen-BG.

Entrée Ensembles FF_k des *k*-groupes fréquents des *k*-générateurs;
Sortie : Ensemble *BG* des règles d'association exactes de la base de générique;

1) $BG \leftarrow \emptyset$;
2) **pour chaque** ensemble FF_k **faire**
3) **pour chaque** *k*-générateur $g \in FF_k$ **tel que** $g \neq \gamma(g)$ **faire**
4) $BG \leftarrow BG \cup \{(r : g \rightarrow (\gamma(g) \setminus g), \gamma(g).support)\}$;
5) **fin pour**
6) **fin pour**
7) **retourner** *BG*;

Algorithme 1 :Génération de la base générique avec Gen-BG.

5.2 Réduction transitive de la base informative pour les règles d'association approximatives

Le pseudo-code de l'algorithme Gen-RI [Pas00] de construction de la réduction transitive de la base informative pour les règles d'association approximatives à partir de l'ensemble des itemsets fermés fréquents et de leurs générateurs est présenté dans l'Algorithme 2. Les notations utilisées sont présentées dans la Table 5.

L'algorithme commence par initialiser l'ensemble *RI* avec l'ensemble vide (ligne 1). Chaque ensemble FF_k de *k*-groupes fréquents est ensuite examiné successivement dans l'ordre des valeurs de *k* croissantes (lignes 2 à 14). Pour chaque *k*-générateur $g \in FF_k$ de l'itemset fermé fréquent $\gamma(g)$ (lignes 3 à 18), l'ensemble $Succ_g$ des successeurs de la fermeture de $\gamma(g)$ est initialisé avec l'ensemble vide (ligne 4) et les ensembles S_j des *j*-itemsets fermés fréquents qui sont des sur-ensembles de $\gamma(g)$ pour $|\gamma(g)| < j \leq \mu$ sont constuits (lignes 5 à 7). Les ensembles S_j

sont ensuite considérés dans l'ordre croissant des valeurs de j (lignes 8 à 17). Pour chaque itemset $f \in S_j$ dont aucun successeur immédiat de $\gamma(g)$ dans $Succ_g$ n'est un sous-ensemble (ligne 10), f est inséré dans $Succ_g$ (ligne 11) et la confiance de la règle $r : g \rightarrow (f \setminus g)$ est calculée (ligne 12). Si la confiance de r est supérieure ou égale au seuil minimal de confiance *minconfiance*, la règle r est insérée dans RI (lignes 13 à 15). Lorsque tous les générateurs de taille inférieure à μ ont été considérés, l'algorithme retourne l'ensemble RI (ligne 20).

FF_k	Ensemble de k-groupes fréquents des k-générateurs. Chaque élément de cet ensemble possède trois champs : *générateur*, *fermé* et *support*.
$Succ_g$	Ensemble des itemsets fermés fréquents qui sont des successeurs immédiats de la fermeture du générateur g considéré.
RI	Ensemble des règles d'association approximatives de la réduction transitive de la base informative.

Table 5 : Notations utilisées dans l'algorithme Gen-RI.

Entrée : Ensembles FF_k des k-groupes fréquents des k-générateurs; seuil minimal de confiance *minconfiance*;

Sortie : Ensemble RI des règles d'association approximatives de la réduction transitive de la base de informative;

1) $RI \leftarrow \varnothing$;
2) **pour** ($k \leftarrow 1$; $k \leq \mu$-1; k++) **faire**
3) **pour chaque** k-générateur $g \in FF_k$ **faire**
4) $Succ_g \leftarrow \varnothing$;
5) **pour** ($j = |\gamma(g)|$; $j \leq \mu$; j++) **faire**
6) $S_j \leftarrow \{f \in FF \mid f \supset \gamma(g) \wedge |f| = j\}$;
7) **fin pour**
8) **pour** ($j = |\gamma(g)|$; $j \leq \mu$; j++) **faire**
9) **pour chaque** itemset fermé fréquent $f \in S_j$ **faire**
10) **si** ($\nexists s \in Succ_g \mid s \subset f$) **alors faire**
11) $Succ_g \leftarrow Succ_g \cup f$;
12) $r.confiance \leftarrow f.support$ / $g.support$;
13) **si** $r.confiance \geq minconfiance$
14) **alors** $RI \leftarrow RI \cup \{(r : g \rightarrow (f \setminus g), r.confiance, g.support\}$;
15) **finsi**
16) **fin pour**
17) **fin pour**
18) **fin pour**
19) **fin pour**
20) **retourner** RI;

Algorithme 2 :Génération de la réduction transitive de la base informative avec Gen-RI.

6 Résultats expérimentaux

Les algorithmes ont été implémentés en C++ et les expérimentations ont été réalisées sur un PC Pentium II possédant une vitesse d'horloge de 350 Mhz et 128 Megaoctets de mémoire, fonctionnant sous le système d'exploitation Linux. Un fichier de mémoire virtuelle d'une taille de 128 Megaoctets a été utilisé, portant à 256 Megaoctets l'espace mémoire total utilisable par les programmes. Nous avons utilisé les quatre jeux de données suivants lors de ces expérimentations :

- T20I6D100K[4], constitué de données synthétiques construites selon les propriétés des données de ventes, qui contient 100 000 objets d'une taille moyenne de 20 items pour une taille moyenne des itemsets fréquents maximaux potentiels de 6 items.
- Mushrooms[5] décrivant les caractéristiques de champignons. Il est constitué de 8 416 objets d'une taille moyenne de 23 attributs (23 items par objets et 127 items au total) correspondant aux caractéristiques des champignons.
- C20D10K et C73D10K[6] qui sont des échantillons du fichier Public Use Microdata Samples contenant des données du recensement du Kansas effectué en 1990. Ils sont constitués des 10 000 objets correspondant aux 10 000 premières personnes recensées, chaque objet contenant 20 attributs (20 items par objets et 386 items au total) pour C20D10K et 73 attributs (73 items par objets et 2 178 items au total) pour C73D10K.

  Les temps de réponse présentés dans le Paragraphe 6.1 correspondent aux temps d'extraction des itemsets fréquents et de génération de toutes les règles d'association valides pour Apriori et aux temps d'extraction des itemsets fermés fréquents et de génération de la base générique et de la réduction de la base informative pour les algorithmes Close et A-Close. Dans tous les cas, les temps de génération des règles d'association ou des bases pour les règles d'association sont très faibles (de l'ordre de quelques secondes) devant les temps d'extraction des itemsets fréquents ou des itemsets fermés fréquents (qui varient de quelques minutes à plusieurs heures).

6.1 Temps d'extraction des règles d'association

Les temps d'extraction des règles d'association avec Apriori et des bases pour les règles d'association avec Close et A-Close pour les jeux de données T20I6D100K, Mushrooms, C20D10K et C73D10K sont présentés dans la Figure 4. Le seuil *minconfiance* a été fixé à 0,5 % pour T20I6D100K, 30 % pour Mushrooms et C20D10K, et 80 % pour C73D10K.

Pour T20I6D100K, les temps de réponse de Apriori et A-Close sont identiques et inférieurs à ceux de Close pour les seuils de support de 2 % à 0,75 %. Les données de ventes étant éparses et faiblement corrélées, pour ces exécutions tous les itemsets fréquents sont des itemsets fermés fréquents (le treillis des itemsets fermés est identique au treillis des itemsets), ce qui correspond au pire des cas pour Close qui réalise plus d'opérations que Apriori et A-Close afin de déterminer les fermetures des générateurs. L'algorithme A-Close ne calcule pas les fermetures des générateurs dans cette situation et fournit donc des temps de réponse identiques à Apriori. Ceci n'est plus le cas pour les seuils de support de 0,5 % à 0,25 % pour lesquels certains itemsets (générateurs) ne sont pas fermés et où les temps de réponse de A-Close rejoignent ceux de Close.

---

[4] `http://www.almaden.ibm.com/cs/quest/syndata.html`

[5] `ftp://ftp.ics.uci.edu/~cmerz/mldb.tar.Z`

[6] `ftp://ftp2.cc.ukans.edu/pub/ippbr/census/pums/pums90ks.zip`

Malgré ces différences, les temps d'exécution des trois algorithmes, qui varient de quelques secondes à quelques minutes, restent acceptables dans tous les cas.

Pour Mushrooms, C20D10K et C73D10K les temps d'exécution de Close et A-Close sont très inférieurs à ceux de Apriori. Ces résultats s'expliquent par les caractéristiques des données de ces jeux qui sont corrélées et denses. En conséquence, le nombre d'itemsets fréquents est important et la proportion d'itemsets fermés fréquents parmi ces derniers est faible. L'espace de recherche des algorithmes Close et A-Close (le treillis des itemsets fermés) est de taille très inférieure à l'espace de recherche de l'algorithme Apriori (le treillis des itemsets). Contrairement aux temps de réponses obtenus pour T20I6D100K, les différences des temps de réponse entre Close et Apriori se mesurent en minutes ou en dizaines de minutes pour Mushrooms et C20D10K, et en dizaines de minutes ou en heures pour C73D10K. De plus, Apriori et A-Close n'ont pu être exécutés sur le jeu Mushrooms pour des seuils de support inférieurs à 7,5 % car ils dépassent alors la limite de 256 Mégaoctets correspondant à la quantité maximale de mémoire utilisable par les programmes. Pour la même raison, Apriori et A-Close n'ont pu être exécutés pour des seuils de support inférieurs à 65 % sur le jeu C73D10K.

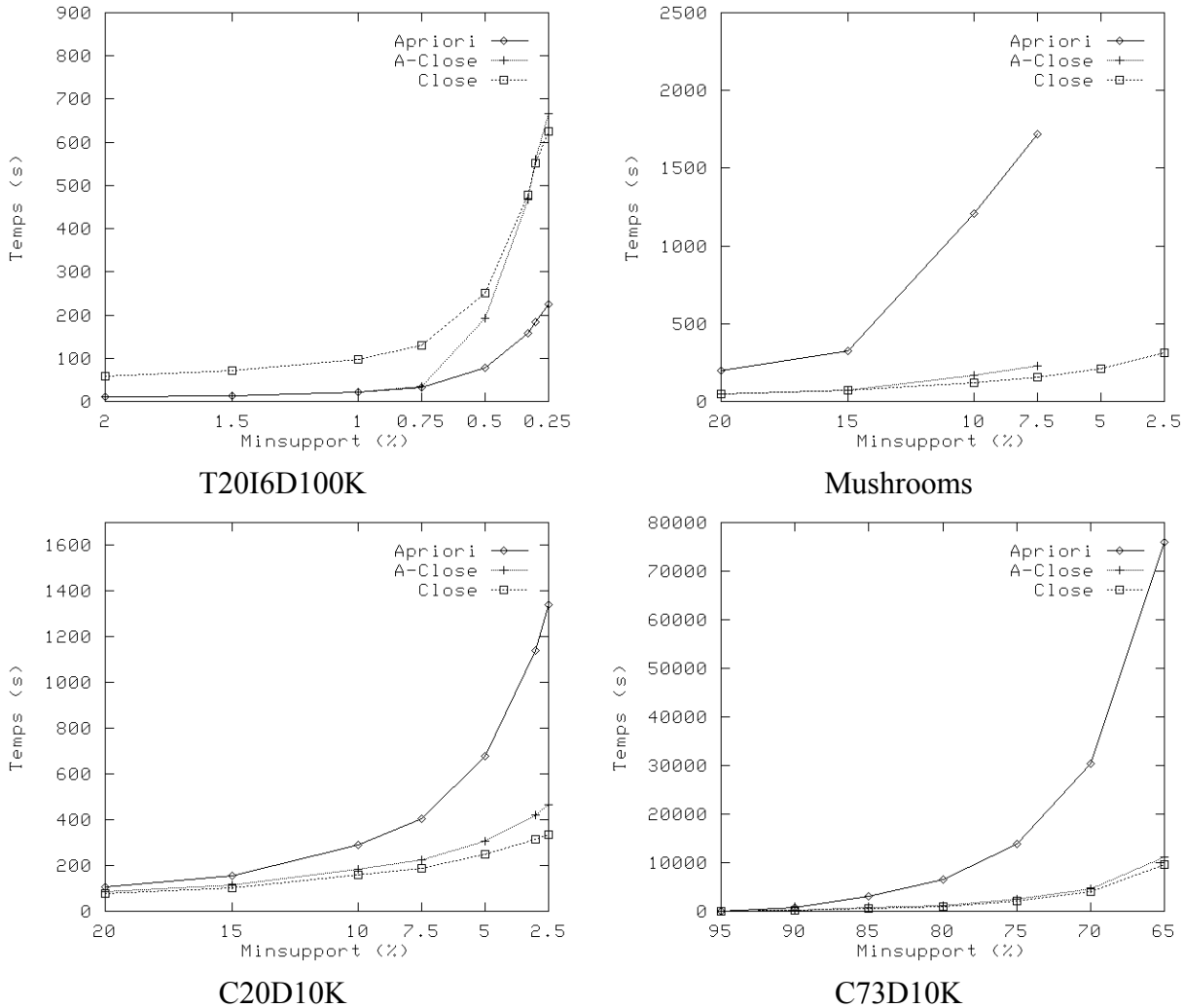

Figure 4 :Temps d'extraction des règles d'association.

6.2 Nombre de règles d'association exactes extraites

Le nombre total de règles d'association exactes valides et le nombre de règles dans la base de Duquenne-Guigues sont présentés dans la Table 4. Aucune règle d'association exacte n'est extraite du jeu T10I4D100K car pour ce seuil de support, tous les itemsets fréquents sont des itemsets fermés fréquents : ils possèdent tous des supports différents et sont donc eux-mêmes leur propre générateur unique. En conséquence, il n'existe aucune règle de la forme $l_1 \rightarrow (l_2 \setminus l_1)$ entre deux itemsets fréquents l_1 et l_2 dont les fermetures sont identiques : $\gamma(l_1) = \gamma(l_2)$ qui sont les règles d'association exactes valides dans le contexte.

Pour les trois autres jeux de données, constitués de données denses et corrélées, le nombre total de règles exactes valides varie de plus de 2 000 à plus de 52 000, ce qui est considérable et rend difficile la découverte de relations intéressantes dans ces ensembles. La base de Duquenne-Guigues représente une réduction importante du nombre de règles exactes extraites mais représente également une perte d'information importante. La base générique de taille supérieure, mais acceptable, ne représente aucune perte d'information et apporte donc une connaissance complète, pertinente et facilement utilisable du point de vue de l'utilisateur.

Jeu de données	Minsupport	Règles exactes	Base de Duquenne-Guigues	Base générique
T10I4D100K	0,5 %	0	0	0
Mushrooms	30 %	7 476	69	543
C20D10K	50 %	2 277	11	457
C73D10K	90 %	52 035	15	1 369

Table 5 : Nombre de règles d'association exactes extraites.

6.3 Nombre de règles d'association approximatives extraites

Le nombre total de règles d'association approximatives valides et le nombre de règles dans la réduction transitive de la base de Luxenburger et la réduction transitive de la base informative sont présentés dans la Table 6. Le nombre total de règles d'association approximatives valides est pour les quatre jeux de données très important puisqu'il varie de près de 20 000 règles pour T20I6D100K à plus de 2 000 000 de règles pour C73D10K. Il est donc indispensable de réduire l'ensemble des règles extraites afin de le rendre utilisable. Pour T20I6D100K, les deux bases sont de taille identique et représentent une division par un facteur de 5 environ du nombre de règles approximatives extraites. Pour Mushrooms, C20D10K et C73D10K, le nombre total de règles d'association approximatives valides est bien plus important que pour les données synthétiques car ces données étant denses et corrélées, le nombre d'itemsets fréquents est bien plus élevé et il en est donc de même pour le nombre de règles approximatives valides. La proportion d'itemsets fermés fréquents parmi les itemsets fréquents étant faible, ces deux bases pour les règles approximatives de tailles sensiblement équivalentes permettent de réduire considérablement (par un facteur variant de 40 à 500) le nombre de règles extraites.

L'examen des règles extraites dans chacune des deux bases par rapport a l'ensemble des règles valides a permis de vérifier qu'aucune de ces bases ne contient de règles redondantes. Considérant l'exemple présenté dans le Paragraphe 1 concernant les neuf règles approximatives extraites du jeu de données Mushrooms, seule la règle 4 est générée parmi ces neuf règles dans les bases. En effet, les itemsets {lamelles libres} et {lamelles libres, comestible, voile partiel,

voile blanc} sont deux itemsets fermés fréquents dont le premier est un prédécesseur immédiat du second, la règle 4 appartient donc à la réduction transitive de la base de Luxenburger. Ces deux itemsets sont les seuls itemsets fermés fréquents de l'intervalle [∅, {lamelles libres, comestible, voile partiel, voile blanc}]. De plus, l'itemset fermé fréquent {lamelles libres} étant lui-même son seul et unique générateur, la règle 4 appartient également à la réduction transitive de la base informative et est donc bien la règle non redondante d'antécédent minimal et de conséquence maximale parmi ces neuf règles.

Jeu de données	Minconfiance	Règles approximatives	Réduction base de Luxemburger	Réduction base informative
T10I4D100K	70 %	20 419	4 004	4 004
(0,5 %)	30 %	22 952	4 519	4 519
Mushrooms	70 %	37 671	968	1 221
(30 %)	30 %	71 412	1 260	1 578
C20D10K	70 %	89 601	1 948	1 957
(50 %)	30 %	116 791	1 948	1 957
C73D10K	90 %	2 053 896	4 089	5 718
(90 %)	80 %	2 053 936	4 089	5 718

Table 6 : Nombre de règles d'association approximatives extraites.

7 Conclusion et perpectives

Dans cet article, le problème de l'utilité et de la pertinence des règles d'association extraites est traité en utilisant la sémantique basée sur la fermeture de la connexion de Galois. Utilisant les itemsets fermés fréquents et leurs générateurs extraits par les algorithmes Close, A-Close et Close$^+$, la base générique pour les règles d'association exactes et la réduction transitive de la base informative pour les règles d'association approximatives sont définies. L'union de ces bases fournit un ensemble générateur non redondant pour toutes les règles d'association valides, leurs supports et leurs confiances. Elle est constituée des règles d'association non redondantes minimales (d'antécédent minimal et de conséquence maximale) et ne représente aucune perte d'information du point de vue de l'utilisateur : ce sont les règles d'association les plus utiles et les plus pertinentes. Toutes les informations convoyées par l'ensemble des règles d'association valides sont également convoyées par l'union de ces deux bases. Deux algorithmes de génération de la base générique et de la réduction transitive de la base informative à partir des itemsets fermés fréquents et leurs générateurs, sont également présentés. Ces bases présentent également un fort intérêt pour :

- La visualisation des règles extraites car le nombre réduit de règles dans ces bases ainsi que la distinction des règles exactes et des règles approximatives facilitent la présentation des règles à l'utilisateur. De plus, l'absence de règles redondantes dans les bases et la génération des règles non redondantes minimales seulement présentent un intérêt important du point de vue de l'utilisateur [KMR+94].
- L'identification des règles d'association non redondantes minimales parmi l'ensemble des règles d'association valides extraites, à partir de la Définition 2. Il est ainsi possible d'étendre une implémentation existante d'extraction des règles d'association ou bien d'intégrer cette méthode dans le système de visualisation afin de présenter les règles d'association non

redondantes minimales à l'utilisateur. Des algorithmes permettant d'identifier les itemsets fermés fréquents et leurs générateurs parmi les itemsets fréquents, et donc permettant de générer les règles d'association minimales non-redondantes à partir du résultat d'une implémentation existante, sont décrits dans [Pas00].

- L'analyse de données et l'analyse formelle de concepts car elles ne représentent aucune perte d'information par rapport à l'ensemble des règles d'implication valides et sont constituées des règles les plus utiles et les plus pertinentes. La Définition 2 des règles non redondantes minimales étant également valide dans le cadre des règles d'implications globales et partielles entre ensembles d'attributs binaires, les Définitions 3 de la base générique et 4 de la base informative sont également valides pour les règles d'implications globales et partielles respectivement.

Les perspectives de travaux ultérieurs concernent l'étude des diverses techniques d'implémentation et structures de données afin d'améliorer les processus d'extraction de connaissances dans les bases de données selon leurs propriétés et les différents types de données [Bas00]; le développement d'un algorithme efficace d'extraction des ensembles fréquents de littéraux spécifiant l'occurrence ou la non occurrence d'un item dans l'ensemble [Bas00]. Ces itemsets permettent la génération de règles d'association avec négation des occurrences des items et l'amélioration de la mesure de la précision des règles approximatives en utilisant des mesures statistiques autres que la confiance telles que la conviction.L'extraction des ensembles fréquents de tels littéraux pose des problèmes plus importants en termes de temps d'exécution et d'espace mémoire que l'extraction des itemsets fréquents, deux littéraux devant être considérés pour chaque item.

Références :

[AS94] : R. Agrawal, R. Srikant. *Fast algorithms for mining association rules in large databases*. Proc. VLDB conf., pp 478–499, September 1994.

[BAG99] : R. J. Bayardo, R. Agrawal, D. Gunopulos. *Constraint-based rule mining in large, dense databases*. Proc. ICDE conf., pp 188–197, March 1999.

[Bas00] : Y. Bastide. *Algorithmique de data mining : techniques d'implémentation et négation*. Thèse de doctorat, Université de Clermont-Ferrand II. En préparation.

[Bay98] : R. J. Bayardo. *Efficiently mining long patterns from databases*. Proc. SIGMOD conf., pp 85–93, June 1998.

[BMS97] : S. Brin, R. Motwani, C. Silverstein. *Beyond market baskets : Generalizing association rules to correlation*. Proc. SIGMOD conf., pp 265–276, May 1997.

[BMUT97] : S. Brin, R. Motwani, J. D. Ullman, S. Tsur. *Dynamic itemset counting and implication rules for market basket data*. Proc. SIGMOD conf., pp 255–264, May 1997.

[DP94] : B. A. Davey, H. A. Priestley. *Introduction to lattices and order*. Cambridge University Press, Fourth edition, 1994.

[DG86] : V. Duquenne, J.-L. Guigues. *Famille minimale d'implications informatives résultant d'un tableau de données binaires*. Mathématiques et Sciences Humaines, 24(95):5–18, 1986.

[FPSSU96] : U. M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, R. Uthurusamy, editors. *Advances in Knowledge Discovery and Data Mining*. AAAI Press, 1996.

[GM94] : R. Godin, R. Missaoui. *An incremental concept formation approach for learning from databases*. Theoretical Computer Science: Special issue on formal methods in databases and software engineering, 133(2):387−419, October 1994.

[GW99] : B. Ganter, R. Wille. *Formal Concept Analysis : Mathematical foundations*. Springer, 1999.

[Hec96] : D. Heckerman. *Bayesian networks for knowledge discovery*. Advances in Knowledge Discovery and Data Mining, pp 273−305. AAAI Press, 1996.

[HF95] : J. Han, Y. Fu. Discovery of multiple-level association rules from large databases. Proc. VLDB conf., pp 420−431, September 1995.

[KMR+94] : M. Klemettinen, H. Mannila, P. Ronkainen, H. Toivonen, A. I. Verkamo. *Finding interesting rules from large sets of discovered association rules*. Proc. CIKM conf., pp 401−407, November 1994.

[LK98] : D. Lin, Z. M. Kedem. *Pincer-Search : A new algorithm for discovering the maximum frequent set*. Proc. EBDT conf., LNCS 1377, pp 105−119, March 1998.

[Lux91] : M. Luxenburger. *Implications partielles dans un contexte*. Mathématiques, Informatique et Sciences Humaines, 29(113):35−55, 1991.

[MTV94] : H. Mannila, H. Toivonen, A. I. Verkamo. *Efficient algorithms for discovering association rules*. AAAI KDD workshop, pp 181−192, July 1994.

[NLHP98] : R. T. Ng, V. S. Lakshmanan, J. Han, A. Pang. *Exploratory mining and pruning optimizations of constrained association rules*. Proc. SIGMOD conf., pp 13−24, June 1998.

[Pas00] : N. Pasquier. *Data mining : algorithmes d'extraction et de réduction des règles d'association dans les bases de données*. Thèse de doctorat, Université de Clermont-Ferrand II, January 2000.

[PBTL98] : N. Pasquier, Y. Bastide, R. Taouil, L. Lakhal. *Pruning closed itemset lattices for association rules*. Proc. BDA conf., pp 177−196, October 1998.

[PBTL99a] : N. Pasquier, Y. Bastide, R. Taouil, L. Lakhal. *Efficient mining of association rules using closed itemset lattices*. Information Systems, 24(1):25−46, 1999.

[PBTL99b] : N. Pasquier, Y. Bastide, R. Taouil, L. Lakhal. *Discovering frequent closed itemsets for association rules*. Proc. ICDT conf., LNCS 1540, pp 398−416, January 1999.

[PBTL99c] : N. Pasquier, Y. Bastide, R. Taouil, L. Lakhal. *Closed set based discovery of small covers for association rules*. Proc. BDA conf., pp 361−381, October 1999.

[PS91] : G. Piatetsky-Shapiro. *Discovery, analysis and presentation of strong rules*. Knowledge Discovery in Databases, pp 229−248. AAAI Press, 1991.

[PSM94] : G. Piatetsky-Shapiro, C.J. Matheus. *The interestingness of deviations*. AAAI KDD workshop, pp 25−36, July 1994.

[SA95] : R. Srikant, R.Agrawal. *Mining generalized association rules*. Proc. VLDB conf., pp 407−419, September 1995.

[SBM98] : C. Silverstein, S. Brin, R. Motwani. *Beyond market baskets : Generalizing association rules to dependence rules*. Data Mining and Knowledge Discovery, 2(1):39–68, January 1998.

[SON95] : A. Savasere, E. Omiecinski, S. Navathe. *An efficient algorithm for mining association rules in large databases*. Proc. VLDB conf., pp 432–444, September 1995.

[ST96] : A. Silberschatz, A. Tuzhilin. *What makes patterns interesting in knowledge discovery systems*. IEEE Transactions on Knowledge and Data Engineering, 8(6):970–974, December 1996.

[SVA97] : R. Srikant, Q. Vu, R. Agrawal. *Mining association rules with item constraints*. Proc. KDD conf., pp 67–73, August 1997.

[TKR+95] : H. Toivonen, M. Klemettinen, P. Ronkainen, K. Hatonen, H. Mannila. *Pruning and grouping discovered association rules*. ECML MLnet workshop, pp 47–52, April 1995.

[ZPOL97] : M. J. Zaki, S. Parthasarathy, M. Ogihara, W. Li. *New algorithms for fast discovery of association rules*. Proc. KDD conf., pp 283–286, August 1997.