Extension de l'algorithme Apriori et des règles d'association au cas des données symboliques diagrammes et sélection des meilleures règles par la régression linéaire symbolique

Filipe Afonso

Lamsade et Ceremade-Unversité Paris 9 Dauphine/ Place du Maréchal de Lattre de Tassigny, 75775 Paris Cedex 16, France
afonso@ceremade.dauphine.fr

Résumé. Cet article présente l'extension de l'algorithme Apriori et des règles d'association au cas des données symboliques diagrammes. La méthode proposée nous permet de découvrir des règles au niveau des concepts. Notamment, plutôt que d'extraire des règles entre différents articles appartenant à des mêmes transactions enregistrées dans un magasin comme dans le cas classique, nous extrayons des règles d'association au niveau des clients afin d'étudier leurs comportements d'achat. Enfin, nous proposons une méthode de sélection des meilleures règles d'association selon la régression linéaire symbolique.

1 Introduction

L'algorithme Apriori défini par [Agrawal et Srikant, 1994] a pour but de découvrir des règles d'association à partir de données nominales issues du panier de la ménagère. Les règles extraites sont du type lait→beurre traduisant le fait que si du lait est présent dans le panier de la ménagère alors il y a aussi du beurre. A partir de là, des travaux ont voulu exploiter la complexité des données afin d'accélérer l'exécution de l'algorithme Apriori ou d'enrichir les règles d'association. Ainsi, [Wang et al., 2000] et [Cai et al., 1998] découvrent des règles pondérées par l'importance d'un même article présent dans le panier de la ménagère alors que [Srikant et al., 1997] et [Han et Fu, 1995] exploitent les relations de taxonomie dans les données. De plus, les règles d'association sont étendues aux données quantitatives et intervalles, notamment [Srikant et Agrawal, 1996] et [Miller et Yang, 1997]. [Kuok et al., 1998] font de même en exploitant les ensembles flous. Ainsi, cet article s'inscrit dans le prolongement de ces travaux. Nous traitons la découverte de règles d'association et l'extension de l'algorithme Apriori au cas des variables symboliques diagrammes afin d'extraire des règles non plus au niveau des individus mais au niveau des concepts. Ainsi, après avoir rappelé certaines définitions à propos des données symboliques, nous présentons l'algorithme Apriori et son extension aux cas des données diagrammes, c'est-à-dire lorsque chaque case de notre matrice de données contient plusieurs modalités pondérées telles que la somme des poids soit égale à un. Par la suite, nous étendons les règles d'association au cas de ces données et nous utilisons la régression linéaire afin d'étudier la qualité des règles d'association symboliques. Finalement, nous terminons par un exemple d'application où nous étudions les comportements d'achat dans des magasins non plus au niveau

des transactions comme dans le cas classique mais au niveau des clients. Pour chaque client, nous agrégeons tous les articles achetés grâce à un diagramme construit avec la proportion de chaque article par rapport aux achats totaux du client.

1.1 Données en entrée de l'analyse des données symboliques

L'intérêt principal de l'analyse des données symboliques (ADS) est de passer de l'étude des individus à l'étude des concepts. Par exemple, si les habitants d'un pays sont décrits par la région, le sexe et la classe d'âges alors nous avons la table 1 classique. Supposons que nous désirons étudier chaque région. Nous supprimons alors la première colonne de la table 1 et nous décrivons, table 2, chaque région par des diagrammes construits avec les catégories de chaque variable et non plus par une valeur unique. Ainsi, les habitants sont des individus de premier niveau et les régions sont des individus de deuxième niveau appelés concepts. L'ADS s'étend non seulement aux variables diagrammes mais aussi aux variables intervalles, multi-valuées, et histogrammes pour lesquelles les opérateurs numériques standards $\times, +, -$ ne peuvent être appliqués directement (voir [Bock et Diday, 2000]). Nous obtenons alors une matrice symbolique où chaque ligne définit la "description " d'une région et chaque colonne est associée à une variable symbolique. Afin de construire et d'étudier les concepts, le logiciel SODAS d'analyse de données symboliques a été développé. Ce logiciel est disponible à l'adresse http://www.ceremade.dauphine.fr/%7Etouati/sodas-pagegarde.htm

1.2 Concepts, objets symboliques et assertions

Un objet symbolique (OS) modélise des concepts. Un concept est généralement défini par un ensemble de propriétés appelé intension et un ensemble d'individus satisfaisant ces propriétés appelé extension (voir [Bock et Diday, 2000]).

Définition 1 Un OS est un triplet s=(a,R,d) où R est une relation entre descriptions, d est une description et "a", allant de Ω (ensemble des individus) dans L, dépend de R et d.

Nous avons deux sortes d'OS pour deux ensembles L différents. Les OS booléens sont tels que $[y(w)Rd] \in L = \{vrai, faux\}$. Exemple: a(w)=[couleur(w)$\subseteq${noir, bleu}] $\vee$ [poids(w)$\subseteq$[60,75]] = (vrai$\vee$faux) = vrai où $w \epsilon \Omega$, couleur et poids sont deux variables qui décrivent w. Les OS modaux sont tels que [y(w)Rd] $\in$ L=[0,1]. Nous n'utilisons pas ce type d'OS par la suite. Une assertion est alors un OS défini par $[d'Rd] = \wedge_{i=1,p}[d'_i R_i d_i]$, $p \geq 1$. Nous donnons, par exemple, l'assertion booléenne: a(w)=[âge(w)$\subseteq$ {20,30,35}] $\wedge$ $[CSP(w) \subseteq \{cadre, ouvrier\}]$ où âge et CSP sont deux variables qui décrivent w. Finalement, l'extension d'un OS est donné par $Ext(s) = \{w \in \Omega / a(w) = vrai\}$ dans le cas booléen.

2 Algorithme Apriori et règles d'association

Depuis [Agrawal et al., 1993], la recherche d'algorithmes capables d'extraire des règles d'association dans de grandes bases de données a été un thème très étudié. La découverte de règles d'association entre différents produits présents dans le panier de

N° résident	Région	Sexe	Age	N° résident	Région	Sexe	Age
10001	Picardie	H	[0,20]	15001	Alsace	F	[60,80]
10002	Picardie	H	[0,20]	15002	Alsace	H	[20,40]
10003	Picardie	F	[60,80]				

TAB. 1 – *Données classiques décrivant les habitants d'une région*

Région	sexe	Age
Picardie	(2/3) H, (1/3) F	(2/3) [0,20], (1/3) [60,80]
Alsace	(1/2) F, (1/2) H	(1/2) [60,80], (1/2) [20,40]

TAB. 2 – *Matrice de données symboliques où chaque case contient un diagramme*

la ménagère a été un exemple d'application particulièrement exploité. Par la suite, les articles du panier de la ménagère sont appelés items alors que les sous-ensembles d'items sont appelés itemsets. Une transaction est un sous-ensemble d'items enregistré à la caisse d'un supermarché. Ainsi, en entrée de ces algorithmes, nous avons un ensemble de n items $I = \{i_1,...,i_n\}$ et un ensemble de m transactions $T = \{t_1,...,t_m\}$ avec $t_i \in P(I) - \oslash$ (voir table 3). Une règle d'association est alors définie par deux itemsets X et Y tels que $X \rightarrow Y$ avec $X \subset I$, $Y \subset I$ et $X \cap Y = \oslash$. Dans [Agrawal et Srikant, 1994], les auteurs suggèrent l'algorithme Apriori. L'idée est de générer les règles d'association ayant un support *sup* et une confiance *conf* supérieurs à deux seuils minimum *minsup* et *minconf* respectivement où: $sup(X \rightarrow Y) = \frac{card(t \in T / X \cup Y \subseteq t)}{card(T)}$,$conf(X \rightarrow Y) = \frac{card(t \in T / X \cup Y \subseteq t)}{card(t \in T / X \subseteq t)} = \frac{sup(X \cup Y)}{sup(X)}$.

L'algorithme Apriori recherche les sous-ensembles ayant un support supérieur à *minsup* appelés itemsets fréquents. Cette recherche a tout de même une complexité forte, de l'ordre de 2^n, où n est le nombre d'items. En fait, dans la pratique, nous avons beaucoup moins de n items par transaction et par conséquent, la complexité réelle est bien moindre. De plus, grâce à la propriété 2 suivante et à la détermination d'un support minimum nous supprimons les itemsets non-fréquents avant la génération de plus grands itemsets.

Propriété 2 [Agrawal et al., 1993]: Tout itemset inclus dans un itemset fréquent est lui-même fréquent.

Nous rappelons les différentes étapes de l'algorithme Apriori:

1. Apriori recherche tous les 1-itemsets (itemsets composés d'un item) fréquents $L_{k=1}$. Les supports sont calculés avec un passage dans la base de données.
2. Tant que $L_k \neq \oslash$ (L_k: ensemble des k-itemsets, itemsets de k items, fréquents):
   (a) L'algorithme génère les k+1-itemsets "candidats" en faisant le produit cartésien entre les itemsets de L_k. L'ensemble des candidats C_{k+1} est généré. Grâce à la propriété 2, Apriori supprime tout itemset $I \in C_{k+1}$ tel qu'il existe un k-itemset non-fréquent $J \subset I$. Par exemple, avec (1,2), (1,3), (1,4), (2,3), (3,4) fréquents, l'algorithme génère (1,2,3), (1,2,4) et (1,3,4). (1,2,4) est alors supprimé car (2,4) est non fréquent.

Transaction	Client	X=items	Transaction	Client	X=items
t_1	1	v	t_7	2	v
t_2	1	v,p,c	t_8	3	v,p
t_3	1	v,p,c	t_9	3	v
t_4	1	v	t_{10}	4	p,c
t_5	2	v,p	t_{11}	4	p
t_6	2	v,p,c			

TAB. 3 – *Matrice de transactions pour l'algorithme Apriori classique*

(b) Pour tout $c \in C_{k+1}$, le support est calculé avec un passage dans la base.
(c) les k+1-itemsets fréquents de C_{k+1} sont ajoutés à L_{k+1}.

3 Algorithme Apriori étendu aux données diagrammes

Nous étendons l'algorithme Apriori au cas des données diagrammes. Concrètement, au lieu d'avoir une valeur unique par case dans notre matrice de données ou bien un ensemble d'items par transaction comme dans le cas classique, nous avons un diagramme dans chaque case, i.e. des valeurs multiples pondérées telles que la somme des poids soit égale à un. Cet "Apriori Diagramme" va nous permettre d'étudier des concepts. Par exemple, dans l'Apriori classique, les unités statistiques sont des transactions. Par opposition, avec notre méthode nous sommes capables d'étudier les clients plutôt que les transactions.

Ainsi, nous considérons la matrice classique, table 3, avec 11 transactions répertoriées dans un supermarché. Ces onze transactions proviennent de quatre clients différents. Dans ces transactions, nous nous intéressons aux associations entre 3 catégories d'items v = viande, p = poissons, c = pâtes et céréales. Pour appliquer l'analyse symbolique sur les concepts clients nous créons ces concepts (table 4). Pour chaque client, cette matrice agrège tous les items achetés sous forme d'un diagramme. Ainsi, chaque diagramme est construit avec la proportion de chaque article par rapport aux achats totaux du client.

Remarque: Nous pouvons si nous le désirons prendre en compte la quantité de chaque item dans une même transaction alors que dans le cas classique on regarde simplement si un item a été acheté ou pas. Par exemple, si nous avons, pour un même client, deux transactions (a,a,b,c) et (a,b,b,b) où a, b, c sont trois items alors le cas classique considère deux transactions (a,b,c) et (a,b) alors que la méthode symbolique considère le diagramme {3/8a, 4/8b, 1/8c}.

Par la suite, nous alons donc utiliser l'exemple table 4 avec quatre concepts et une seule variable diagramme X. Cependant, l'algorithme que nous présentons se généralise en présence de plusieurs variables. Nous explicitons les étapes nécessaires à l'obtention des règles table 5 où par exemple la règle 1, $1/3 < P_v \leq 2/3 \rightarrow 0 < P_p \leq 1/3$ se lit: "Si pour le concept client, la proportion d'achats de viandes est comprise entre 1 produit sur 3 et 2 produits sur 3 alors la proportion d'achats de poissons est strictement positive et inférieure à 1 produit sur 3".

Concepts=Clients	X=items	Concepts=Clients	X=items
1	1/2v, 1/4p, 1/4c	3	2/3v, 1/3p
2	1/2v, 1/3p, 1/6c	4	2/3p,1/3c

TAB. 4 – *Matrice de données symboliques composée d'une variable diagramme*

N°	Règles	Sup %	Conf %	CD %
1	$1/3 < \mathrm{P}_v \leq 2/3 \rightarrow 0 < \mathrm{P}_p \leq 1/3$	75	100	75
2	$0 < \mathrm{P}_p \leq 1/3 \rightarrow 1/3 < \mathrm{P}_v \leq 2/3$	75	100	75
3	$0 < P_c \leq 1/3 \rightarrow 0 < \mathrm{P}_p \leq 2/3$	75	100	60
4	$0 < \mathrm{P}_p \leq 2/3 \rightarrow 1/3 < \mathrm{P}_v \leq 2/3$	75	75	56
5	$0 < \mathrm{P}_p \leq 2/3 \rightarrow 0 < \mathrm{P}_c \leq 1/3$	75	75	56

TAB. 5 – *Règles d'association symboliques*

3.1 Principe de la méthode

Pour étendre l'algorithme Apriori, nous "discrétisons" les fréquences de chaque catégorie des diagrammes. Nous découpons en intervalles les fréquences $P_{Xi,c}$ pour chaque catégorie c de chaque variable X_i. Ainsi, nous regardons les supports des intervalles de fréquences $0 < P_{Xi,c} \leq 1/h$,$1/h < P_{Xi,c} \leq 2/h$,$2/h < P_{Xi,c} \leq 3/h$,...,$(h-1)/h < P_{Xi,c} \leq 1$ où h détermine la précision du découpage. Dans un deuxième temps, nous faisons l'union 2 à 2 des intervalles de poids contigus ayant des supports strictement positifs $0 < P_{Xi,c} \leq 2/h$,$1/h < P_{Xi,c} \leq 3/h$,...,$(h-2) < P_{Xi,c} \leq 1$. Nous répétons l'opération jusqu'à obtenir un unique intervalle $0 < P_{Xi,c} \leq 1$. Par la suite, toute cette taxonomie d'intervalles sera considérée. Ainsi, nous travaillons avec des objets symboliques (OS) booléens et non plus avec des itemsets où les intervalles de fréquences sont les propriétés des OS qui ont donc pour intensions a(w)=$[\frac{a}{h} < P_{Xi,c}(w) \leq \frac{b}{h}]$ (a=0..h-1, b=1..h, a<b). Finalement, un k-OS est une assertion booléenne définie à partir de k propriétés. Par exemple, si a et a' sont deux catégories de deux variables diagrammes X et X' avec P_{X_a} et $P_{X'_{a'}}$ leurs fréquences respectives alors $[\frac{1}{3} < P_{X_a} \leq \frac{2}{3}] \wedge [0 < P_{X'_{a'}} \leq \frac{1}{3}]$ sera un 2-OS. Ces k-OS ne seront pas totalement traités comme des catégories de l'algorithme classique. Il ne faut pas croiser des intervalles de même catégorie et nous devons à chaque fois utiliser les plus petits intervalles de fréquences possibles pour un même support.

3.2 Choix de la précision du découpage h

L'utilisateur peut choisir une valeur de h en fonction du nombre de modalités des variables à étudier et de son besoin de résultats plus ou moins précis. Bien évidemment, plus la précision est grande plus le nombre de 1-OS sera grand par rapport au nombre d'items dans le cas classique. En contrepartie, la transformation de la matrice des données classiques en données symboliques aura réduit le nombre d'individus à étudier. Dans notre exemple, table 4, nous pouvons utiliser une précision h égale au nombre

moyen (h=2.5) ou au nombre maximum (h=3) de catégories par concept dans nos données ou toute autre précision jugée conforme au besoin de l'étude.

3.3 Définitions du support, de la confiance, de la "confiance diagramme" (CD) dans le cas de données diagrammes.

Soient Ω un ensemble d'individus (concepts), X et Y deux OS ayant pour intensions $a_x(w) = \bigwedge_{i,u}[\frac{a_{i,u}}{h} < P_{Xi,u}(w) \leq \frac{b_{i,u}}{h}]$ et $a_y(w) = \bigwedge_{j,v}[\frac{c_{j,v}}{h} < P_{Yj,v}(w) \leq \frac{d_{j,v}}{h}]$ avec $\forall i,u,j,v\ X_{i,u} \neq Y_{j,v}$ où $P_{x_{i,u}}$ ($P_{y_{j,v}}$) est la fréquence de la catégorie u (v) de la variable diagramme X_i (Y_j), $\frac{a_{i,u}}{h}$ et $\frac{b_{i,u}}{h}$ ($\frac{c_{j,v}}{h}$ et $\frac{d_{j,v}}{h}$) les bornes des intervalles de fréquences.

Définitions 3

A) Support. $Sup(X \rightarrow Y) = \frac{card(ext(X \wedge Y) = \{w \in \Omega / a_x(w) = vrai, a_y(w) = vrai\}}{card(\Omega)}$

B) Confiance. $Conf(X \rightarrow Y) = \frac{card(ext(X \wedge Y) = \{w \in \Omega / a_x(w) = vrai, a_y(w) = vrai\}}{card(ext(X) = \{w \in \Omega / a_x(w) = vrai\}} = \frac{sup(X \rightarrow Y)}{sup(X)}$

C) CD. De plus, dans le cas de variables diagrammes, il est intéressant de définir un nouvel indicateur de qualité (confiance diagramme ou CD) pénalisant les règles ayant les plus grands intervalles de fréquences et donc la plus grande imprécision en conclusion: $CD(X \rightarrow Y) = conf(X \rightarrow Y)/(1 + \frac{\sum_{j,v}(d_{j,v} - c_{j,v})}{n_v \times h})$ où n_v est le nombre de propriétés en conclusion.

Ce coefficient CD est tel que : $\frac{1}{2}$Conf(X→Y)≤CD(X→Y)≤ $\frac{h}{h+1}$Conf(X→ Y). Nous définissons alors un CD minimum *minCD* pour la génération des règles.

3.4 Algorithme Apriori Diagramme

Dans ce paragraphe, nous détaillons les différentes étapes de l'algorithme "Apriori diagramme" à l'aide de l'exemple table 4. Nous donnons alors une précision h=3 (= nombre maximum de catégories pour un concept), un support miminum $minsup$ = 35% (i.e. 2 unités):

1. Discrétiser les fréquences de chaque catégorie (voir section 3.1). Nous donnons aux intervalles de fréquences les codes 1, 2, 3, 4... afin de faciliter l'écriture.
   Pour la matrice, table 4, nous considérons les poids P_v, P_p et P_c des catégories v, p et c. Ces poids sont discrétisés table 6 colonnes C_1 (OS 1 à 9).
2. Calculer les supports des intervalles de poids précédents avec un passage dans la matrice des données. Nous faisons alors l'union 2 à 2 des intervalles contigus de supports strictement positifs (voir section 3.1). Nous répétons les unions 2 à 2 de nos nouveaux intervalles jusqu'à obtenir un unique intervalle $0 < P_{Xi,c} \leq 1$. Les supports de ces intervalles sont calculés sans passage dans la matrice des données car si A et B sont des intervalles contigus alors Sup(A∪B) = Sup(A)+Sup(B). Nous ajoutons à $L_{k=1}$ les 1-OS de support supérieur au seuil *minsup*.
   Dans notre exemple, les supports des intervalles du point 1. sont calculés (table 6 colonne sup). Nous remarquons que $0 < P_p \leq 1/3$ et $1/3 < P_p \leq 2/3$ ont un support supérieur à 0. Par conséquent, $0 < P_p \leq 2/3$ devient candidat (OS numéro 10) et il est fréquent car son support est égal à la somme des supports

OS	C_1	Sup	OS	C_1	Sup	OS	C_2	Sup	OS	C_3	Sup
1	$0<P_v\leq\frac{1}{3}$	0	6	$\frac{2}{3}<P_p\leq 1$	0	11	2∧4	3	16	2∧4∧7	2
2	$\frac{1}{3}<P_v\leq\frac{2}{3}$	3	7	$0<P_c\leq\frac{1}{3}$	3	12	2∧7	2			
3	$\frac{2}{3}<P_v\leq 1$	0	8	$\frac{1}{3}<P_c\leq\frac{2}{3}$	0	13	2∧10	3			
4	$0<P_p\leq\frac{1}{3}$	3	9	$\frac{2}{3}<P_c\leq 1$	0	14	4∧7	2			
5	$\frac{1}{3}<P_p\leq\frac{2}{3}$	1	10	$0<P_p\leq\frac{2}{3}$	4	15	7∧10	3			

TAB. 6 – *k-Objets Symboliques fréquents*

des intervalles précédents, soit 3+1=4. Finalement, nous ajoutons à l'ensemble L_1 les intervalles fréquents 2, 4, 7 et 10.

3. Faire tant que l'ensemble des k-OS (assertion définie avec la conjonction de k intervalles de fréquences) fréquents $L_k \neq \emptyset$ $(k \geq 1)$:

   (a) Générer les k+1-OS candidats en calculant le produit cartésien entre les k-OS de L_k. Dans le cas des diagrammes, nous générons les k+1-OS entre intervalles de catégories différentes (et "non marqués" voir point (c)). Ainsi, l'ensemble des candidats C_{k+1} est généré. Du fait de la propriété 2, nous supprimons de C_{k+1} tout k+1-OS I tel qu'il existe un k-OS $J \subset I$ n'appartenant pas à L_k.

   Pour notre exemple, nous calculons le produit Cartésien entre les OS de L_1 pour des intervalles de catégories différentes. Ainsi, l'algorithme génère les candidats C_2: (2∧4), (2∧7), (2∧10), (4∧7) et (7∧10) (voir table 6, OS=11 à 15). (4∧10) n'est pas généré car 4 et 10 sont des intervalles de la même catégorie.

   (b) Pour tout $c \in C_{k+1}$, calculer le support avec un passage dans la matrice de données. Tout k+1-OS $I \in C_{k+1}$ fréquent est ajouté à L_{k+1}.

   (2∧4), (2∧7), (2∧10), (4∧7) et (7∧10) sont fréquents.

   (c) Marquer tout k+1-OS $I \in L_{k+1}/\exists J \in L_{k+1}$ avec $J \subset I$ et $sup(I) = sup(J)$. Il s'agit de k+1-OS définis avec les mêmes catégories mais avec des intervalles de poids différents et nous conservons uniquement les plus petits intervalles pour un même support. Nous les marquons au lieu de les supprimer car ces k+1-OS ne sont pas utilisés pour la génération de k+2-OS mais ils sont utilisés pour la génération de règles.

   Dans notre exemple, (2∧10) ne sera pas utilisé pour la génération de 3-OS car (2∧10) ⊃ (2∧4) et sup(2∧10) = sup(2∧4). Par contre, les règles 2→10 et 10→2 seront considérées.

   (d) Générer les règles avec un CD supérieur à *minCD*, voir section 4.

Finalement, à l'itération suivante, le 3-OS fréquent (2∧4∧7) (voir table 6, OS=16) est généré à partir des 2-OS (2∧4) et (2∧7) et l'algorithme s'arrête. Nous remarquons que (4∧7) et (7∧10) ne génèrent pas (4∧7∧10) car (4∧10) n'est pas fréquent.

4 Règles d'association symboliques et étude de ces règles à l'aide de la régression linéaire

4.1 Base de règles symboliques

Dans le cas classique, pour tous les itemsets fréquents X et $Y \subset X$, nous générons la règle $Y \rightarrow X - Y$. L'algorithme classique génère uniquement les règles ayant une confiance supérieure à un seuil minimum *minconf*. Dans le cas diagramme, nous générons les règles ayant un CD supérieur à *minCD*. Aussi, nous générons des règles avec une unique propriété en conclusion: $\bigwedge_{i,u}[\frac{a_{i,u}}{h} < P_{Xi,u} \leq \frac{b_{i,u}}{h}] \rightarrow [\frac{c_{j,v}}{h} < P_{Yj,v} \leq \frac{d_{j,v}}{h}]$ où $\forall i,u\ X_{i,u} \neq Y_{j,v}$.

Nous générons des règles sans prémisses redondantes, soient $X \rightarrow Y$ telles que:

1. $\forall\ Z \subset X$ avec $Z \rightarrow Y$, conf($X \rightarrow Y$)>conf($Z \rightarrow Y$) ou sup(X)>sup(Z).
   Nous remarquons qu'il peut exister des OS $Z \subset X$ avec $sup(X) > sup(Z)$ car nous avons plusieurs intervalles de fréquences pour une même catégorie. Dans l'exemple table 6, nous avons $4 \subset 10$ et $sup(4) < sup(10)$;
2. $\forall W$ 1-OS, $W \subset Y$, $conf(X \rightarrow W) < conf(X \rightarrow Y)$. Lorsque W et Y sont deux 1-OS, $W \subset Y$ signifie que W et Y sont deux intervalles de la même catégorie. Nous cherchons alors le plus petit intervalle en conclusion pour une même confiance.

Ainsi, la base de règles R est alors définie comme suit:

Définition 4 R={(r,sup(r),conf(r),CD(r)): r=$X - Y \rightarrow Y$ / X k-OS ($k > 1$), Y 1-OS, $Y \subset X$; ($\forall Z \subset X - Y$, $conf(Z \rightarrow Y) < conf(r)$ ou $sup(Z) < sup(X - Y)$); ($\forall W$ 1-OS, $W \subset Y$, $conf(X - Y \rightarrow W) < conf(r)$); $sup(r) \geq minsup$; $CD(r) \geq minCD$}.

Nous générons les k-règles (règles avec k-1 prémisses et une conclusion, $k \geq 2$) immédiatement après la génération des k-OS (voir l'algorithme précédent section 3.4):

1. Pour chaque k-OS fréquent X, calculer l'indicateur CD de toute règle $X - Y \rightarrow Y$ où Y 1-OS, $Y \subset X$;
2. Pour chaque règle ayant un CD supérieur à *minCD*:
   (a) Vérifier qu'il n'existe pas une J-règle ($j < k$), $Z \rightarrow Y$ avec $Z \subset X - Y$ et conf($Z \rightarrow Y$)=conf($X - Y \rightarrow Y$) (sup(Z)≥sup(X-Y) toujours vrai grâce au point 3(c) de l'algorithme Apriori diagramme),
   (b) Vérifier qu'il n'existe pas une k-règle $X - Y \rightarrow W$ avec $W \subset Y$ et $conf(X - Y \rightarrow W) = conf(X - Y \rightarrow Y)$,
   (c) Si (a) et (b) sont vérifiées: ajouter $X - Y \rightarrow Y$ à la base de règles R.

Dans notre exemple, en choisissant un *minCD* égal à 55%, nous obtenons les cinq 2-règles (une prémisse et une conclusion) de la table 5. La règle $2 \rightarrow 10$ n'est pas générée car conf($2 \rightarrow 10$)=conf($2 \rightarrow 4$) et $4 \subset 10$. De plus, malgré la découverte d'un 3-OS fréquent (2∧4∧7) (table 6, OS=16), l'algorithme ne génère pas de 3-règles (2 prémisses, 1 conclusion) car aucune 3-règle n'améliore la confiance par rapport aux 2-règles: conf(4∧7→2) = conf(4→2), conf(2∧7→4) = conf(2→4), conf(2∧4→7) = conf(4→7).

	X_1	X_2	X_3		X_1	X_2	X_3
individus de la régression	P_v	P_p	P_c	individus	P_v	P_p	P_c
1	1/2	1/4	1/4	3	2/3	1/3	0
2	1/2	1/3	1/6	4	0	2/3	1/3

TAB. 7 – *De la matrice diagramme vers une matrice classique*

4.2 Discrimination des règles d'association symboliques à l'aide de la régression linéaire symbolique

Les règles symboliques obtenues comprennent de la variation puisqu'elles sont définies avec des intervalles de fréquences en prémisses et en conclusion. Ceci engendre de l'imprécision dans nos règles. Par exemple, si nous regardons la première règle ($1/3 < P_v \leq 2/3 \rightarrow 0 < P_p \leq 1/3$) table 5, nous ne pouvons pas savoir si lorsque P_v est proche de 1/3 alors P_p est plutôt proche de 0, ou proche de 1/3 ou bien varie dans l'intervalle]0,1/3] sans distinction. Par la suite, pour étudier ces variations, nous utilisons la régression linéaire symbolique.

[Afonso et al., 2004], [Afonso et al., 2003] et [Billard et Diday, 2002] étendent la régression linéaire aux cas des données symboliques et notamment au cas des variables à valeurs diagrammes. Nous expliquons brièvement les points de la régression linéaire symbolique importants pour notre étude. Pour plus d'informations, le lecteur pourra se référer aux articles cités ci-dessus. En fait, pour le cas des données diagrammes, nous faisons une régression classique en considérant les catégories de nos diagrammes comme les variables de la régression. Par exemple, nous pouvons partir de la variable X de la table 4. Il faut transformer cette matrice en matrice à valeur unique par case afin de pouvoir faire la régression. Ainsi, nous construisons une matrice où les poids de chaque catégorie de la variable (ou des variables) deviennent des variables classiques. Ceci nous donne la table 7 où les poids P_v, P_p et P_c de chaque catégorie deviennent 3 variables classiques prenant, par exemple, pour valeurs 1/2, 1/4 et 1/4 respectivement pour le premier concept. Cette méthode est étendue au cas de plusieurs variables diagrammes auxquelles nous pouvons adjoindre d'autres variables classiques et symboliques. Nous pouvons alors choisir les variables explicatives et la variable dépendante et faire une régression normalement. Il faut cependant veiller à ne pas faire la régression avec toutes les catégories d'une même variable diagramme, car ces catégories sont linéairement dépendantes et par conséquent la matrice ne serait pas inversible.

Ainsi, en addition aux indicateurs définis section 3.3, nous pouvons discriminer les règles symboliques de la forme $\bigwedge_{i,u}[\frac{a_{i,u}}{h} < P_{Xi,u} \leq \frac{b_{i,u}}{h}] \rightarrow [\frac{c_{j,v}}{h} < P_{Yj,v} \leq \frac{d_{j,v}}{h}]$ à l'aide de la régression linéaire symbolique. En effet, pour des règles avec une seule propriété en conclusion (une seule catégorie), nous calculons la régression des fréquences en prémisse sur la fréquence en conclusion en ne conservant uniquement que les individus dans l'extension de l'OS. Nous obtenons alors une équation linéaire:

$$P_{Y_{j,v}} = \beta_0 + \sum_{i,u} \beta_{i,u} P_{X_{i,u}} + \varepsilon$$

Dans la boule:

$$(\bigwedge_{i,u}[\frac{a_{i,u}}{h} < P_{Xi,u} \leq \frac{b_{i,u}}{h}]) \bigwedge ([\frac{c_{j,v}}{h} < P_{Yj,v} \leq \frac{d_{j,v}}{h}])$$

Règle	Equation	R^2	F-test	Règle	Equation	R^2	F-test
1	P_p=0.16+0.25P_v	0.25	0.33	4	P_v=0.25+P_p	0.25	0.33
2	P_v=0.25+P_p	0.25	0.33	5	P_c=0.13+0.3P_p	0.57	1.33
3	P_p=-0.1+2P_c	0.57	1.33				

TAB. 8 – *Etude des règles d'association symboliques à l'aide de la régression symbolique*

où ε désigne le résidu de la régression, β_0 la constante et les $\beta_{i,u}$ désignent les coefficients des variables $P_{X_{i,u}}$ de la régression.

Nous sommes alors capables de mesurer la qualité des règles grâce aux indicateurs classiques de la régression linéaire:

1. Le coefficient de détermination $R^2 = \frac{\sum_{i=1..n}(y_i^* - y_m)^2}{\sum_i (y_i - y_m)^2} = \frac{SSE}{SST}$ (où n est le nombre d'individus dans la régression, les y_i sont les valeurs de la variable dépendante Y, les y_i^* les prédictions de Y à partir de l'équation linéaire et y_m la moyenne des y_i). R^2 nous donne la part de la variation de Y expliquée par les variables explicatives de la régression. Plus la part de variation de Y expliquée est forte plus le R^2 est proche de 1.
2. Le test de Fisher-Snedecor (F-test) de validité de la régression.

Nous donnons table 8, les équations, les coefficients de détermination R^2, et les F-tests des règles découvertes table 5. Pour la règle 1 ($1/3 < P_v \leq 2/3 \rightarrow 0 < P_p \leq 1/3$), nous faisons la régression des poids P_v sur les poids P_p en conservant uniquement les individus avec un poids P_v dans l'intervalle]1/3,2/3] et un poids P_p dans l'intervalle]0,1/3], c'est-à-dire les individus 1, 2 et 3 (voir table 7). Nous obtenons $P_p = 0.16 + 0.25P_v$ lorsque P_v est dans l'intervalle]1/3,2/3] avec une part de variation de P_p expliquée par P_v de $R^2 = 0.25\%$. Ces résultats sont donnés à titre d'exemple étant donné que le nombre d'individus est ici trop faible pour obtenir des résultats significatifs (les R^2 sont faibles et les F-tests rejettent les régressions).

5 Applications

5.1 Règles d'association classiques versus symboliques

Nous comparons les règles générées à partir des itemsets fréquents issus de l'algorithme classique appliqué à la matrice table 3 et les règles générées à partir des objets symboliques issus de l'algorithme "diagramme" appliqué à la matrice table 4. Nous donnons les règles obtenues table 9 pour le cas classique avec $minsup = 35\%$ et $minconf = 55\%$. Pour le cas diagramme, nous rentrons comme paramètres à l'algorithme, la précision h=3 (correspondant au plus grand nombre d'items pour un même client), le support minimum $minsup = 35\%$ (i.e. 2 clients). Les OS fréquents sont donnés table 6 où P_v, P_p et P_c sont les poids de v = viande, p = poissons, c = pâtes et céréales. Nous donnons table 5 les règles symboliques pour $minDC = 55\%$.

Dans les deux cas, nous remarquons que l'achat de pâtes et céréales implique, avec une confiance de 100%, l'achat de poissons. Toutefois, la méthode diagramme nous

N°	Règle	Support %	Confiance %	N°	Règle	Support %	Confiance %
1	c → p	36	100	3	p → c	36	57
2	p → v	45	71	4	v → p	45	55

TAB. 9 – *Règles d'association classiques*

fournit plus d'informations que la méthode classique. En effet, nous savons en plus que les clients de pâtes et céréales achètent plus de poissons que de pâtes : $0 < P_c \leq 1/3 \rightarrow 0 < P_p \leq 2/3$ avec un support de 75%, une confiance de 100% et une DC de 60%. Nous pouvons alors calculer une relation linéaire entre les prémisses et la conclusion afin d'étudier les variations à l'intérieur de la règle (voir table 8). Nous obtenons que les clients de pâtes et céréales achètent environ deux fois plus de poissons que de pâtes puisque nous avons l'équation P_p=-0.1+2P_c. En fait, le test de Fisher de validité de la régression rejette cette équation du fait du nombre trop faible d'individus dans le calcul de la régression.

Deuxièmement, avec l'étude classique, nous obtenons les règles $v \rightarrow p$ avec $conf(v \rightarrow p) = 55\%$ et $p \rightarrow v$ avec $conf(p \rightarrow v) = 71\%$. Par conséquent, la meilleure règle, selon la confiance, serait $p \rightarrow v$ alors que dans le cas symbolique nous obtenons "l'inverse". En effet, la règle $1/3 < P_v \leq 2/3 \rightarrow 0 < P_p \leq 2/3$ est meilleure que la règle $0 < P_p \leq 2/3 \rightarrow 1/3 < P_v \leq 2/3$ selon la confiance (100%, 75% resp.). Ainsi, nous voyons que si le "degré d'inclusion" de l'achat de poissons dans l'achat de viandes dans les transactions est grand, l'analyse symbolique nous montre qu'en fait ce sont plutôt les clients de viandes qui sont aussi clients de poissons et non l'inverse. Et comme le montre la règle 1, les clients de viandes sont aussi clients de poissons bien qu'ils achètent plus de viandes que de poissons. Si nous prenons un autre exemple, dans un tabac la vente de cigarettes est très importante et par conséquent le "degré d'inclusion" de l'achat de jeux à gratter dans l'achat de cigarettes dans les transactions est grand mais, en fait, ce sont les clients de cigarettes qui pourront être amenés à acheter des jeux et non l'inverse comme l'aurait suggéré le cas classique.

6 Conclusions et perspectives

Nous avons étendu l'algorithme Apriori au cas des variables symboliques diagrammes dans le but d'extraire des règles d'association à partir d'une matrice de concepts. Nous avons pris comme exemple des clients de magasins quelconque où nous trouvons des règles entre les articles achetés au niveau des clients et non plus au niveau des transactions. Nous avons constaté que nous découvrions des informations supplémentaires par rapport aux règles classiques. De plus, nous avons proposé une manière d'étudier la variation à l'intérieur de ces règles d'association à l'aide de la régression linéaire symbolique. Il serait alors intéressant d'étendre cette méthode à d'autres variables symboliques afin d'extraire des règles d'association plus riches.

Références

[Afonso et al., 2004] F. Afonso, L. Billard et E. Diday. Régression Linéaire Symbolique avec Variables Taxonomiques. *Actes des 4èmes journées d'Extraction et de Gestion des Connaissances*, EGC'04, Clermont-Ferrand, Cépadues, 2004.

[Afonso et al., 2003] F. Afonso, L. Billard et E. Diday. Extension des Méthodes de Régression Linéaire aux cas des Variables Symboliques Taxonomiques et Hiérarchiques. *Actes des XXXVèmes journées de Statistiques*, SFDS-03, Lyon,Vol. 1, 89-92, 2003.

[Agrawal et Srikant, 1994] R. Agrawal et R. Srikant. Fast Algorithms for Mining Association Rules. *Proc. of the 20th Int'l Conf. on Very Large Databases*, 1994.

[Agrawal et al., 1993] R. Agrawal, T. Imielinski et A. Swami. Mining Association Rules between Sets of Items in Large Databases. *ACM SIGMOD Records*, 1993.

[Billard et Diday, 2002] L. Billard et E. Diday. Symbolic Regression Analysis. *Classification, Clustering, and Data Analysis*, K. Jajuga, A. Sokolowski, et H.H. Bock eds., Berlin, Springer-Verlag, 281-288, 2002.

[Bock et Diday, 2000] H-H. Bock et E. Diday. Analysis of Symbolic Data. Exploratory methods for extracting statistical information from complex data, Springer Verlag, Heidelberg, 2000.

[Cai et al., 1998] C.H. Cai, A.W.C. Fu, C.H. Cheng et W.W. Kwong. Mining Association Rules With Weighted Items. *Proc. of the 1998 Int'l Database Engineering and Applications Symposium (IDEAS'98)*, 68-77, 1998.

[Han et Fu, 1995] J. Han et Y. Fu. Discovery of Multiple-Level Association Rules from Large Databases. *Proc. of the 21th Int'l Conf. on Very Large Data Bases*, 1995.

[Kuok et al., 1998] C.M. Kuok, A. Fu et M.H. Wong. Mining Fuzzy Association Rules in Databases. *ACM SIGMOD Record*, Vol. 27, 41-46, 1998.

[Miller et Yang, 1997] R.J. Miller et Y. Yang. Association Rules over Interval Data. *Proc. of the 1997 ACM SIGMOD int'l conf. on Management of data*, 452-461, 1997.

[Srikant et al., 1997] R. Srikant, Q. Vu et R. Agrawal. Mining Association Rules with Item Constraints. *Proc. of the 3rd Int'l Conf. on Knowledge Discovery in Databases and Data Mining*, 1997.

[Srikant et Agrawal, 1996] R. Srikant et R. Agrawal. Mining Quantitative Association Rules in Large Relational Tables. *Proc. of the ACM-SIGMOD 1996 Conf. on Management of Data*, 1996.

[Wang et al., 2000] W. Wang, J. Yang et P. Yu. Efficient Mining of Weighted Association Rules (WAR). *Proc. of the sixth ACM SIGKDD int'l conf. on Knowledge discovery and data mining* , 270-274, 2000.

Summary

This paper deals with the extension of the Apriori algorithm and of the association rules to the symbolic histogram-valued data. We suggest a method that will enable us to discover rules at the level of the concepts. For example, instead of mining rules between different items of some transactions recorded in a retail organization like in the classical case, we will mine rules at the level of the customers in order to study their purchase behavior. Finally, we suggest a method in order to evaluate the quality of the association rules according to the symbolic linear regression.