Extraction de propriétés correctes dans des bases de données incomplètes

François Rioult et Bruno Crémilleux

GREYC, CNRS - UMR 6072, Université de Caen
F-14032 Caen Cédex France
{Francois.Rioult,Bruno.Cremilleux}@info.unicaen.fr

Résumé :
Les valeurs manquantes dans les bases de données posent de nombreuses difficultés lors de processus d'extraction de connaissances et les propriétés extraites sont parfois incorrectes. Nous proposons à l'aide de calculs menés dans une base incomplète de caractériser des propriétés de la base complète dont la base de calcul est issue. Les propriétés de k-liberté sont étudiées, elles sont fondamentales pour le calcul des motifs fréquents, des représentations condensées de ces motifs et la construction de règles d'association généralisées. Nous donnons pour les bases de données incomplètes une définition de la k-liberté et montrons un résultat de correction pour cette propriété.

1 Introduction

Le problème de la gestion des valeurs manquantes dans les bases de données est aussi ancien que l'avènement de ces structures de stockage. Tout comme les techniques relevant de l'apprentissage automatique, des statistiques ou de l'analyse de données, celles de la fouille de données peinent à prendre en compte ces éléments particuliers. Pourtant, les dispositifs utilisant les règles d'association sont très populaires, fournissant à l'aide d'algorithmes efficaces des connaissances potentiellement intéressantes et aisément interprétables par les experts. Ces règles et plus généralement les motifs fréquents autorisent la mise en œuvre de méthodes d'exploration des données, comme la classification supervisée ou non supervisée.

Préalablement à la mise en œuvre de toute méthode, il est souhaitable d'évaluer la qualité des connaissances extraites à partir d'une base de données incomplète. En particulier, toute propriété calculée dans une base incomplète se doit d'être cohérente avec les propriétés calculées dans toute base complète dont la base de calcul est issue, et nous parlerons de *correction* de ces propriétés. Pour caractériser les bases complètes possibles, nous modélisons une base incomplète grâce à un opérateur de transformation agissant sur une base complète.

Nous cherchons à obtenir dans des bases incomplètes des propriétés correctes de k-*liberté*, centrales pour le calcul des motifs fréquents, des règles d'associations, et plus

généralement des *représentations condensées* de motifs. En utilisant les définitions classiques dans des données incomplètes, les propriétés calculées sont parfois incorrectes. Il est souvent impossible de certifier les calculs effectués car les valeurs manquantes ne permettent pas de déterminer précisément les supports des motifs utilisés pour rendre les décisions. Notre travail consiste donc à définir les modes de calculs des propriétés de k-liberté dans des bases de données incomplètes, puis à montrer que les propriétés obtenues sont correctes et caractérisent toute complétion de la base incomplète.

La présentation est organisée comme suit : la section 2 fournit les pré-requis techniques nécessaires sur la k-liberté des motifs, montre rapidement les dégâts causés par les valeurs manquantes lors du calcul des motifs k-libres et explique notre positionnement pour cette problématique. La section 3 définit le calcul des motifs k-libres lorsqu'il y a des valeurs manquantes. Cette méthode est évaluée sur des données de test à la section 4.

2 Préliminaires

Après quelques rappels de vocabulaire pour la fouille de données, nous introduisons les règles d'association généralisées et les motifs k-libres. Nous montrons ensuite les problèmes posés lors de l'extraction de ces motifs dans une base incomplète et terminons ces préliminaires en expliquant notre positionnement face à ces problèmes.

Classiquement, les bases de données décrivent des *objets* à l'aide d'*attributs* quantitatifs ou qualitatifs. Le format correspondant est qualifié d'*attribut/valeur*. La table 1 montre l'exemple de huit objets décrits par trois attributs X_1, X_2, X_3. Dans le domaine des règles d'association et plus généralement de la fouille de données orientée *motifs*, les attributs quantitatifs doivent être segmentés puis binarisés, afin de disposer de contextes booléens (cet article ne discute pas de cette étape).

	attributs		
objets	X_1	X_2	X_3
o_1	$+$	$\rightarrow$	0.2
o_2	$-$	$\rightarrow$	0
o_3	$+$	$\rightarrow$	0.1
o_4	$+$	$\leftarrow$	0.4
o_5	$-$	$\rightarrow$	0.6
o_6	$-$	$\rightarrow$	0.5
o_7	$+$	$\leftarrow$	1
o_8	$-$	$\leftarrow$	0.8

TAB. 1 – Base de données au format attribut/valeur.

Nous disposons d'une base de données r sous la forme d'un *contexte booléen* $(\mathcal{A}, \mathcal{O}, R)$ où $\mathcal{O}$ est l'ensemble des objets étudiés, $\mathcal{A}$ est l'ensemble des attributs de ces objets et R est une relation binaire qui indique la présence ou l'absence de chaque attribut binaire dans chaque objet. R peut être représentée par une matrice booléenne mais

l'on considérera également chaque objet comme un sous ensemble de $\mathcal{A}$ (par exemple, $o_1 = \{a_1, a_3, a_5\}$, que l'on notera sous forme de chaîne $a_1a_3a_5$). Par abus d'écriture, nous noterons $|r|$ pour désigner le nombre d'objets de r, soit $|r| = |\mathcal{O}|$. La table 2 reprend notre précédent exemple, où l'attribut X_3 est codé par les attributs binaires a_5 à a_7.

	attributs						
objets	a_1	a_2	a_3	a_4	a_5	a_6	a_7
o_1	×		×		×		
o_2		×	×		×		
o_3	×		×		×		
o_4	×			×		×	
o_5		×	×			×	
o_6		×	×			×	
o_7	×			×			×
o_8		×		×			×

TAB. 2 – Exemple d'un contexte booléen r.

Un *motif* X est un sous ensemble de $\mathcal{A}$, son *support* est l'ensemble des objets qui le contiennent (nous noterons $supp(X) = r_X = \{o \in \mathcal{O} \mid X \subseteq o\}$) et sa *fréquence* $\mathcal{F}(X) = |supp(X)|$ est le nombre d'objet du support. Une règle d'association *classique* (Agrawal & Srikant, 1994) est une expression $X \rightarrow Y$, où X et Y sont deux motifs. Elle est quantifiée par sa fréquence, c'est-à-dire celle de $X \cup Y$, et sa confiance qui est la probabilité conditionnelle de présence de Y dans les objets contenant X, ou $conf(X \rightarrow Y) = \mathcal{F}(X \cup Y)/\mathcal{F}(X)$. Elle est exacte dans r et nous notons $\models_r X \rightarrow Y$ quand sa confiance vaut 1.

Par exemple, la règle $a_7 \rightarrow a_4$ est exacte dans les données de la table 2 car chaque objet contenant l'attribut a_7 contient également a_4. Du point de vue des fréquences, $\mathcal{F}(a_7a_4) = \mathcal{F}(a_7)$.

2.1 Règles d'association généralisées et k-liberté

Nous considérons ici une forme *généralisée* de règles d'association, appelées *règles disjonctives* dans (Calders & Goethals, 2003), par opposition à la forme classique. Ces règles ont la particularité de conclure sur une disjonction d'attributs plutôt que sur une conjonction. L'utilisation de *motifs généralisés* est nécessaire pour étudier ces règles. Ces motifs contiennent des attributs booléens mais également des négations d'attributs booléens. Par exemple, $Z = \{a_1, \overline{a_2}, a_3\}$ est un motif généralisé, que l'on écrira comme l'union d'une partie *positive* $X = \{a_1, a_3\}$ et d'une partie *négative* $\overline{Y}$ où $Y = \{a_2\}$, chaque partie étant un motif *classique*. Un objet o supporte $Z = X \cup \overline{Y}$ si $X \subseteq o$ et $Y \cap o = \emptyset$ (o supporte la partie positive du motif, mais ne contient aucun attribut de la partie négative). Pour alléger les notations, nous omettrons par la suite le signe *union* et écrirons $X\overline{Y}$ à la place de $X \cup \overline{Y}$.

La fréquence de $Z = X\overline{Y}$ est fondamentale : si elle est nulle, cela signifie que l'un des éléments de Y est toujours présent avec X, et valide une association généralisée entre X et Y :

Définition 1
Une règle d'association généralisée basée sur $Z = X \cup Y$ est une expression de la forme $X \rightarrow \vee Y$ où X et Y sont deux motifs classiques. Elle est exacte dans une base de données r si tout objet de r contenant la prémisse X contient également un attribut de la conclusion Y. Nous notons $\models_r X \rightarrow \vee Y \iff \mathcal{F}(X\overline{Y}, r) = 0$. La fréquence de la règle $X \rightarrow \vee Y$, notée $\mathcal{F}(X \rightarrow \vee Y)$, est le nombre d'objets contenant X et au moins un des attributs de Y[1]*. $\mathcal{F}(X \rightarrow \vee Y) = \mathcal{F}(X) - \mathcal{F}(X\overline{Y})$. Sa profondeur est la longueur de la conclusion.*

Le calcul des règles généralisées est complexe et on limitera dans la pratique la longueur des conclusions potentielles à un entier k, afin d'éviter l'explosion combinatoire.

Les motifs k-libres, récemment formalisés par Calders et Goethals (Calders & Goethals, 2003), sont utiles pour calculer les règles d'association généralisées. Ils expriment l'absence de corrélation entre les attributs qui les constituent. Jusqu'ici, ils ont surtout été employés pour calculer des représentations condensées des motifs fréquents (Calders & Goethals, 2002).

Définition 2 (Motif k-libre)
Un motif Z est k-libre dans une base de données complète (sans valeurs manquantes) r et nous notons $kLibre(Z, r)$ si il n'existe aucune règle d'association généralisée exacte basée sur Z dans r, soit : $\forall X \cup Y = Z,\ |Y| \leq k \Rightarrow \mathcal{F}(X\overline{Y}) \neq 0$.

La k-liberté est une propriété anti-monotone et les motifs peuvent être extraits par niveaux (Mannila & Toivonen, 1997). Pour décider de la k-liberté d'un motif candidat à la phase d'examen de la base qui calcule les supports, la fréquence de $X\overline{Y}$ est calculée grâce au principe d'inclusion-exclusion (Jaroszewicz & Simovici, 2002) utilisant la fréquence des sous-ensembles de XY : $\mathcal{F}(X\overline{Y}) = \sum_{\emptyset \subseteq J \subseteq Y} (-1)^{|J|} \mathcal{F}(XJ)$

Les motifs k-libres ont d'excellentes propriétés pour résumer les collections de motifs fréquents. Par exemple, dans la base `mushroom` (Blake & Merz, 1998), il y a $2,7 \cdot 10^9$ fréquents de fréquence 1, 426134 1-libres, 224154 2-libres[2]. Au-delà de $k = 5$, le nombre de k-libres stagne à 214530, extraits en deux minutes. La difficulté de calcul des supports à l'aide du principe d'inclusion-exclusion est supportable car k reste faible.

Ils permettent également de construire les règles d'association généralisées. En particulier, le rôle des motifs 1-libres est bien connu pour calculer les règles d'association classiques non redondantes (Bastide *et al.*, 2000; Zaki, 2000). Ces règles sont constituées d'une prémisse X 1-libre et d'une conclusion qui est la fermeture $h(X)$ au sens de GALOIS (elle rassemble les attributs présents dans chaque objet contenant X).

[1] La définition de la fréquence d'une règle diverge ici avec la fréquence d'une règle d'association classique.

[2] également introduits par (Bykowski & Rigotti, 2001) sous l'appellation *disjonction-free sets*.

La mise en évidence de règles généralisées non redondantes est plus complexe et procède de deux manières différentes. La première consiste à extraire des motifs X 1-libres et à calculer leur *fermeture généralisée* (Rioult, 2005), rassemblant les motifs Y qui partagent un attribut avec chaque objet contenant X. Pour avoir des conclusions disjonctives non redondantes, on ne conserve que les motifs Y minimaux. Cette opération revient, pour chaque X à construire les conclusions obtenues par un calcul des traverses minimales (Gunopulos *et al.*, 1997) des objets contenant X (Rioult, 2005).

Le deuxième procédé utilise la propriété d'*anti monotonie* de la k-liberté et le formalisme des bordures de la théorie de ce type de propriétés (Mannila & Toivonen, 1997). Les règles sont construites à partir des motifs non k-libres minimaux, constituant la *bordure négative des motifs k-libres* et matérialisant l'apparition des corrélations (Rioult, 2005). La bordure négative s'obtient soit par les traverses minimales sur les complémentaires des k-libres maximaux (Demetrovics & Thi, 1995; Mannila & Räihä, 1986) ; soit lors de l'extraction des k-libres, ce sont les motifs exclus lors de la phase de génération car ils ne vérifient pas la contrainte de k-liberté, ou bien leur fréquence est nulle. Sur Z non k-libre, on construira une règle d'association généralisée $X \rightarrow \vee Y$ où X est le plus petit motif inclus dans Z (et $Y = Z \backslash X$) tel que $\mathcal{F}(X\overline{Y}) = 0$.

Les règles généralisées expriment des corrélations avec un formalisme plus riche que celui des règles classiques. Des règles positives et négatives (Antonie & Zaïane, 2004b), concluant sur un attribut ou la négation de cet attribut, peuvent ainsi être déduites et trouvent leur utilité dans la mise au point de méthodes de classification supervisée (Antonie & Zaïane, 2004a). Par exemple, la règle $a_1 \rightarrow a_4 \vee a_5$ est exacte dans les données de la table 2 car chaque objet contenant a_1 contient soit a_4, soit a_5. Ainsi, on peut construire la règle positive $a_1\overline{a_4} \rightarrow a_5$ et la règle négative $\overline{a_4 a_5} \rightarrow \overline{a_1}$. Les motifs k-libres jouent donc un rôle central pour la classification supervisée.

2.2 Dégâts des valeurs manquantes sur les motifs k-libres

Supposons que certaines variables de notre exemple de la table 1 n'aient pu être mesurées, ignorant par exemple pour certains objets si $X_1 = a_1$ ou $X_1 = a_2$. Une valeur manquante apparaît alors et nous utilisons le caractère `'?'` pour indiquer que cette valeur n'est ni présente, ni absente, et ce pour **chaque** attribut binaire issu de l'attribut multi-valué correspondant. Dans notre exemple de la table 1, nous avons artificiellement introduit trois valeurs manquantes, afin de simuler le processus réel qui rend les données incomplètes. La base r' résultant de cette opération ainsi que le codage binaire des sept valeurs manquantes sont indiqués à la table 3.

Le calcul du support d'un motif classique X dans une base incomplète est aménagé de la façon suivante : un objet contient X si tous ses attributs sont présents. Si l'un des attributs est manquant ou absent, l'objet n'appartient pas au support. La table 4 met en évidence les incorrections introduites par les valeurs manquantes lors du calcul des motifs 1-libres de fréquence minimale 2. La partie gauche concerne l'extraction réalisée dans la base complète (table 2). Pour chaque motif, nous indiquons sa fermeture. Nous pouvons par exemple déduire la règle informative $a_1 a_3 \rightarrow a_5$, présente deux fois dans les données (l'attribut a_5 est toujours présent avec le motif $a_1 a_3$).

Comment calculer les supports pour les motifs généralisés, lorsqu'il y a des valeurs

	attributs						
objets	a_1	a_2	a_3	a_4	a_5	a_6	a_7
o_1	×		×		×		
o_2		×	×		×		
o_3	×		×		**?**	**?**	**?**
o_4	×			×		×	
o_5		×	×			×	
o_6	**?**	**?**	×			×	
o_7	×			×			×
o_8		×	**?**	**?**			×

TAB. 3 – Base incomplète r' issue de r.

X	$h(X)$	X	$h(X)$
a_1		a_1a_3	a_5
a_2		a_1a_4	
a_3		a_1a_5	a_3
a_4		a_2a_3	
a_5	a_3	a_2a_6	a_3
a_6		a_3a_6	a_2
a_7	a_4		

Base complète r

X	$h(X)$	X	$h(X)$
a_1		a_1a_3	
a_2		a_1a_5	
a_3		a_2a_3	
a_4	a_1	a_2a_5	a_3
a_5	a_3	a_2a_6	a_3
a_6		a_3a_6	
a_7		a_4a_7	

Base incomplète r'

TAB. 4 – Liste des motifs 1-libres de fréquence minimale 2 et leur fermeture.

manquantes ? La définition 2 ne prévoit pas ce cas courant et le problème se pose en particulier pour déterminer la fréquence de $X\overline{Y}$. En l'absence de préconisation relative aux bases incomplètes, nous avons mené les calculs en ignorant les valeurs manquantes, les considérant absentes. La partie droite de la table 4 liste les motifs 1-libres extraits de la base incomplète r'. On constate par comparaison avec la partie gauche que le motif a_1a_4 n'est plus 1-libre. En outre, cette table contient des motifs, par exemple a_2a_5 et a_4a_7, qui ne sont pas présents dans l'extraction obtenue depuis la base originale r : nous qualifions ces derniers motifs d'*incorrects*.

Les valeurs manquantes produisent des dégâts à la fois sur les motifs libres et sur les attributs des fermetures. Concernant un motif X 1-libre, supposons qu'un attribut a appartienne dans la base complète à la fermeture de X : cela signifie que a est toujours présent avec X. Si des valeurs manquantes se produisent sur a, il existe donc des objets pour lesquels cette association est rompue : a sort alors de la fermeture de X (dégât sur la fermeture) et Xa peut devenir libre (dégât sur le libre). Sur notre exemple, a_4 est dans la fermeture de a_7 dans r, tandis que a_4 sort de cette fermeture dans r' à cause de la valeur manquante de l'objet o_8. Ainsi, a_4a_7 est incorrectement déclaré 1-libre.

Des expériences sur des données de test de l'UCI (Blake & Merz, 1998) mettent également en évidence ces dégâts. Partant d'une base complète, nous introduisons artifi-

ciellement des valeurs manquantes selon une probabilité uniforme. Puis nous extrayons les motifs 3-libres et mesurons le nombre de motifs incorrects relativement au nombre de motifs corrects obtenus depuis le contexte original (cf. figure 1).

Le nombre de motifs incorrects varie suivant les bases : il est inférieur à 10 % pour les bases `pima`, `wine`, `liver-disorders`, `servo` et `tic-tac-toe` (le graphique correspondant n'est pas reporté). Pour les bases de la partie gauche de la figure, le nombre de motifs incorrect se situe entre 10 et 90 % de l'extraction de référence. Enfin, pour la partie droite de la figure, cette quantité monte jusqu'à 300 %, ce qui signifie que pour quatre motifs calculés, trois motifs sont incorrects.

Dans les conditions réelles où on ne connaît pas la base complète, il est impossible de discerner les bons motifs des mauvais, ni de dire à l'avance s'il y aurait une petite ou une grande proportion de motifs incorrects. Notre travail vise à éviter ces dégâts en calculant correctement la propriété de k-liberté dans un contexte incomplet.

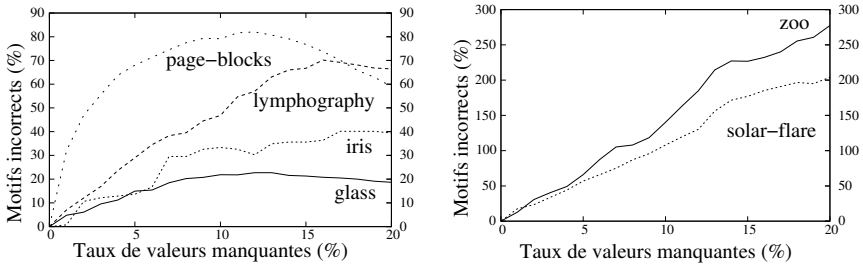

FIG. 1 – Motifs incorrects lors de l'extraction des 3-libres dans des bases de l'UCI.

2.3 Position du travail

Si les valeurs manquantes posent des problèmes classiques lors de l'étude des bases de données (Dyreson, 1997; Levene & Loizou, 1999), les contributions dans le domaine de la fouille de données sont peu nombreuses. Arnaud RAGEL (Ragel & Crémilleux, 1999) a étudié l'extraction des règles en présence de valeurs manquantes à des fins de complétion et a proposé de redéfinir les notions de support et de confiance. Plus récemment, (Nayak & Cook, 2001) expose une méthode simple d'imputation, fondée sur la probabilité des différentes valeurs d'un attribut. De cette façon, le support d'un motif par un objet n'est plus booléen, mais probabiliste. Enfin, (Jami *et al.*, 2004) calcule dans la partie de la base de données qui est complète des règles de prédiction pour les valeurs manquantes. Elles fournissent des intervalles pour les attributs continus.

Nous avons décidé d'orienter notre travail selon deux grands principes :

– nous ne voulons pas compléter les valeurs manquantes avant d'effectuer l'étape d'extraction de connaissances car c'est une opération difficile à réaliser sans connaissance préliminaire.
– nous souhaitons extraire des connaissances à partir de la base incomplète, sans la réduire à une quelconque portion complète. Par exemple, nous ne voulons pas ignorer certains objets sous le prétexte qu'ils ne sont pas complètement décrits.

Nous réserverons donc un traitement particulier pour chaque valeur manquante et ne ferons pas d'hypothèse statistique concernant le modèle de probabilité d'apparition des

valeurs manquantes. Nous définissons pour cela à la section suivante un *opérateur de modélisation* des valeurs manquantes, noté $mv()$. Cette formalisation est utile car elle permet de définir une base incomplète comme le résultat d'une opération d'effacement de certaines valeurs, réalisée sur une base complète. Dans ces conditions, des calculs menés dans une base incomplète peuvent caractériser les propriétés communes à toutes les bases complètes dont la base de calcul est issue. Cela évite ainsi de considérer toutes les bases originales possibles.

Dans la suite de notre présentation, nous verrons qu'il est, sous ces hypothèses, possible de découvrir des connaissances valides dans la base complète, comme cela avait été signalé dans (Bosc *et al.*, 2002). Ce principe n'est pas si surprenant : si l'on considère que les valeurs manquantes occultent la véritable valeur d'une donnée, les fréquences d'apparition de certains motifs vont diminuer, car la décision de présence n'est plus possible pour certains objets. Un motif fréquent détecté par un calcul dans une base incomplète ne peut donc *a fortiori* qu'être fréquent dans la base complète.

Nous avons vu (cf. section 2.2) que calculer sans précaution les motifs k-libres d'une base produit des motifs *incorrects*, n'ayant pas cette propriété dans toute complétion de la base incomplète disponible. Notre travail poursuit le but de définir correctement le calcul de la propriété de k-liberté dans une base de données incomplète et la définition suivante précise cette notion :

Définition 3 (motif k-correct)
Soit r' une base de donnée incomplète et $mv()$ un opérateur de modélisation de valeurs manquantes. Un motif Z est k-correct dans r' si pour toute base complète r, $(mv(r) = r') \Rightarrow kLibre(Z, r)$.

Lorsqu'un motif est k-correct dans r', on est certain qu'il est k-libre dans toute complétion de r'. Or, la propriété de k-liberté, exposée à la définition 2 et restreinte aux bases complètes, ne peut être calculée sans précaution dans une base incomplète. Il est souhaitable que cette opération fournisse des motifs k-corrects.

3 Calcul de motifs k-libres dans les bases incomplètes

Nous proposons ici une définition de la propriété de k-liberté dans une base incomplète. Nous montrons qu'elle permet de calculer des motifs k-corrects, c'est-à-dire k-libres dans toute complétion de la base de calcul.

3.1 Opérateur de modélisation des valeurs manquantes

Comme nous l'avons explicité à la section précédente, notre positionnement sur le problème des valeurs manquantes requiert l'utilisation d'un opérateur de modélisation des valeurs manquantes. Celui-ci définit la relation entre une base incomplète et toute complétion possible.

Définition 4 (Opérateur de modélisation de valeurs manquantes)
Soit $r = (\mathcal{A}, \mathcal{O}, R)$ un contexte booléen. Un opérateur $mv()$ est appelé opérateur de modélisation de valeurs manquantes s'il transforme une base complète r en $mv(r) = (\mathcal{A}, \mathcal{O}, mv(R))$. La nouvelle relation binaire $mv(R)$ prend ses valeurs dans $\{present, absent, manquant\}$ et satisfait les propriétés suivantes, pour tout attribut a de $\mathcal{A}$ et tout objet o de $\mathcal{O}$, et $valeur \in \{present, absent\}$:

1. $mv(R)(a, o) = valeur \Rightarrow R(a, o) = valeur$;
2. $R(a, o) = valeur \Rightarrow mv(R)(a, o) \in \{valeur, absent\}$;

L'opérateur $mv()$ modélise un effacement des données. Dans le cas où une valeur est manquante dans $mv(r)$, il est donc impossible de connaître la valeur originale dans r et c'est une propriété de compatibilité forte. Quand la valeur est connue dans $mv(r)$, elle l'est également dans la base complète et c'est la même valeur. En revanche, la deuxième propriété de $mv()$ assure qu'une valeur présente ou absente dans r conservera cette qualité à l'identique dans $mv(r)$, ou sera manquante.

3.2 Désactivation temporaire d'objets

Nous introduisons ici la *désactivation* temporaire d'objets d'une base incomplète, qui différencie d'une part les objets qui supportent ou ne supportent pas un motif donné, et d'autre part les objets incomplets pour lesquels la décision de support ne peut être rendue. La désactivation permet de quantifier l'écart de fréquence constaté entre les bases complètes et incomplètes. En effet, en présence de valeurs manquantes, les fréquences décroissent. Sur notre exemple (table 2), $\mathcal{F}(a_3a_5, r) = 3$ mais $\mathcal{F}(a_3a_5, mv(r)) = 2$ (table 3). Pour pouvoir calculer correctement la fréquence d'un motif X dans $mv(r)$, il est nécessaire de distinguer les objets de $mv(r)$ qui ont une valeur manquante parmi les attributs de X. Ces objets vont être temporairement désactivés pour calculer une estimation de $supp(X, r)$ à l'aide de $supp(X, mv(r))$, car il est impossible de décider si oui ou non ils contiennent X.

Nous commençons par définir formellement les objets désactivés relativement à un motif classique :

Définition 5 (Objet désactivé)
Pour un motif classique $X \subseteq \mathcal{A}$, un objet $o \in \mathcal{O}$ est désactivé si $\forall a \in X, mv(R)(a, o) \neq absent$ et $\exists a \in X$ t.q. $mv(R)(a, o) = manquant$. Nous notons $\mathcal{DES}(X, mv(r))$ pour les objets de $mv(r)$ désactivés pour X.

La figure 2 illustre la notion de désactivation, en représentant simultanément la base complète r (sur la gauche) et la base incomplète $mv(r)$ (sur la droite). On suppose que chaque objet de la moitié supérieure de la base contient X et cette moitié de la base est repérée r_X. La moitié inférieure est repérée $r_{\overline{X}}$.

Sur la droite, la zone hachurée désigne les objets de $mv(r)$ qui contiennent des valeurs manquantes. Elle est constituée de six ensembles d'objets décrits ci-dessous (leur composition exacte est indiquée pour notre exemple de la table 3, avec $X = a_2a_3$) :

Région A : (o_2, o_5) les objets sans valeur manquante, contenant X ;

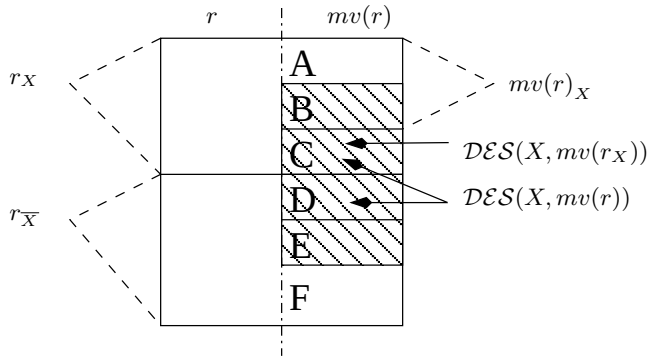

FIG. 2 – Base $mv(r)$ et objets désactivés pour X.

Région B : (aucun objet dans notre exemple) les objets contenant initialement X, dont les valeurs manquantes n'occultent pas la présence de X. Ces objets appartiennent à $mv(r)_X$;

Région C : (o_6) les objets contenant initialement X, dont les valeurs manquantes cachent la présence de X et constituent $\mathcal{DES}(X, mv(r_X))$;

Région D : (o_8) les objets ne contenant pas X dans la base complète mais qui pourraient le contenir avec un remplacement adéquat des valeurs manquantes. Dans notre exemple, l'objet o_8 ne contient pas le motif a_2a_3 dans la base complète que nous avons choisie pour exemple et ces objets sont désactivés par précaution ;

Région E : (o_3) les objets incomplets ne contenant X ni dans la base originale ni après une quelconque substitution des valeurs manquantes ;

Région F : (o_1, o_4, o_7) les objets complets qui ne contiennent pas X.

À l'examen de la base de données incomplète $mv(r)$, les objets seront donc répartis en trois groupes pour décider du support de X :

Régions A et B : $mv(r)_X$ objets supportant X, malgré les valeurs manquantes de B ;

Régions C et D : $\mathcal{DES}(X, mv(r))$ les objets où le support de X est indécidable ;

Régions E et F : les objets qui ne supportent pas X.

La notion de désactivation permet de caractériser précisément à la différence de support pour X entre la base incomplète et la base complète :

Proposition 1
Soit X un motif classique, r une base de données et mv un opérateur de modélisation de valeurs manquantes, alors $\mathcal{DES}(X, mv(r_X)) = r_X \setminus mv(r)_X$. D'un point de vue fréquentiel, $|\mathcal{DES}(X, mv(r_X))| = \mathcal{F}(X, r) - \mathcal{F}(X, mv(r))$.

Détaillons ce principe sur notre exemple pour le motif a_2a_3 : $r_{a_2a_3} = \{o_2, o_5, o_6\}$ et sa fréquence vaut 3. Dans la base incomplète, sa fréquence n'est plus que 2 et $\mathcal{DES}(a_2a_3, mv(r_{a_2a_3})) = \{o_6\}$: nous avons bien l'égalité de la proposition 1. Quand

on ne connaît pas la base complète r, on ne connaît pas non plus r_X, encore moins $|\mathcal{DES}(X, mv(r_X))|$. Mais cette quantité peut être bornée en considérant les objets désactivés dans $mv(r)$, car cette base contient plus d'objets que $mv(r_X)$. Sur notre exemple $\mathcal{DES}(a_2a_3, mv(r)) = \{o_6, o_8\}$, à cause de la confusion induite dans l'objet o_8 par la valeur manquante qui touche a_3 et a_4. $\mathcal{F}(a_2a_3, r)$ est donc compris entre $\mathcal{F}(a_2a_3, mv(r))$ et $\mathcal{F}(a_2a_3, mv(r)) + |\mathcal{DES}(a_2a_3, mv(r))|$, soit entre 2 et 4.

Pour la suite de l'exposé, il est nécessaire de définir la désactivation pour les motifs généralisés. Nous utilisons pour cela le principe d'inclusion-exclusion :

Définition 6 (Désactivation généralisée)
$des(X\overline{Y}, mv(r_{X\overline{Y}})) = \sum_{\emptyset \subseteq J \subseteq Y} (-1)^{|J|} |\mathcal{DES}(XJ, mv(r_{XJ}))|$.

L'ensemble $\mathcal{DES}(X\overline{Y}, mv(r_{X\overline{Y}}))$ n'est pas défini et c'est pourquoi nous notons la désactivation généralisée avec des minuscules. Toutefois, elle permet de qualifier la différence de fréquence d'un motif généralisé entre la base complète et la base incomplète.

Proposition 2
$des(X\overline{Y}, mv(r_{X\overline{Y}})) = \mathcal{F}(X\overline{Y}, r) - \mathcal{F}(X\overline{Y}, mv(r))$.

Cet écart de fréquence peut être négatif. Lorsque l'association entre X et Y existe dans la base complète ($\mathcal{F}(X\overline{Y}, r) = 0$), la présence d'une valeur manquante peut la faire disparaître dans la base incomplète ($\mathcal{F}(X\overline{Y}, mv(r)) > 0$). Dans ce cas, la différence est négative. Dans notre exemple, $des(a_7\overline{a_4}) = 0 - 1 = -1$.

Concernant les objets désactivés pour une association $X \rightarrow \vee Y$, nous définissons $|\mathcal{DES}(X \rightarrow \vee Y, mv(r_{X \rightarrow \vee Y}))| = |\mathcal{DES}(X, mv(r_X))| - des(X\overline{Y}, mv(r_{X\overline{Y}}))$. Nous retrouvons alors pour une association le comportement de la désactivation, analogue à celui déjà mis en évidence par les propositions 1 et 2 : $|\mathcal{DES}(X \rightarrow \vee Y, mv(r_{X \rightarrow \vee Y}))| = \mathcal{F}(X \rightarrow \vee Y, r) - \mathcal{F}(X \rightarrow \vee Y, mv(r))$.

De plus, un objet est désactivé pour une association $X \rightarrow \vee Y$ s'il est désactivé pour X, ou s'il contient X mais tous les attributs de Y sont manquants. En notant $\mathcal{DES}(\wedge Y, mv(r_{X \rightarrow \vee Y})_X)$ pour ces objets, nous avons $|\mathcal{DES}(X \rightarrow \vee Y, mv(r_{X \rightarrow \vee Y})) = |\mathcal{DES}(X, mv(r_{X \rightarrow \vee Y}))| + |\mathcal{DES}(\wedge Y, mv(r_{X \rightarrow \vee Y})_X)|$.

3.3 Définition et correction de la k-liberté dans les bases incomplètes

À l'aide de la désactivation des objets incomplets, la fréquence de $X\overline{Y}$ dans r peut être bornée par des quantités calculées dans $mv(r)$:

Propriété 1
$\mathcal{F}(X\overline{Y}, mv(r)) - |\mathcal{DES}(\wedge Y, (mv(r))_X)| \leq \mathcal{F}(X\overline{Y}, r) \leq \mathcal{F}(X\overline{Y}, mv(r)) + |\mathcal{DES}(X, mv(r))|$.

Preuve :
La proposition 2 indique que $\mathcal{F}(X\overline{Y}, r) = \mathcal{F}(X\overline{Y}, mv(r)) + des(X\overline{Y}, mv(r_{X\overline{Y}}))$. La définition de la désactivation d'une association permet d'écrire $des(X\overline{Y}, mv(r_{X\overline{Y}})) = |\mathcal{DES}(X, mv(r_X))| - |\mathcal{DES}(X \rightarrow \vee Y, mv(r_{X \rightarrow \vee Y}))|$. D'une part, on a la majoration

$des(X\overline{Y}, mv(r_{X\overline{Y}})) \leq |\mathcal{DES}(X, mv(r_X))|$, et donc en retirant la restriction sur la base de désactivation, $des(X\overline{Y}, mv(r_{X\overline{Y}})) \leq |\mathcal{DES}(X, mv(r))|$. D'autre part, on peut décomposer en $des(X\overline{Y}, mv(r_{X\overline{Y}})) = |\mathcal{DES}(X, mv(r_X))| - (|\mathcal{DES}(X, mv(r_{X\to\vee Y}))| + |\mathcal{DES}(\wedge Y, mv(r_{X\to\vee Y})_X)|) = (|\mathcal{DES}(X, mv(r_X))| - |\mathcal{DES}(X, mv(r_{X\to\vee Y}))|) - |\mathcal{DES}(\wedge Y, mv(r_{X\to\vee Y})_X)|$. La différence $|\mathcal{DES}(X, mv(r_X))| - |\mathcal{DES}(X, mv(r_{X\to\vee Y}))|$ est positive donc on a la minoration $des(X\overline{Y}, mv(r_{X\overline{Y}})) \geq |\mathcal{DES}(\wedge Y, mv(r_{X\to\vee Y})_X)|$. Sans restreindre la base de désactivation, $des(X\overline{Y}, mv(r_{X\overline{Y}})) \geq |\mathcal{DES}(\wedge Y, mv(r)_X)|$. □

Grâce à ces bornes, il est possible de définir la notion de k-liberté dans les bases de données incomplètes en bornant la fréquence de $X\overline{Y}$. La désactivation permet cette opération dans un contexte incomplet et nous définissons pour cela la *k-liberté* dans une base incomplète :

Définition 7 (k-liberté dans une base incomplète)
- *Un motif Z est k-libre dans $mv(r)$ et nous notons $kLibre(Z, mv(r))$ si et seulement si* $\forall XY = Z,\ |Y| \leq k,\ \mathcal{F}(X\overline{Y}, mv(r)) - |\mathcal{DES}(\wedge Y, (mv(r))_X)| > 0$.
- *Un motif Z est k-dépendant dans $mv(r)$ et nous notons $kDepdt(Z, r)$ si et seulement si* $\exists XY = Z,\ |Y| \leq k,\ \mathcal{F}(X\overline{Y}, mv(r)) + |\mathcal{DES}(X, mv(r))| = 0$.

Les notions de k-liberté et k-dépendance sont introduites indépendamment l'une de l'autre. La section 3.4 justifiera cette distinction, car ces deux définitions ne sont pas contraires, à cause des valeurs manquantes.

Notons tout d'abord que dans le cas d'une base complète, la notion de k-liberté selon notre définition est **compatible** avec la k-liberté au sens classique de la définition 2. Dans ce cas, les ensembles d'objets désactivés sont vides lorsqu'il n'y a pas de valeurs manquantes. C'est un point important pour développer des algorithmes qui travaillent indifféremment sur les contextes complets ou incomplets.

La k-liberté dans la base incomplète est reliée à la k-liberté dans la base complète grâce à l'important théorème suivant :

Théorème 1 (Correction de la k-liberté)
Soit r' une base de donnée incomplète et $mv()$ un opérateur de modélisation de valeurs manquantes. Pour toute base complète r telle que $mv(r) = r'$ et tout motif Z,
- $kLibre(Z, r') \Longrightarrow kLibre(Z, r)$;
- $kDepdt(Z, r') \Longrightarrow \neg kLibre(Z, r)$.

Les motifs k-libres de r' sont k-corrects.

Preuve : La propriété 1 montre que $\mathcal{F}(X\overline{Y}, r)$ est bornée par $\mathcal{F}(X\overline{Y}, r') - |\mathcal{DES}(\wedge Y, r'_X)|$ et $\mathcal{F}(X\overline{Y}, r') + |\mathcal{DES}(X, r')|$. Si la borne inférieure est strictement positive, $\mathcal{F}(X\overline{Y}, r)$ est alors strictement positive donc non nulle et le motif est k-libre dans r. De même, si la borne supérieure de $\mathcal{F}(X\overline{Y}, r)$ est nulle, $\mathcal{F}(X\overline{Y}, r)$ est nulle et le motif n'est pas k-libre dans r. □

Les motifs k-libres calculés dans une base incomplète avec la définition 7 sont donc k-corrects, c'est-à-dire qu'ils sont k-libres dans toute complétion. Dans (Rioult & Crémilleux, 2003; Rioult & Crémilleux, 2004), cette correction est montrée dans le cas particulier où $k = 1$, à l'aide de considérations sur la constitution des fermetures au sens de GALOIS.

Ces définitions de la k-liberté et de la k-dépendance permettent de déterminer des propriétés vraies dans toute complétion. En cela, nous parlons de **correction** de nos

définitions. Elles sont également *complètes* dans la mesure à elles caractérisent chaque motif k-libre dans toute complétion :

Théorème 2 (Complétude de la k-liberté)
Soit r' une base de données incomplète. Si Z est k-libre dans toute base complète r telle qu'il existe un opérateur de modélisation des valeurs manquantes $mv()$ avec $mv(r) = r'$, alors Z est k-libre dans r'.

Preuve : Par l'absurde, soit Z k-libre dans toute r telle que $mv(r) = r'$ et non k-libre dans r'. $\exists XY = Z \mid \mathcal{F}(X\overline{Y}, r') - Des(\wedge Y, r'_X) \leq 0$. Soit r_0 la base de données déduite de r' en replaçant toutes les valeurs manquantes par une valeur absente, soit $mv(r_0) = r'$. Dans r_0, la désactivation est nulle car r_0 est complète, et le calcul de $\mathcal{F}(X\overline{Y}, r_0)$ donne le même résultat que dans r' où il est effectué avec les fréquences des attributs présents. $\mathcal{F}(X\overline{Y}, r_0)$ est donc nulle et Z n'est pas k-libre dans r_0 : contradiction. □

Dans une base incomplète, tous les motifs k-libres découverts sont k-corrects et chaque motif k-libre dans toute complétion de cette base est couvert par cette définition.

3.4 Propriétés de la k-liberté dans les bases incomplètes

Les notions de k-liberté et de k-dépendance ne sont pas complémentaires : certains motifs ne seront ni k-libres ni k-dépendants car il est parfois impossible de décider s'ils sont présents ou pas dans un objet. La table ci-dessous détaille le calcul de la 1-liberté pour le motif a_4a_7. Ce motif n'est ni 1-libre, ni 1-dépendant.

X	Y	$\mathcal{F}(X\overline{Y}, mv(r))$	$\lvert\mathcal{DES}(\wedge Y, mv(r)_X)\rvert$	$\lvert\mathcal{DES}(X, mv(r))\rvert$	1-libre ?	1-dépendant ?
a_4	a_7	1	1	1	$1-1 \not> 0$: non	$1+1 \neq 0$: non
a_7	a_4	1	1	1	$1-1 \not> 0$: non	$1+1 \neq 0$: non

Nous donnons maintenant une propriété capitale pour le développement d'algorithmes d'extraction des motifs k-libres. Elle concerne l'(anti)monotonie des nouvelles définitions pour la k-liberté. *A priori*, la k-liberté n'a pas de bonne propriété. Ce n'est pas le cas pour la k-dépendance :

Théorème 3 (Monotonie de la k-dépendance)
La propriété de k-dépendance est monotone, i.e. pour tout motif Z et toute base de données r', $Z \subseteq Z' \Rightarrow (kDepdt(Z, r') \Rightarrow kDepdt(Z', r'))$

Preuve : Soit Z un motif k-dépendant. Alors $\exists XY = Z$, $\mathcal{F}(X\overline{Y}, mv(r)) + \lvert\mathcal{DES}(X, mv(r))\rvert = 0$ soit $\mathcal{F}(X\overline{Y}, mv(r)) = 0$ et $\lvert\mathcal{DES}(X, mv(r))\rvert = 0$. $\mathcal{F}(X\overline{Y}, mv(r)) = 0$ signifie que pour tout objet $o \in \mathcal{O}$, $X \subseteq o \Rightarrow Y \cap o \neq \emptyset$. *A fortiori*, $X \subseteq o \Rightarrow aY \cap o \neq \emptyset$ pour tout $a \in \mathcal{A}$, donc $\mathcal{F}(X\overline{aY}, mv(r)) = 0$. Par induction sur tous les attributs de $Z' \backslash Z$, on déduit que Z' est également k-dépendant. □

Grâce à ce résultat, le cadre des algorithmes par niveaux d'extraction sous contrainte antimonotone est exploitable, en utilisant la négation de la contrainte de k-dépendance. Nous avons réalisé un algorithme appelé `MV-k-miner`, qui extrait par niveaux des motifs qui ne sont pas k-dépendants mais ne présente finalement à l'utilisateur que les motifs k-libres. Il est détaillé par l'algorithme 1 d'extraction, un classique parcours par niveau de l'espace de recherche, et par l'algorithme 2 de génération des candidats.

Données : une base incomplète $mv(r)$, une fréquence minimum γ et k un entier naturel (fixe la profondeur des règles)
Résultat : l'ensemble $\mathcal{S}$ des motifs vérifiant $kLibre$
$\mathcal{D}_l$ est l'ensemble des motifs de longueur l, k-dépendants ou non fréquents ;
$l = 1$; initialiser $\mathcal{C}and_1$ avec la liste des singletons ;
répéter
  $\mathcal{D}_l = \{X \in \mathcal{C}and_l \; t.q. \; kDepdt(X, mv(r)) \vee \neg frequent(X, mv(r))\}$;
  $\mathcal{S}_l = \{X \in \mathcal{C}and_l \backslash \mathcal{D}_l \mid kLibre(X, mv(r))\}$;
  générer les candidats dans $\mathcal{C}and_{l+1}$ (cf. algorithme 2);
  l = l + 1 ;
jusqu'à $\mathcal{C}and_l = \emptyset$;
retourner $\mathcal{S} = \bigcup_l \mathcal{S}_l$;

Algorithme 1 – `MV-k-miner`- extraction de motifs k-libres.

Données : un ensemble $\mathcal{S}_l$ de motifs k-libres de longueur l
Résultat : l'ensemble $\mathcal{C}and_{l+1}$ des motifs candidats à la vérification de $kDepdt$
pour *chaque candidat Z, généré par fusion de deux k-libres ayant un préfixe commun de longueur $l-1$* **faire**
  début
    vérifier que les $Z' \subsetneq Z$ de longueur $|Z| - 1$ sont k-libres ;
    construire l'arbre des motifs X et leurs fréquences tels que $|Z \backslash X| \leq k$;
    pour chaque X de l'arbre, calculer la somme alternée des fréquences de tous ses sur-ensembles, qui constitue une version préliminaire $\sigma(X, Y) = \sum_{\emptyset \subseteq J \subsetneq Y} (-1)^{|J|} \mathcal{F}(XJ)$ de la borne de $\mathcal{F}(Z)$;
    calculer $\sigma(X, Y) - |\mathcal{DES}(\wedge Y, mv(r_X))|$ et $\sigma(X, Y) + |\mathcal{DES}(X, mv(r))|$;
    mémoriser le minimum des bornes $\sigma(X, Y) + |\mathcal{DES}(X, mv(r))|$ pour $\mathcal{F}(Z)$;
    en cas d'égalité des bornes, refuser le candidat ;
  fin
fin

Algorithme 2 – Génération des candidats de longueur $l + 1$.

4 Expérimentations sur des données de l'UCI

Nous avons reproduit les expériences de la section 2.2 (introduction de valeurs manquantes dans des bases r de l'UCI selon une loi uniforme) et mesuré, proportionnellement à la base complète, le nombre de motifs 3-libres obtenus avec `MV-k-miner` dans la base incomplète $mv(r)$. Les résultats sont reportés à la figure 3. Pour des raisons de place, seuls les graphiques correspondant aux bases `solar-flare` et `zoo` sont indiqués. Cependant, les résultats mesurés dans d'autres bases confirment les tendances sur ces exemples. Le temps pour une extraction est de l'ordre d'une dizaine de secondes.

Comme prévu, notre méthode retrouve une quantité faible de motifs en fonction du nombre de valeurs manquantes. En effet, chaque motif extrait est k-correct dans toute complétion possible de la base de calcul et il y a un nombre de complétions exponen-

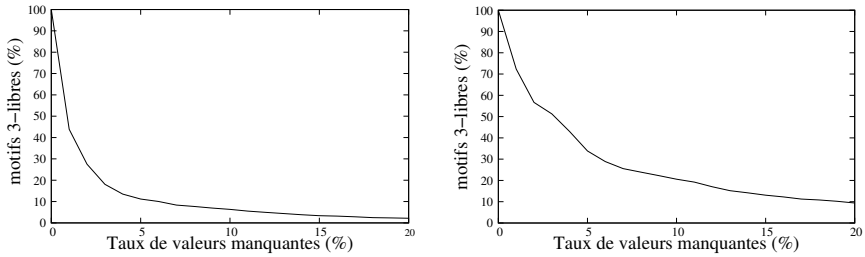

FIG. 3 – Proportion des 3-libres extraits dans la base incomplète. La base complète détermine la référence des 100 %.

tiel en nombre de valeurs manquantes. Pour autant, `MV-k-miner` ne calcule que des motifs k-corrects. Dans un contexte de fouille de données qui produit généralement de grandes quantités de motifs, cet argument est déterminant car il assure que chaque motif découvert serait également découvert dans toute complétion de la base disponible. Les dégâts induits par les valeurs manquantes sont ainsi évités et ce résultat ouvre la voie aux usages des motifs k-libres cités à la section 2.1.

5 Conclusion

Nous avons proposé une définition pour la propriété de k-liberté dans une base de données incomplète. Les motifs extraits selon nos préconisation sont k-corrects dans toute complétion de la base de calcul, ce qui permet d'éviter les dégâts causés par les valeurs manquantes. Nos perspectives concernent d'une part la contribution au large domaine de la classification supervisée, d'autre part la combinaison de cette technique d'extraction avec une méthode de décision utilisant des règles généralisées, dans le but de compléter les valeurs manquantes.

Références

AGRAWAL R. & SRIKANT R. (1994). Fast algorithms for mining association rules. In *Intl. Conference on Very Large Data Bases (VLDB'94), Santiago de Chile, Chile*, p. 487–499.

ANTONIE M.-L. & ZAÏANE O. (2004a). An associative classifier based on positive and negative rules. In *ACM SIGMOD Workshop on Research Issues in Data Mining and Knowledge Discovery (DMKD'04), Paris, France*.

ANTONIE M.-L. & ZAÏANE O. (2004b). Mining positive and negative association rules : An approach for confined rules. In *Conference on Principles and Practice of Knowledge Discovery in Databases (PKDD'04), Pisa, Italy*, p. 27–38.

BASTIDE Y., TAOUIL R., PASQUIER N., STUMME G. & LAKHAL L. (2000). Mining minimal non-redundant association rules using frequent closed itemsets. In *International Conference on Deductive and Object Databases (DOOD'00)*, p. 972–986.

BLAKE C. & MERZ C. (1998). UCI repository of machine learning databases.

BOSC P., CHOLVY L., DUBOIS D., MOUADDIB N., PIVERT O., PRADE H., RASCHIA G. & ROUSSET M.-C. (2002). Les informations incomplètes dans les bases de données et en intelligence artificielle. In *Actes des 2è assises nationales du GRD i3*.

BYKOWSKI A. & RIGOTTI C. (2001). A condensed representation to find frequent patterns. In *ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems, Santa Barbara, USA*, p. 267–273.

CALDERS T. & GOETHALS B. (2002). Mining all non-derivable frequent itemsets. In *Conference on Principles and Practice of Knowledge Discovery in Databases (PKDD'02), Helsinki, Finland*.

CALDERS T. & GOETHALS B. (2003). Minimal k-free representations of frequent sets. In *Conference on Principles and Practice of Knowledge Discovery in Databases (PKDD'03), Cavtat-Dubrovnik, Croatia*, p. 71–82.

DEMETROVICS J. & THI V. (1995). Some remarks on generating armstrong and inferring functionnal dependencies relation. *Acta Cybernetica*, **12**(2), 167–180.

DYRESON C. E. (1997). *Uncertainty Management in Information Systems*, chapter A Bibliography on Uncertainty Management in Information Systems. Kluwer Academic Publishers.

GUNOPULOS D., MANNILA H., KHARDON R. & TOIVONEN H. (1997). Data mining, hypergraph transversals, and machine learning. In *ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems (PODS'97), Tucson, USA*.

JAMI S., JEN T., LAURENT D., LOIZOU G. & SY O. (2004). Extraction de règles d'association pour la prédiction de valeurs manquantes. In *Colloque Africain sur la Recherche en Informatique (CARI)*.

JAROSZEWICZ S. & SIMOVICI D. (2002). Support approximations using bonferroni-type inequalities. In *Principles of Data Mining and Knowledge Discovery (PKDD'02), Helsinki, Finland*, p. 212–224.

LEVENE M. & LOIZOU G. (1999). Database design for incomplete relations. *ACM Transactions on Database Systems*, **24**(1), 80–126.

MANNILA H. & RÄIHÄ K.-J. (1986). Inclusion dependencies : Application to logical database tuning. In *International Conference on Data Engineering, Los Angeles, USA*.

MANNILA H. & TOIVONEN H. (1997). Levelwise search and borders of theories in knowledge discovery. *Data Mining and Knowledge Discovery*, **1**(3), 241–258.

NAYAK J. & COOK D. (2001). Approximate association rule mining. In *Florida Artificial Intelligence Research Symposium, Key West, Florida, USA*, p. 259–263.

RAGEL A. & CRÉMILLEUX B. (1999). Mvc - a preprocessing method to deal with missing values. *Knowledge-Based Systems*, **12**(5-6), 285–291.

RIOULT F. (2005). *Extraction de connaissances dans les bases de données comportant des valeurs manquantes ou un grand nombre d'attributs*. PhD thesis, Université de Caen Basse-Normandie, France.

RIOULT F. & CRÉMILLEUX B. (2003). Condensed representations in presence of missing values. In *Symposium on Intelligent Data Analysis, Berlin, Germany*, p. 578–588.

RIOULT F. & CRÉMILLEUX B. (2004). Représentation condensée en présence de valeurs manquantes. In *XXIIè congrès Inforsid, Biarritz, France*, p. 301–317.

ZAKI M. (2000). Generating non-redundant association rules. In *ACM SIGKDD international conference on Knowledge discovery and data mining, Boston, USA*, p. 34–43.