Extraction de règles d'association pour la prédiction de valeurs manquantes

Sylvie Jami[1]— Tao-Yan Jen[2]— Dominique Laurent[2]—
George Loizou[1]— Oumar Sy[2,3]

1. Birkbeck College - University of London
London - United Kingdom
{s.jami, george}@dcs.bbk.ac.uk
2. LICP - Université de Cergy-Pontoise Cergy-Pontoise - France
{tao-yuan.jen, dominique.laurent}@dept-info.u-cergy.fr
3. UFR Sciences Appliquées - Université Gaston Berger
Saint Louis - Sénégal
oumar.sy@ugb.sn

RÉSUMÉ. La présence de valeurs manquantes ou valeurs nulles dans les bases de donnnées a suscité de nombreuses recherches dans le domaine de la découverte des connaissances, notamment en ce qui concerne la prédiction. Cependant, à notre connaissance, peu de telles approches utilisent les règles d'association pour la prédiction des valeurs manquantes. Dans cet article, il est montré comment adapter les différents concepts et algorithmes par niveau liés aux règles d'association, afin d'obtenir des règles fréquentes et de confiance 1, permettant la prédiction de valeurs manquantes dans une table relationnelle. La particularité des règles extraites dans notre approche est que leurs conséquents se présentent sous la forme d'intervalles ou d'ensembles de valeurs, selon que le domaine de l'attribut sur lequel les valeurs sont prédites est soit continu soit discret.

ABSTRACT. Missing values in databases have motivated many researches in the field of KDD, specially concerning prediction. However, to the best of our knowledge, few appraoches based on association rules have been proposed so far. In this paper, we show how to adapt the levelwise algorithm for the mining of association rules in order to mine frequent rules with a confidence equal to 1 from a relational table. In our approach, the consequents of extracted rules are either an interval or a set of values, according to whether the domain of the predicted attribute is continuous or discrete.

MOTS-CLÉS : Bases de données, valeurs manquantes, règles d'association, prédiction.

KEYWORDS : Databases, missing values, association rules, prediction.

Novembre 2005, Numéro spécial CARI'04

Revue ARIMA **pages 103 à 124**

1. Introduction

La présence de valeurs manquantes ou valeurs nulles dans les bases de données a suscité de nombreuses recherches dans le domaine de la découverte de connaissances. Parmi ces travaux, de nombreuses approches ont été proposées dans le cadre de la prédiction. Les principales techniques utilisées par la prédiction sont la régression, les arbres de décision et les réseaux de neurones (voir [4] pour une synthèse de ces travaux).

La régression permet de faire la prédiction de valeurs continues ou de valeurs ordonnées, et utilise des méthodes statistiques parmi lesquelles on peut noter la régression linéaire, la régression multiple et les modèles logarithmiques linéaires [4]. Ces méthodes ont le défaut de ne pouvoir traiter que des valeurs numériques, alors que dans la majorité des cas pratiques, les données peuvent également être symboliques.

Appliqués à la prédiction, les algorithmes basés sur les arbres de décision tels que *C4.5* ([11]) permettent de prédire les valeurs d'un attribut discret ou continu, mais avec un succès limité notamment dans le cas de gros volumes de données [6]. De plus, dans le cas de données numériques, celles-ci doivent être discrétisées au préalable.

Dérivés de l'intelligence artificielle, les réseaux de neurones utilisent une technique de régression généralisée et permettent un apprentissage non-supervisé qui peut être utilisé à des fins de prédiction. Toutefois, contrairement aux techniques utilisant les arbres de décision, les réseaux de neurones n'offrent pas une bonne représentation des connaissances apprises, ce qui limite leur application dans le cadre de la prédiction ([4]).

D'autres algorithmes comme *KID3* ([10]) ont été proposés dans le but d'extraire des règles. Cependant comme *Apriori* et *C4.5*, ces algorithmes ne peuvent être appliqués qu'après discrétisation de l'attribut prédit lorsque celui-ci est continu [6].

On notera d'autre part que dans [12, 13, 18], l'extraction de règles d'association en présence de valeurs manquantes est étudiée. Alors que l'objectif de [12] est d'adapter les définitions de support et de confiance afin de prendre en compte le fait que les données sont incomplètes, dans [13, 18], les règles d'association sont utilisées à des fins de prédiction. Dans ces travaux, les auteurs utilisent les règles d'association pour remplacer chaque valeur manquante par une seule valeur. Ces méthodes donnent lieu à des conflits, puisque en général, pour un cas donné de prédiction, plusieurs règles ayant des conséquents différents peuvent être appliquées. Dans [13], ce problème est traité grâce à l'intervention de l'utilisateur, qui est censé pouvoir résoudre au moins partiellement de tels conflits, alors que dans [18], des mesures autres que le support et la confiance, telles que le lift, sont utilisées.

L'approche présentée dans cet article, initialement introduite dans [6, 7] a pour but d'extraire des règles permettant la prédiction de valeurs manquantes dans les cas où l'attribut sur lequel la prédiction est faite est soit continu soit discret ; de plus, dans le premier

cas, aucune discrétisation *a priori* du domaine de l'attribut prédit n'est nécessaire. Il est également important de noter que notre approche se distingue de celles de [13] et de [18], car les cas de conflits évoqués ci-dessus ne peuvent apparaître. Cet article présente cette approche et montre que, contrairement à ce qui est fait dans [6, 7], les règles permettant la prédiction peuvent être extraites en utilisant un algorithme par niveau de type *Apriori* [1].

Étant donné une table R définie sur un schéma relationnel $\{A_1, A_2, ..., A_n\}$ et contenant des valeurs manquantes, la partie de données complètes de R (*i.e.*, l'ensemble des n-uplets de R ne contenant aucune valeur manquante), notée $\overline{R}$, est utilisée afin de calculer des règles d'association de la forme

$$\rho : (A_{i_1} = v_1, A_{i_2} = v_2, ..., A_{i_k} = v_k) \Rightarrow (A_{i_0} \in E)$$

telles que :

1) l'attribut A_{i_0} est fixé, et appelé attribut prédit,

2) E est soit un intervalle soit un ensemble de valeurs selon que l'attribut prédit A_{i_0} est de type continu ou discret,

3) la partie gauche de ρ est fréquente dans $\overline{R}$ (*i.e.*, la proportion dans $\overline{R}$ de n-uplets dont les valeurs sont égales aux v_i apparaissant dans la partie gauche de ρ est supérieure à un seuil donné), et

4) la confiance de ρ est 1 (*i.e.*, la règle est satisfaite dans $\overline{R}$).

Les règles ainsi obtenues peuvent alors être utilisées pour la prédiction de valeurs manquantes sur l'attribut A_{i_0} dans R.

Le but de notre méthode consiste ainsi à extraire de telles règles qui soient valides dans la table considérée de façon à ce que la partie gauche soit suffisamment fréquente et à ce que l'intervalle (ou ensemble de valeurs) associé soit le plus petit possible. L'exemple suivant illustre notre approche et sera repris dans la suite de l'article.

Exemple 1 *Considérons la relation appelée R et représentée par la table 1 (a). Cette relation contient des résultats d'analyses d'échantillons de semences de riz, un échantillon de semence étant décrit par un numéro d'analyse (TID), la variété (VAR) et la catégorie (CAT) de riz auxquelles il appartient. De plus, un échantillon possède une certaine pureté spécifique (PURETE) (deux échantillons de même variété et même catégorie peuvent avoir la même pureté spécifique ou des puretés différentes), et une certaine faculté germinative (GERM).*

Dans la table 1 (a) ci-dessous, certaines valeurs sont inconnues, en particulier sur l'attribut GERM. Afin de prédire ces valeurs manquantes, la table complète 1 (b) est utilisée pour extraire des règles d'association afin de pouvoir prédire les valeurs de GERM à partir de valeurs sur un ou plusieurs des autres attributs, à savoir TID, VAR, CAT, ou PURETE.

Pour un seuil de support de 0.14 (correspondant à 2 lignes dans la table 1(b)), les règles suivantes ont toutes une confiance de 1 et leurs parties gauches ont toutes un support supérieur ou égal à 0.14.

1) $\rho_1 : (\mathit{VAR} = \text{JAYA}) \Rightarrow (\mathit{GERM} \in [90, 95])$

2) $\rho_2 : (\mathit{CAT} = \text{BASE}) \Rightarrow (\mathit{GERM} \in [90, 93])$

3) $\rho_3 : (\mathit{CAT} = \text{CERT R1}) \Rightarrow (\mathit{GERM} \in [90, 95])$

4) $\rho_4 : (\mathit{VAR} = \text{IR1529}) \Rightarrow (\mathit{GERM} \in [90, 98])$

5) $\rho_5 : (\mathit{VAR} = \text{IKP}) \Rightarrow (\mathit{GERM} \in [90, 93])$

6) $\rho_6 : (\mathit{VAR} = \text{IKP}, \ \mathit{CAT} = \text{BASE}) \Rightarrow (\mathit{GERM} \in [90, 93])$

7) $\rho_7 : (\mathit{VAR} = \text{JAYA}, \ \mathit{CAT} = \text{BASE}) \Rightarrow (\mathit{GERM} \in [90, 93])$

8) $\rho_8 : (\mathit{VAR} = \text{JAYA}, \ \mathit{CAT} = \text{CERT R1}) \Rightarrow (\mathit{GERM} \in [92, 95])$

Intuitivement, la règle ρ_1 est interprétée comme suit : à partir de la table $\overline{R}$, les données concernant la variété JAYA *sont suffisamment fréquentes pour conclure, avec une confiance de 1, que la valeur correspondante pour l'attribut GERM est comprise entre 90 et 95.*

De manière similaire, la règle ρ_8 indique que les données concernant à la fois la variété JAYA *et la catégorie* CERT R1 *sont suffisamment fréquentes dans $\overline{R}$ pour conclure, avec une confiance de 1, que la valeur correspondante pour l'attribut GERM est comprise entre 92 et 95.*

Il est important de remarquer, à partir de l'exemple ci-dessus, que toutes les règles obtenues ne présentent pas un intérêt pour l'utilisateur. Par exemple, la règle ρ_4 peut être considérée comme inutile, puisque l'intervalle associé contient toutes les valeurs possibles qui peuvent être présentes pour l'attribut prédit *GERM* dans la table $\overline{R}$.

De même, si l'on considère les règles ρ_2, ρ_5 et ρ_6, on constate d'une part, que leurs parties droites sont identiques et, d'autre part, que ρ_6 est plus spécifique que ρ_2 et que ρ_5. En d'autres termes, ρ_6 autorise moins de cas de prédiction que ρ_2 et que ρ_5, sans pour autant améliorer la qualité de ces prédictions. Dans le cas où les règles ρ_2 et ρ_5 seraient retenues à des fins de prédiction, la règle ρ_6 n'a donc pas à être séléctionnée.

Afin de prendre en compte les remarques ci-dessus, une mesure spécifique, appelée *gain de précision*, est utilisée en plus des mesures habituelles de support et de confiance. Intuitivement, le gain de précision entre deux règles ρ_1 et ρ_2 telles que ρ_2 est plus spécifique que ρ_1 permet de ne retenir ρ_2 que si celle-ci donne une prédiction plus précise que ρ_1, *i.e.*, une réduction significative de l'intervalle ou de l'ensemble de prédiction. Si les intervalles ou ensembles prédits respectivement par ρ_1 et ρ_2 sont notés E_1 et E_2, le gain de précision entre ρ_1 et ρ_2 est défini comme suit :

TID	VAR	CAT	PURETE	GERM
1	JAYA	BASE	BONNE	93
2	JAYA	BASE	BONNE	?
3	JAYA	BASE	BONNE	90
4	JAYA	PBASE	FAIBLE	95
5	JAYA	CERT R1	MOYENNE	93
6	JAYA	CERT R1	MOYENNE	95
7	JAYA	CERT R2	BONNE	95
8	IR1529	PBASE	MOYENNE	95
9	IR1529	PBASE	MOYENNE	98
10	IR1529	CERT R1	BONNE	90
11	IKP	BASE	MOYENNE	?
12	IKP	BASE	BONNE	90
13	IKP	BASE	BONNE	93
14	JAYA	CERT R1	MOYENNE	?
15	JAYA	CERT R1	BONNE	92
16	JAYA	CERT R1	?	93
17	JAYA	CERT R2	?	?
18	IR1529	CERT R1	MOYENNE	92
19	JAYA	BASE	MOYENNE	92

(a) La table R

TID	VAR	CAT	PURETE	GERM
1	JAYA	BASE	BONNE	93
3	JAYA	BASE	BONNE	90
4	JAYA	PBASE	FAIBLE	95
5	JAYA	CERT R1	MOYENNE	93
6	JAYA	CERT R1	MOYENNE	95
7	JAYA	CERT R2	BONNE	95
8	IR1529	PBASE	MOYENNE	95
9	IR1529	PBASE	MOYENNE	98
10	IR1529	CERT R1	BONNE	90
12	IKP	BASE	BONNE	90
13	IKP	BASE	BONNE	93
15	JAYA	CERT R1	BONNE	92
18	IR1529	CERT R1	MOYENNE	92
19	JAYA	BASE	MOYENNE	92

(b) La table $\overline{R}$

Tableau 1. *(a) Données incomplètes ; (b) Données complètes*

$$gain(\rho_1, \rho_2) = \frac{|E_1| - |E_2|}{|E_1|}$$

où $|Y|$ désigne la taille de Y, *i.e.,* soit la cardinalité de *Y* si A_{i_0} est de type discret, soit la longueur de l'intervalle *Y* si A_{i_0} est de type continu. Le gain de précision ainsi calculé mesure la réduction de la taille de E_2 relativement à celle de E_1.

On notera que dans le cas où on considère une règle dont la partie gauche comporte une seule condition, le gain de précision correspondant est calculé en fonction de la taille du *domaine actif* de A_{i_0} dans $\overline{R}$, noté $adom(A_{i_0})$. Comme dans [14, 8, 17], on appelle domaine actif de A_{i_0} dans $\overline{R}$ l'ensemble des valeurs sur A_{i_0} présentes dans $\overline{R}$. De plus, dans la suite de l'article, $|adom(A_{i_0})|$ désigne soit la cardinalité de $adom(A_{i_0})$, si A_{i_0} est de type discret, soit la taille du plus petit intervalle contenant $adom(A_{i_0})$, si A_{i_0} est de type continu. Ainsi, dans ce cas, le gain de précision est exprimé par :

$$gain(\emptyset, \rho_2) = \frac{|adom(A_{i_0})| - |E_2|}{|adom(A_{i_0})|}$$

D'autre part, il est montré dans cet article que la confiance des règles extraites est toujours 1. La qualité de ces règles est par conséquent évaluée selon deux seuils minimaux spécifiés par l'utilisateur : un seuil de support et un seuil de gain de précision.

Exemple 2 *Dans l'exemple 1, les valeurs du domaine actif de GERM appartiennent à l'intervalle* $[90, 98]$. *Donc, si on considère un seuil de gain de précision égal à 0.15, pour la règle* ρ_1 *:* $(VAR = \text{JAYA}) \Rightarrow (GERM \in [90, 95])$, *on obtient :*

$$\frac{|98 - 90| - |95 - 90|}{|98 - 90|} = 0.375$$

Ce quotient étant supérieur à 0.15, la règle est retenue. Les règles ρ_2 *et* ρ_4 *donnent lieu à des calculs similaires et il est facile de voir que :*

– $\rho_2 : (CAT = \text{BASE}) \Rightarrow (GERM \in [90, 93])$ *donne lieu à un gain de précision égal à 0.625. Par conséquent, cette règle est retenue.*

– $\rho_4 : (VAR = \text{IR1529}) \Rightarrow (GERM \in [90, 98])$ *donne lieu à un gain de précision égal à* 0. *Par conséquent, cette règle n'est pas retenue, bien que le support soit suffisant.*

Pour la règle $\rho_7 : (VAR = \text{JAYA}, CAT = \text{BASE}) \Rightarrow (GERM \in [90, 93])$ *on a alors*

$$gain(\rho_1, \rho_7) = \frac{|95 - 90| - |93 - 90|}{|95 - 90|} = 0.4$$

Par conséquent, cette règle permet d'affiner la prédiction faite par ρ_1 *au delà du seuil de* 0.15, *et donc peut être retenue.*

Néanmoins, si on calcule le gain de précision de ρ_7 relativement à ρ_2, i.e., $gain(\rho_2, \rho_7)$, on trouve un gain nul. Ceci signifie que ρ_7 n'apporte aucune précision supplémentaire par rapport à ρ_2, et donc, il n'y a aucune raison de retenir ρ_7.

D'autre part, pour la règle ρ_8 : (VAR $=$ JAYA, CAT $=$ CERT R1) $\Rightarrow$ (GERM $\in [92, 95]$), on a $gain(\rho_1, \rho_8) = gain(\rho_3, \rho_8) = 0.4$. Cette règle, apportant une meilleure précision que ρ_1 et ρ_3, sera donc retenue.

Ainsi, de manière générale, pour s'assurer qu'une règle apporte effectivement une réduction de la taille de l'ensemble prédit, tous les ordres possibles d'écriture des conditions élémentaires de la partie gauche de la règle doivent être considérés. Par conséquent, les tests pour savoir si une règle candidate dont le corps contient *n* conditions de la forme $A_{i_j} = v_j$ peut être retenue nécessitent de calculer $n * n!$ gains de précision.

Un tel critère engendre par conséquent une explosion combinatoire des calculs de gain. De plus, comme il a été vu dans [6], son application nécessite un parcours en profondeur de l'espace de recherche, empêchant ainsi l'utilisation d'un algorithme standard par niveau de type *Apriori* ([1]) pour l'extraction des fréquents.

Il est montré dans cet article que l'extraction des règles peut être effectuée selon les critères ci-dessus, tout en appliquant un algorithme standard par niveau de type *Apriori*, évitant en particulier l'explosion combinatoire évoquée ci-dessus concernant des calculs de gain de précision.

L'article est organisé comme suit : Dans la section 2, les notations et définitions nécessaires, ainsi que la forme des règles extraites sont introduites. De plus, il est montré comment les mesures de support et de confiance s'appliquent à ces règles. La section 3 présente la mesure utilisée pour l'évaluation des règles, ainsi que la propriété fondamentale de cette mesure. Les algorithmes de notre méthode sont donnés à la section 4. La section 5 présente notre proposition pour effectuer des prédictions à partir des règles calculées par notre algorithme. De plus, dans cette section, certains résultats expérimentaux obtenus antérieurement dans [6] sont rappelés. La section 6 conclut l'article et propose quelques directions pour de futures recherches envisagées à partir de ces travaux.

2. Règles de prédiction

Dans cette section, sont présentées les définitions de base, la forme des règles extraites, ainsi que les définitions de support et de confiance [1] adaptées à notre approche. Dans la suite de l'article, le lecteur est supposé être familier avec la terminologie du modèle relationnel [14, 8, 17].

En particulier, on considère un ensemble fini d'attributs $U = \{A_1, A_2, \ldots, A_n\}$ et on suppose qu'à chaque A_i ($i = 1, 2, \ldots, n$) est associé un ensemble de valeurs, appelé

domaine de A_i et noté $dom(A_i)$. Pour tout $i = 1, 2, \ldots, n$, $dom(A_i)$ est soit un ensemble discret soit un ensemble continu.

Dans notre approche, une *relation* sur U est un ensemble fini de n-uplets d'arité n pouvant éventuellement contenir des valeurs manquantes. On rappelle que différents types de valeurs manquantes ont été étudiés dans le cadre des bases de données relationnelles. Dans [8], ces différents types sont classés en quatre catégories qui sont : (i) la valeur existe mais est actuellement inconnue, (ii) la valeur n'existe pas, (iii) aucune information n'est disponible pour cette valeur (ce qui signifie que l'on ignore si cette valeur existe ou non), et (iv) la valeur est inconsistante.

Dans la mesure où notre approche consiste à prédire des valeurs manquantes, nous supposons que celles-ci sont du type (i) ci-dessus, et nous signifions la présence d'une valeur manquante, inconnue mais qui est supposée exister, par le symbole ' ?'.

Soit R une telle relation sur U, on note $\overline{R}$ la relation sur U obtenue en éliminant de R tous les n-uplets contenant au moins une valeur manquante.

2.1. Définitions et notations

Afin de définir la forme des règles de prédiction considérées dans cet article, on introduit tout d'abord les notions de *condition de prédiction* et d'*ensemble prédit*, qui constituent respectivement les parties gauche et droite des règles extraites.

Définition 1 - Condition de prédiction. *Une* condition élémentaire *est une expression de la forme $A_i = v_i$ où $A_i \in U$ et $v_i \in dom(A_i)$. Un n-uplet t sur U satisfait la condition élémentaire $A_i = v_i$, noté $t \models (A_i = v_i)$, si la restriction de t à A_i, notée $t.A_i$, est égale à v_i.*

Une condition de prédiction *(ou simplement* condition*) est soit une condition élémentaire soit une conjonction de conditions élémentaires de la forme $(A_{i_1} = v_1 \wedge A_{i_2} = v_2 \wedge \ldots \wedge A_{i_k} = v_k)$ telle que, si j et j' sont deux entiers distincts de $\{1, \ldots, k\}$, alors $A_{i_j} \neq A_{i_{j'}}$.*

Soit Γ une condition de prédiction. Un n-uplet t sur U satisfait Γ, noté $t \models \Gamma$, si t satisfait chacune des conditions élémentaires de Γ.

Afin d'alléger les notations, toute condition de prédiction de la forme $(A_{i_1} = v_1 \wedge A_{i_2} = v_2 \wedge \ldots \wedge A_{i_k} = v_k)$ sera plus simplement notée $(A_{i_1} = v_1, A_{i_2} = v_2, \ldots, A_{i_k} = v_k)$.

De plus, une condition est assimilée à l'*ensemble* des conditions élémentaires qui la constituent et les notations ensemblistes (union, inclusion, appartenance d'une condition élémentaire à une condition) sont également utlisées à ce propos. Ainsi, dans le cadre de l'exemple 1, $\Gamma = (VAR = \text{JAYA}, CAT = \text{BASE})$ est une condition constituée des deux conditions élémentaires $VAR = \text{JAYA}$ et $CAT = \text{BASE}$. On pourra donc écrire des formules telles que $VAR = \text{JAYA} \in \Gamma$ ou encore $\Gamma = (VAR = \text{JAYA}) \cup (CAT = \text{BASE})$.

Afin de définir la notion d'*ensemble prédit,* on définit le *domaine actif* d'un attribut A de U, noté $adom(A)$, comme étant :

– soit l'ensemble des valeurs de $dom(A)$ présentes dans $\overline{R}$, si $dom(A)$ est de type discret,

– soit l'intervalle $[\mu, \nu]$ où μ et ν sont respectivement les plus petite et plus grande valeurs de $dom(A)$ présentes dans $\overline{R}$.

Définition 2 - Ensemble prédit. *Soit A_i un attribut de U. Un* ensemble prédit *sur A_i est*

– une partie de $adom(A_i)$ si $dom(A_i)$ est discret, ou

– un intervalle de $adom(A_i)$ si $dom(A_i)$ est continu.

Si E est un ensemble prédit sur A_i et t un n-uplet sur U, on note $t \models E$ le fait que la restriction de t à A_i appartient à E, i.e., $t.A_i \in E$.

Dans toute la suite de l'article, on suppose fixé l'attribut sur lequel les ensembles prédits sont calculés ; cet attribut est noté A_{i_0}. La forme des règles extraites dans notre approche est définie comme suit.

Définition 3 - Règle de prédiction. *On appelle* règle de prédiction *ou simplement* règle, *toute règle de la forme $(\Gamma \Rightarrow A_{i_0} \in E_\Gamma)$, notée $\langle \Gamma, E_\Gamma \rangle$, où :*

– Γ est une condition de prédiction dans laquelle A_{i_0} n'a aucune occurrence.

– E_Γ est l'ensemble prédit défini par $E_\Gamma = \{v \in dom(A_{i_0}) \mid (\exists t \in \overline{R})(t \models \Gamma$ et $t.A_{i_0} = v)\}$, si $dom(A_{i_0})$ est discret, ou, si $dom(A_{i_0})$ est continu, par $E_\Gamma = [\mu_\Gamma, \nu_\Gamma]$ avec

- $\mu_\Gamma = \min\{v \in dom(A_{i_0}) \mid (\exists t \in \overline{R})(t \models \Gamma$ et $t.A_{i_0} = v)\}$ et

- $\nu_\Gamma = \max\{v \in dom(A_{i_0}) \mid (\exists t \in \overline{R})(t \models \Gamma$ et $t.A_{i_0} = v)\}$.

Il est important de remarquer que, d'après la définition 3 ci-dessus, E_Γ est le plus petit ensemble ou intervalle tel que :

$$(t \in \overline{R} \wedge t \models \Gamma) \Longrightarrow (t \models E_\Gamma)$$

De plus, pour toute règle de prédiction $\langle \Gamma, E_\Gamma \rangle$, l'ensemble E_Γ est complètement déterminé par la seule connaissance de Γ. Par conséquent, afin d'alléger les notations, lorsque E_Γ n'est pas nécessaire au discours, toute règle de prédiction $\langle \Gamma, E_\Gamma \rangle$ sera notée plus simplement $\langle \Gamma \rangle$.

Exemple 3 *Les définitions ci-dessus sont illustrées dans le cadre de l'exemple 1 de l'introduction. Dans ce cas, l'univers U considéré est constitué des cinq attributs TID, VAR, CAT, PURETE et GERM. Les domaines des quatre premiers sont supposés discrets et celui du dernier continu. Les relations utilisées dans cet exemple sont R et $\overline{R}$ données dans les tables 1 (a) et 1 (b).*

En utilisant la définition 3, la règle $\rho_1 : (VAR = \text{JAYA}) \Rightarrow (GERM \in [90, 95])$ *ou* $\langle VAR = \text{JAYA}, [90, 95]\rangle$ *est une règle de prédiction puisque dans* $\overline{R}$*, la valeur* JAYA *est associée sur l'attribut GERM à des valeurs comprises entre 90 et 95.*

2.2. Support et confiance

Le support d'une condition ou d'une règle est défini, de manière analogue à [1], comme suit.

Définition 4 - Support. *Soit* $\overline{R}$ *une relation sur* U *ne contenant aucune valeur manquante,* Γ *une condition et* $\langle \Gamma, E_\Gamma\rangle$ *une règle.*

– *Le* support de Γ dans $\overline{R}$*, noté* $sup(\Gamma, \overline{R})$*, est le rapport :*

$$\frac{|\{t \mid t \models \Gamma\}|}{|\overline{R}|}$$

– *Le* support de $\langle \Gamma, E_\Gamma\rangle$ dans $\overline{R}$*, noté* $sup(\langle \Gamma, E_\Gamma\rangle, \overline{R})$*, est le rapport :*

$$\frac{|\{t \mid t \models \Gamma \; et \; t \models E_\Gamma\}|}{|\overline{R}|}$$

Étant donné un seuil de support S*, une condition* Γ *(respectivement une règle* $\langle \Gamma, E_\Gamma\rangle$*) est dite* fréquente dans $\overline{R}$ par rapport à S *(ou simplement* fréquente*, si* S *et* $\overline{R}$ *sont fixés) si* $sup(\Gamma, \overline{R}) \geq S$ *(respectivement* $sup(\langle \Gamma, E_\Gamma\rangle, \overline{R}) \geq S$*).*

Comme dans [1], la confiance d'une règle est définie dans notre approche à partir du support de la manière suivante.

Définition 5 - Confiance d'une règle. *Soit* $\overline{R}$ *une relation sur* U *ne contenant aucune valeur manquante, et soit* $\langle \Gamma, E_\Gamma\rangle$ *une règle. La* confiance de $\langle \Gamma, E_\Gamma\rangle$ dans $\overline{R}$*, notée* $conf(\langle \Gamma, E_\Gamma\rangle, \overline{R})$*, est le rapport :* $sup(\langle \Gamma, E_\Gamma\rangle, \overline{R})/sup(\Gamma, \overline{R})$.

Si l'on reprend la règle $\langle VAR = \text{JAYA}, [90, 95]\rangle$ de l'exemple 3 ci-dessus, il est facile de voir que son support dans $\overline{R}$ est égal à $8/14$ soit environ 0.57. Donc, pour un seuil de support de $S = 0.14$, cette règle est fréquente dans $\overline{R}$ par rapport à S. De plus, comme le support de la condition $VAR = \text{JAYA}$ est égal à $8/14$, la confiance de cette règle dans $\overline{R}$ est égale à 1.

Dans la mesure où dans l'article, les relations R et $\overline{R}$ sont *fixées,* les notations pour le support et la confiance sont simplifiées en omettant le second argument, *i.e.,* $sup(\Gamma, \overline{R})$, $sup(\langle \Gamma\rangle, \overline{R})$ et $conf(\langle \Gamma\rangle, \overline{R})$ seront respectivement notés $sup(\Gamma)$, $sup(\langle \Gamma\rangle)$ et $conf(\langle \Gamma\rangle)$.

La proposition suivante énonce deux propriétés concernant les règles de prédiction. La première propriété indique que toute règle de prédiction a une confiance de 1, et la seconde propriété permet de comparer les ensembles prédits de deux règles dont les conditions sont elles-mêmes comparables.

Proposition 1 *Soit* Γ *une condition, alors :*

1) $sup(\langle \Gamma, E_\Gamma \rangle) = sup(\Gamma)$ *et donc* $conf(\langle \Gamma, E_\Gamma \rangle) = 1$.

2) Si Γ' *est une condition telle que* $\Gamma \subseteq \Gamma'$*, alors* $E_{\Gamma'} \subseteq E_\Gamma$.

PREUVE : 1. D'après la définition 3, pour tout n-uplet t de $\overline{R}$, on a : $t \models \Gamma$ si et seulement si $t \models \langle \Gamma, E_\Gamma \rangle$. Le résultat est ainsi une conséquence immédiate des définitions 4 et 5.
2. Soit $F = \{t \in \overline{R} \mid t \models \Gamma\}$ et $F' = \{t \in \overline{R} \mid t \models \Gamma'\}$. Comme $\Gamma \subseteq \Gamma'$, on a $F' \subseteq F$. Donc, d'après la remarque qui suit la définition 3, on obtient $E_{\Gamma'} \subseteq E_\Gamma$. □

3. Gain de précision

Comme il a été souligné dans l'introduction, en plus des mesures de support et de confiance, une mesure spécifique appelée *gain de précision* est utilisée dans notre approche. La sémantique et l'utilisation de cette mesure dans le calcul des règles telles qu'elles ont été définies dans [6, 7] sont respectivement données par les définitions 6 et 7. De plus, la proposition 2, qui constitue l'aspect essentiel de notre contribution dans cet article, montre que le calcul des règles peut être effectué plus efficacement que par la méthode de [6, 7].

3.1. Définitions

Définition 6 - Gain de précision. *Soit* $\langle \Gamma, E_\Gamma \rangle$ *et* $\langle \Gamma', E_{\Gamma'} \rangle$ *deux règles telles que* $\Gamma \subseteq \Gamma'$. *Le* gain de précision *de* $\langle \Gamma', E_{\Gamma'} \rangle$ *par rapport à* $\langle \Gamma, E_\Gamma \rangle$*, noté* $gain(\Gamma, \Gamma')$*, est le quotient défini par :*

$$gain(\Gamma, \Gamma') = \frac{|E_\Gamma| - |E_{\Gamma'}|}{|E_\Gamma|}$$

Dans le cas particulier où Γ *est la condition vide, on définit le gain de précision de* Γ' *par rapport à* Γ *par le quotient :*

$$gain(\emptyset, \Gamma') = \frac{|adom(A_{i_0})| - |E_{\Gamma'}|}{|adom(A_{i_0})|}.$$

On notera que d'après la seconde propriété de la proposition 1 précédente, $gain(\Gamma, \Gamma')$ est toujours positif. Comme il est facile de voir que l'on a toujours $gain(\Gamma, \Gamma') \leq 1$, le gain de précision est un nombre toujours compris entre 0 et 1.

On définit maintenant le critère utilisant la mesure de gain de précision pour sélectionner une règle.

Définition 7 - Règle retenue. *Soit G un seuil de gain et soit $\langle\Gamma\rangle$ une règle telle que $\Gamma = \{\gamma_1, \gamma_2, \ldots, \gamma_k\}$ où, pour $i = 1, 2, \ldots, k$, γ_i est une condition élémentaire.* Γ *est* retenue par rapport à G *si :*

– Lorsque $k = 1$ alors $gain(\emptyset, \Gamma) \geq G$

– Lorsque $k > 1$ alors, pour toute permutation θ de $\{1, 2, \ldots, k\}$, et pour tout $j = 1, 2, \ldots, k-1$, alors $gain(\Gamma_j, \Gamma_j \cup \{\gamma_{\theta(j+1)}\}) \geq G$, avec $\Gamma_j = \{\gamma_{\theta(1)}, \gamma_{\theta(2)}, \ldots, \gamma_{\theta(j)}\}$.

On constate ainsi que ce critère prend en considération les remarques faites dans l'introduction, à savoir que pour s'assurer que la règle apporte effectivement une réduction de la taille de l'ensemble prédit, tous les ordres possibles d'écriture des conditions élémentaires de Γ doivent être considérés. Par conséquent, si Γ contient k conditions élémentaires, le nombre de calculs de gain de précision est $k * k!$. L'exemple suivant illustre cette définition.

Exemple 4 *Dans le cadre de l'exemple 1, considérons de nouveau les règles*

– ρ_1 : (*VAR* = JAYA) $\Rightarrow$ (*GERM* $\in [90, 95]$)

– ρ_2 : (*CAT* = BASE) $\Rightarrow$ (*GERM* $\in [90, 93]$)

– ρ_7 : (*VAR* = JAYA, *CAT* = BASE) $\Rightarrow$ (*GERM* $\in [90, 93]$)

Pour un seuil de gain de précision de 0.15, l'application de la définition 7 ci-dessus nécessite les calculs de gain de précision suivants :

1) $gain(\emptyset, \rho_1)$*. Ce gain étant de 0.375, la règle ρ_1 est retenue*

2) $gain(\rho_1, \rho_7)$*. Ce gain étant de 0.4, on considère tous les ordres possibles d'écriture de la condition de ρ_7 pour calculer les gains suivants*

a) $gain(\emptyset, \rho_2) = 0.625$*, ce qui implique que la règle ρ_2 est retenue*

b) $gain(\rho_2, \rho_7) = 0$

Puisque le second gain ci-dessus est inférieur à 0.15, la règle ρ_7 n'est pas retenue.

On montre par la suite qu'il est possible de réduire significativement le nombre de calculs de gains pour retenir les règles de prédiction.

3.2. Propriété fondamentale

La proposition ci-dessous montre que le nombre de calculs de gains de précision pour une règle donnée, au lieu d'être exponentiel comme indiqué par la définition 7, peut être ramené à la cardinalité de la condition de la règle testée

Proposition 2 *Soit G un seuil de gain de précision et $\langle\Gamma\rangle$ une règle telle que Γ contient au moins deux conditions élémentaires. $\langle\Gamma\rangle$ est retenue par rapport à G si et seulement si pour tout γ de Γ :*

– *$\langle\Gamma \setminus \{\gamma\}\rangle$ est retenue par rapport à G, et*

– *$gain(\Gamma \setminus \{\gamma\}, \Gamma) \geq G$.*

PREUVE : Soit $\Gamma = \{\gamma_1, \ldots, \gamma_n\}$ telle que $\langle\Gamma\rangle$ est retenue par rapport à G. Alors, pour toute condition γ_i de Γ et toute permutation θ de $\{1, \ldots, n\}$ telle que $\theta(i) = n$, on sait que $gain(\Gamma \setminus \{\gamma_i\}, \Gamma) \geq G$. Le second point de la proposition est donc satisfait pour γ_i. Afin de montrer que $\langle\Gamma \setminus \{\gamma_i\}\rangle$ est forcément retenue par rapport à G, on note Γ_i la condition $\Gamma \setminus \{\gamma_i\}$ et on suppose que Γ_i n'est pas retenue par rapport à G. Dans ce cas, d'après la définition 7, il existe une permutation θ' de $\{1, \ldots, n\} \setminus \{i\}$ et un entier j tels que $1 \leq j < n-1$ et $gain(\Gamma_{i_j}, \Gamma_{i_j} \cup \{\gamma_{\theta'(j+1)}\}) < G$, avec $\Gamma_{i_j} = \{\gamma_{\theta'(1)}, \ldots, \gamma_{\theta'(j)}\}$. La permutation θ_0 de $\{1, \ldots, n\}$ définie par : $\theta_0(j) = \theta'(j)$ pour $j \neq i$ et $\theta_0(i) = n$ montre alors une contradiction avec notre hypothèse selon laquelle $\langle\Gamma\rangle$ est retenue par rapport à G. Par conséquent, $\langle\Gamma \setminus \{\gamma_i\}\rangle$ est retenue par rapport à G, et donc le premier point de la proposition est satisfait.

Réciproquement, supposons que pour tout γ_i de Γ, $\langle\Gamma \setminus \{\gamma_i\}\rangle$ est retenue par rapport à G, et que $gain(\Gamma \setminus \{\gamma_i\}, \Gamma) \geq G$. Soit θ une permutation de $\{1, \ldots, n\}$ et j un entier tel que $1 \leq j < n$. On montre alors que $gain(\Gamma_j, \Gamma_j \cup \{\gamma_{\theta(j+1)}\}) \geq G$, avec $\Gamma_j = \{\gamma_{\theta(1)}, \ldots, \gamma_{\theta(j)}\}$. Pour cela, on distingue les cas suivants :
1. Si Γ_j contient moins de $n-1$ conditions élémentaires, alors le résultat est une conséquence immédiate du fait que l'on suppose que, pour tout $i = 1, \ldots, n$, $\langle\Gamma \setminus \{\gamma_i\}\rangle$ est retenue par rapport à G.
2. Si Γ_j contient $n-1$ conditions élémentaires, alors il existe γ_0 tel que $\Gamma = \Gamma_j \cup \{\gamma_0\}$ et $\gamma_0 = \gamma_{\theta(j+1)}$. Dans ce cas, on a $gain(\Gamma_j, \Gamma_j \cup \{\gamma_{\theta(j+1)}\}) = gain(\Gamma \setminus \{\gamma_0\}, \Gamma) \geq G$.
D'où le résultat. □

Exemple 5 *Si l'on reprend l'exemple 4, avec les règles*

– ρ_1 : (*VAR* = JAYA) $\Rightarrow$ (*GERM* $\in [90, 95]$)

– ρ_2 : (*CAT* = BASE) $\Rightarrow$ (*GERM* $\in [90, 93]$)

– ρ_7 : (*VAR* = JAYA, *CAT* = BASE) $\Rightarrow$ (*GERM* $\in [90, 93]$)

et le seuil de gain de précision de 0.15, l'application à ρ_7 de la proposition 2 ci-dessus nécessite

1) de savoir si ρ_1 et ρ_2 sont retenues, et puisque c'est le cas

2) de calculer $gain(\rho_1, \rho_7)$ et $gain(\rho_2, \rho_7)$.

Puisque le second gain ci-dessus est inférieur à 0.15, la règle n'est pas retenue. D'autre part, il est important de noter par rapport aux calculs de l'exemple 4, que si l'on sait que l'une des règles ρ_1 ou ρ_2 n'est pas retenue, alors le premier point de la proposition

2 ci-dessus n'est pas satisfait. Dans ce cas, il est inutile de vérifier le second point de cette proposition, et donc aucun calcul supplémentaire n'est nécessaire pour savoir que ρ_7 n'est pas retenue.

Le corollaire suivant généralise la remarque de l'exemple ci-dessus.

Corollaire 1 *Soit G un seuil de gain de précision et $\langle\Gamma\rangle$ une règle telle que Γ contient au moins deux conditions élémentaires. Si $\langle\Gamma\rangle$ est retenue par rapport à G alors pour tout $\Gamma' \subset \Gamma$, $\langle\Gamma'\rangle$ est retenue par rapport à G.*

PREUVE : Conséquence immédiate du premier point de la proposition 2. □

Le corollaire 1 ci-dessus est utilisé dans les algorithmes de façon à faire des coupures, comme dans le cas de la seule mesure de support [1] : si, lors de l'examen de la règle $\langle\Gamma\rangle$, l'une des règles $\langle\Gamma'\rangle$ avec $\Gamma' \subset \Gamma$ n'est pas retenue, alors $\langle\Gamma\rangle$ ne peut être retenue.

4. Algorithmes d'extraction des règles de prédiction

L'algorithme d'extraction des règles de prédiction présenté figure 1 est basé sur *Apriori* [1]. Cet algorithme prend en entrée une relation sur U ne contenant aucune valeur manquante, un seuil de support et un seuil de gain de précision, et retourne toutes les règles de prédiction fréquentes et retenues. Il est important de noter les deux points suivants :

1) Comme l'algorithme *Apriori*, notre algorithme effectue un parcours par niveau du treillis de l'ensemble des parties de l'ensemble de toutes les conditions élémentaires et utilise l'anti-monotonie du support et le corollaire 1 afin d'effectuer des coupures.

2) Contrairement à l'algorithme *Apriori*, les règles sont engendrées dans la même phase que l'extraction des fréquents. Cette particularité est due au fait que les règles calculées ont une confiance de 1.

Les deux algorithmes de la figure 2 détaillent les calculs de RC_k et L_k dans l'algorithme 1. Ces algorithmes sont présentés dans le cas où le domaine de l'attribut prédit A_{i_0} est continu, le cas discret étant similaire, n'est pas décrit dans l'article.

On remarquera que la complexité de l'algorithme 1 est identique à celle de l'algorithme *Apriori*, *i.e.*, linéaire en nombre de passes sur la table $\overline{R}$ mais exponentielle par rapport au nombre N de conditions élémentaires que l'on peut former à partir des données présentes dans la table $\overline{R}$.

En effet, pour chaque niveau du treillis, donc pour chaque valeur de k dans l'algorithme 1, une seule passe sur les données suffit à évaluer les supports et les intervalles (ou ensembles) associés aux conditions de C_k. Puisque le nombre maximal de conditions

Algorithme 1
Entrée : *une table de données complètes $\overline{R}$, un seuil de support S, un seuil de gain de précision G*
Sortie : *l'ensemble des règles de prédiction fréquentes et retenues*
DEBUT
Calculer $RC_1 = \{(\langle\Gamma_1\rangle, support(\langle\Gamma_1\rangle)) \mid |\Gamma_1| = 1\}$
//Nécessite un balayage de la table de données
$L_1 = \{\langle\Gamma_1\rangle \in RC_1 \mid support(\langle\Gamma_1\rangle) \geq S$ *et* $gain(\emptyset, \Gamma_1) \geq G\}$
k=2
Tant que $L_{k-1} <> \emptyset$ **faire**
//Génération des conditions de cardinalité k à partir de L_{k-1} selon la méthode de [1]
$C_k = \{\Gamma_k \mid |\Gamma_k| = k\}$ *à partir des* Γ_{k-1} *dans* L_{k-1}
//Phase d'élagage de C_k analogue à la méthode de [1]
$C_k = C_k \setminus \{\Gamma_k \in C_k \mid (\exists\gamma \in \Gamma_k)(\langle\Gamma_k \setminus \{\gamma\}\rangle \notin L_{k-1})\}$
//Calcul du support et construction de l'ensemble prédit
//Nécessite une passe sur la table de données
$RC_k = \{(\langle\Gamma_k\rangle, support(\langle\Gamma_k\rangle)) \mid \Gamma_k \in C_k\}$
//Sélection des règles fréquentes et retenues
$L_k = \{\langle\Gamma_k\rangle \in RC_k \mid support(\langle\Gamma_k\rangle) \geq S$ *et* $(\forall\gamma \in \Gamma_k)(gain(\Gamma_k \setminus \{\gamma\}, \Gamma_k) \geq G)\}$
$k = k + 1$
Fin Tantque
Retourner $\bigcup_{i=1..k} L_i$
FIN

Figure 1. *Algorithme d'extraction des règles de prédiction*

élémentaires dans une condition est égal à $|U| - 1$, où $|U|$ est le nombre d'attributs de $\overline{R}$, le nombre de passes sur les données est au plus égal à $|U| - 1$. D'autre part, la cardinalité de chaque ensemble C_k est au pire égale à $\binom{k}{N}$. Donc la taille de RC_k est en $O(N^k)$. On rappelle toutefois que les nombreuses expérimentations utilisant un algorithme par niveau montrent que ce type d'algorithme reste performant en présence de gros volumes de données.

L'exemple ci-dessous illustre l'application de ces algorithmes sur les données présentées dans la table 1 (b), pour un seuil de support $S = 0.14$ et un seuil de gain de précision $G = 0.14$.

Exemple 6 *Pour* $k = 1$*, l'ensemble des candidats est initialisé (première boucle* **Pour** *de l'algorithme 2) :* $sup(\Gamma_k) = 0$, $\min(E_{\Gamma_k}) = \max(adom_{A_{i_0}})$, *et* $\max(E_{\Gamma_k}) = \min(adom_{A_{i_0}})$*. Le parcours de la table de données (deuxième boucle* **Pour** *de l'algorithme 2) permet de déterminer les valeurs de* $sup(\Gamma_k)$, $\min(E_{\Gamma_k})$ *et* $\max(E_{\Gamma_k})$*. Ainsi, la première ligne de données* (JAYA, BASE, BONNE, 93) *permet de mettre à 1 les supports des conditions VAR* = JAYA, *CAT* = BASE, *et PURETE* = BONNE*. De plus, en ce qui concerne les intervalles prédits, on a* $t_1.A_{i_0} = 93$*. Donc,* $\min(E_{\Gamma_k})$ *et* $\max(E_{\Gamma_k})$ *prennent la valeur* 93*.*

Algorithme 2 : Calcul de RC_k
Entrée : *l'ensemble C_k des candidats*
Sortie : *le support et les règles associées aux éléments de RC_k*
DEBUT
Pour tout Γ_k **de** C_k **faire**
$min(E_{\Gamma_k}) = max(adom(A_{i_0}))$
$max(E_{\Gamma_k}) = min(adom(A_{i_0}))$
$sup(\langle\Gamma_k\rangle) = 0$
Fin Pourtout
Pour tout t **de** $\overline{R}$ **faire**
Pour tout Γ_k **de** C_k **faire**
Si $t \models \Gamma_k$ **alors**
$sup(\langle\Gamma_k\rangle) = sup(\langle\Gamma_k\rangle) + 1$
Si $t.A_{i_0} < min(E_{\Gamma_k})$ **alors** $min(E_{\Gamma_k}) = t.A_{i_0}$ **Fin Si**
Si $t.A_{i_0} > max(E_{\Gamma_k})$ **alors** $max(E_{\Gamma_k}) = t.A_{i_0}$ **Fin Si**
Fin Si
Fin Pourtout
Fin Pourtout
Retourner $RC_k = \{(\langle\Gamma_k\rangle, sup(\Gamma_k)) \mid \Gamma_k \in C_k\}$
FIN

Algorithme 3 : Calcul de L_k
Entrée : *l'ensemble RC_k*
Sortie : *l'ensemble L_k des règles fréquentes et retenues*
DEBUT
$L_k = \emptyset$
Pour tout $\langle\Gamma_k\rangle$ **de** RC_k **faire**
stocker=faux
Si $sup(\langle\Gamma_k\rangle) \geq S$ **alors**
stocker=vrai
Pour tout γ **de** Γ_k **faire**
Si $gain(\Gamma_k \setminus \{\gamma\}, \Gamma_k) < G$ **alors** *stocker=faux* **Fin Si**
Fin Pourtout
Fin Si
Si *stocker=vrai* **alors** $L_k = L_k \cup \{\langle\Gamma_k\rangle\}$ **Fin Si**
Fin Pourtout
Retourner L_k
FIN

Figure 2. *Algorithmes de calcul de RC_k et de L_k*

La deuxième ligne concernant les mêmes conditions élémentaires, on incrémente les supports et on effectue la mise à jour pour l'intervalle prédit : on a $t_2.A_{i_0} = 90$ *et alors* $\min(E_{\Gamma_k})$ *devient 90 et* $\max(E_{\Gamma_k})$ *est inchangé. Donc le nouvel intervalle prédit est* $[90, 93]$*. Ce procédé est répété jusqu'au balayage complet de la table et on obtient :*

$$\begin{aligned} RC_1 = \{ \quad & (\langle \textit{VAR} = \text{JAYA}, [90, 95]\rangle, 8/14), \\ & (\langle \textit{VAR} = \text{IR1529}, [90, 98]\rangle, 4/14), \\ & (\langle \textit{VAR} = \text{IKP}, [90, 93]\rangle, 2/14), \\ & (\langle \textit{CAT} = \text{BASE}, [90, 93]\rangle, 5/14) \\ & (\langle \textit{CAT} = \text{PBASE}, [95, 98]\rangle, 3/14), \\ & (\langle \textit{CAT} = \text{CERT R1}, [90, 95]\rangle, 5/14), \\ & (\langle \textit{CAT} = \text{CERT R2}, [95, 95]\rangle, 1/14), \\ & (\langle \textit{PURETE} = \text{BONNE}, [90, 95]\rangle, 7/14), \\ & (\langle \textit{PURETE} = \text{FAIBLE}, [95, 95]\rangle, 1/14), \\ & (\langle \textit{PURETE} = \text{MOYENNE}, [92, 98]\rangle, 6/14) \quad \} \end{aligned}$$

Les deux conditions $\textit{CAT} = \text{CERT R2}$ *et* $\textit{PURETE} = \text{FAIBLE}$ *ne sont pas fréquentes et donc n'apparaissent pas dans* L_1 *lors de l'application de l'algorithme 3. Concernant les règles associées aux autres conditions, il est facile de vérifier que* $\langle \textit{VAR} = \text{IR1529}, [90, 98]\rangle$ *a un gain de précision nul. Après exécution complète de l'algorithme 3, on obtient :*

$$\begin{aligned} L_1 = \{ \quad & \langle \textit{VAR} = \text{JAYA}, [90, 95]\rangle, \langle \textit{VAR} = \text{IKP}, [90, 93]\rangle, \\ & \langle \textit{CAT} = \text{BASE}, [90, 93]\rangle, \langle \textit{CAT} = \text{PBASE}, [95, 98]\rangle, \\ & \langle \textit{CAT} = \text{CERT R1}, [90, 95]\rangle, \langle \textit{PURETE} = \text{BONNE}, [90, 95]\rangle, \\ & \langle \textit{PURETE} = \text{MOYENNE}, [92, 98]\rangle \quad \} \end{aligned}$$

À partir des règles ainsi obtenues, les candidats de niveau 2 sont générés et leurs supports et intervalles sont calculés. Prenons la première candidate : $\langle (\textit{VAR} = \text{JAYA}, \textit{CAT} = \text{BASE}), [90, 93]\rangle$*, dont le support est supérieur ou égal à* 0.14*. Le premier calcul de gain effectué est :* $gain((\textit{CAT} = \text{BASE}), (\textit{VAR} = \text{JAYA}, \textit{CAT} = \text{BASE}))$*. Ce gain étant nul, la règle n'est pas retenue.*

On considère maintenant $\langle (\textit{VAR} = \text{JAYA}, \textit{CAT} = \text{CERT R1}), [92, 95]\rangle$*. Le support étant supérieur ou égal à* 0.14*, on calcule :* $gain((\textit{VAR} = \text{JAYA}), (\textit{VAR} = \text{JAYA}, \textit{CAT} = \text{CERT R1}))$ *et* $gain((\textit{CAT} = \text{CERT R1}), (\textit{VAR} = \text{JAYA}, \textit{CAT} = \text{CERT R1}))$*. Les deux gains étant supérieurs à 0.15, la règle est retenue. Après exécution complète de l'algorithme 3, on obtient :*

$$\begin{aligned} L_2 = \{ \quad & \langle \textit{VAR} = \text{JAYA}, \textit{CAT} = \text{CERT R1}, [92, 95]\rangle, \\ & \langle \textit{VAR} = \text{JAYA}, \textit{PURETE} = \text{MOYENNE}, [92, 95]\rangle, \\ & \langle \textit{CAT} = \text{CERT R1}, \textit{PURETE} = \text{MOYENNE}, [92, 95]\rangle \quad \} \end{aligned}$$

Il est facile de voir qu'au troisième niveau, on a : $C_3 = \{(VAR = \text{JAYA}, CAT = \text{CERT R1}, PURETE = \text{MOYENNE})\}$. *La condition* $(VAR = \text{JAYA}, CAT = \text{CERT R1}, PURETE = \text{MOYENNE})$ *étant fréquente, on évalue les gains suivants :*

$$\begin{aligned} gain(\ & (CAT = \text{CERT R1}, PURETE = \text{MOYENNE}), \\ & (VAR = \text{JAYA}, CAT = \text{CERT R1}, PURETE = \text{MOYENNE})) \\ gain(\ & (VAR = \text{JAYA}, PURETE = \text{MOYENNE}), \\ & (VAR = \text{JAYA}, CAT = \text{CERT R1}, PURETE = \text{MOYENNE})) \\ gain(\ & (VAR = \text{JAYA}, CAT = \text{CERT R1}), \\ & (VAR = \text{JAYA}, CAT = \text{CERT R1}, PURETE = \text{MOYENNE})) \end{aligned}$$

Ces trois gains étant égaux à 0.33*, la règle*

$$\langle VAR = \text{JAYA}, CAT = \text{CERT R1}, PURETE = \text{MOYENNE}, [93, 95]\rangle$$

est retenue.

5. Prédiction et résultats expérimentaux

5.1. Méthode de prédiction

Étant donné un n-uplet t de R dont la valeur sur A_{i_0} est inconnue, on peut prédire une approximation de $t.A_{i_0}$ en utilisant les règles de prédiction sur A_{i_0}, extraites selon la méthode vue précédemment. La méthode de prédiction proposée dans cet article est la suivante.

Soit $\bar{t} = \langle v_{i_1}, v_{i_2}, \ldots, v_{i_k}\rangle$ le n-uplet sur le schéma $\{A_{i_1}, A_{i_2}, \ldots, A_{i_k}\}$ obtenu à partir de t en ne considérant que les attributs où t est défini. Si l'on assimile $\bar{t}$ à la condition $\Gamma_t = \langle A_{i_1} = v_{i_1}, A_{i_2} = v_{i_2}, \ldots, A_{i_k} = v_{i_k}\rangle$, alors l'intervalle (ou l'ensemble) prédit sur A_{i_0} pour t est l'intersection de tous les intervalles (ou ensembles) E_i tels que :

– $\langle \Gamma_i, E_i\rangle$ est une règle extraite, et

– $\Gamma_i \subseteq \Gamma_t$.

Dans le cas où cette intersection est vide, alors aucune prédiction n'est possible, ce qui signifie intuitivement que les données de la table $\overline{R}$ ne permettent pas de traiter le cas.

L'exemple suivant illustre notre méthode dans le cadre de l'exemple 1.

Exemple 7 *Si l'on reprend les règles de prédiction obtenues à l'exemple 6, les prédictions suivantes peuvent alors être faites concernant les valeurs manquantes sur GERM dans la table* R *du tableau 1(a) :*

– Pour TID $= 2$*, le n-uplet à considérer est* $\langle 2, \text{JAYA}, \text{BASE}, \text{BONNE}\rangle$*. Dans ce cas, grâce aux règles* $\langle$*VAR* $= \text{JAYA}, [90, 95]\rangle$*,* $\langle$*CAT* $= \text{BASE}, [90, 93]\rangle$*, et* $\langle$*PURETE* $= \text{BONNE}, [90, 95]\rangle$*, l'intervalle prédit sur GERM est* $[90, 93]$.

– Pour TID $= 11$*, le n-uplet à considérer est* $\langle 11, \text{IKP}, \text{BASE}, \text{MOYENNE}\rangle$*. Dans ce cas, grâce aux règles* $\langle$*VAR* $= \text{IKP}, [90, 93]\rangle$*,* $\langle$*CAT* $= \text{BASE}, [90, 93]\rangle$ *et* $\langle$*PURETE* $= \text{MOYENNE}, [92, 98]\rangle$*, l'intervalle prédit sur GERM est* $[92, 93]$.

– Pour TID $= 14$*, le n-uplet à considérer est* $\langle 14, \text{JAYA}, \text{CERT R1}, \text{MOYENNE}\rangle$*. Dans ce cas, seule la règle* $\langle$*VAR* $= \text{JAYA}, $*CAT* $= \text{CERT R1}, $*PURETE* $= \text{MOYENNE}, [93, 95]\rangle$ *peut être appliquée, et donc l'intervalle prédit sur GERM est* $[93, 95]$.

– Pour TID $= 17$*, le n-uplet à considérer est* $\langle 17, \text{JAYA}, \text{CERT R2}\rangle$*. Dans ce cas, seule la règle* $\langle$*VAR* $= \text{JAYA}, [90, 95]\rangle$ *peut être appliquée, et donc l'intervalle prédit sur GERM est* $[90, 95]$.

On notera que, selon la proposition 1, seules les règles $\langle\Gamma\rangle$ telles que Γ est maximale (au sens de l'inclusion) sont à considérer dans le calcul de la prédiction. De plus, comme le montre la proposition suivante, notre méthode de prédiction est *cohérente* avec le contenu de $\overline{R}$.

Proposition 3 *En reprenant les notations introduites ci-dessus, soit* t *un n-uplet de* R *dont la valeur sur* A_{i_0} *est inconnue, et supposons que les règles de prédiction extraites conduisent, pour* t*, à l'intervalle (ou à l'ensemble) de prédiction* E*. Alors, pour tout n-uplet* t' *de* $\overline{R}$ *tel que* t' *est un sur-n-uplet de* $\bar{t}$*, on a :* $t'.A_{i_0} \in E$.

PREUVE : Soit $\langle\Gamma, E_\Gamma\rangle$ une règle utilisée pour la prédiction. Puisque t' est un sur-n-uplet de $\bar{t}$, pour tout attribut A sur lequel $\bar{t}$ est défini, si A apparaît dans Γ, alors Γ contient la condition $A = t.A$. Comme t' est un sur-n-uplet de $\bar{t}$, on a $t.A = t'.A$, et par conséquent, $t'.A_{i_0} \in E_\Gamma$. Ce résultat étant vrai pour toute règle $\langle\Gamma, E_\Gamma\rangle$ ayant participé au calcul de E, $t'.A_{i_0}$ appartient à tous les intervalles considérés et donc, $t'.A_{i_0} \in E$. □

Afin d'illustrer la proposition 3 ci-dessus, on considère à nouveau les tables R et $\overline{R}$ du tableau 1, ainsi que $\bar{t} = \langle\text{JAYA}, \text{CERT R2}\rangle$. Comme il a été vu à l'exemple 7 ci-dessus, dans ce cas, l'intervalle prédit sur *GERM* est $[90, 95]$. On peut de plus constater que $\overline{R}$ contient le n-uplet $\langle 7, \text{JAYA}, \text{CERT R2}, \text{BONNE}, 95\rangle$, dont la valeur sur *GERM* appartient à l'intervalle prédit.

Pour terminer cette section, on remarquera que dans [6], la méthode de prédiction utilisée diffère de celle exposée ci-dessus, car dans [6], la mesure de Piatetski-Shapiro ([10]) est utilisée pour calculer l'intervalle (ou l'ensemble) prédit. Les liens entre les deux méthodes de prédiction sont actuellement à l'étude, et il semble en particulier que l'intervalle (ou l'ensemble) prédit par notre méthode est un *sous-ensemble* de l'intervalle (ou l'ensemble) prédit par la méthode de [6].

5.2. Résultats expérimentaux

Les résultats expérimentaux décrits dans [6] ont été réalisés sur un serveur Ultra-SPARC (64 bit CPU et 512 M mémoire centrale) et les programmes ont été écrits en C++. Les tests ont été effectués à partir de données synthétiques et de données réelles, puis les résultats ont été comparés aux résultats obtenus par l'algorithme *C4.5*. Cette comparaison, exprimée en pourcentage, est faite sur la base d'une mesure appelée *précision* et définie de la manière suivante :

$$\left(1 - \frac{|E|}{|adom(A_{i_0})|}\right) 100$$

où $|E|$ est la longueur ou la cardinalité de l'ensemble prédit.

Dans le cas de données synthétiques, le pourcentage de prédictions correctes est de l'ordre de 98%, et la précision moyenne varie entre 89% et 100%, lorsque le nombre de valeurs manquantes est strictement inférieur à 50%. Lorsque le nombre de valeurs manquantes est de 50%, la précision moyenne est égale à 67%.

Les données réelles sur lesquelles des expériences ont été faites concernent la reconnaissance de caractères. L'ensemble de données compte 20 000 lignes dont 16 000 ont été utilisées pour l'apprentissage. Pour un seuil de support de 0.03 et un seuil de gain de précision de 0.1, 410 règles ont été extraites en *10h 30mn*. Parmi toutes les valeurs manquantes, 99.62% d'entre elles peuvent donner lieu à une prédiction et 99.55% des règles extraites sont correctes.

Il est important de rappeler en ce qui concerne les performances, que dans [6, 7], un parcours en profondeur de l'espace de recherche est effectué. Une implantation de l'algorithme 1, qui effectue un parcours par niveau, est en cours.

6. Conclusion

En conclusion, rappelons que l'approche présentée dans cet article permet l'extraction de règles de prédiction de confiance 1 et dont la partie droite est un ensemble de valeurs. Outre les mesures de support et de confiance, la notion de gain de précision a été introduite afin d'améliorer la pertinence des règles extraites. Il a de plus été vu que, bien que l'évaluation de cette mesure nécessite *a priori* un nombre exponentiel de calculs de gains de précision, il est possible de diminuer ce nombre de calculs en appliquant un algorithme par niveau de type *Apriori*.

Nos travaux en cours sont liés aux aspects de performance : l'implantation des algorithmes présentés dans cet article devrait sensiblement améliorer les performances rapportées dans [6]. Les points suivants sont également étudiés :

1) Étude précise des règles retenues, notamment en considérant les liens entre notre approche et, d'une part les concepts de motifs clés et de motifs fermés ([9, 16]), et d'autre part les bases génériques de règles associatives exactes proposées dans [3].

2) Étude des liens entre notre approche et les travaux proposés dans [15]. En effet, l'étude des représentations condensées de motifs fréquents en présence de valeurs manquantes pourrait permettre de réduire le nombre de règles de prédiction, qui en pratique peut être important.

3) Étude des liens entre la méthode de prédiction donnée dans cet article et celle introduite dans [6], qui utilise la mesure de Piatetski-Shapiro ([10]) pour calculer l'intervalle (ou l'ensemble) prédit.

4) Étude d'un critère pour retenir une règle qui soit moins restrictif que celui proposé dans cet article (définition 3). En effet, si l'on considère par exemple le cas où l'on a deux conditions élémentaires γ_1 et γ_2 telles que les gains $gain(\emptyset, \{\gamma_i\})$ $(i = 1, 2)$ sont tous les deux inférieurs au seuil G, il se peut néanmoins que $gain(\emptyset, \{\gamma_1, \gamma_2\})$ soit *supérieur* à G. Dans une telle situation, l'approche présentée dans cet article ne retient pas la règle $\langle\{\gamma_1, \gamma_2\}\rangle$, alors qu'une telle règle présente un gain suffisant pour permettre des prédictions.

7. Bibliographie

[1] AGRAWAL R., MANNILA H., SRIKANT R., TOIVONEN H., VERKAMO A.I., « Fast Discovery of Association Rules », *Advances in Knowledge Discovery and Data Mining*, pp 309-328, AAAI-MIT Press, 1996.

[2] FUJIKAWA Y., « Efficient Algorithms for Dealing with Missing Values in Knowledge Discovery », Ph.D Thesis, School of Knowledge Science, Japan Advanced Institute of Science and Technology, 2001.

[3] GUIGUES J.-L., DUQUENNE V., « Familles Minimales d'Implications Informatives Résultant d'un Tableau de Données Binaires », *Math. Sci. Humaines*, vol. 95, pp 5-18, 1986.

[4] HAN J., KAMBER M., *Data Mining : Concepts and Techniques*, Morgan Kaufmann Publishers, 2000.

[5] INGUNN M., STENRUD E., OLSSON U., « Analyzing Data Sets with Missing Data : An Empirical Evaluation of Imputation Methods and Likelihood-Based Methods », *IEEE Transactions on Software Engineering*, vol. 27, n° 11, pp 999-1013, 2001.

[6] JAMI S., « Learning Quality Rules from Sparse and Uncertain Data », Ph.D Thesis, University of London, Birkbeck College, 2000.

[7] JAMI S., LIU X., LOIZOU G., « Learning from an Incomplete and Uncertain Data Set : The Identification of Variant Haemoglobins », *Workshop on Intelligent Data Analysis in Medicine and Pharmacology, ECAI'98*, 1998.

[8] LEVENE M., LOIZOU G., *A Guided Tour of Relational Databases and Beyond*, Springer-

Verlag, 1999.

[9] PASQUIER N., BASTIDE Y., TAOUIL R., LAKHAL L., « Efficient Mining of Association Rules using Closed Itemsets Lattices », *Information Systems*, vol. 24, n° 1, pp 25-46, 1999.

[10] PIATETSKY-SHAPIRO G., « Discovery, Analysis and Presentation of Strong Rules », *Knowledge Discovery in Databases*, pp 229-248, AAAI-MIT Press, 1991.

[11] QUINLAN R., *C4.5 : Programs for Machine Learning*, Morgan Kaufmann, 1993.

[12] RAGEL A., CRÉMILLEUX B., « Treatment of Missing Values for Association Rules », *Pacific-Asia Conference on Knowledege Discovery and Data Mining (PAKDD)*, LNCS 1394, pp 258-270, Springer-Verlag, 1998.

[13] RAGEL A., CRÉMILLEUX B., « MVC- A Preprocessing Method to Deal With Missing Values », *Knowledege-based Systems*, vol. 12, n° 5/6, pp 285-291, 1999.

[14] RAMEZ E., SHAMKANT N.B., *Fundamentals of Databases Systems*, Addison-Wesley, 3ème Edition, 2000.

[15] RIOULT F., CRÉMILLEUX B., « Condensed Representations in Presence of Missing Values », *International Conference on Intelligent Data Analysis, IDA'03*, LNCS 2810, pp 578-588, Springer-Verlag, 2003.

[16] STUMME G., TAOUIL R., BASTIDE Y., PASQUIER N., LAKHAL L., « Intelligent Structuring and Reducing of Association Rules with Formal Concept Analysis », *KI 2001 : Advances in Artificial Intelligence, KI/ÖGAI*, LNCS 2174, pp 335-350, Springer-Verlag, 2001.

[17] ULLMAN J.D., *Principles of Databases and Knowledge-Base Systems*, vol. 1-2, Computer Science Press, 1989.

[18] WU C.-H., WUN C.-H., CHOU H.J., « Using Association Rules for Completing Missing Data », *Fourth IEEE Int'l Conf. on Hybrid Intelligent Systems (HIS'04)*, pp 236-241, IEEE Computer Society, 2004.