Chapitre 4

Contrôle du risque de fausse découverte en extraction des connaissances

Sommaire

1 Introduction 121
1.1 Cas des règles d'association 122
1.2 Validation des règles d'association 123
2 État de l'art : le contrôle du risque multiple 124
2.1 Risque et erreurs de 1^e espèce 124
2.2 Procédures de contrôle 125
3 BSFD 127
3.1 Algorithme *BSFD* 128
3.2 *BSFD* bilatéral 128
3.3 Cas de plusieurs critères 130
3.4 Justification de la méthode 130
3.5 Complexité de *BSFD* 131
3.6 Application des nombres quasi-aléatoires au bootstrap. 131
4 Sélection de règles d'association significatives 135
4.1 Notion de règle d'association significative 135
4.2 Test de signification d'une règle 136
4.3 Expérimentation 136
4.4 Validation des résultats 138
5 Découvertes de règles d'association différentiellement exprimées : cas des données génomiques 139
5.1 Notion de règles d'association différentiellement exprimées 140
5.2 Résultats et interprétation 142
6 Discussion 145

Contrôle du risque de fausse découverte en extraction des connaissances

Publications

Lallich S., O. Teytaud et E. Prudhomme (2007), Association rules interestingness : measure and validation, *Quality Measures in Data Mining* (eds. Guillet F. and Hamilton H. J.), Studies in Computational Intelligence, 43, Springer-Verlag Berlin Heidelberg, pp. 251-275.

Lallich S., O. Teytaud et E. Prudhomme (2006), Statistical inference and data mining : false discoveries control. In *Proc. of 17th Computational Statistique Symposium (COMPSTAT)*, pp. 325-336, La Sapienza, Rome, Italy.

Teytaud O., S. Gelly, S. Lallich et E. Prudhomme (2006), Quasi-random bootstrap, with applications to rule extraction and (sub)bagging, In *Proc. of International Workshop on Intelligent Information Access, IIIA-2006*, July 2006, Helsinki, Finlande.

Vaillant B., P. Meyer, E. Prudhomme, S. Lallich, P. Lenca et S. Bigaret (2005), Mesurer l'intérêt des règles d'association, In *Actes de l'Atelier Qualité des Données et des Connaissances (DQK), associé à la* 5e *Conférence Extraction et Gestion des Connaissances, EGC 2005*, pp. 69-78. Cépaduès Editions.

Lallich S., E. Prudhomme et O. Teytaud (2004), Contrôle du risque multiple en sélection de règles d'association significatives, In *Extraction des connaissances et apprentissage (Actes de EGC'2004*, Clermont-Ferrand, France, RNTI-E-2, pp. 305-316. Cépaduès Editions.

1 Introduction

Comme nous l'avons énoncé dans la section 4.3 du chapitre 1, l'extraction de connaissances doit être accompagnée d'une mesure de qualité capable de juger de la pertinence de la connaissance découverte. Dans certains contextes, telles que la sélection d'attributs, les algorithmes d'extraction s'intéressent à un ensemble de connaissances. La pertinence de chacune est alors jugée en comparant leur valeur pour une mesure donnée à un seuil prédéterminé. Ce test est ainsi répété autant de fois qu'il y a de connaissances potentielles, par exemple d'attributs à examiner. Comme les mesures sont calculées sur un échantillon de la population de référence, chaque test est associé à un risque d'erreur de première

espèce, de sorte que l'augmentation du nombre de tests conduit à une augmentation du nombre de fausses découvertes, et donc dans le cas de la sélection d'attributs, à des attributs sélectionnés à tort.

La sélection d'attributs discriminants en apprentissage supervisé illustre bien ce problème. Pour chaque attribut, cela revient à tester s'il prend des valeurs différentes suivant les modalités de la variable de classe (en comparant les moyennes conditionnelles à la classe (ANOVA) ou en comparant leur rang moyen (test des rangs de Friedman) ou encore en effectuant un test de permutations). En se plaçant dans une situation où l'on teste m attributs non-discriminants, et en acceptant un risque α_0 de sélectionner à tort un attribut, on sélectionne en moyenne $m\alpha_0$ attributs. Si $m = 10\,000$ et avec un risque $\alpha_0 = 0,05$, on sélectionne en moyenne 500 attributs à tort. Ici, le problème vient donc de la reconduction du risque associé à chaque test autant de fois qu'il y a de tests.

1.1 Cas des règles d'association

Ce biais ne se produit pas seulement dans le cas de la sélection d'attributs. Il est présent pour toutes méthodes de fouille qui produisent un ensemble de connaissances à valider. Dans ce chapitre, nous nous sommes plus particulièrement intéressé aux règles d'association. Comme nous l'avons souligné dans l'introduction générale, la recherche des règles d'association intéressantes est un problème classique de l'ECD qui fait suite aux travaux de Agrawal et al. (1993) dans le cadre des bases de données transactionnelles. Ces bases sont des matrices attributs/valeurs booléennes (notée $\mathcal{T}$, de taille $n \times d$), où chaque exemple représente une transaction dont les attributs correspondent aux articles disponibles. L'intersection d'une transaction et d'un article est une variable booléenne qui symbolise l'acte d'achat ou *item*. De même, la conjonction des actes d'achat (ou *itemset*) associés à un ensemble d'articles est également vue comme une variable booléenne.

À partir cette matrice booléenne, on veut extraire une base de règles (notée $\mathcal{R}$, avec $m = card(\mathcal{R})$ du type « si un client achète du pain et du fromage, alors probablement il achète aussi du vin ». Plus généralement, chaque règle r est une expression du type $A \rightarrow B$, où l'antécédent A et le conséquent B sont des *itemsets* qui n'ont pas d'*items* communs. On note n_a et n_b les nombres de transactions qui réalisent respectivement les items de A et de B et n_{ab} le nombre de celles qui réalisent à la fois A et B. Les proportions correspondantes sont désignées par p_a, p_b et p_{ab}. Ce formalisme se généralise à toute base de données dont on a extrait une matrice booléenne cas-attributs.

Pour extraire les règles d'association à partir d'une base de transactions, on a généralement recours à des algorithmes de type support-confiance, comme *Apriori* l'algorithme fondateur (Agrawal et al., 1993) et les améliorations qui en ont été proposées (voir par exemple celle proposée par Borgelt utilisée dans Tanagra (Rakotomalala, 2005)). Le support d'une règle est la proportion de transactions qui réalisent à la fois A et B,

$Supp(A \rightarrow B) = p_{ab} = \frac{n_{ab}}{n}$, alors que sa confiance est la proportion de transactions qui réalisent B, parmi celles qui réalisent A, $Conf(A \rightarrow B) = \frac{p_{ab}}{p_a} = \frac{n_{ab}}{n_a} = 1 - \frac{n_{a\bar{b}}}{n_a}$. Les algorithmes d'extraction support-confiance parcourent le treillis des *itemsets* pour rechercher les *itemsets* fréquents, dont le support dépasse un seuil min_{supp}, avec une efficacité liée à l'antimonotonie du support sur le treillis des *itemsets*. On déduit ensuite de ce parcours les règles dont la confiance dépasse le seuil min_{conf}, obtenant la base de règles admissibles aux seuils choisis. Il résulte de cette approche des bases comportant un grand nombre de règles, dont toutes ne sont pas intéressantes pour l'utilisateur.

À partir de cette base de règles admissibles, la sélection des règles intéressantes nécessite d'évaluer celles-ci à l'aide de mesures ayant les qualités requises compte tenu de la nature des règles d'association et des attentes de l'utilisateur. À ce stade, il est important de connaître les mesures disponibles et les critères d'intérêt qui permettent de les évaluer. Ce travail a notamment été réalisé par Lallich et Teytaud (2004). En outre, Lenca et al. (2003) proposent une procédure d'aide à la décision pour choisir les mesures les plus adaptées au problème. Une fois les mesures choisies, un seuil minimal est fixé pour chacune d'entre elles, seuil à partir duquel une règle est sélectionnée (sur ce point, une solution alternative consiste à retenir un nombre fixé des meilleures règles, classées suivant une mesure d'intérêt correctement choisie).

1.2 Validation des règles d'association

La sélection des règles d'association intéressantes ne constitue pas une étape de validation mais plutôt une nouvelle étape de fouille de données qui s'applique cette fois à une base de règles. Cependant, avant de sélectionner les règles intéressantes, il est important de valider le résultat de l'algorithme d'extraction, afin de s'assurer que toutes les règles de la base soient valides au sens des critères qui ont amené à leur découverte. Dans ce chapitre, nous nous intéressons plus particulièrement à ce problème. En effet, valider une base de règles revient à conduire un test statistique sur chacune, ce qui nous ramène au problème du contrôle du risque multiple.

Plusieurs méthodes de contrôle du risque multiple ont été proposées dans la littérature afin de corriger le biais introduit par une multitude de tests. Nous les présentons dans la section 2. Ces méthodes ont souvent pour origine le contrôle du risque multiple en sélection d'attributs. Face aux problèmes rencontrés pour la validation des règles d'association, nous proposons $BSFD$ une méthode de contrôle du risque multiple flexible : elle est non-paramétrique, s'applique à la fois au cas des tests unilatéraux et bilatéraux et permet une validation simultanément sur plusieurs critères (section 3). Ces caractéristiques la rendent particulièrement intéressante dans le contexte de la sélection de règles d'association. Pour illustrer cet intérêt, nous proposons deux applications. La première concerne la sélection de

règles d'association significatives (contexte non-supervisé, section 4) alors que la seconde vise la sélection de règles d'association différentiellement exprimées (contexte supervisé, section 5).

2 État de l'art : le contrôle du risque multiple

Cette section présente les différentes stratégies de contrôle du risque multiple en séparant le risque contrôlé sur l'ensemble des tests, des procédures qui le contrôlent. Nous nous plaçons dans le cas où ces procédures sont utilisées pour sélectionner des règles d'associations à partir d'une base $\mathcal{R}$ de m règles, dont on connait pour chacune la *p-value* du critère statistique contrôlé.

2.1 Risque et erreurs de 1^e espèce

Face au problème du contrôle du risque multiple, la correction de Bonferroni consiste à pratiquer chaque test au risque $\frac{\alpha}{m}$, afin que le risque de rejeter au moins une fois à tort H_0, noté par la suite $FWER$ (pour *Family Wise Error Rate* ou taux d'erreur en famille complète) soit égal à α. Cette correction n'est pas satisfaisante pour deux raisons :

- le $FWER$ n'est pas contrôlé par cette procédure. Il est compris entre $\frac{\alpha}{m}$ et α et ne vaux α que si toutes les règles sont indépendantes.
- le $FWER$ est très conservateur en faveur de H_0, ce qui augmente considérablement le risque de 2^e espèce ou risque de ne pas sélectionner une règle pertinente.

Face à ce dernier point, d'autres auteurs ont cherché à contrôler une quantité moins sévère que le $FWER$ sur le risque de 1^e espèce, notamment sous la condition de dépendance entre les tests. Dans ce domaine, les principaux travaux ont émergé de la bio-informatique et de l'analyse des données génomiques, où le problème consiste à sélectionner des gènes (les attributs) dont le niveau d'expression diffère suivant la classe du patient analysé (dans la plupart des cas, ces classes correspondent souvent à des patients sains *vs* malades). On trouvera une remarquable synthèse de ces travaux dans Ge et al. (2003). L'idée fondamentale proposée par Benjamini et Hochberg (1995) consiste à considérer non plus le risque d'erreur de la procédure de test lorsqu'elle est pratiquée une fois, mais plutôt le nombre d'erreurs commises lorsque l'on réitère m fois la procédure. À partir du tableau 4.1 (où les quantités en majuscules sont des variables aléatoires observables, celles en minuscules étant fixes mais inconnues en ce qui concerne m_0 et m_1), on peut définir différents indicateurs des erreurs commises. Nous présentons ici les deux plus connus, le $FWER$ (introduit précédemment) et le FDR (*False discovery rate* ou taux de fausses découvertes) qui ont chacun donné lieu à des procédures de contrôle.

Le $FWER$ est la probabilité de rejeter au moins une fois à tort H_0. Avec les notations du tableau 4.1, il s'écrit $FWER = \Pr(V > 0)$. Il a l'inconvénient d'être trop sévère pour

Réalité \ Décision	Acceptation	Rejet	Total
H_0 est vraie	U	V	m_0
H_1 est vraie	T	S	m_1
Total	W	R	m

TAB. 4.1: Synthèse des résultats de m tests

une multiplicité de tests. Cependant, il nous a suggéré une variante flexible et originale, où l'on s'autorise V_0 fausses découvertes et que nous appelons *User Adjusted Family Wise Error Rate*, $UAFWER = \Pr(V > V_0)$ (voir section 3). Pour remédier aux inconvénients du *FWER*, diverses quantités reposant sur l'espérance de V, le nombre de fausses découvertes, éventuellement normalisé, ont été proposées. La plus connue est le *FDR* (Benjamini et Hochberg, 1995), qui correspond dans le cas des règles d'association à la proportion attendue de règles sélectionnées à tort parmi les règles sélectionnées. Ces auteurs définissent $FDR = E(Q)$, avec $Q = \frac{V}{R}$ si $R > 0$, 0 sinon. Il s'ensuit :

$$FDR = E(\frac{V}{R} \mid R > 0)P(R > 0)$$

En suivant cette idée, Storey (2002) a proposé le *pFDR*, une variante du *FDR* adaptée à l'estimation du taux d'erreur sachant que l'hypothèse nulle a été refusée au moins une fois :

$$pFDR = E(\frac{V}{R} \mid R > 0)$$

Ces deux quantités présentent l'avantage d'être moins sévères sur le résultat des m tests, au prix de l'acceptation de quelques sélections à tort dont on contrôle la proportion, ce qui augmente pour chaque test la probabilité pour qu'une règle pertinente soit sélectionnée (c'est-à-dire la puissance du test). Du point de vue de la sévérité du contrôle, il est possible d'ordonner ces quantités de la manière suivante : $FDR \leq pFDR \leq FWER$ (Ge et al., 2003). Une fois choisie une définition du risque multiple de 1^e espèce se pose le problème de son contrôle.

2.2 Procédures de contrôle

Dans cette section, nous examinons successivement les procédures de contrôle pour le *FWER*, le *FDR* et le *pFDR*. Les procédures de contrôle relatives au *FWER* et au *FDR* sont fondées sur le principe des *p-values* ajustées. Ce principe consiste à corriger la valeur de la *p-value* pour rendre compte de l'ensemble des autres *p-value*. Ge et al. (2003) définissent la *p-value* ajustée comme « *the level of the entire test procedure at which* H_0 *would just be rejected, given the values of all test statistics involved* ». Pour les procédures relatives au *FWER* et au *FDR* que nous présentons dans la section suivante, les *p-values* ajustées sont ensuite comparées à un seuil de rejet α_0.

2.2.1 Contrôle du *FWER*

Correction de Bonferroni. La correction de Bonferroni consiste à calculer des *p-values* ajustées afin de prendre en compte la multiplicité des tests. Étant données la statistique de test T_i relative à une règle i $(i = 1, 2, ..., m)$ et la *p-value* correspondante p_i, on définit la *p-value* ajustée, notée $\widetilde{p}_i$, par $\widetilde{p}_i = \min(mp_i, 1)$. On sélectionne alors toutes les règles ayant une *p-value* ajustée inférieure au risque α_0.

En pratique, cette procédure ne contrôle pas le $FWER$ dans tous les cas. Il se montre en effet que $FWER = 1 - \Pr(\bigcap_{r=1}^{m} \left(p_i > \frac{\alpha_0}{m}\right) \mid H_0)$. Sous des conditions d'indépendance entre les règles, on obtient bien $FWER = 1 - (1 - \frac{\alpha_0}{m})^m \approx \alpha_0$ ce qui est le but de la correction de Bonferroni. En revanche, lorsque les règles sont dépendantes, on obtient seulement $\frac{\alpha_0}{m} \leq FWER \leq \alpha_0$ et la correction de Bonferroni devient trop sévère.

Procédure pas à pas. Holm (1979) propose de faire évoluer le seuil de contrôle au fur et à mesure que des règles sont selectionnées. En effet, une règle sélectionnée correspondant à une situation où H_0 est fausse, un nouveau seuil peut donc être établi sur la base des règles restant à examiner. Pour cela, il met en place une procédure pas à pas au cours de laquelle les règles sont testées par ordre croissant de leur *p-value*. En partant de la *p-value* la plus faible, H_0 est refusée tant que $p_{(k)} < \frac{\alpha_0}{m-k+1}$, $k = 1, 2, \ldots, m$. À l'inverse, pour toutes les *p-values* qui suivent la première acceptation, H_0 est acceptée. Cette procédure, facile à mettre en oeuvre, donne de bons résultats lorsque le nombre de tests est faible, la correction du seuil ayant alors de l'importance. Elle reste cependant mal adaptée à un grand nombre de tests.

Procédure minP. Cette procédure d'ajustement des *p-values* a été initiée par Westfall et Young (1993) pour prendre en compte les dépendances entre les différents tests (Ge et al., 2003). Pour contrôler le $FWER$, ces auteurs calculent des *p-values* ajustées telles que $\widetilde{p}_i = \Pr(\min_{k=1,2,\ldots,m} P_k \leq p_i \mid H_0)$. L'estimation de cette probabilité se fait en deux étapes. Ils estiment d'abord chaque *p-value* P_k à partir d'un échantillon obtenu par randomisation des étiquettes. $\widetilde{p}_i$ est ensuite calculée comme la proportion de $P_k \leq p_i$ sur un ensemble d'échantillons. Parce qu'elle tient compte de certaines dépendances, cette procédure contrôle moins fortement le $FWER$ que la correction de Bonferroni.

2.2.2 Contrôle du *FDR* et du *pFDR*

Procédure BH. Benjamini et Hochberg (1995) ont mis en place une procédure séquentielle de contrôle du FDR en cas d'indépendance. Elle consiste à trier les *p-values* des règles dans l'ordre croissant puis à rejeter l'hypothèse nulle tant que la *p-value* examinée $p_{(i)}$ est inférieure à $\frac{i\alpha_0}{m}$, pour i croissant à partir de 1. Les auteurs ont également montré que cette

procédure assure que $FDR \leq \frac{m_0}{m}\alpha_0 \leq \alpha_0$ en cas d'indépendance entre les règles. Enfin, pour prendre en compte les dépendances entre les règles, Benjamini et Yekutieli (2001) remplacent le critère de contrôle $\frac{i\alpha_0}{m}$ par $\frac{i\alpha_0}{m\sum_{l=1}^{m} 1/l}$. La contre-partie de cette correction est que la procédure devient trop stricte pour m grand (Ge et al., 2003).

pFDR. Le $pFDR$ est défini comme $E(\frac{V}{R} \mid R > 0)$, ce qui conduit à $pFDR = 1$ lorsque $m = m_0$. Ce cas particulier empêche le $pFDR$ d'être contrôlé par une procédure fondée sur les *p-values* ajustées puisqu'il n'est plus possible de garantir que $pFDR \leq \alpha_0$ pour un $\alpha_0 < 1$ (Benjamini et Hochberg, 1995). Pour contrôler malgré tout le $pFDR$, Storey (2002) le définit en fonction d'une zone de rejet Γ par l'approximation suivante :

$$\hat{pFDR}(\Gamma) = \frac{\hat{\pi_0}.m.\Gamma}{\#\{p_i \leq \Gamma, i = 1, \ldots, m\}}$$

Ici, π_0 correspond à la proportion d'hypothèses nulles $\frac{m_0}{m}$. Storey (2002) estime cette proportion, pour une zone d'acceptation λ :

$$\hat{\pi_0}(\lambda) = \frac{\#\{p_i \geq \lambda, i = 1, \ldots, m\}}{m(1-\lambda)}$$

Pour choisir automatiquement parmi les différents $\hat{\pi}_0(\lambda)$, Storey et Tibshirani (2003) définissent une procédure automatique qui repose sur l'estimation de $\lim_{\lambda \to 1} \hat{\pi}_0(\lambda)$. Pour ce faire, ils calculent $\hat{f}(\lambda)$ une *cubic spline* de λ sur $\hat{\pi}_0(\lambda)$ et estiment $\hat{\pi}_0$ par $\hat{f}(1)$.

Le $pFDR$ est donc défini par rapport à une zone de rejet Γ qu'il faut choisir par avance. Pour cette zone de rejet, il est possible de déterminer des *q-values* qui vont permettre de contrôler le $pFDR$ sur chaque *p-value*. Elles sont obtenues de la manière suivante :

$$\hat{q}(p_{(i)}) = \min_{k=i,\ldots,m} pFDR(p_{(k)})$$

La *q-value* est donc au $pFDR$ ce que la *p-value* est à l'erreur de 1^e espèce ou ce que la *p-value* ajustée est au $FWER$. Ainsi, pour toute règle dont la *p-value* a une *q-value* inférieure au $pFDR$, on rejette H_0 et on sélectionne la règle.

3 BSFD

Les procédures de contrôle que nous venons de présenter sont paramétriques et requièrent de connaître la loi du critère contrôlé sous l'hypothèse nulle. Or, cette loi n'est pas toujours connue, en particulier lors de la sélection des règles d'association où les mesures statistiques ont parfois des formes élaborées. Dans ce cas, on veut alors sélectionner parmi les règles r de la base $\mathcal{R}$ celles qui sont statistiquement significatives pour un critère c, au sens où leur évaluation $c(r)$ est significativement plus élevée que $c_0(r)$, valeur attendue

sous H_0, l'hypothèse d'indépendance entre les *itemsets* A et B. On note $\mathcal{R}^*$, la base des règles significatives.

Au cours de travaux précédents, Lallich et al. (2003b) ont suggéré différents algorithmes qui utilisent les outils de l'apprentissage statistique pour garantir que 100% des règles trouvées soient significatives à un risque α donné, ainsi l'algorithme *BS* fondé sur le bootstrap. Les expérimentations ont confirmé que cette approche était trop prudente, ce qui la rend peu puissante. Prenant en compte l'idée d'accepter de façon contrôlée un certain nombre de fausses découvertes, à l'instar des travaux de Benjamini sur le FDR, nous proposons *BSFD* (Lallich et al., 2004) qui adapte l'algorithme *BS* au contrôle du nombre de faux positifs.

L'algorithme *BSFD* repose sur le contrôle du $UAFWER$ (*User Adjusted FWER*) que nous avons défini comme $P(V > V_0)$, c'est à dire la probabilité de rejeter à tort V_0 fois l'hypothèse nulle (pour rappel, on a $FWER = P(V > 0)$), avec V_0 un paramètre contrôlé par l'utilisateur. L'algorithme *BSFD* va permettre de contrôler cette probabilité pour un risque δ donné, de sorte qu'il garantit que $P(V > V_0)$ converge vers δ, à la limite d'un grand échantillon de transactions.

3.1 Algorithme *BSFD*

Le principe de $BSFD$ est d'estimer par une procédure *bootstrap* de combien peut varier le critère c tout en maintenant V_0 fausses découvertes à un risque δ. Ainsi, pour garantir que $c(r) > c_0(r)$, $BSFD$ raisonne sur $c'(r)$, l'évaluation empirique de $c(r)$ sur la base $\mathcal{T}$ des transactions et sur $c''(r)$, l'évaluation empirique de $c(r)$ sur un échantillon *bootstrap* de $\mathcal{T}$. Par ailleurs, plutôt que de considérer $c(r) > c_0(r)$, on se limite sans perte de généralité à garantir $c(r) > 0$, en remplaçant $c(r)$ par le critère translaté $c(r) - c_0(r)$. L'algorithme 6 résume la procédure à suivre. On note par la suite # l'opérateur cardinal, associant à un ensemble son cardinal, i.e. $\#E$ est le cardinal de l'ensemble E.

3.2 *BSFD* bilatéral

Dans Lallich et al. (2006b), nous avons étendu *BSFD* aux tests bilatéraux. Il est alors nécessaire de définir $V_{0\mathrm{g}}$ et $V_{0\mathrm{d}}$, le nombre de fausses découvertes tolérées respectivement à gauche et à droite de sorte que $V_{0\mathrm{g}} + V_{0\mathrm{d}} = V_0$. Dans ce cas, $BSFD$ va estimer par une procédure *bootstrap* de combien peut varier le critère c tout en maintenant $V_{0\mathrm{g}}$ fausses découvertes à gauche et $V_{0\mathrm{d}}$ fausses découvertes à droite, à un risque global δ. À cette fin, les étapes 2.b, 3 et 4 de $BSFD$ sont modifiées. L'algorithme 7 détaille ces modifications.

L'étape 3 de l'algorithme requiert le classement des couples $(\varepsilon(V_{0\mathrm{g}}, i), \varepsilon(V_{0\mathrm{d}}, i))$. Une solution consiste à utiliser l'ordre suivant : $(a, b) > (c, d) \Leftrightarrow (a > c)$ et $(b > d)$. Cependant, cet ordre n'est pas total ce qui conduit à plusieurs maximums. Parmi ces différents maxi-

Algorithme 6 : Algorithme BSFD

1. *Évaluation empirique.* Définir $c'(r)$, l'évaluation empirique de $c(r)$ sur l'ensemble $\mathcal{T}$ de transactions pour $r \in \mathcal{R}$.
2. *Bootstrap.* Les opérations suivantes sont répétées un grand nombre l de fois :
   a. Tirer au sort, avec remise, une liste $\mathcal{T}'$ d'éléments de $\mathcal{T}$, de même cardinal que $\mathcal{T}$. Certaines transactions de $\mathcal{T}$ ne seront pas dans $\mathcal{T}'$ alors que d'autres seront présentes plusieus fois.
   b. Définir $c''(r)$, l'évaluation de $c(r)$ sur la liste $\mathcal{T}'$ de transactions.
   c. À partir des différences $c''(r) - c'(r)$, calculer $\varepsilon(V_0, i)$, la plus petite valeur telle que $\#\{c''(r) > c'(r) + \varepsilon(V_0, i)\} \leq V_0$. En pratique, $\varepsilon(V_0, i)$ est le $(V_0 + 1)^{\text{e}}$ plus grand élément parmis les différences $c''(r) - c'(r)$ de la i^{e} itération du bootstrap.
3. *Résumé des échantillons bootstrap.* À partir des l valeurs $\varepsilon(V_0, i)$, calculer $\varepsilon(\delta)$, le quantile $(1 - \delta)$ des $\varepsilon(V_0, i)$. De cette manière, $\varepsilon(V_0, i)$ est plus grand que $\varepsilon(\delta)$ seulement $l\delta$ fois sur l.
4. *Décision.* Enfin, garder dans $\mathcal{R}^*$ toutes les règles r de $\mathcal{R}$ telles que $c'(r) > \varepsilon(\delta)$.

Algorithme 7 : Algorithme *BSFD* bilatéral

2.c Déterminer $V_{0\text{g}}$ et $V_{0\text{d}}$ de sorte que $V_{0\text{g}} + V_{0\text{d}} = V_0$. Ensuite, à partir des différences $c''(r) - c'(r)$, calculer :
- $\varepsilon(V_{0\text{d}}, i)$, la plus petite valeur telle que $\#\{c''(r) > c'(r) + \varepsilon(V_{0\text{d}}, i)\} \leq V_{0\text{d}}$ et
- $\varepsilon(V_{0\text{g}}, i)$, la plus petite valeur telle que $\#\{c''(r) < c'(r) - \varepsilon(V_{0\text{g}}, i)\} \leq V_{0\text{g}}$.

En pratique, $\varepsilon(V_{0\text{d}}, i)$ est le $(V_{0\text{d}} + 1)^{\text{e}}$ plus grand élément parmi les différences $c''(r) - c'(r)$ et $\varepsilon(V_{0\text{g}}, i)$ est le $(V_{0\text{g}} + 1)^{\text{e}}$ plus grand élément parmi les différences $c'(r) - c''(r)$.

3. *Résumé des échantillons bootstrap.* À partir des l paires de valeurs $(\varepsilon(V_{0\text{g}}, i), \varepsilon(V_{0\text{d}}, i))$, calculer $\varepsilon(\delta) = (\varepsilon(V_{0\text{g}}), \varepsilon(V_{0\text{d}}))$, le quantile $(1 - \delta)$ des $(\varepsilon(V_{0\text{g}}, i), \varepsilon(V_{0\text{d}}, i))$. De cette manière, $(\varepsilon(V_{0\text{g}}, i), \varepsilon(V_{0\text{d}}, i))$ est plus grand que $(\varepsilon(V_{0\text{g}}), \varepsilon(V_{0\text{d}}))$ seulement $l\delta$ fois sur l.
4. *Decision.* Enfin, garder dans $\mathcal{R}^*$ toutes les règles r de $\mathcal{R}$ telles que $c'(r) > \varepsilon(V_{0\text{d}})$ ou $c'(r) < -\varepsilon(V_{0\text{g}})$.

mums, nous proposons de choisir celui qui maximise le nombre de découvertes, de manière à augmenter la puissance du test.

Notons que la solution proposée est à la fois efficace et flexible, mais difficile à implémenter. À la place, une solution moins puissante mais plus facile à mettre en œuvre consiste à équilibrer le nombre de fausses découvertes et le risque d'erreur à gauche et à droite. Pour ce faire, on réalise l'étape 2 avec $V_{0\mathrm{g}} = V_{0\mathrm{d}} = \frac{V_0}{2}$ afin de calculer $\varepsilon(V_{0\mathrm{g}}, i)$ et $\varepsilon(V_{0\mathrm{d}}, i)$. Ensuite, à l'étape 3, $\varepsilon(V_{0\mathrm{g}})$ est estimé par le quantile $1 - \frac{\delta}{2}$ de $\varepsilon(V_{0\mathrm{g}}, i)$ et $\varepsilon(V_{0\mathrm{d}})$, par le quantile $1 - \frac{\delta}{2}$ de $\varepsilon(V_{0\mathrm{d}}, i)$. Dans les expérimentations menées en section 5, nous utilisons cette version de l'algorithme.

3.3 Cas de plusieurs critères

En pratique, il arrive fréquemment que l'on s'intéresse à plusieurs critères. Dans le cas des règles d'association, ces critères peuvent être la confiance, le support et un critère de non-indépendance. L'extension de l'algorithme *BSFD*, notée *BSFD_ mc*, se fait alors simplement en utilisant comme critère unique le minimum des différents critères. Ainsi pour un travail sur 3 critères c_1, c_2 et c_3, considère-t-on $c(r) = min\{c_1(r), c_2(r), c_3(r)\}$. Utiliser *BSFD_ mc* sur c au risque δ fournit ainsi des règles valides à la fois pour les critères c_1, c_2 et c_3 au risque δ.

Notons qu'afin d'optimiser le risque de seconde espèce, on gagnera à travailler sur des transformations (différentiables au sens de Hadamard) des c_i qui rendent ces critères homogènes, par exemple des *p-values* ou des réductions, en divisant l'écart empirique de chaque critère à sa valeur de référence par l'estimation de l'écart-type issue des $c'(r)$.

3.4 Justification de la méthode

Les méthodes de bootstrap (Efron, 1979) ont d'abord un aspect très intuitif de par leur idée d'approcher l'écart entre la loi empirique et la loi réelle par l'écart entre la loi obtenue par *bootstrap* et la loi empirique. En outre, elles ont de profondes justifications mathématiques qui nécessitent une formalisation précise de la question posée (Giné et Zinn, 1984). Formellement, l'objectif est que la fonction de répartition du nombre de règles telles que $c(r) < 0$ malgré $c'(r) > \epsilon$ ait pour valeur au moins $1 - \delta$ en V_0. On a $\#\{c(r) \leq 0$ et $c'(r) > \epsilon\}$ majoré par $\#\{c'(r) \geq c(r) + \epsilon\}$. Les théorèmes sur le bootstrap appliqué à une famille de fonctions vérifiant des hypothèses minimales (Van Der Vaart et Wellner, 1996), nous permettent d'approcher cette quantité par $\#\{c''(r) \geq c'(r) + \epsilon\}$.

3.5 Complexité de *BSFD*

La complexité de *BS_FD* est proportionnelle à $l \times m \times n$, en considérant que le générateur de nombres au hasard fonctionne en temps constant. En effet, la complexité de recherche du k^e plus grand élément d'un tableau est proportionnelle à la taille du tableau. L'algorithme est donc globalement linéaire en $m \times n$, avec une forte constante l, liée au *bootstrap*.

3.6 Application des nombres quasi-aléatoires au bootstrap.

Hasard et rééchantillonnage. En pratique, la valeur de l doit être assez grande pour que l'imprécision liée à la finitude de l ne nuise pas à la confiance globale. En effet, chacun des l échantillons étant issus d'un tirage aléatoire avec remise, rien ne garantit que ces échantillons soient bien distribués. Augmenter le nombre d'échantillon améliore ainsi l'espérance de l'estimation, en lissant l'effet provoqué par les aléas du rééchantillonnage. Pour améliorer la stabilité du bootstrap afin d'en accélérer la convergence (et ainsi réduire l), nous proposons de biaiser le tirage alétoire de manière à choisir des échantillons bien distribués (Teytaud et al., 2006).

Nombres quasi-aléatoires et *discrepancy*. L'idée de faire mieux que le hasard pour simuler des distributions uniformes n'est pas nouvelle en informatique. Elle part d'une observation simple : la génération aléatoire de points dans $[0,1]^2$ conduit presque toujours à l'obtention de régions de l'espace plus denses en points que d'autres. Pour contrer cette effet, une stratégie consiste à définir une mesure d'uniformité (*discrepancy*) à optimiser lors de la génération des points (voir Hickernell, 1998, pour plus de détails). Pour un ensemble de points $X = (x_1, x_2, \ldots, x_n)$ dans $[0,1)^d$, la *discrepancy* maximise, parmi les régions de l'espace, l'écart absolu entre le nombre de points de X dans une région et le nombre de points que cette région aurait comporté si la distribution de X était uniforme. La mesure *star-discrepancy* (notée D^*) est l'une des mesures de *discrepancy* les plus connues. Elle est définie telle que :

$$D_n^*(X) = \sup_{B \in J^*} \left| \frac{Card_X(B)}{n} - \lambda_d(B) \right| ,$$

avec J^* l'ensemble des hyper-rectangles, incluant le point $(0)^d$, dans l'espace $[0,1)^d$, B un hyper-rectangle de J^*, $Card_X(B)$ le nombre de points de X contenu dans B et $\lambda_d(B)$ la mesure d-dimensionnelle de Lebesgue, qui correspond ici au produit de la longueur des arètes de B par n (et qui est une estimation du nombre de points de l'hyper-rectangle B si les points sont répartis uniformément dans $[0,1)^d$). Le principal défaut de D^* est de

choisir des hyper-rectangles ancrés en $(0)^d$, ce qui a conduit à définir la mesure *L_2-star-discrepancy*, plus flexible quant au choix des régions de l'espace à considérer (Hickernell, 1998).

Diverses méthodes génèrent des séquences de points optimisant ces mesures de *discrepancy*, ainsi la séquence de Van Der Corput en dimension 1 (van der Corput, 1935), généralisée par les séquences de Halton et *scrambled*-Halton à d dimensions. En pratique, cette dernière séquence génère des points quasi-aléatoirement dans $[0,1)^d$ en optimisant la *L_2-star-discrepancy*. On parle alors de générateur quasi-aléatoire[1]. Il peut être montré que ces générateurs assurent une convergence de la distribution plus rapide que les nombres pseudo-aléatoires.

Bootsrap et nombres quasi-aléatoires. Le bootstrap est fondé sur un l rééchantillonnages successifs avec remise. À partir d'un échantillon de points $X = (x_1, x_2, \ldots, x_n)$, le bootsrap génère l nouveaux échantillons $X' = \{x_{j_i}\}$ où $(j_i)_{i\in[1,n]}$ correspond à des indices de points dans X, tirés aléatoirement et uniformément dans $[1, n]$. Pour améliorer la distribution du bootstrap, il est donc tentant de biaiser le tirage des indices à l'aide d'un générateur de nombres quasi-aléatoires en n dimensions, de sorte que les échantillons du bootstrap dans cette espace soient uniformisés. Malheureusement, cette solution qui présente l'avantage de la simplicité n'est pas correcte. Pour le comprendre, il faut considérer chaque échantillon *bootstrap* comme un point dans l'espace $[1, n]$ où chaque dimension de l'espace correspond à un indice d'un point de X. En utilisant un générateur de nombres pseudo-aléatoires pour choisir l points (un par échantillon requis par le bootstrap), on obtient bien une distribution presque uniforme des échantillons dans cet espace. Cependant, rien ne garantit que deux points différents de cet espace mènent à deux échantillons différents du *bootstrap*. Pour s'en convaincre, prenons le cas où $n = 5$, soit $X = (x_1, x_2, \ldots, x_5)$. À l'aide d'un générateur quasi-aléatoire, on va chercher à obtenir 5 nouveaux indices pour chaque échantillon du *bootstrap*. Ainsi, le premier échantillon pourra être composé des indices $(1, 1, 3, 5, 2)$. Il s'agit bien d'un point dans un espace à 5 dimensions et le tirage suivant va faire en sorte d'optimiser la *discrepancy* de cette espace. Imaginons que le deuxième tirage donne les indices suivant $(5, 3, 1, 2, 1)$. Il s'agit d'un point de l'espace différent du premier et de nature à diminuer la *discrepancy*. Pourtant, du point de vue du bootstrap, ces deux échantillons sont identiques.

Le bootstrap doit donc être formalisé différemment pour que l'espace du tirage aléatoire conduise systématiquement à des échantillons différents pour le bootstrap. Teytaud et al. (2006) remarquent que la distribution des indices j_i est multinomiale. Or, une distribution multinomiale le reste même après que les premiers éléments aient été choisis. Plutôt que

[1] Nous ne détaillons pas ici ces générateurs dont les algorithmes sont librement disponibles. Par exemple, on pourra trouver le générateur de *scrambled-Halton* au sein de la *Gnu Scientifique Library*, disponible à l'adresse http ://www.gnu.org/software/gsl/

d'orienter le tirage aléatoire vers le choix des indices j_i, nous proposons d'introduire l'aléa dans le nombre de réplications (≥ 0) de chaque indice. L'algorithme 8 détaille la procédure à suivre pour générer un échantillon *bootstrap*.

Algorithme 8 : Procédure de choix d'un échantillon *bootstrap* en vue de l'introduction de nombres quasi-aléatoires.

entrées : $\mathbf{X} = (x_1, x_2, \ldots, x_n)$, l'échantillon des n points à *bootstraper*.
sorties : $\mathbf{X}' = (x'_1, x'_2, \ldots, x'_n)$, un échantillon *bootstrap* de X.

Tirer aléatoirement $z \in]0,1]^n$;
$r \leftarrow n$;
pour $i \leftarrow 1$ **à** n **faire**
  Définir le plus petit z_i tel que $P(\text{binomial}(r, \frac{1}{(n+1-i)} \leq z_i) \geq z_i$;
  $r \leftarrow r - z_i$;
fin-pour
$j \leftarrow 1$;
pour $i \leftarrow 1$ **à** n **faire**
  pour $k \leftarrow 1$ **à** z_i **faire**
    $x'_j = x_i$;
    $j \leftarrow j + 1$;
  fin-pour
fin-pour
renvoie X';

À partir de l'algorithme 8, il est facile d'introduire les nombres quasi-aléatoires, simplement en remplaçant le tirage aléatoire des $z \in]0,1]^n$ par un générateur de nombres pseudo-aléatoires.

Problème des données volumineuses. La procédure que nous venons de définir permet d'uniformiser le tirage des échantillons *bootstrap* afin de réduire le nombre d'itérations nécessaires à une bonne estimation. L'objectif pratique est de réduire le temps de calcul du bootstrap. La génération de nombres quasi-aléatoires a cependant un coût qui devient important lorsque le nombre de points de l'échantillon de départ est grand. Rappelons ici que le nombre de points de l'échantillon (les exemples de nos matrices attributs/valeurs) devient la dimension de l'espace dans lequel la distribution est uniformisée. Pour 2000 exemples (ce qui est loin d'être un cas anecdotique de la fouille de données), on se retrouve à générer des points dans des espaces à 2000 dimensions, ce qui requiert un temps de calcul important[(2)]. Dans ce cas, il est nécessaire de réaliser une réduction de la dimentionnalité en regroupant les exemples en classes de sorte que la génération quasi-aléatoire soit appli-

[2] De toute façon, le générateur de la *Gnu Scientifique Library* s'arrête à des espaces de 1229 dimensions.

quée au choix des classes et non plus aux exemples de ces classes. L'algorithme 9 explicite cette stratégie.

Algorithme 9 : Adaptation du tirage quasi-aléatoire des exemples en bootstrap aux données de grande taille.

Grouper les exemples en k classes à l'aide des k-means ;
pour $i \leftarrow 1$ **à** k **faire**
    Déterminer n_i, le nombre d'exemples par classe à l'aide des nombres quasi-aléatoires ;
    Tirer aléatoirement n_i exemples parmi ceux de la i^{e} classe ;
fin-pour

Résultats. Cette dernière recherche reste en marge de nos travaux de thèse. C'est pourquoi les résultats ne sont pas détaillé dans la partie expérimentation mais font l'objet d'une synthèse ici.

Ces procédures de bootstrap reposant sur les nombres quasi-aléatoires ont été évaluées dans Teytaud et al. (2006) à l'aide de $BSFD$. Les expérimentations ont été menées sur des jeux de données artificielles pour lesquels les attributs sont déterminés indépendemment suivant une loi $N(0,1)$. Chaque jeu de données est composé de 16 exemples et d'un nombre d'attributs variant de 10 à 50 (par pas de 10). Sur ces données, les attributs dont la moyenne est supérieure à 0 ont été sélectionnés à l'aide $BSFD$ avec $V_0 \in (1,\ldots,6)$ et $\delta = 0,25$. Compte tenu de la définition des données, aucun attribut n'est à sélectionner en théorie avec ce critère.

Pour chaque jeu de données et chaque valeur de V_0, la procédure de *bootstrap* de $BSFD$ a été appliquée avec 32 échantillons soit avec un générateur quasi-aléatoire (de type *scrambling*-Halton), soit avec un générateur pseudo-aléatoire. Les résultats obtenus montrent que l'utilisation de nombres quasi-aléatoires accélère la convergence de l'algorithme tout en diminuant la variance de l'estimateur (augmentation de la stabilité).

Enfin, cette stratégie est également transposable à d'autres méthodes de rééchantillonnage comme le *bagging* et le *subagging*. Les expérimentations menées montrent là encore une amélioration de la convergence pour le *bagging* et le *subagging* lors de l'utilisation de nombres quasi-aléatoires.

4 Sélection de règles d'association significatives

4.1 Notion de règle d'association significative

Parmi les mesures objectives (celles qui ne font intervenir que les effectifs observés), il faut distinguer les mesures statistiques des mesures descriptives qui s'opposent sur le critère « prise en compte de n, le nombre d'observations ». Il est logique *a priori* de souhaiter qu'une mesure soit statistique, les résultats observés étant d'autant plus fiables que n est grand. Cependant, compte tenu de la taille des bases sur lesquelles on recherche des règles d'association, de telles mesures perdent leur pouvoir discriminant, ainsi l'indice d'implication (Lerman et al., 1981) et l'intensité d'implication (Gras, 1979). Des solutions très intéressantes ont été proposées : l'indice probabiliste discriminant (Lerman et Azé, 2003) qui centre et réduit les valeurs de l'indice d'implication sur une base de règles et l'intensité d'implication entropique (Gras et al., 2001) qui affecte l'intensité d'implication d'un facteur correctif tenant compte de l'entropie des expériences B/A et $\overline{A}/\overline{B}$. Mais il s'ensuit un mélange des notions de signification et d'intérêt, une perte d'intelligibilité de la mesure et sa loi est plus difficile à étudier.

Paradoxalement, le nombre de transactions n est le même pour toutes les règles de la base, ce qui milite pour d'abord tester la signification statistique des règles, au sens de savoir si elles renforcent réellement la probabilité du conséquent. On testera alors l'hypothèse d'indépendance de l'antécédent A et du conséquent B, notée H_0, en direction d'une dépendance positive (hypothèse alternative unilatérale notée H_1), pour ensuite utiliser des mesures descriptives intelligibles et discriminantes de l'intérêt des règles sur la base filtrée. Dans Lallich et al. (2004), nous proposons ainsi une nouvelle démarche qui dissocie la signification statistique de l'évaluation de l'intérêt. Elle comporte trois étapes :

- étape 1 : application d'un algorithme support-confiance à la base de transactions, pour constituer une base de règles admissibles aux seuils choisis.
- étape 2 : validation de la base de règles par le test d'indépendance de A et B pour chaque règle, soit une multitude de tests ; on pourra tester tout à la fois l'indépendance et le dépassement significatif des seuils de support et de confiance.
- étape 3 : analyse des règles figurant dans la base filtrée par des mesures descriptives vérifiant les critères retenus par l'utilisateur (Lenca et al., 2003; Vaillant et al., 2005).

L'étape 2 pose le problème de contrôler une multitude de tests. Comme nous l'avons expliqué précédemment, si chaque test est pratiqué au risque de 1^e espèce α, on engendre mécaniquement des faux positifs, c'est-à-dire ici des règles sélectionnées alors qu'elles ne renforcent pas la probabilité du conséquent. Dans la section suivante, nous appliquons $BSFD$ pour contrôler la multiplicité de ces tests.

4.2 Test de signification d'une règle

Considérons une règle $r : A \rightarrow B$ et une mesure de qualité c, croissante avec n_{ab} à marges fixées. On dira que r est significative au sens de la mesure c si son évaluation empirique $c'(r)$ remet en cause H_0 dans le sens de H_1, au risque de 1^e espèce α. Pour décider de la signification, on calcule la *p-value* de $c'(r)$ qui se définit comme la probabilité d'obtenir une valeur aussi grande que $c'(r)$ sous H_0 et on sélectionne la règle si l'on observe *p-value*$< \alpha$. Cette démarche impose de connaître la loi de $c(A \rightarrow B)$ sous H_0.

Dans le cadre de cette modélisation à marges fixées, la *p-value* de $c(r)$ se calcul comme celle de n_{ab}. Supposons que sous H_0 les transactions qui réalisent A et celles qui réalisent B sont réparties au hasard, indépendamment, en respectant les marges. Il se démontre que le nombre d'exemples suit alors une loi hypergéométrique, $N_{ab} \equiv H(n, np_a, p_b) \equiv H(n, np_b, p_a)$ (par convention, les variables aléatoires sont en majuscules). Pour une valeur observée n_{ab}, toujours à marges fixées :

$$p - value = \Pr(N_{ab} \geq n_{ab}) = \Pr(H(n, np_a, p_b) \geq n_{ab})$$

On opére l'approximation normale de cette loi hypergéométrique sous des conditions peu contraignantes, à savoir $n_a n_b \geq 5n$ et $n_a n_{\overline{b}} \geq 5n$, où Φ est la fonction de répartition de la loi normale centrée réduite $N(0,1)$. En notant $t_{ab} = \frac{n_{ab}}{n_a n_b}$, la valeur attendue de N_{ab} sous H_0, R le coefficient de corrélation entre A et B, il vient :

$$p - value = 1 - \Phi\left(\frac{n_{ab} - t_{ab}}{\sqrt{\frac{n_{\overline{a}}}{n-1} t_{ab} p_{\overline{b}}}}\right) \approxeq 1 - \Phi\left(\frac{n_{ab} - t_{ab}}{\sqrt{n p_a p_b p_{\overline{a}} p_{\overline{b}}}}\right) = 1 - \Phi\left(R\sqrt{n}\right)$$

Le coefficient de corrélation R est donc une mesure privilégiée pour tester l'indépendance (H_0) face à une dépendance positive (H_1).

4.3 Expérimentation

4.3.1 Description des données

Les méthodes de filtrage présentées ici ont été appliquées à cinq bases de règles disponibles sur la plateforme HERBS (Vaillant et al., 2003). Celles-ci ont été extraites à l'aide d'*Apriori* suivant l'implémentation de Borgelt et Kruse (2002) à partir de bases de transactions du site de l'UCI (Blake et Merz, 1998) : Contraceptive Method Choice (CMC), Flags (Flags), Wisconsin breast Cancer (WBC), Solar Flare I (SFI) et Solar Flare II (SFII). Les caractéristiques de ces bases sont données dans la partie supérieure du tableau 4.2. Sur chacune de ces bases, nous avons calculé le taux de réduction après retrait des règles non significatives à l'aide de $BSFD$ ainsi que celui des méthodes présentées en section 2.2.

Caractéristiques	CMC	Flags	WBC	SF I	SFII
Nb cas	1473	194	699	323	1066
Nb règles	2878	3329	3095	5402	3596
Tx couverture	100%	100%	96.2%	100%	100%
Tx recouvrement	259	1848	646	1828.6	2277
Seuil support	5%	50%	10%	20%	20%
Seuil confiance	60%	90%	70%	85%	85%
Résultats	CMC	Flags	WBC	SF I	SF II
contrôle à 5%	1401	2181	3094	2544	2558
pFDR	916 (3)	1200 (3)	3095 (0)	900 (5)	1625 (4)
***FDR* (Benj.)**	913 (0,003)	1198 (0,0027)	/	899 (0,006)	1626 (0,0022)
BSFD(R)	794 (3)	1074 (3)	3093 (0)	604 (5)	738 (4)
Holm	742	564	3094	432	1020
Bonferroni	731	539	3042	427	1006

TAB. 4.2: Validation des bases de règles CMC, Flags, WBC, SFI et SFII

4.3.2 Paramétrage des procédures de contrôles

Pour appliquer ces procédures, nous avons utilisé la définition de la *p-value* donnée en section 4.2. Sur ces *p-values*, les procédures de Holm et de Bonferroni ont été appliquées avec un seuil de 5%. Pour paramétrer le $pFDR$, nous avons choisi une zone de rejet de $0,1\%$. Cette zone a été déterminée de sorte que l'estimation du nombre de fausses découvertes qui en découle soit la plus acceptable possible. Ce nombre de fausses découvertes est indiqué entre parenthèses, dans la partie « Résultats » du tableau 4.2. Le seuil utilisé pour le FDR est égal à celui de la dernière *q-value* sélectionnée par le $pFDR$, de sorte qu'il soit possible de comparer grossièrement ces deux procédures. La valeur de cette *q-value* a également été reportée entre parenthèses dans le tableau 4.2. De même, pour pouvoir comparer $BSFD$ et le $pFDR$ nous avons choisi un nombre de fausses découvertes V_0 égal au résultat du $pFDR$, contrôlé à un risque $\delta = 0,05$, sur le cœfficient de corrélation R comparé à 0. Le choix de cette mesure est du à son lien avec la *p-value* (voir section 4.2).

4.3.3 Interprétation

À partir du tableau 4.2, nous opérons plusieurs constatations. Sur la base WBC, d'abord, la procédure de validation ne filtre aucune règle, y compris la correction de Bonferroni. Sur cette base, une seule des règles extraites a une *p-value* supérieure à 0,05 (0,184), une autre à une *p-value* de 0,023, et les autres *p-values* sont toutes inférieures à 0,01 (dont 3036 d'entre elles égalent à 0,0000 !). Pour les 4 autres bases, le filtrage par la simple ré-

pétition du test d'indépendance réduit notablement le nombre de règles respectivement de 51%, 34%, 53%, 39% (dans l'ordre du tableau de résultats). Cependant, dans une base ainsi filtrée, il reste encore beaucoup de faux positifs. L'application des différentes méthodes de contrôle du risque multiple va permettre de limiter fortement ces erreurs.

Comme prévu, la procédure de contrôle du risque la plus sévère est la correction de Bonferroni, donnant sur les 4 bases des taux de réduction de 75%, 81%, 65% et 60%. Par sa sévérité même, cette procédure est peu puissante, limitant certes les faux positifs, mais au prix d'une augmentation des faux négatifs. La procédure de Holm donne des résultats voisins, son inefficacité venant du très grand nombre de règles qui rend inutile la correction du seuil pas à pas. En revanche, les méthodes $pFDR$, FDR (procédure de Benjamini) et notre méthode $BSFD(R)$ donnent des résultats intermédiaires, correspondant à ce que l'on pouvait attendre. La méthode $BSFD$ apparaît comme la plus sévère des 3, plus particulièrement sur Solar Flare II. La raison de cette sévérité vient du critère contrôlé. En effet, le $pFDR$ et le FDR assurent un nombre moyen de fausses découvertes égal à V_0, alors que $BSFD$ assure que V_0 n'est dépassé qu'avec un risque de 0,05, ce qui est plus exigeant.

Notons également que la procédure de validation que nous proposons ici est d'autant plus nécessaire qu'elle permet d'éliminer des règles qui seraient sélectionnées avec bon nombre de mesures de qualité. Ainsi les règles logiques dont le conséquent est très fréquent (*e.g.* Solar Flare II) ont-elles une valeur maximale pour toutes les mesures qui donnent une valeur fixe maximale aux règles logiques, alors qu'elles n'ont aucune espèce d'intérêt et que leur *p-value* est non significative. À l'inverse, le calcul des *p-values* ne préjuge pas du classement ultérieur des règles par des mesures descriptives qui favorisent les règles les plus intéressantes, par exemple des mesures à la fois dissymétriques et qui avantagent les règles dont le conséquent est rare.

4.4 Validation des résultats

Alors que le découpage de la base des exemples en ensemble d'apprentissage et ensemble de test est courant en apprentissage supervisé, il n'est jamais pratiqué dans le domaine des règles d'association. La raison est sans doute que les règles d'association découlent de l'apprentissage non supervisé et que leur extraction suivant le support et la confiance est une tâche déterministe clairement définie, comme le souligne Freitas (2000). Dans la mesure où nous avons introduit un test de signification, il devient logique de distinguer base d'apprentissage et base de test pour étudier la pertinence sur la base de test des règles sélectionnées sur la base d'apprentissage.

La base de transactions CMC a été séparée aléatoirement en 2 bases de même taille, CMC_{app} et CMC_{test}. Sur CMC_{app}, nous avons extrait une base de règles admissibles au sens de l'algorithme *Apriori* que nous avons ensuite filtrée à l'aide des différentes méthodes

Résultats	**CMC$_{app}$**	**CMC$_{test}$**	**croisement**	**Taux**
sans contrôle	3391	2742	2742	0,81
contrôle à 5%	1835	1462	1238	0,68
pFDR	1347 (3)	996 (4)	938	0,70
BSFD(R)	1302 (3)	955 (4)	903	0,69
Holm	1149	858	795	0,69
BSFD_mc	159 (3)	276 (4)	125	0,79

TAB. 4.3: Validation des procédures de contrôle du risque multiple sur la base CMC

présentées. Nous avons alors examiné le comportement des règles admissibles issues de cette base lorsqu'elles étaient appliquées à la base CMC$_{test}$, en calculant pour chaque méthode de contrôle du risque multiple combien de règles sélectionnées sur CMC$_{app}$ le sont encore par la même méthode sur CMC$_{test}$. Le tableau 4.3 donne les résultats de ces différents traitements. Pour chaque méthode de contrôle du risque, la première colonne indique le nombre de règles sélectionnées sur CMC$_{app}$, la seconde le nombre de règles sélectionnées sur CMC$_{test}$ et la troisième, le croisement entre les bases de règles issues de CMC$_{app}$ et CMC$_{test}$.

On constate ainsi que 80% des règles admissibles pour CMC$_{app}$ le sont encore pour CMC$_{test}$, alors que ce taux descend à 70% lorsque l'on filtre la base de règle pour ne garder que les règles significatives, quelle que soit la méthode. En revanche, l'application de BSFD_mc) avec les trois critères, support, confiance et R, permet de revenir à ce taux de 80% au prix il est vrai d'une sérieuse diminution du nombre de règles.

5 Découvertes de règles d'association différentiellement exprimées : cas des données génomiques

Dans cette section, nous abordons l'extraction de règles d'association dans un contexte supervisé, où une étiquette de classe est associée à chaque transaction. Dans ce contexte, une approche fréquente consiste à rechercher des règles dont le conséquent correspond à un seul *item* qui prend ces valeurs sur la variable de classe (voir Liu et al., 1998, pour une implémentation). Dans cette section, néanmoins, nous développons une approche différente qui consiste à rechercher des règles d'association qui s'expriment différemment suivant la variable de classes. Dans la section 5.1, nous présentons d'abord le contexte de cette recherche, les données génomiques, puis la notion de règles différentiellement exprimées à laquelle nous avons eu recours pour les analyser.

5.1 Notion de règles d'association différentiellement exprimées

5.1.1 Données génomiques

Nous nous sommes intéressés au cas des données génomiques issues des puces à ADN, dans Lallich et al. (2006b). Cette technologie permet de mesurer le niveau d'expression de gènes (à partir de l'ARN produites par ces gènes) chez un patient. Grâce à elle, les biologistes peuvent mesurer l'expression simultanée de plusieurs gènes chez un même patient, afin de les étudier sous des conditions différentes, souvent liées à la présence ou l'absence d'une maladie. L'objectif est bien sûr de pouvoir déterminer quelles sont les gènes qui sont spécifiques de telle ou telle maladie, afin de mettre en œuvre des outils de dépistage ou d'orienter la recherche d'un traitement. Afin de ne pas omettre un gène important dans la différenciation des classes de patients sous les différentes conditions d'intérêt, les biologistes ont pris l'habitude de mesurer le niveau d'expression d'un très grand nombre de gènes à chaque étude. De ce fait, les données d'expression de gènes concernent souvent plusieurs milliers de gènes pour chaque patient. Par ailleurs, le coût de cette technologie limite souvent son application à quelques dizaines ou quelques centaines de patients.

On se retrouve alors face à des données particulières pour l'ECD, qui comportent un très grand nombre d'attributs (les gènes) pour un tout petit nombre d'exemples (les patients). Ces données sont évidemment touchées par les problèmes spécifiques des EGD (voir la section 4.1 du chapitre 1). Certaines de ces données sont accessibles publiquement. En particulier, dans Lallich et al. (2006b), nous nous sommes servis des données tirées de Golub et al. (1999). Ces données représentent le niveau d'expression (ou valeurs d'expression) de 7 129 gènes chez des patients atteints de leucémie, avec 34 patients de type *Acute Myeloid Leukemia* (AML) et 48 patients de type *Acute Lymphoblastic Leukemia* (ALL). Ces données sont disponibles sous la forme de données *Affymetrix*® standardisées, ce qui permet de connaître pour chaque patient, en plus de sa classe d'appartenance à l'un des deux types de cancer, la présence ou l'absence de l'expression d'un gène ainsi que le niveau d'expression de ces gènes. On dispose donc de deux matrices attributs/valeurs, la première étant composée de valeurs booléennes (1, lorsque le gène est exprimé chez le patient, 0 sinon[3]) et la deuxième de valeurs numériques normalisées par *Affymetrix* qui nous renseignent sur le niveau d'expression du gène.

[3]En fait, *Affymetrix* fournit trois valeurs pour chaque gène : présent, absent ou marginal. Ces valeurs sont déterminées suivant un test statistique (test des rangs de Wilcoxon) conduit sur les intensités des sondes spécifiques du gène considéré (aussi appelé *probe-set*), en tenant compte des hybridations qui sont correctes (*perfect match* dans la terminologie *Affymetrix*) et incorrectes (*mismatch*). Cette procédure est détaillée dans le document technique disponible à l'adresse : http ://www.affymetrix.com/support/technical/whitepapers/sadd_whitepaper.pdf . Dans nos travaux, nous avons considéré les gènes dont la valeur d'expression est marginale comme absents, afin de nous retrouver dans le cas booléen.

5.1.2 Problématique.

Face à ces données, les biologistes sont bien sûr intéressés par des méthodes de classification automatique, par exemple pour mettre en place des procédures de dépistage. Mais, compte tenu du prix des puces à ADN, il est important de cibler les gènes les plus impliqués pour réduire le nombre de mesures, tout en gardant une qualité prédictive au moins égale. Ce ciblage est également important pour comprendre le mécanisme d'une maladie en vue de son étude. Ici, la découverte des gènes impliqués dans tel ou tel type de pathologie est donc primordiale.

D'un autre côté, les gènes sont loin d'être indépendants les uns des autres. Au contraire, l'expression des gènes est souvent contrôlée par des réseaux de régulation (Gutierrez-Rios et al., 2003) qui entraînent la co-expression ou co-répression de plusieurs gènes. Il en résulte qu'une simple sélection des gènes différentiellement exprimés n'est pas suffisante pour découvrir les gènes les plus spécifiques d'une maladie. Par exemple, sur les données ALL/AML de Golub et al. (1999), une sélection des gènes différentiellement exprimés conserve près de 1000 gènes, même avec un contrôle du risque multiple (voir section 5.2). Si ces gènes sont bien caractéristiques d'une maladie, ils sont encore trop nombreux pour apporter une information réellement exploitable par un biologiste.

Partant de cette réalité biologique, nous nous sommes intéressés à la co-expression des gènes en étudiant des règles simples, du type « si le gène A est présent, alors il est très probable que le gène B le soit aussi ». Comme nous sommes dans un contexte supervisé, où chaque gène est étudié dans deux classes de cancer, nous avons recherché parmi ce type de règles, celles qui s'expriment différemment suivant la catégorie de cancer. Nous parlons alors de règles différentiellement exprimées pour caractériser des règles qui sont présentes dans un cancer et non dans l'autre.

5.1.3 Procédure d'extraction

Dans Lallich et al. (2006b), nous avons proposé une procédure d'extraction de ces règles. Par la suite, nous utilisons les notations suivantes :

- n : le nombre de patients ; n_i : le nombre de patient de classe i ; d : le nombre de gènes.
- X^i : une matrice booléenne de taille $n_i \times d$. $X^i_{jk} = 1$ si le patient j exprime le gène k, 0 sinon.
- p_{a_i} : la probabilité de l'antécédent sur X^i et p_{b_i} celle du conséquent sur X^i.
- $Sup_i(r)$ et $Conf_i(r)$: le support et la confiance de la règle r sur X^i ; m : le nombre de règles extraites.

Pour identifier les règles différentiellement exprimées, nous procédons en 2 étapes.

Étape [a] : sélection des gènes différentiellement exprimés. Au cours de cette étape, nous recherchons les gènes différentiellement exprimés suivant la classe du cancer. L'objectif de cette étape est de limiter la recherche des règles à ces seuls gènes afin de réduire le nombre de règles découvertes, tout en se focalisant sur les gènes qui expliquent le mieux la classe du cancer. Pour déterminer si un gène est différentiellement exprimé, nous avons utilisé un test-t de Student sur la matrice du niveau d'expression des gènes. Cela revient bien sûr à conduire une multitude de tests. Pour contrôler le risque de 1$^{\text{e}}$ espèce à un niveau de 5%, nous avons utilisé la procédure FDR proposée par Benjamini et Hochberg (1995) (voir section 2.2). Par la suite, d devient le nombre de gènes différentiellement exprimés et X^i la matrice réduite à ces seuls gènes.

Étape [b] : sélection des règles différentiellement exprimées. À partir de cette matrice, nous avons extrait les règles différentiellement exprimées. Pour cela, nous proposons la procédure suivante :

- *Extraction des règles.* Construire $\mathcal{R}_i$, la base des règles d'association de X^i en utilisant un algorithme de type support-confiance tel que $\forall r \in \mathcal{R}_i,\ Sup_i(r) > 0.25$ et $Conf_i(r) > 0.5$.
- *Filtrage des règles admissibles.* En déduire $\mathcal{R} = \{r \in \mathcal{R}_1 \cup \mathcal{R}_2 \mid\ p_{a_1} > 0.5 \text{ et } p_{a_2} > 0.5\}$. De cette manière, on s'assure que chaque règle découverte sur une classe de cancer particulière soit valable sur l'autre classe de cancer, en imposant la présence de l'antécédent chez un minimum de 50% des patients, quelle que soit la classe considérée.
- *Mesure de l'intérêt d'une règle.* On considère ici la mesure d'intérêt $c(r) = Conf_1(r) - Conf_2(r),\ \forall r \in \mathcal{R}$. Une règle est donc d'autant plus intéressante que sa confiance sur une classe de cancer est élevée alors qu'elle est faible sur l'autre classe, de sorte que les règles caractéristiques d'une classe soient privilégiées par la mesure c.
- *Sélection des règles différentiellement exprimées.* La base $\mathcal{R}^*$ des règles réellement intéressantes suivant c est déterminée par l'application de $BSFD$ bilatéral sur $\mathcal{R}$ avec $V_0 = 10$ et $\delta = 0,05$.

5.2 Résultats et interprétation

En appliquant le FDR, le nombre de gènes est réduit de 7 129 à 980. La recherche des règles d'association par *Apriori* sur ces 980 gènes (en se limitant aux règles dont l'antécédent et le conséquent ne sont composés que d'un *item*) conduit à la découverte de 174 412 règles admissibles. Ici, on comprend donc bien la nécessité de réaliser une sélection des gènes. Enfin, en appliquant la procédure $BSFD$ bilatéral sur le critère c (avec un partage équitable du nombre de fausses découvertes V_0 et du risque δ à gauche et à droite, voir section 3.2), nous avons identifé 799 règles différentiellement exprimées (et validées

statistiquement). Dans notre cas, la procédure $BSFD$ bilatéral a sélectionné les règles telles que $c(r) < -0.76$ or $c(r) > 0.74$. Parmi ces règles, 235 sont caractéristiques de la classe AML et 564 de la classe ALL.

Bien qu'il soit possible d'imaginer une procédure de prédiction fondée sur ces règles, leur découverte ne constitue pas en soit un résultat directement exploitable par un biologiste. Nous avons donc continué l'étude en nous intéressant aux gènes impliqués dans ces règles. Les résultats sont particulièrement intéressants. Sur les 799 règles découvertes, seulement 26 gènes différents sont présents en tant que conséquentspatient des règles. Le tableau 4.4 en donne une liste exhaustive, en fournissant pour chacun d'entre eux leur nom, le nombre de règles dans lesquelles ils apparaissent (sur les 799, colonne #), la classe pour laquelle ces règles sont valides (classe), leur probabilité de présence dans chaque classe (p_{b_1} et p_{b_2}) et enfin leur rang lorsque les gènes sont classés par ordre croissant de leur *p-value* sur le test-t (rang).

Ces conséquents correspondent à des gènes différentiellement exprimés, à cause de l'étape [a] de la procédure. Pour l'un des deux cancers, ils sont fortement co-exprimés avec des gènes (les antécédents des règles) eux-mêmes différentiellement exprimés (toujours à cause de l'étape [a]). Par exemple, le gène M31211-s est co-exprimé avec 228 autres gènes pour les patients atteints par le cancer ALL. Mais le réel intérêt de ces co-expressions, c'est de n'être valable que pour une classe de cancer, alors même que les antécédents de la règle sont présents pour les deux classes (puisque nous avons sélectionné des règles dont $p_{a_1} > 0.5$ et $p_{a_2} > 0.5$). L'intérêt de ces 26 gènes n'est donc pas seulement une différence de leur niveau d'expression suivant les classes de cancer, mais le fait que le contexte qui permet leur expression à lui aussi disparu. Ce dernier point est important car il fait de ces gènes des indicateurs plus spécifiques et plus robustes de la différence entre les classes ALL et AML que d'autres gènes. Pourtant, en triant les gènes dans l'ordre croissant des *p-value*, ils ne correspondent pas aux 26 premiers (colonne rang de la table 4.4). Leur découverte apporte donc une information qualitative (ces gènes sont des indicateurs robustes des classes ALL et AML dont l'expression est déterminée par un certain contexte défini par les antécédents des règles) à une information seulement quantitative (la probabilité d'être différentiellement exprimé). Parce que ces gènes sont en petits nombres, qu'ils sont extraits avec leur contexte, il est possible d'envisager une exploitation de ce résultat par des biologistes qui permettrait de valider l'intérêt de ce résultat. Néanmoins, ce travail montre l'importance de la validation des découvertes et comment celle-ci conduit à la production d'une connaissance exploitable par un utilisateur final.

TAB. 4.4: Conséquents des règles intéressantes et valides découvertes avec $BSFD$

Nom	#	**Classe**	p_{b_1}	p_{b_2}	**Rang**
X95735	30	AML	0,09	0,82	1
M23197	4	AML	0,18	0,86	3
M84526	14	AML	0	0,71	4
U46499	2	AML	0,20	0,89	8
S50223	3	ALL	0,66	0,07	14
M31211-s	**228**	**ALL**	**0,95**	**0,21**	**22**
U05259-rna1	16	ALL	0,82	0,18	28
M96326-rna1	**185**	**AML**	**0,05**	**0,82**	**30**
M92287	30	ALL	0,91	0,25	41
L47738	**93**	**ALL**	**0,86**	**0,14**	**71**
U53468	1	ALL	0,59	0,04	73
M83233	19	ALL	0,73	0,07	75
M11722	**103**	**ALL**	**0,91**	**0,18**	**83**
M89957	2	ALL	0,68	0,11	136
M12959-s	3	ALL	0,93	0,32	183
D88270	8	ALL	0,73	0,11	197
L08895	2	ALL	0,77	0,21	224
L41870	2	ALL	0,84	0,25	258
Z14982-rna1	1	ALL	0,59	0,07	261
U79285	9	ALL	0,73	0,14	277
AB000449	10	ALL	0,77	0,14	298
D21262	11	ALL	0,82	0,18	343
D86983	2	ALL	0,73	0,11	472
X62535	15	ALL	0,86	0,21	581
U37352	1	ALL	0,84	0,32	617
X79865	5	ALL	0,84	0,21	629

6 Discussion

Dans ce chapitre, nous avons étudié la validation des procédures d'extraction de règles d'association. Ces procédures fournissent généralement à l'utilisateur un grand nombre de règles (d'autant plus grand, d'ailleurs, que les données sont volumineuses), dont toutes ne sont pas intéressantes. Pour filtrer la base des règles admissibles, nous proposons une étape de contrôle de leur validité à l'aide d'un critère statistique. Ce contrôle amène à réaliser un grand nombre de tests (un par règle admissible), ce qui pose le problème du contrôle du risque multiple. Face à ce problème, nous proposons $BFSD$, une procédure de contrôle du risque multiple fondée sur le *bootstrap* dont l'originalité est de contrôler le nombre maximum de faux positifs à un risque δ près, alors que les procédures de type FDR contrôlent la moyenne du nombre de faux positifs. Ce critère, appelé $UAFWER$, augmente la puissance du test. En outre, $BSFD$ présente 2 avantages :

- non-paramétrique : l'application de $BSFD$ ne requiert pas de connaître la loi du critère contrôlé sous l'hypothèse nulle.
- multi-critères : $BSFD$ est applicable à plusieurs critères simultanément.

Ces avantages font de $BSFD$ une procédure de contrôle du risque multiple particulièrement adaptée à la sélection de règles d'association. Nous l'avons donc exploitée dans ce contexte dans deux situations, l'une supervisée et l'autre non.

En situation non-supervisée, nous définissons une règle valide comme une règle qui renforce la probabilité du conséquent. Nous avons montré que le cœfficient de corrélation R est une mesure priviligiée pour contrôler cette validité. Lors de nos expérimentations, nous avons comparé le comportement de $BSFD$ avec plusieurs autres méthodes de contrôle du risque multiple sur différentes bases de transactions. Ces expérimentations ont d'abord montré que la validation des règles est une étape importante qui permet de réduire de manière conséquente la base des règles à étudier. Ensuite, elles ont confirmé que $BSFD$ contrôle de manière légèrement plus sévère le nombre de fausses découvertes acceptées. Enfin, la validation simultanée par $BSFD$ des critères d'extraction (support et confiance) et de R a permis d'obtenir des règles moins spécifiques des transactions sur lesquelles elles ont été extraites. Cependant, cette dernière procédure réduit très fortement le nombre de règles validées.

En situation supervisée, nous avons introduit une notion nouvelle, celle des règles d'association différentiellement exprimées. Une règle différentiellement exprimée est une règle extraite à partir des transactions d'une modalité de la variable de classe et qu'on ne retrouve pas à partir des transactions des autres modalités, alors même que l'antécédent de la règle est rencontré fréquemment dans toutes les modalités. L'intérêt d'une telle règle est d'être spécifique d'une modalité de la variable de classe. Nous avons proposé une procédure pour extraire ces règles tout en s'assurant qu'elles soient valides. Cette validation

repose sur des mesures de la confiance, estimée chacune sur les transactions relatives à une modalité. En composant ces différentes confiances pour une règle, nous avons construit un critère de validation des règles différentiellement exprimées. Enfin, nous avons eu recours à $BSFD$ pour contrôler le risque qui résulte de la multiplicité des tests sur ce critère. Ici, $BSFD$ montre tout son intérêt puisqu'on ne connaît pas la loi sous l'hypothèse nulle du critère contrôlé, qui plus est bilatéralement. Cette procédure d'extraction de règles différentiellement exprimées a été appliquée sur des données génomiques. Grâce à la validation statistique des connaissances, nous avons pu extraire au final un petit ensemble de connaissances valides, ici des gènes, exploitables par un utilisateur final.

Ce chapitre illustre bien l'intérêt de l'étape de validation dans le processus d'ECD. Grâce à elle, la connaissance extraite par les méthodes de fouille est réduite, ce qui aide à se concentrer sur les connaissances potentiellement intéressantes. Dans le cas de données volumineuses, cette étape devient critique pour pouvoir exploiter un résultat, comme le montre nos expérimentations sur les données génomiques.