Echantillonnage de motifs avec une contrainte de fréquence

Arnaud Soulet

Université de Tours, LIFAT, Blois
firstname.lastname@univ-tours.fr

Résumé. L'échantillonnage de motifs est une technique récente de découverte de motifs favorisant l'interactivité avec l'utilisateur. Son principe est de tirer aléatoirement un motif proportionnellement à son intérêt. Malheureusement, les tirages peuvent se focaliser sur une partie de l'espace de recherche avec des motifs peu fréquents mais extrêmement nombreux. Il serait bien possible d'échantillonner des motifs et d'éliminer ceux non-fréquents, mais le taux de rejet s'avère souvent trop élevé. Dans cet article, nous proposons la première méthode efficace d'échantillonnage de motifs avec une contrainte de fréquence minimale. Elle s'appuie sur la suppression des items non-fréquents (opération de réduction) et sur la projection de la base de données sur chacun des items (opération de projection). En combinant ces deux opérations, nous proposons une méthode générique qui revient à éliminer tous les motifs contenant un couple non-fréquent d'items. Nos expérimentations montrent que notre méthode réduit considérablement le taux de rejet.

1 Introduction

Ces dernières années, une large partie des méthodes de découverte de motifs s'est orientée vers des approches favorisant l'interaction avec l'utilisateur afin d'intégrer ses retours dans le processus de découverte (Leeuwen, 2014). Pour cela, il est nécessaire de disposer de techniques d'extraction non-exhaustives afin d'être assez rapide pour extraire des motifs à la volée tout en s'appuyant sur des mesures d'intérêt complexes pouvant être mises à jour à chaque itération. Typiquement, ces méthodes d'extraction s'appuient sur des algorithmes de recherche en faisceau (Leeuwen, 2014), de recherche arborescente Monte-Carlo (Bosc et al., 2018) ou encore, d'échantillonnage de motifs (Boley et al., 2011). En particulier, les méthodes fondées sur l'échantillonnage de motifs ont reçu beaucoup d'attention avec la proposition de systèmes interactifs (Giacometti et Soulet, 2017; Dzyuba et al., 2017) ou d'algorithmes anytime pour extraire des données aberrantes (Giacometti et Soulet, 2016).

Plus précisément, l'échantillonnage de motifs consiste à tirer un motif X avec une probabilité proportionnelle à son intérêt $m(X)$. Par exemple avec la fréquence, un motif X deux fois plus fréquent qu'un motif Y aura deux fois plus de chance d'être tiré. Malheureusement, l'échantillonnage de motifs a souvent tendance à se focaliser sur des parties de l'espace de recherche avec une densité forte de motifs peu intéressants (i.e., beaucoup de motifs avec des valeurs faibles pour m). Pour minimiser ce phénomène, appelé « malédiction de la longue traine », Diop et al. (2018) ont proposé d'ajouter une contrainte de longueur maximale sur les

motifs tirés car les motifs courts sont les plus généraux et ils ont tendance à être plus fréquents. Il n'en demeure pas moins que certains motifs courts sont inintéressants au sens de la mesure m et surtout, que ce filtrage supprime des motifs longs intéressants. Par ailleurs, il est difficile de choisir le seuil de longueur adéquat inférant une mesure m suffisante. Il serait donc particulièrement intéressant de pouvoir ajouter une contrainte retirant seulement les motifs qui ont une faible mesure pour m. Nous montrons dans cet article comment pousser la contrainte de fréquence minimale dans l'échantillonnage de motifs.

Nos contributions dans cet article sont les suivantes :

— Nous proposons une méthode générique d'échantillonnage qui intègre un seuil minimal de fréquence pour retirer les motifs non-fréquents. Pour cela, nous décomposons la base de données sur plusieurs bases de données projetées desquelles nous supprimons les items non-fréquents. Notre méthode est générique puisqu'elle permet de pousser la contrainte de fréquence minimale dans n'importe quelle méthode d'échantillonnage de motifs selon la fréquence.

— Nous évaluons notre méthode sur les benchmarks de l'UCI. Ces résultats expérimentaux montrent la réduction significative du taux de rejet par rapport à une méthode naïve montrant la faisabilité de l'échantillonnage de motifs sous contrainte de fréquence.

La suite de cet article est organisée de la manière suivante. La section 2 situe notre travail dans le domaine de l'extraction de motifs. La section 3 introduit les notations et définitions de l'article et elle formule notre problème. Nous présentons notre proposition dans la section 4 en soulignant les défis techniques à relever et en introduisant nos deux principaux outils à savoir la réduction et la projection de la base de données. Nous évaluons le taux de rejet de la procédure avec projection et le volume de données qu'elle requiert dans la section 5. Enfin, nous concluons l'article.

2 Travaux relatifs

Al Hasan et Zaki (2009) ont introduit le principe de l'échantillonnage de motifs qui vise à tirer des motifs avec une distribution de probabilité proportionnelle à leur intérêt. Actuellement, les techniques d'échantillonnage se répartissent principalement en deux grandes familles : les méthodes stochastiques (Al Hasan et Zaki, 2009) et les méthodes en plusieurs étapes (Boley et al., 2011). La première famille (Al Hasan et Zaki, 2009) repose sur les méthodes de Monte-Carlo par chaînes de Markov. L'idée est que la loi stationnaire de la marche aléatoire corresponde à la distribution à échantillonner. L'avantage de telles approches stochastiques est de pouvoir considérer des mesures variées et même des contraintes. Malheureusement, leur vitesse de convergence est souvent très lente. La seconde famille (Boley et al., 2011) repose sur l'enchainement de plusieurs tirages successifs. En choisissant judicieusement les différentes distributions de tirage, il est alors possible d'obtenir un tirage exact selon la distribution désirée. Dans ce travail, nous avons opté pour une telle approche pour sa rapidité et son exactitude.

Dans les méthodes en plusieurs étapes, chacune des étapes consiste à répartir les occurrences de motifs en différents groupes et à tirer un groupe proportionnellement à son poids. Au niveau de la dernière étape, une occurrence est choisie aléatoirement au sein de son groupe proportionnellement à son poids. Quelles que soient les spécificités du problème, la difficulté est donc de trouver une décomposition judicieuse en plusieurs étapes. Par exemple, la méthode originelle de Boley et al. (2011) regroupe les occurrences par transaction. Pour traiter les motifs

séquentiels, Diop et al. (2018) ajoute une étape supplémentaire pour regrouper les occurrences selon leur longueur au sein de chaque transaction. Dans le contexte des bases de données distribuées, Diop et al. (2022) utilise une étape préliminaire pour tirer la bonne base de données distribuée. D'un point de vue technique, notre proposition s'inscrit dans cette direction avec une étape de tirage dans une base de données projetée. Mais, de manière originale, l'occurrence finalement retournée agrège des parties d'occurrences obtenues à différentes étapes.

A la différence de l'échantillonnage, la découverte de motifs classique énumère tous les motifs satisfaisant un prédicat logique de sélection (aussi appelé contrainte) pour éliminer les motifs inintéressants (Mannila et Toivonen, 1997). La contrainte de fréquence minimale introduite par Agrawal et al. (1993) est l'une des plus populaires parmi l'ensemble des contraintes. Cette popularité s'explique par l'efficacité de cet élagage éliminant de nombreux motifs dont la probabilité d'apparition est trop faible pour être jugés pertinents (Agrawal et al., 1994). Même lorsque d'autres contraintes de filtrage sont requises, la contrainte de fréquence minimale est souvent utilisée conjointement (Ng et al., 1998). La technique d'échantillonnage proposée par Al Hasan et Zaki (2009) intègre naturellement la contrainte de fréquence minimale lors de sa marche aléatoire. A l'inverse, aucune méthode d'échantillonnage de motifs en plusieurs étapes n'a été proposée permettant de conjuguer l'efficacité de la proposition de Boley et al. (2011) avec un élagage suivant la fréquence. A notre connaissance, seule la contrainte de longueur maximale et ses variantes (Diop et al., 2022) ont été poussées dans ces méthodes d'échantillonnage de motifs pour les séquences et pour les itemsets. Il est bien plus difficile de pousser la contrainte de fréquence minimale qui n'est pas une contrainte syntaxique i.e., vérifier la contrainte requiert de considérer la base de données dans son intégralité.

3 Préliminaires

3.1 Définitions

Soit $\mathcal{I}$ un ensemble de littéraux distincts appelés *items*, un itemset (ou un motif) est un sous-ensemble de $\mathcal{I}$. Nous considérons une relation d'ordre totale arbitraire sur $\mathcal{I}$ dénotée par $<_{\mathcal{I}}$ (par exemple, l'ordre alphabétique). Le langage des itemsets correspond à $\mathcal{L} = 2^{\mathcal{I}}$. Une base de données transactionnelles est un multi-ensemble d'itemsets de $\mathcal{L}$. Chacun de ces itemsets, généralement appelé *transaction*, est une observation des données. Une transaction t contient l'*occurrence* du motif X ssi $X \subseteq t$. La *fréquence* d'un itemset X dans la base de données $\mathcal{D}$ est son nombre d'occurrences : $\mathsf{freq}(X, \mathcal{D}) = |\{t \in \mathcal{D} \ : \ X \subseteq t\}|$. Étant donné un seuil minimal de fréquence γ, un motif est fréquent lorsque sa fréquence est supérieure ou égale à γ. Pour intégrer cette contrainte à la mesure de fréquence, nous définissons la *fréquence contrainte* comme $\mathsf{freq}_\gamma(X, \mathcal{D}) = \mathsf{freq}(X, \mathcal{D})$ si $\mathsf{freq}(X, \mathcal{D}) \geq \gamma$, et 0 sinon. La table 1 montre une base de données jouet contenant 5 transactions $t_1, \ldots, t_5$ décrites par 8 items $A, \ldots, H$. Pour $\gamma = 3$, l'itemset FGH est fréquent car sa fréquence est 3 (i.e., $\mathsf{freq}_3(FGH, \mathcal{D}) = 3$) tandis que l'itemset AB n'est pas fréquent avec $\mathsf{freq}(AB, \mathcal{D}) = 1$ (i.e., $\mathsf{freq}_3(AB, \mathcal{D}) = 0$).

3.2 Formulation du problème

Soit Ω une population et $f : \Omega \to [0, 1]$ une mesure, la notation $x \sim f(\Omega)$ signifie que l'élément x est tiré au hasard dans Ω avec une distribution de probabilité $\pi(x) =$

$\mathcal{D}$

Trans.	Items							
t_1				D			G	H
t_2			C			F	G	H
t_3	A	B	C	D	E	F	G	H
t_4		B			E	F	G	H
t_5			C	D	E	F		

TAB. 1 – Une base de données $\mathcal{D}$ avec en gris les items non-fréquents pour $\gamma = 3$

$f(x)/\sum_{y\in\Omega} f(y)$. Le problème usuel de l'échantillonnage de motifs selon la fréquence revient à tirer avec remise un motif X avec $X \sim \mathsf{freq}(\mathcal{L}, \mathcal{D})$. Dans cet article, nous ajoutons une contrainte minimale de fréquence qui pourrait s'écrire : $X \sim \mathsf{freq}(\mathcal{L}, \mathcal{D})$ subject to $\mathsf{freq}(X, \mathcal{D}) \geq \gamma$. Pour éviter toute ambiguïté, nous préférons formaliser notre problème avec la fréquence contrainte freq_γ de la manière suivante :

Etant donnés une base de données $\mathcal{D}$ et un seuil minimal de fréquence γ, notre objectif est de proposer une procédure d'échantillonnage $\mathcal{S}_{\mathsf{freq}_\gamma}$ qui sélectionne un itemset X dans $\mathcal{L}$ avec une probabilité déterminée par son poids relatif $\mathsf{freq}_\gamma(X, \mathcal{D})$: $X \sim \mathsf{freq}_\gamma(\mathcal{L}, \mathcal{D})$.

4 Procédure d'échantillonnage avec des bases projetées

4.1 Défis et idée clé de notre approche

Une approche naïve pour échantillonner des itemsets avec une contrainte de fréquence minimale consisterait à utiliser une procédure d'échantillonnage selon la fréquence pour tirer des itemsets et rejeter ceux qui ne sont pas fréquents. Plus formellement, nous considérons $\mathcal{S}_{\mathsf{freq}}$ une procédure d'échantillonnage selon la fréquence comme la procédure aléatoire en deux étapes de Boley et al. (2011). Nous introduisons alors une procédure de rejet $\mathcal{R}_\gamma$ qui répète $\mathcal{S}_{\mathsf{freq}}$ sur $\mathcal{D}$ tant que le motif tiré $\mathcal{S}_{\mathsf{freq}}(\mathcal{L}, \mathcal{D})$ a une fréquence inférieure à γ. On a donc l'équivalence suivante : $\mathcal{S}_{\mathsf{freq}_\gamma}(\mathcal{L}, \mathcal{D}) \Leftrightarrow \mathcal{R}_\gamma(\mathcal{S}_{\mathsf{freq}}, \mathcal{L}, \mathcal{D})$. Malheureusement, le taux de rejet de la procédure $\mathcal{R}_\gamma(\mathcal{S}_{\mathsf{freq}}, \mathcal{L}, \mathcal{D})$ augmente très rapidement avec le seuil minimal de fréquence (voir les expérimentations de la section 5). L'échantillonnage est une procédure extrêmement rapide mais le calcul de la fréquence pour vérifier l'acceptation ou le rejet entraine un surcout important. Dans notre exemple, avec la base de données de la table 1 et un seuil $\gamma = 3$, le taux de rejet est de 86% notamment à cause des nombreux motifs non-fréquents qui contiennent soit A, soit B. Bien entendu, il faudrait éviter de considérer ces itemsets non-fréquents dans notre procédure d'échantillonnage. La propriété suivante formalise cette observation :

Propriété 1 (Elagage d'itemsets) *Etant donnés une base de données $\mathcal{D}$, un seuil minimal de fréquence γ et une procédure $\mathcal{S}_{\mathsf{freq}}$ d'échantillonnage de motifs suivant la fréquence (sans contrainte), l'équivalence $\mathcal{S}_{\mathsf{freq}_\gamma}(\mathcal{L}, \mathcal{D}) \Leftrightarrow \mathcal{R}_\gamma(\mathcal{S}_{\mathsf{freq}}, \mathcal{L} \setminus R, \mathcal{D})$ est vraie pour tout ensemble R d'itemsets non-fréquents (i.e., $\forall X \in R$, nous avons $\mathsf{freq}(X, \mathcal{D}) < \gamma$) et le taux de rejet,*

dénoté $\rho_{\gamma,\mathcal{D}}$, de la procédure $\mathcal{R}_\gamma(\mathcal{S}_{\mathsf{freq}}, \mathcal{L} \setminus R, \mathcal{D})$ est :

$$\rho_{\gamma,\mathcal{D}}(R) = 1 - \frac{\sum_{X \in \mathcal{L}} \mathsf{freq}_\gamma(X, \mathcal{D})}{\sum_{X \in \mathcal{L} \setminus R} \mathsf{freq}(X, \mathcal{D})}$$

Cette propriété signifie qu'un tirage aléatoire reste proportionnel à la fréquence contrainte même si certains motifs non-fréquents sont ignorés. Par manque de place, les preuves ont été omises dans cet article. Néanmoins, ce résultat s'explique par la valeur nulle pour freq_γ des motifs contenus dans R. De manière intéressante, il est facile de voir que le taux de rejet sera d'autant plus faible que l'ensemble R sera grand. Revenons à l'exemple de la table 1 où $\gamma = 3$, la procédure naïve $\mathcal{R}_\gamma(\mathcal{S}_{\mathsf{freq}}, \mathcal{D})$ va considérer 203 motifs de fréquence 1 et 40 de fréquence 2 qui seront rejetés, ce qui donne bien un taux de rejet de $1 - (8 \times 3 + 4 \times 4 + 1 \times 5)/(203 \times 1 + 40 \times 2 + 8 \times 3 + 4 \times 4 + 1 \times 5) = 1 - 45/328 = 86\%$ avec la procédure $\mathcal{R}_\gamma(\mathcal{S}_{\mathsf{freq}}, \mathcal{L}, \mathcal{D})$. Idéalement, nous voudrions supprimer les 243 motifs non-fréquents.

Dans la suite, nous cherchons à trouver le plus grand ensemble de motifs non-fréquents R afin de minimiser le taux de rejet. Cette tâche est rendue ardue par deux contraintes :

C1 Pour rester efficace, l'échantillonnage doit se faire sans la matérialisation des motifs fréquents. Nous devrons donc isoler un maximum de motifs non-fréquents à moindre coût.

C2 En réalité, les procédures d'échantillonnage traditionnelles $\mathcal{S}_{\mathsf{freq}}(\mathcal{L}, \mathcal{D})$ considèrent $\mathcal{L}$ dans son intégralité. Nous ne pourrons pas retirer les motifs non-fréquents R : $\mathcal{S}_{\mathsf{freq}}(\mathcal{L} \setminus R, \mathcal{D})$.

Par conséquent, l'idée clé de notre approche est de modifier la base de données $\mathcal{D}$ pour supprimer directement les motifs non-fréquents de l'échantillonnage. La section 4.2 propose de supprimer les items non-fréquents de $\mathcal{D}$ (opération de réduction) et la section 4.3 amplifie cette réduction en l'appliquant sur les bases projetées de $\mathcal{D}$ (opération de projection).

4.2 Suppression des items non-fréquents

Pour rappel, la fréquence décroit avec la spécialisation à savoir si un itemset est non-fréquent, tous ses sur-ensembles le sont également. Or, la suppression d'un item i de $\mathcal{D}$ retire aussi tous ses sur-ensembles de $\mathcal{D}$. Cela signifie que la suppression d'un item non-fréquent retire aussi d'autres itemsets non-fréquents. Pour notre exemple de la table 1 où $\gamma = 3$, la base de données réduite $\mathcal{D}_3$ revient à supprimer les items grisés à savoir A et B. Cela entraine la suppression de 196 itemsets incluant A ou B : AC, AD, etc. Nous formalisons cette réduction de la base de données de la manière suivante :

Définition 1 (Base de données réduite) *Soient une base de données $\mathcal{D}$ et un seuil minimal de fréquence γ, la base de données réduite $\mathcal{D}_\gamma$ reprend la base de données $\mathcal{D}$ en retirant les items non-fréquents : $\mathcal{D}_\gamma = \{t_\gamma \subseteq t \ : \ t \in \mathcal{D} \wedge (i \in t_\gamma \Leftrightarrow \mathsf{freq}(\{i\}, \mathcal{D}) \geq \gamma)\}$*

Cette réduction de la base de données est une technique déjà exploitée par les méthodes d'extraction de motifs sous contrainte de fréquence minimale (Han et al., 2000; Bonchi et al., 2003). La propriété suivante démontre son utilité pour l'échantillonnage sous contrainte :

Propriété 2 *Soient une base de données $\mathcal{D}$, un seuil minimal de fréquence γ et une procédure $\mathcal{S}_{\mathsf{freq}}$ d'échantillonnage de motifs suivant la fréquence (sans contrainte), un processus de tirage avec rejet d'un motif tiré dans $\mathcal{D}_\gamma$ est équivalent à un processus de tirage avec rejet d'un motif tiré dans $\mathcal{D}$: $\mathcal{R}_\gamma(\mathcal{S}_{\mathsf{freq}}, \mathcal{L}, \mathcal{D}_\gamma) \Leftrightarrow \mathcal{R}_\gamma(\mathcal{S}_{\mathsf{freq}}, \mathcal{L}, \mathcal{D})$*

$\mathcal{D}^{(C)}$

Trans.	Items				
t_2			F	G	H
t_3	D	E	F	G	H
t_5	D	E	F		

$\mathcal{D}^{(D)}$

Trans.	Items			
t_1			G	H
t_3	E	F	G	H
t_5	E	F		

$\mathcal{D}^{(E)}$

Trans.	Items		
t_3	F	G	H
t_4	F	G	H
t_5	F		

$\mathcal{D}^{(F)}$

Trans.	Items	
t_2	G	H
t_3	G	H
t_4	G	H
t_5		

$\mathcal{D}^{(G)}$

Trans.	Items
t_1	H
t_2	H
t_3	H
t_4	H

$\mathcal{D}^{(H)}$

Trans.	Items
t_1	
t_2	
t_3	
t_4	

TAB. 2 – Bases de données projetées de $\mathcal{D}$ avec en gris les items non-fréquents pour $\gamma = 3$

Cette propriété se démontre facilement en s'appuyant sur la propriété 1 et en observant que $\mathcal{R}_\gamma(\mathcal{S}_{\mathsf{freq}}, \mathcal{L}, \mathcal{D}_\gamma) \Leftrightarrow \mathcal{R}_\gamma(\mathcal{S}_{\mathsf{freq}}, \mathcal{L} \setminus R, \mathcal{D})$ où les motifs de $\mathcal{D}$ absents dans $\mathcal{D}_\gamma$ forment l'ensemble $R : R = \{X \subseteq t \ : \ t \in \mathcal{D} \wedge (\exists i \in X)(\mathsf{freq}(\{i\}, \mathcal{D}) < \gamma)\}$. En comparaison de $\mathcal{R}_\gamma(\mathcal{S}_{\mathsf{freq}}, \mathcal{L}, \mathcal{D})$, la suppression des motifs non-fréquents diminue le taux de rejet : $1 - (\sum_{X \in \mathcal{L}} \mathsf{freq}_\gamma(X, \mathcal{D}))/(\sum_{t \in \mathcal{D}_\gamma} 2^{|t|})$.

Cette réduction respecte bien les deux contraintes définies dans la section précédente. D'une part, les items non-fréquents sont déterminés lors de la lecture de la base de données et la réduction ne nécessite aucun surcout (respect de la contrainte C1). D'autre part, une procédure d'échantillonnage traditionnelle $\mathcal{S}_{\mathsf{freq}}$ peut bénéficier directement de la réduction en opérant directement sur la base de données (respect de la contrainte C2). Néanmoins, la réduction de la base de données manque de subtilité. En pratique, l'élagage induit par la contrainte de fréquence minimale est plus avantageux à partir des itemsets de taille 2.

4.3 Projection de la base de données suivant les items

Base de données projetée Nous proposons d'étendre l'élagage à tous les motifs contenant au moins une paire d'items non-fréquente en projetant la base de données sur chacun de ses items – revenant à une base de données de profondeur 1 dans (Pei et al., 2004). Pour commencer, nous rappelons la notion de base de données projetée :

Définition 2 (Base de données projetée) *Soient une base de données $\mathcal{D}$ et un item $i \in \mathcal{I}$, la base de données projetée $\mathcal{D}^{(i)}$ regroupe toutes les transactions contenant i en ne conservant que les items j plus grand que i (au sens de la relation d'ordre $<_\mathcal{I}$) :* $\mathcal{D}^{(i)} = \{t_i \subseteq t \ : \ t \in \mathcal{D} \wedge i \in t \wedge (j \in t_i \Leftrightarrow i <_\mathcal{I} j)\}$

Une base de données projetée suivant l'item i correspond donc tout simplement aux transactions contenant i où tous les items inférieurs à i ont été supprimés. Bien sûr, il est aussi possible de réduire une base de données projetée $\mathcal{D}^{(i)}$ pour donner $\mathcal{D}^{(i)}_\gamma$. La table 2 explicite les 6 bases de données projetées correspondant aux 6 items fréquents de $\mathcal{D}$ (voir la table 1).

Procédure avec projection Comme annoncé dans la section 2, nous allons ajouter une étape de tirage avec les bases de données projetées. Plus précisément, il suffit de tirer une base de données projetée $\mathcal{D}_\gamma^{(i)}$ proportionnellement au nombre d'occurrences quelle contient, puis d'y tirer un itemset Y comme suffixe pour former le motif $\{i\} \cup Y$. Admettons que la base de données projetée $\mathcal{D}_\gamma^{(C)}$ soit tirée, on pourra alors échantillonner soit le motif $\emptyset$, soit le motif F pour former au final C ou CF. Il est clair que l'élimination des items D, E, G et H empêche de tirer des motifs qui aurait été tirés avec $\mathcal{S}_{\mathsf{freq}}(\mathcal{L}, \mathcal{D}_\gamma)$ (comme CD, CDE, etc). L'algorithme 1 retourne un motif fréquent par rapport à γ qui a été échantillonné dans $\mathcal{D}$ selon la fréquence. Naturellement, cette procédure prend en entrée le jeu de données $\mathcal{D}$ et le seuil minimal de fréquence γ. De plus, par généricité, elle prend aussi en argument une méthode d'échantillonnage suivant la fréquence dans un jeu de données $\mathcal{D}$ (sans considérer de contrainte de fréquence). Dans nos expérimentations, nous utilisons par exemple la procédure aléatoire en deux étapes proposée par Boley et al. (2011).

Algorithm 1 Procédure avec projection de tirage de motifs sous contrainte de fréquence

Input: Une base de données $\mathcal{D}$, un seuil minimal de fréquence γ et une procédure $\mathcal{S}_{\mathsf{freq}}$ d'échantillonnage de motifs suivant la fréquence (sans contrainte)
Output: Un itemset X tiré aléatoirement suivant la fréquence tel que $\mathsf{freq}(X, \mathcal{D}) \geq \gamma$
1: Soit $\omega(i) := \sum_{t \in \mathcal{D}_\gamma^{(i)}} 2^{|t|}$ pour tout $i \in \mathcal{I}$
2: Soit $\Omega := |\mathcal{D}| + \sum_{i \in \mathcal{I}} \omega(i)$
3: **repeat**
4: Tirer uniformément un entier u entre 1 et Ω : $u \sim \mathsf{unif}(\{1, \ldots, \Omega\})$
5: **if** $u \leq |\mathcal{D}|$ **then return** $\emptyset$
6: Tirer un item i proportionnellement à ω : $i \sim \omega(\mathcal{I})$
7: Tirer un itemset suffixe Y dans $\mathcal{D}_\gamma^{(i)}$ suivant la fréquence : $Y := \mathcal{S}_{\mathsf{freq}}(\mathcal{L}, \mathcal{D}_\gamma^{(i)})$
8: $X := \{i\} \cup Y$
9: **until** $\mathsf{freq}(X, \mathcal{D}) \geq \gamma$
10: **return** X

Les lignes 1 et 2 initialisent la procédure en faisant un prétraitement qui peut être effectué une seule fois pour échantillonner plusieurs motifs. Il consiste à calculer le nombre d'occurrences présentes dans chacune des bases de données projetées $\mathcal{D}_\gamma^{(i)}$ où i est un item (ligne 1). On notera que $\omega(i) = 0$ si l'item i n'est pas fréquent car $\mathcal{D}_\gamma^{(i)}$ est vide. Ensuite, ces valeurs sont immédiatement exploitées à la ligne 2 pour calculer Ω qui est le nombre d'occurrences contenues dans la base de données $\mathcal{D}$ (i.e., les $|\mathcal{D}|$ occurrences de l'ensemble vide plus les occurrences contenues dans chacune des bases projetées). L'échantillonnage effectif d'un motif X est réalisé entre les lignes 4 et 8 imbriquées dans une boucle qui se répète tant que la fréquence de X est inférieure au seuil minimal de fréquence γ (condition de la ligne 9). Chaque répétition correspond donc au rejet d'un motif. Pour commencer, la ligne 4 tire un entier u entre 1 et Ω. Si cet entier u est inférieur à la cardinalité de la base de données $\mathcal{D}$, la ligne 5 retourne l'ensemble vide comme motif. Sinon, les lignes 6 à 8 s'appuient sur les bases de données projetées pour construire un motif X. Pour cela, la ligne 6 tire un item correspondant à la base de données projetée choisie aléatoirement en faisant un tirage proportionnel au poids ω. La ligne 7 tire alors un itemset Y dans la base de données projetée $\mathcal{D}_\gamma^{(i)}$ en utilisant la procédure $\mathcal{S}_{\mathsf{freq}}$. Enfin, le motif X est composé de i suivi de l'itemset Y comme suffixe (ligne 8).

Analyse théorique L'algorithme 1 propose une procédure de tirage aléatoire exact :

Théorème 1 (Justesse de l'algorithme 1) *Soient une base de données $\mathcal{D}$, un seuil minimal de fréquence γ et une procédure $\mathcal{S}_{\mathsf{freq}}$ d'échantillonnage de motifs suivant la fréquence, l'algorithme 1 retourne un itemset X tiré aléatoirement suivant la fréquence tel que $\mathsf{freq}(X, \mathcal{D}) \geq \gamma$.*

Ce résultat découle de la propriété 2 au niveau de la ligne 7 et les motifs non-fréquents éliminés diminuent le rejet sans remettre en cause la justesse conformément à la propriété 1. Comme indiqué dans la section 3.1, la relation d'ordre total sur les items $<_{\mathcal{I}}$ est arbitraire et elle n'a pas d'impact sur la justesse de l'algorithme. En revanche, il est judicieux de choisir une relation $<_{\mathcal{I}}$ énumérant les items du plus rare au plus fréquent afin de minimiser le nombre d'occurrences dans chacune des bases de données projetées. Cette même heuristique est utilisée dans les méthodes d'énumération de motifs en profondeur (Han et al., 2000). Quel que soit $<_{\mathcal{I}}$, il est clair que le taux de rejet de l'algorithme 1 est inférieur à celui de $\mathcal{R}_\gamma(\mathcal{S}_{\mathsf{freq}}, \mathcal{L}, \mathcal{D}_\gamma)$:

Propriété 3 *Le taux de rejet de la procédure avec projection (algorithme 1) est inférieur à celui de la procédure avec suppression des items non-fréquents ($\mathcal{R}_\gamma(\mathcal{S}_{\mathsf{freq}}, \mathcal{L}, \mathcal{D}_\gamma)$) :*

$$1 - \frac{\sum_{X \in \mathcal{L}} \mathsf{freq}_\gamma(X, \mathcal{D})}{|\mathcal{D}| + \sum_{i \in \mathcal{I}} \sum_{t \in \mathcal{D}_\gamma^{(i)}} 2^{|t|}} \leq 1 - \frac{\sum_{X \in \mathcal{L}} \mathsf{freq}_\gamma(X, \mathcal{D})}{\sum_{t \in \mathcal{D}_\gamma} 2^{|t|}}$$

En plus d'avoir un bon taux de rejet, l'algorithme 1 satisfait à nouveau nos deux exigences initiales. Même si l'application d'une projection et d'une réduction pour chacun des items fréquents de $\mathcal{I}$ alourdit le prétraitement par rapport à la procédure $\mathcal{R}_\gamma(\mathcal{S}_{\mathsf{freq}}, \mathcal{L}, \mathcal{D}_\gamma)$, le coût reste raisonnable même pour les bases de données les plus larges (respect de la contrainte C1). L'utilisation exclusive d'opérations sur la base de données $\mathcal{D}$ permet à nouveau l'utilisation de $\mathcal{S}_{\mathsf{freq}}$ (respect de la contrainte C2).

5 Expérimentations

Cette étude expérimentale évalue les performances de notre approche d'échantillonnage de motifs sous contrainte de fréquence en analysant l'évolution du taux de rejet et du volume de données selon le seuil minimal de fréquence.

Nous utilisons 16 bases de données de référence provenant du UCI Machine Learning repository et du FIMI repository dont la taille et la densité sont variées (voir leurs caractéristiques dans la table 3). Nous comparons 3 méthodes s'appuyant toutes sur la procédure en deux étapes de Boley et al. (2011) pour $\mathcal{S}_{\mathsf{freq}}$: la procédure naïve $\mathcal{R}_\gamma(\mathcal{S}_{\mathsf{freq}}, \mathcal{L}, \mathcal{D})$ (Naive), la procédure avec suppression des items non-fréquents $\mathcal{R}_\gamma(\mathcal{S}_{\mathsf{freq}}, \mathcal{L}, \mathcal{D}_\gamma)$ (Del) et la procédure avec projection de l'algorithme 1 (Proj). Les méthodes sont implémentées avec le langage Java. Le code source est disponible en ligne [1]. Toutes les expériences sont réalisées sur un processeur Xeon de 2,5 GHz avec le système d'exploitation Linux et 2 Go de mémoire RAM. Même si nous ne reportons pas les temps d'exécution du prétraitement (tous inférieurs à la minute), ils sont proportionnels au volume de données étudié ci-après. De même, le temps d'exécution du tirage d'un motif dépend du temps de tirage de la procédure $\mathcal{S}_{\mathsf{freq}}$ utilisée et du nombre de rejets $1/(1 - \rho_{\gamma,\mathcal{D}})$ où $\rho_{\gamma,\mathcal{D}}$ est le taux de rejet. Pour calculer $\rho_{\gamma,\mathcal{D}}$, nous utilisons la propriété 1 avec

1. https://github.com/asoulet/egc23freqsamp

Base de données $\mathcal{D}$	$\lvert\mathcal{D}\rvert$	$\lvert\mathcal{I}\rvert$	AUC Naive	AUC Del	AUC Proj	Taille Proj
abalone	4 177	28	0.886	0.216	0.081	4.
austral	690	55	0.97	0.561	0.24	7.
chess	3 196	75	0.974	0.931	0.835	18.
cmc	1 473	28	0.941	0.452	0.13	4.5
connect	67 557	129	0.975*	0.945*	0.872*	21.
crx	690	59	0.971	0.589	0.254	7.46
hypo	3 163	47	0.942	0.526	0.299	8.7
iris	150	15	0.778	0.133	0.008	2.
mushroom	8 124	119	0.973	0.603	0.271	11.
page	941	35	0.962	0.376	0.122	5.
pumsb	49 046	7 117	0.975*	0.948*	0.911*	36.5
retail	88 162	16 470	0.975	0.052	0.001	7.89
sick	2 800	58	0.957	0.589	0.353	10.72
T40I10D100K	97 182	999	0.975	0.16	0.	20.21
vehicle	846	58	0.973	0.352	0.17	9.
waveform	5 000	67	0.975	0.3	0.184	10.5
		Moyenne :	0.950	0.483	0.296	

TAB. 3 – AUC du taux de rejet et taille pour 16 bases de données (*dénote des valeurs qui sont des approximations surévaluées)

$\gamma = 0$ sur $\mathcal{D}$ pour Naive et avec $\gamma = 0$ sur $\mathcal{D}_\gamma$ pour Del, et nous nous appuyons sur la partie gauche de la propriété 3 pour Proj.

La table 3 (colonnes 4 à 6) donne l'aire sous la courbe (AUC) du taux de rejet pour les 16 bases de données. L'aire sous la courbe du taux de rejet correspond à l'aire sous une courbe du taux de rejet tracé en fonction du seuil minimal de fréquence [2]. Par exemple, la figure 1 détaille l'évolution du taux de rejet en fonction du seuil minimal de fréquence pour 4 bases de données : abalone, chess, mushroom et sick. Plus l'AUC est faible, plus la méthode est efficace car le rejet est minimisé. Premièrement, nous observons d'abord que la méthode naïve n'est pas exploitable en pratique avec un taux de rejet toujours très élevé et une moyenne de 0.950. Plus précisément, l'AUC du taux de rejet est toujours supérieure à 0.778. Deuxièmement, la procédure Del diminue énormément le taux de rejet lorsque le seuil γ est élevé ce qui explique des AUC raisonnables avec une moyenne de 0.483. Mais, en observant les variations du taux de rejet, on constate que la méthode est inefficace pour les seuils minimaux de fréquence peu élevés. Dans cette configuration, il n'y a plus d'items qui sont supprimés. Clairement, la procédure Proj parvient davantage à conserver un taux de rejet faible pour ces seuils donnant une AUC moyenne du taux de rejet de 0.296. Ce phénomène s'observe bien sur les différents graphiques de la figure 1. En particulier, pour mushroom, le taux de rejet de Del remonte rapidement dès que $\gamma = 0.6$ tandis que la méthode Proj résiste mieux. Il y a aussi des gains plus marginaux pour les seuils de fréquence élevés comme pour sick autour de 0.9.

Pour la procédure Proj, la table 3 présente le volume maximal pour les différentes bases

2. Ce calcul nécessite de disposer du nombre d'occurrences complet pour un seuil minimal γ. Comme il n'a pas été possible de calculer ce nombre pour les seuils les plus faibles pour connect et pumsb, le taux de rejet a été surévalué en utilisant le nombre d'occurrences le plus élevé extrait.

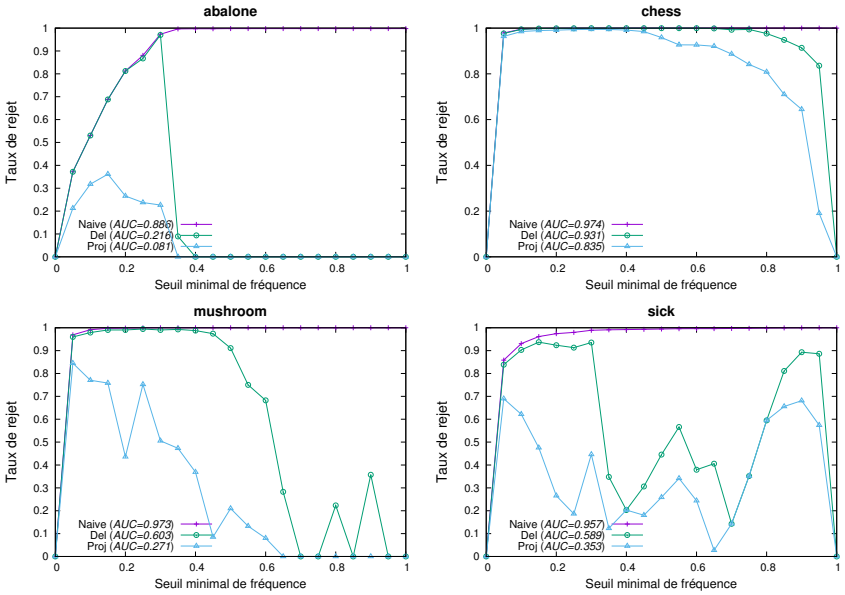

FIG. 1 – Evolution du taux de rejet en fonction du seuil minimal de fréquence γ

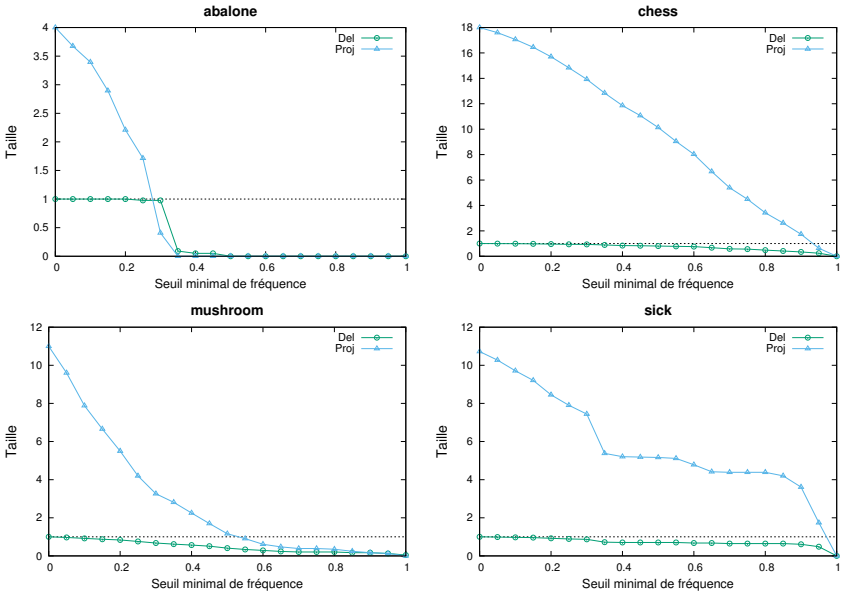

FIG. 2 – Evolution du volume de données en fonction du seuil minimal de fréquence γ

de données (i.e., pour $\gamma = 0$ même si l'intérêt d'une contrainte est plutôt de viser un seuil supérieur à 0). Plutôt que de mesurer la mémoire vive dépendant de l'implémentation, ce volume est donné en nombre de fois le volume de la base de données initiale. Il est à noter que ce volume maximal pour Naive et Del est non-reporté car toujours égal à 1. Au pire, ce volume est 37 fois supérieur au volume initial montrant que le stockage des différentes bases projetées est faisable même sur des grands jeux de données. Par ailleurs, dès que le seuil minimal de fréquence augmente, ce volume décroit rapidement comme nous pouvons l'observer sur la figure 2 détaillant l'évolution en fonction de γ pour 4 bases de données. Occasionnellement, le volume de données pour Proj est moins important que pour la procédure Naive (e.g., sur `abalone` pour $\gamma \geq 0.30$ ou sur `mushroom` pour $\gamma \geq 0.55$). Sur ces graphiques, on peut aussi noter que la procédure Del est très avantageuse par rapport à la procédure Naive. C'est par exemple le cas sur `abalone` pour $\gamma \geq 0.35$. Au final, le volume de données n'est pas une limite forte empêchant l'utilisation de la procédure Proj.

6 Conclusion

Cet article propose la première méthode d'échantillonnage de motifs avec une contrainte de fréquence. Pour cela, nous avons repris des mécanismes bien connus de l'extraction de motifs fréquents en prenant garde à ne pas matérialiser une collection de motifs et en conservant une procédure d'échantillonnage traditionnelle. Les expérimentations soulignent l'intérêt de l'approche pour réduire significativement le taux de rejet par rapport à une approche naïve ou limitée à la seule suppression des items non-fréquents. Ces résultats montrent qu'il est désormais possible d'appliquer l'échantillonnage de motifs avec une contrainte de fréquence minimale. Nous pensons que la levée de cette limite (souvent et légitimement entendue comme critique) augmentera l'intérêt pratique de l'échantillonnage de motifs.

En perspective, il serait intéressant d'étudier l'impact de l'ajout d'une étape supplémentaire de projection des bases projetées lorsque le taux de rejet reste trop élevé. Cela permettrait d'éliminer des itemsets non-fréquents de taille 3 manqués par notre méthode. De plus, notre méthode dédiée au langage des itemsets et à la fréquence peuvent s'étendre plus ou moins facilement à d'autres configurations. La généralisation à d'autres langages comme les séquences ou les graphes semble plutôt naturelle. En effet, des approches d'extraction de motifs fréquents ont déjà bénéficié de la réduction et de la projection de bases de données. En revanche, il sera plus compliqué de généraliser cette approche à toute mesure d'intérêt en remplacement de la fréquence (que cela soit au niveau de la probabilité de tirage ou de la contrainte). En première approche, nous pensons qu'il est possible d'étendre l'approche pour l'échantillonnage de motifs suivant les mesures fondées sur la longueur (Diop et al., 2022).

Références

Agrawal, R., T. Imieliński, et A. Swami (1993). Mining association rules between sets of items in large databases. In *Proceedings of the 1993 ACM SIGMOD*, pp. 207–216.

Agrawal, R., R. Srikant, et al. (1994). Fast algorithms for mining association rules. In *Proc. 20th int. conf. very large data bases, VLDB*, Volume 1215, pp. 487–499. Santiago, Chile.

Al Hasan, M. et M. J. Zaki (2009). Output space sampling for graph patterns. *Proceedings of the VLDB Endowment 2*(1), 730–741.

Boley, M., C. Lucchese, D. Paurat, et T. Gärtner (2011). Direct local pattern sampling by efficient two-step random procedures. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 582–590.

Bonchi, F., F. Giannotti, A. Mazzanti, et D. Pedreschi (2003). Exante : Anticipated data reduction in constrained pattern mining. In *European Conference on Principles of Data Mining and Knowledge Discovery*, pp. 59–70. Springer.

Bosc, G., J.-F. Boulicaut, C. Raïssi, et M. Kaytoue (2018). Anytime discovery of a diverse set of patterns with monte carlo tree search. *Data mining and knowledge discovery 32*(3), 604–650.

Diop, L., C. T. Diop, A. Giacometti, D. Li, et A. Soulet (2018). Sequential pattern sampling with norm constraints. In *2018 IEEE International Conference on Data Mining (ICDM)*, pp. 89–98. IEEE.

Diop, L., C. T. Diop, A. Giacometti, et A. Soulet (2022). Pattern on demand in transactional distributed databases. *Information Systems 104*, 101908.

Dzyuba, V., M. van Leeuwen, et L. De Raedt (2017). Flexible constrained sampling with guarantees for pattern mining. *Data Mining and Knowledge Discovery 31*(5), 1266–1293.

Giacometti, A. et A. Soulet (2016). Anytime algorithm for frequent pattern outlier detection. *International Journal of Data Science and Analytics 2*(3), 119–130.

Giacometti, A. et A. Soulet (2017). Interactive pattern sampling for characterizing unlabeled data. In *International Symposium on Intelligent Data Analysis*, pp. 99–111. Springer.

Han, J., J. Pei, et Y. Yin (2000). Mining frequent patterns without candidate generation. *ACM sigmod record 29*(2), 1–12.

Leeuwen, M. v. (2014). Interactive data exploration using pattern mining. In *Interactive knowledge discovery and data mining in biomedical informatics*, pp. 169–182. Springer.

Mannila, H. et H. Toivonen (1997). Levelwise search and borders of theories in knowledge discovery. *Data mining and knowledge discovery 1*(3), 241–258.

Ng, R. T., L. V. Lakshmanan, J. Han, et A. Pang (1998). Exploratory mining and pruning optimizations of constrained associations rules. *ACM Sigmod Record 27*(2), 13–24.

Pei, J., J. Han, B. Mortazavi-Asl, J. Wang, H. Pinto, Q. Chen, U. Dayal, et M.-C. Hsu (2004). Mining sequential patterns by pattern-growth : The prefixspan approach. *IEEE Transactions on knowledge and data engineering 16*(11), 1424–1440.

Summary

Pattern sampling is a recent technique for discovering patterns that promotes interactivity with the user. Its principle is to randomly draw a pattern in proportion to its interestingness. Unfortunately, the draws can focus on a part of the search space with non-frequent but extremely numerous patterns. It would be possible to sample patterns and eliminate those that are not frequent, but the rejection rate is often too high. In this paper, we propose the first pattern sampling method with a minimum frequency constraint. It is based on (i) the deletion of the non-frequent items and (ii) the projection of the database on each item. We propose a generic method that removes all occurrences containing a non-frequent pair of items. Our experiments show that our method significantly reduces the rejection rate.