Extraction et sélection de motifs émergents minimaux : application à la chémoinformatique

Mouhamadou Bamba Kane

Normandie Université

THESE

Pour obtenir le diplôme de doctorat

Spécialité Informatique

Préparée au sein de l'Université de Caen Normandie

Extraction et sélection de motifs émergents minimaux : application à la chémoinformatique

Présentée et soutenue par

Mouhamadou Bamba Kane

Thèse soutenue publiquement le 6 Septembre 2017 devant le jury composé de		
Mme Céline Rouveirol	Professeur Université Paris Nord	Rapporteur
M. Yannick Toussaint	Professeur Université de Lorraine - LORIA	Rapporteur
M. Sébastien Adam	Professeur Université de Rouen	Examinateur
Mme Christine Largeron	Professeur Université Jean Monnet(Saint Etienne)	Examinateur
M. Bruno Crémilleux	Professeur Université de Caen Normandie	Directeur de thèse
M. Bertrand Cuissart	MC Université de Caen Normandie	Examinateur

Thèse dirigée par Bruno Crémilleux (Laboratoire GREYC) et Alban Lepailleur (Laboratoire CERMN)

UNIVERSITÉ CAEN NORMANDIE

GREYC

Résumé

Cette thèse porte sur la fouille de données dont l'objectif est de découvrir des informations pertinentes dans des jeux de données. L'extraction de motifs intéressants est une étape importante de cette découverte de connaissances, un motif exprimant une relation entre les données. Les motifs extraits sont généralement analysés par l'expert qui espère en tirer une connaissance nouvelle. Dans un cadre où les jeux de données sont partitionnés en plusieurs classes, les motifs fortement corrélés à une classe présentent un intérêt particulier dû notamment à leur fort pouvoir discriminant. Dans un contexte de classification, ces motifs permettent généralement une bonne précision, en terme de prédiction de la classe des objets dans un contexte de classification supervisée et donnent lieu à la découverte de connaissances pertinentes dans les données.

Les Jumping Emerging Patterns (JEPs) minimaux sont un cas très intéressant de ces types de motifs. Un JEP est un motif qui n'apparaît que dans une seule classe. Un JEP minimal correspond quant à lui à un motif dont aucun sous motif n'est un JEP. Les JEPs sont largement utilisés pour la prédiction ou la découverte de connaissance dans beaucoup de domaines d'applications comme en biologie, en chimie ou en médecine. Calculer tous les *JEPs minimaux* permet de réduire considérablement le nombre de motifs par rapport à l'ensemble des JEPs et aussi d'obtenir des classifieurs avec une bonne précision. Cependant, extraire tous les JEPs minimaux peut être une tâche très coûteuse en temps d'exécution. Les méthodes existantes extraient généralement les JEPs qui apparaissent le plus dans un jeu de données, au risque de passer à côté de motifs très intéressants mais peu supportés par les données. Nous proposons une méthode qui permet d'extraire efficacement le totalité des JEPs minimaux. De plus, notre méthode prend en compte l'absence d'attribut qui apporte une nouvelle connaissance intéressante.

Dans la pratique, on constate que l'ensemble des JEPs minimaux ne couvrent pas l'ensemble des données, ce qui constitue un frein pour leur usage en classification. Nous avons montré expérimentalement que plus les JEPs sont supportés dans le jeu de données, plus la couverture de l'ensemble de ces motifs est faible et leur confiance plus grande. La question intéressante est ainsi de savoir comment arriver à améliorer la couverture de l'ensemble JEPs (beaucoup supportés) sans pour autant dégrader leur confiance. Nous avons

résolu ce problème en proposant une méthode de sélection à base de prototypes. Au vu des résultats encourageants obtenus, nous avons appliqué cette méthode de sélection sur un jeu de données chimique : Aquatox. Cela permet ainsi aux chimistes, dans un contexte de classification, de mieux expliquer la classification des molécules, qui sans cette méthode de sélection serait prédites par l'usage d'une règle par défaut.

Table des matières

Liste des tableaux

Table des figures

Introduction générale

1 Contexte

L'extraction de connaissance dans les bases de données ou la fouille de données a pour objectif de découvrir dans les données des connaissances nouvelles, pertinentes et interprétables par un expert du domaine dont sont issues les données [FPSSU96]. Une partie importante de la fouille de données est le domaine de l'extraction de motifs, un motif correspondant à une régularité sur les données. L'extraction de motifs pose un problème algorithmique difficile lié à l'espace de recherche qui est souvent très important [MT96a]. Plusieurs critères peuvent être définis afin d'extraire les motifs les plus intéressants en vue d'une interprétation par l'expert et optimiser au mieux le parcours dans l'espace de recherche. Parmi ces critères, on peut citer la fréquence [AIS93] qui mesure le nombre d'occurrences d'un motif dans les données. Nos travaux se situent dans le domaine de l'extraction de motifs dans un contexte de classification supervisée (les jeux de données sont partitionnés en plusieurs classes). En plus du critère de fréquence, nous nous intéressons à faire ressortir les contrastes entre les descriptions des objets des différentes classes [NLW09]. Dans ce sens, Dong et Li ont défini les *motifs émergents (Emerging Patterns (EP))* [DL99] qui sont les motifs qui apparaissent plus fréquemment dans une classe que dans une autre. Dong et Li ont montré que les motifs émergents possèdent de bonnes caractéristiques dans un contexte de classification [DZWL99]. Ils ont aussi proposé les *JEPs (Jumping Emerging Patterns)* qui sont des motifs émergents qui sont présents dans une classe et jamais dans les autres. Les JEPs minimaux correspondent à des JEPs dont aucun sous ensemble n'est un JEP. Les JEPs minimaux possèdent plusieurs avantages : ils permettent d'éviter la redondance des motifs, réduisent le nombre de motifs extraits et ont une bonne précision dans un contexte de classification. Ils sont au centre du travail mené dans ce mémoire.

Cependant, extraire tous les JEPs minimaux est une tâche ardue et coûteuse en temps d'exécution. Boulicaut et al. [BBR03] ont introduit la notion de motifs *libres*. Ces motifs forment un sur-ensemble des JEPs minimaux, nous verrons qu'il est possible d'extraire les JEPs minimaux à partir de cet algorithme. Ugarte et al. [UBLC15] proposent une

méthode d'extraction de plusieurs types de motifs, dont les JEPs minimaux, fondée sur la programmation par contraintes. Cependant, pour les jeux de données volumineux, ces deux méthodes d'extraction posent un problème de temps de calcul. L'espace de recherche à explorer étant très grand, Fan et Ramamohanarao [FR02, FR06] ont ainsi proposé une méthode d'extraction des JEPs minimaux en ajoutant un seuil minimum de fréquence ; ces motifs sont appelés les *essential JEPs ou strong JEPs*. Dans la même lancée, Terlecki et Walczak [TW08] définissent les *top-k JEPs minimaux* qui correspondent aux k JEPs minimaux les plus supportés, k étant un seuil fixé par l'utilisateur.

Lorsque les objets sont décrits à l'aide d'attributs binaires (présence ou absence), les méthodes d'extraction existantes des JEPs minimaux ne prennent pas en compte l'absence d'attribut. Terlecki et Walczak [TW08] ont démontré, qu'en plus d'apporter une connaissance nouvelle, l'intégration de l'absence d'attribut, ou négation d'attribut, dans l'extraction des motifs émergents permet d'améliorer la précision des classifieurs par rapport au cas où la négation d'attribut n'est pas utilisée. Par exemple, pour un jeu de données chimique, la prise en compte de l'absence d'attributs permettrait de constater que la présence de certains fragments moléculaires, combinée à l'absence d'autres concluent sur une activité chimique d'une molécule.

Nous verrons que notre travail s'inscrit dans le contexte de la chémoinformatique. La chémoinformatique est un domaine où les motifs émergents ont été largement employés pour la découverte de nouvelles connaissances sur les jeux de données chimiques mais aussi aussi pour la prédiction de la toxicité ou de la mutagénicité des molécules [SGJV12, MLB+15, SJH+14, MLB+15]. Pour cette partie de notre travail, nous avons collaboré avec le Centre d'Études et de Recherche sur le Médicament de Normandie (CERMN, UPRES EA 4258, Université de Caen Normandie).

2 Contributions

Dans ce mémoire, nous apportons plusieurs contributions sur le plan de l'extraction des motifs et de l'évaluation de la qualité des motifs extraits. Nous synthétisons maintenant les contributions de notre travail.

Nouvelle méthode d'extraction de tous les JEPs minimaux. Extraire tous les JEPs minimaux est une tâche difficile et coûteuse vu la combinatoire importante et que, d'autre part, leur extraction ne s'appuie pas sur les propriétés classiques d'anti-monotonicité en fouille de données [MT96b]. Les principales méthodes d'extraction calculent généralement les JEPs minimaux avec un seuil de support (cf. définition 1.1.8) fixé par l'utilisateur. Nous proposons un algorithme, appelé *AMinJEP*, qui calcule d'une ma-

nière efficace tous les JEPs minimaux avec ou sans contrainte de support. Une originalité de *AMinJEP* est de définir les conditions caractérisant le fait que chaque attribut d'un JEP minimal est indispensable en exploitant les différences entre les objets du jeu de données. La différence entre deux objets est définie comme étant les attributs présents dans un objet et pas dans l'autre. Pour le calcul des JEPs minimaux, nous introduisons une structure d'arbre dont les nœuds sont construits à partir des différences entre les objets. Pour chaque ajout d'un nœud dans l'arbre, nous définissons des conditions d'arrêt qui permettent d'anticiper si le chemin passant par ce nœud conduira à un JEP minimal. La force de cette structure d'arbre est d'élaguer une bonne partie de l'espace de recherche en n'insérant pas dans l'arbre les nœuds qui ne mèneront pas à un JEP minimal. Grâce à cette structure d'arbre, nous pouvons aussi calculer efficacement les essential JEPs ou les top-k JEPs minimaux. Pour cela, le support est calculé à chaque ajout d'un nœud qui ne vérifie pas les conditions d'arrêt. La mise à jour du support n'étant pas coûteuse en temps d'exécution, les JEPs minimaux sont extraits efficacement avec ou sans contrainte de support.

Nous avons aussi intégré dans *AMinJEP* la possibilité d'extraire les JEPs minimaux avec la négation d'attribut. Pour cela, nous ajoutons pour chaque descripteur sur les différents jeux de données, un descripteur représentant son absence. L'espace de recherche devient ainsi plus important et l'extraction plus coûteuse en temps d'exécution. Mais grâce à l'efficacité de la structure d'arbre proposée, nous montrons qu'il est possible de calculer les JEPs minimaux sur des jeux de données volumineux pour lesquels les méthodes utilisant la programmation par contrainte sont inopérantes. Cette contribution est développée au chapitre 2.

Méthode de sélection de règles à base de prototypes : Un JEP minimal peut être considéré comme une règle, donc une conjonction d'attributs qui conclut sur une classe. Nous utilisons deux mesures pour évaluer la qualité de ces règles ; la *couverture* et la *confiance*. La *couverture* d'une règle correspond à la proportion d'objets qui supportent cette règle. La *confiance* dénote la probabilité que la conclusion de la règle soit vérifiée lorsque cette règle est supportée. La *couverture* (respectivement la *confiance*) pour un ensemble de règles correspond à la moyenne des couvertures (respectivement des confiances) des règles de l'ensemble. La couverture, respectivement la confiance, des JEPs minimaux correspond à la couverture, respectivement la confiance, de l'ensemble des règles associées aux JEPs minimaux. Nous montrons d'une manière expérimentale que plus le support des JEPs minimaux est élevé, plus la couverture de l'ensemble des règles associées à ces JEPs minimaux est faible et plus la confiance de l'ensemble des règles est grande. Ainsi l'usage des JEPs minimaux (où chaque JEP vérifie un seuil élevé de support) constitue

un frein pour la classification car, cet ensemble de JEPs ne couvre pas une partie conséquente des objets. Une question alors intéressante est comment améliorer la couverture de l'ensemble des JEPs tout en obtenant quasiment la même confiance ? Pour cela, nous proposons une méthode de sélection de règles qui considère un ensemble des règles, ici des JEPs, avec un support élevé comme étant les règles *prototypes*. Ces règles possèdent une bonne confiance mais leur couverture est faible. Toute autre règle est considérée comme une règle *candidate*. Grâce à une méthode des plus proches voisins, une règle candidate est *sélectionnée* si elle est assez similaire aux règles prototypes. L'intuition est qu'une règle candidate, même peu supportée, qui possède presque la même description que les règles prototypes, permet de couvrir des objets non couverts par les règles prototypes avec quasiment la même confiance que les règles prototypes. Les tests expérimentaux ont montré que cette méthode de sélection améliore la couverture des règles prototypes avec une très faible perte en confiance.

Nous avons aussi évalué la fiabilité de la méthode de sélection dans un contexte de classification. A cette fin, nous avons utilisé le classifieur CPAR [YH03]. Les résultats expérimentaux ont montré que la sélection des règles, en plus de couvrir plus d'objets, permet d'améliorer la précision du classifieur comparée à celle des règles prototypes.

Nous proposons aussi une autre amélioration de la méthode de sélection appelée *méthode de sélection à plusieurs passes* . Les règles sélectionnées deviennent aussi des règles prototypes lors de la sélection d'un nouvel ensemble de règles candidates. Cette méthode est développée au chapitre 3.

Application de la méthode de sélection sur le jeu de données Aquatox Nous avons appliqué la méthode de sélection sur un jeu de de données chimique sur l'environnement aquatique. Ce jeu de données est partitionné en 362 substances très toxiques et 187 faiblement toxiques. En chémoinformatique, un intérêt de la sélection de règles à base de prototypes est de faciliter l'intelligibilité des résultats d'un classifieur fondé sur les règles. En effet, un classifieur utilise souvent une règle par défaut et l'utilisation de la méthode de sélection comme définie au chapitre 3 diminue l'emploi de la règle par défaut car un plus grand nombre d'objets sont couverts avec la méthode de sélection que si seules les règles prototypes sont utilisées. On peut ainsi mieux repérer quels sont les fragments moléculaires qui sont responsables de la toxicité d'une molécule. Cette contribution est détaillée au chapitre 4.

3 Organisation du mémoire

Le *chapitre 1* de ce mémoire présente les principales notions liées à l'extraction de motifs en fouille de données. Il introduit l'extraction des motifs selon deux critères que sont la fréquence et le taux de croissance. Nous faisons un focus sur les motifs émergents, en particulier les JEPs minimaux qui constituent une partie importante de notre travail. Nous montrons dans ce chapitre la similarité des motifs émergents avec d'autres types de motifs proposés dans la littérature. Les méthodes existantes d'extraction des JEPs minimaux sont aussi indiquées. Nous terminons par une présentation des mesures utilisées pour évaluer la qualité d'un classifieur.

Le *chapitre 2* détaille notre méthode d'extraction des JEPs. Il contient : i) les notations utilisées pour expliquer la méthode, ii) la notion de complémentaire d'objet qui permet d'expliquer la différence entre objets, iii) les conditions que nous avons définies pour vérifier qu'un motif est un JEP et que chaque attribut est nécessaire pour qu'il soit minimal, iv) la structure d'arbre que nous proposons dont l'ensemble des chemins de la racine vers les feuilles "valides" est en bijection avec l'ensemble de tous les JEPs minimaux, v)le lien entre les JEPs minimaux et les traverses minimales d'hypergraphes, vi) la faisabilité de la méthode en comparaison avec les méthodes existantes d'extraction des JEPs minimaux, vii) l'évaluation de la qualité des JEPs minimaux s'ils sont considérés comme des règles concluant sur une classe.

Le *chapitre 3* expose notre méthode de sélection de règles à base de prototypes. Les points développés dans ce chapitre sont : i) le principe de la méthode de sélection, ii) les paramètres utilisés pour l'implémentation de la méthode de sélection tels que la mesure de similarité, les ensembles de règles utilisés, les mesures utilisées pour l'évaluation de la méthode sélection iii) le calcul de ces différents paramètres, iv) l'évaluation expérimentale de la méthode de sélection avec la comparaison entre les différents ensembles de règles étudiés, v) évaluer la qualité des différents ensembles de règles au moyen d'un classifieur, v) l'apport de la méthode de sélection à plusieurs passes en terme de couverture et de confiance.

Le *chapitre 4* analyse l'application de la méthode de sélection sur le jeu de données Aquatox qui est un jeu de données fourni par le *Centre d'Études et de Recherche sur le Médicament de Normandie (CERMN UPRES EA 4258, Université de Caen Normandie).* Ce chapitre détaille les motivations de cette application, ainsi que les paramètres de la méthode de sélection qui sont spécifiques à ce cas d'étude. Une première partie expérimentale fait une étude comparative de la couverture et de la confiance des différents ensembles de règles étudiés. La deuxième partie expérimentale évalue la qualité des différents ensembles de règles dans un contexte de classification. La dernière partie de ce chapitre fait

une évaluation quantitative de la méthode de sélection sur le jeu de données Aquatox. Nous montrons comment cette méthode permet de trouver des toxicophores bien connues dans la littérature.

Le dernier chapitre conclut sur les travaux réalisés et met en lumière quelques perspectives.

Chapitre 1

État de l'art

Sommaire

La fouille de données a pour objectif de découvrir des informations utiles et pertinentes dans les jeux de données. Ce chapitre présente un état de l'art de l'extraction de motifs en fouille de données. Nous introduisons la notion de contraintes qui permet de cibler des motifs intéressants ainsi que celle de représentation condensée de motifs ayant pour but de synthétiser les motifs extraits. Nous enchaînons par la définition des motifs émergents qui sont les motifs sur lesquels nous nous focalisons dans cette thèse et nous montrons que ces motifs partagent le même principe de mise en relief de contraste que d'autres types de motifs définis dans d'autres domaines. Nous terminons ce chapitre par une explication des différentes mesures utilisées pour évaluer un classifieur et les méthodes utilisées pour calculer ces mesures.

1.1 Extraction de motifs en fouille de données

Cette section introduit les termes et notions classiques utilisés dans le cadre de l'extraction de motifs, ainsi que quelques exemples de contraintes utilisés.

1.1.1 Terminologie

Nous introduisons dans cette section les notions de jeu de données, motif, langage et théorie. Les données que nous allons traiter dans ce mémoire sont des données transactionnelles. Une transaction décrit un ensemble d'attributs appelé aussi *items*. La définition 1.1.1 précise la notion de jeu de données.

Définition 1.1.1 - *Définition d'un jeu de données* : Un jeu de données $\mathcal{D}$ correspond à un triplet ($\mathcal{G}$, $\mathcal{M}$, $\mathcal{I}$) avec :

— $\mathcal{G}$: l'ensemble des *objets* (transactions) du jeu de données

— $\mathcal{M}$: les attributs (items) qui servent à décrire les objets

— $\mathcal{I}$ sur $\mathcal{G} \times \mathcal{P}(\mathcal{M})$ avec pour tout objet o et ensemble d'attributs p : $oIp \iff p \subseteq o$.

Nous choisissons, dans ce mémoire, d'utiliser les notations de l'analyse formelle de concepts (AFC) [GSW04].

Exemple 1.1.1 Considérons l'exemple du jeu de données $\mathcal{D}$ (cf tableau 1.1) qui va servir d'exemple tout au long de ce chapitre. Un objet de $\mathcal{G}$ est noté o. "x" correspond à la présence d'un attribut dans un objet. Le jeu de données est composé de 10 objets qui sont décrits par les attributs de $a_1 \dots a_6$. Les objets du jeu de données sont partitionnés en deux classes : $\mathcal{G}_+$ et $\mathcal{G}_-$. Ainsi les objets o_1, o_2, o_3, o_4, o_5 appartiennent à la classe $\mathcal{G}_+$, tandis que o_6, o_7, o_8, o_9, o_{10} appartiennent à la classe $\mathcal{G}_-$.

Tableau 1.1 – Jeu de données $\mathcal{D}$ de 10 objets partitionné en deux classes

$\mathcal{G}$ \ $\mathcal{M}$		a_1	a_2	a_3	a_4	a_5	a_6
$\mathcal{G}_+$	$\mathbf{o}_1$	x	x			x	
	$\mathbf{o}_2$	x		x	x	x	
	$\mathbf{o}_3$	x		x		x	
	$\mathbf{o}_4$	x	x			x	
	$\mathbf{o}_5$			x	x	x	
$\mathcal{G}_-$	$\mathbf{o}_6$		x	x	x		x
	$\mathbf{o}_7$	x	x			x	x
	$\mathbf{o}_8$				x		
	$\mathbf{o}_9$		x			x	
	$\mathbf{o}_{10}$	x			x	x	x

L'objectif de la fouille de données est de découvrir des relations utiles entre les objets du jeu de données. Ces relations évoquent un comportement ou décrivent un phénomène

dans les données. Elles sont caractérisées par les motifs.

Définition 1.1.2 - *Motif* : Un motif ou *itemset* (dans le cas de données transactionnelles) correspond à un sous ensemble de $\mathcal{M}$. On peut aussi dire qu'un motif est une conjonction d'attributs. Par exemple $\{a_1\}$, $\{a_3, a_6\}$, $\{a_2, a_3, a_5\}$ sont des motifs. Nous utilisons la représentation ensembliste pour les motifs à défaut de la représentation sous forme de chaînes où les motifs précédents seraient notés a_1, a_3a_6 ou $a_2a_3a_5$.

Définition 1.1.3 - *Langage* : Un *langage* $\mathcal{L}$ est un ensemble de motifs.

Le langage de motifs ensemblistes $\mathcal{L}_\mathcal{M}$ correspond à tous les sous-ensembles de $\mathcal{M}$: $\mathcal{L}_\mathcal{M} = 2^\mathcal{M}$.

Définition 1.1.4 - *Contrainte* : Une *contrainte* est un prédicat booléen défini sur un langage.

Une contrainte exprime l'intérêt d'un motif selon un critère fixé par l'utilisateur. Elle permet de réduire l'ensemble des motifs extraits tout en ciblant les motifs recherchés par l'utilisateur. Par exemple, une contrainte est appliquée sur une mesure afin d'obtenir les motifs respectant un critère de sélection par rapport à une mesure. Une des mesures les plus classiques est la *fréquence* [AS94] qui permet d'extraire les motifs fréquents. Le calcul de la fréquence s'obtient grâce à la notion d'*extension* (cf définition 1.1.7). Geng et al. [GH06] ont détaillé plusieurs mesures utilisées pour définir l'intérêt d'un motif.

Définition 1.1.5 - *Théorie* : Soit un langage $\mathcal{L}$ et un jeu de données $\mathcal{D} = (\mathcal{G}, \mathcal{M}, \mathcal{I})$, la *théorie* $Th(\mathcal{L},\mathcal{D},q)$ [MT97a] est l'ensemble des motifs dans $\mathcal{L}$ satisfaisant la contrainte q.

Ce cadre permet d'unifier les différents types d'extraction quel que soit le contexte (base de données, langage, contexte). De nombreuses mesures sont utiles pour extraire des motifs pertinents pour un expert. Nous portons dans ce travail un intérêt particulier pour les mesures de contraste tel que le *taux de croissance* qui donne lieu aux motifs émergents (définition 1.4.1) proposés par Dong et Li [DL99].

1.1.2 Mesures sur les motifs

Il existe de nombreuses mesures pour évaluer la qualité des motifs. Nous en donnons maintenant quelques unes. Les définitions suivantes se rapportent à un jeu de données $\mathcal{D}$

$= (\mathcal{G}, \mathcal{M}, \mathcal{I})$.

Définition 1.1.6 - *Taille* : La *taille* d'un motif correspond au nombre d'attributs que contient ce motif. En d'autres termes, il s'agit de sa cardinalité.

Par exemple la taille du motif $\{a_2, a_3, a_5\}$ est égale à 3.

Définition 1.1.7 - *Extension d'un motif* : L'*extension* d'un motif $p \subseteq \mathcal{M}$ dans un jeu de données $(\mathcal{G}, \mathcal{M}, \mathcal{I})$ correspond à l'ensemble des objets de $\mathcal{G}$ qui contiennent p : elle est notée $p'_{\mathcal{G}} = \{o \in \mathcal{G} : oIp\}$.

Par exemple, en considérant $p = \{a_4,a_5\}$, on obtient $p'_{\mathcal{G}} = \{o_2, o_5, o_{10}\}$.

Définition 1.1.8 - *Support d'un motif* : Le *support* d'un motif p dans un jeu de données $(\mathcal{G}, \mathcal{M}, \mathcal{I})$ est égal à la cardinalité de son extension : il est noté $supp_{\mathcal{G}}(p) = |p'_{\mathcal{G}}|$.

On a ainsi pour le motif $p = \{a_4,a_5\}$, $supp_{\mathcal{G}}(p) = |\{o_2,o_5,o_{10}\}| = 3$

Définition 1.1.9 - *Fréquence d'un motif* : La *fréquence* d'un motif p correspond au ratio du support de p sur la cardinalité de $\mathcal{G}$. Elle est notée $\mathcal{F}_{\mathcal{G}}(p) = \frac{supp_{\mathcal{G}}(p)}{|\mathcal{G}|}$

La contrainte des motifs fréquents est définie en sélectionnant les motifs dont la fréquence est supérieure à un seuil de fréquence donné (cf définition 1.1.10).

Définition 1.1.10 - *Motif fréquent* : Soit un seuil de fréquence f_{min}. Un motif p est fréquent dans $\mathcal{D}$ si $\mathcal{F}_{\mathcal{G}}(p) > f_{min}$.

Sur l'exemple du tableau 1.1, l'extension du motif $p = \{a_1,a_2\}$ dans $\mathcal{D}$ est $p'_{\mathcal{G}} = \{t_1, t_2, t_3\}$, son support $supp_{\mathcal{G}}(p) = |p'_{\mathcal{G}}| = 3$, sa fréquence $\mathcal{F}_{\mathcal{G}}(p)$ équivaut à $\frac{supp_{\mathcal{G}}(p)}{|\mathcal{G}|} = \frac{3}{10} = 30\%$. Le motif a_5 est le plus fréquent avec une fréquence de $\frac{8}{10} = 80\%$. Si $f_{min} = 20\%$, alors le motif $p = \{a_2,a_3,a_4\}$ n'est pas fréquent car sa fréquence $\mathcal{F}_{\mathcal{G}}(p) = \frac{1}{10} = 10\%$.

Définition 1.1.11 - *Taux de croissance* : Soit un jeu de données $\mathcal{D} = (\mathcal{G}, \mathcal{M}, \mathcal{I})$ partitionné en deux classes : $\mathcal{G}_+$ et $\mathcal{G}_-$. Le taux de croissance d'un motif p de $\mathcal{G}_-$ vers $\mathcal{G}_+$ correspond au ratio de la fréquence de p dans $\mathcal{G}_+$ sur la fréquence de p dans $\mathcal{G}_-$: il est noté

$$\mathcal{GR}_{\mathcal{G}_+}(p) = \begin{cases} 0 & \text{si } \mathcal{F}_{\mathcal{G}_+}(p) = 0 \text{ et } \mathcal{F}_{\mathcal{G}_-}(p) = 0 \\ \infty & \text{si } \mathcal{F}_{\mathcal{G}_+}(p) \neq 0 \text{ et } \mathcal{F}_{\mathcal{G}_-}(p) = 0 \\ \frac{\mathcal{F}_{\mathcal{G}_+}(p)}{\mathcal{F}_{\mathcal{G}_-}(p)} & \text{sinon.} \end{cases}$$

1.1.3 Règles d'association

Nous considérons toujours un jeu de données $\mathcal{D} = (\mathcal{G}, \mathcal{M}, \mathcal{I})$.

Définition 1.1.12 - *Règle d'association* : Soit p un motif. Une *règle d'association* r fondée sur p est une expression $X \rightarrow Y$ avec $X \subsetneq p$ et $Y = p \backslash X$. X est la *prémisse* de r et Y sa *conclusion*.

Deux mesures sont souvent utilisées pour l'évaluation d'une règle d'association : la fréquence et la confiance

Définition 1.1.13 - *Fréquence d'une règle d'association* : Soit r une règle d'association $X \rightarrow Y$ dans $\mathcal{D}$. La *fréquence* de r dans $\mathcal{D}$ est le ratio du support de $XY = X \cup Y$ sur le nombre d'objets de $\mathcal{D}$: elle est notée $freq_D(r) = \mathcal{F}_{\mathcal{G}}(XY)$.

Définition 1.1.14 - *Confiance d'une règle d'association* : Soit r une règle d'association $X \rightarrow Y$ dans $\mathcal{D}$. La *confiance* de r dans $\mathcal{D}$ est le ratio de la fréquence de r dans $\mathcal{D}$ sur la fréquence de X dans $\mathcal{D}$: elle est notée $conf_D(r) = \frac{\mathcal{F}_{\mathcal{G}}(XY)}{\mathcal{F}_{\mathcal{G}}(X)}$.

Le tableau 1.2 illustre ces notions pour des règles d'association issues du jeu de données $\mathcal{D}$ (cf tableau 1.1) en indiquant les valeurs de fréquence et de confiance de ces règles.

1.2 Espace de recherche

L'extraction de motifs est une tâche difficile du point de vue algorithmique au vu de la grande taille de l'espace de recherche. Pour les motifs ensemblistes, l'espace de recherche correspond à un treillis.

Définition 1.2.1 - *Relation de spécialisation* : Une relation de spécialisation $\preceq$ est un ordre partiel défini sur les motifs de $\mathcal{L}$. Le motif p_1 est plus général (respectivement plus spécifique) que p_2 si $p_1 \preceq p_2$ (respectivement $p_2 \preceq p_1$).

Tableau 1.2 – Exemple de règles de classification dans $\mathcal{D}$ extraites avec $f_{min} = 10\%$.

Règles de clasification	**Fréquence**	**Confiance**
$a_3a_4 \rightarrow c_1$	0,2	0,5
$a_3a_4 \rightarrow c_2$	0,2	0,5
$a_1a_2 \rightarrow c_1$	0,2	0,67
$a_1a_2 \rightarrow c_2$	0,1	0,33

Pour les ensembles d'attributs, l'inclusion $\subseteq$ constitue une relation de spécialisation. Par exemple si $\{a_1\} \subseteq \{a_1,a_2\}$ alors $\{a_1\}$ est plus général que $\{a_1,a_2\}$ et $\{a_1,a_2\}$ est une spécialisation de $\{a_1\}$. On obtient ainsi une structure de l'espace de recherche en treillis comme représentée à la figure 1.1 (correspondant au jeu de données du tableau 1.1). La représentation des motifs sous forme de chaine est utilisée à la figure 1.1.

A la figure 1.1, le motif le plus en haut est l'ensemble vide, le motif composé de l'ensemble des attributs est celui le plus bas. Le passage d'un niveau à un autre est réalisé par l'ajout d'un attribut : le deuxième niveau contient les singletons, le troisième les paires, le quatrième les triplets, etc. Les premiers et derniers niveaux contiennent un seul motif et le nombre maximum de motifs par niveau est atteint au milieu du treillis.

Pour faciliter l'extraction de motifs, certaines propriétés comme l'(anti-)monotonicité [MT97b] sont d'une grande aide.

Définition 1.2.2 - *Contrainte monotone ou anti-monotone* : Une contrainte q est monotone (respectivement anti-monotone) suivant la relation de spécialisation $\preceq$ si et seulement si pour tout motif satisfaisant q, ses spécialisations (respectivement généralisations) satisfont également q.

Propriété 1.2.1 : Élagage fondé sur les contraintes anti-monotones Si un motif m ne satisfait pas la contrainte anti-monotone q, alors toutes les spécialisations de m ne satisfont pas la contrainte q.

Cette propriété implique en pratique que :

— si un motif m ne satisfait pas une contrainte, ses spécialisations ne la satisfont pas : on peut élaguer les branches issues de m lors du parcours de l'espace de recherche,

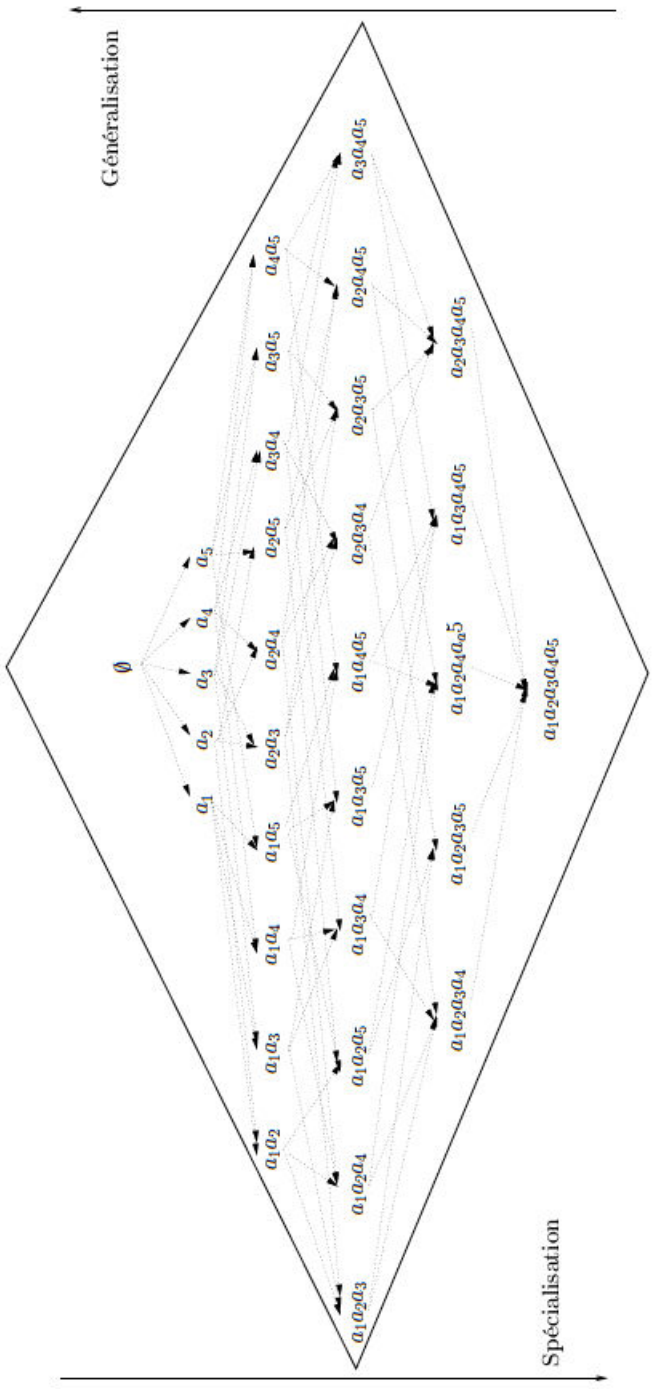

FIGURE 1.1 – Treillis de motifs du jeu de données $\mathcal{D}$

— si un sous-ensemble d'un motif m ne satisfait pas une contrainte, m ne peut pas la satisfaire non plus.

La fréquence est une contrainte anti-monotone qui est classiquement utilisée [AS94]. Considérons toujours l'exemple du jeu de données $\mathcal{D}$ du tableau 1.1. Avec un seuil de fréquence minimum f_{min} fixé à 30% dans $\mathcal{D}$, le motif $\{a_3,a_4,a_5\}$ est fréquent puisque sa fréquence est de 30%. L'anti-monotonicité de la fréquence assure que tous les motifs pouvant être généralisés depuis $\{a_3,a_4,a_5\}$ sont également fréquents. Considérons maintenant le motif $\{a_1,a_4\}$. Celui-ci n'est pas fréquent dans $\mathcal{D}$, alors toutes ses spécialisations (p.e. $\{a_1,a_2,a_4\}$, $\{a_1,a_3,a_4\}$) ne le sont pas. En revanche, comme le montre l'exemple suivant, le taux de croissance ne donne pas lieu à une contrainte monotone ou anti-monotone. Avec un taux de croissance minimum ρ fixé à 1,5, $\{a_5\}$ est émergent de $\mathcal{G}_-$ vers $\mathcal{G}_+$ ($\mathcal{GR}_{\mathcal{G}_+}(\{a_5\})$) = 5/3 = 1,67), $\{a_2,a_5\}$ ne l'est plus ($\mathcal{GR}_{\mathcal{G}_+}(\{a_2,a_5\}) = 1$), alors que $\{a_1,a_2,a_5\}$ est à nouveau émergent ($\mathcal{GR}_{\mathcal{G}_+}(\{a_1,a_2,a_5\}))) = 2$). L'explication provient du fait que le taux de croissance est un ratio de fréquences. Entre un motif et sa généralisation, le numérateur peut être égal ou plus grand, de même pour le dénominateur, et le ratio peut ainsi augmenter ou diminuer. La contrainte d'émergence est ainsi plus difficile à être poussée lors de l'extraction.

1.3 Représentation condensée de motifs

Selon le jeu de données et les contraintes utilisées, le nombre de motifs extraits peut facilement ne plus être interprétable par un humain. Les représentations condensées de motifs ont été introduites afin d'extraire plus efficacement les motifs et faciliter leur interprétation. Nous donnons ici la définition d'une représentation condensée exacte des motifs (i.e., il est possible de régénérer l'ensemble complet des motifs vérifiant la contrainte).

Définition 1.3.1 - *Représentation condensée exacte* : La représentation condensée [MT96b] d'un ensemble de motifs $\mathcal{P}$ selon une contrainte q est un ensemble de motifs $\mathcal{R}$ de cardinalité inférieure à celle de $\mathcal{P}$ tel que pour tout motif $p \subseteq \mathcal{P}$ la valeur de $q(p)$ puisse être déduite à partir de un ou plusieurs motifs de $\mathcal{R}$.

Pour les motifs ensemblistes fréquents, les deux représentations condensées les plus classiques sont les motifs *fermés* fréquents [PBTL99a] et les motifs *libres* fréquents [BBR03] que nous définissons maintenant.

Définition 1.3.2 - *Motif fermé* : Un motif p est un motif fermé si toutes ses spécialisations strictes ont une fréquence strictement inférieure à celle de p.

Définition 1.3.3 - *Motif libre* : Un motif p est un motif libre (appelé également générateur) si toutes ses généralisations strictes ont une fréquence strictement supérieure à celle de p.

Il existe plusieurs méthodes pour le calcul de motifs fermés et libres ainsi que leurs représentations condensées associées. Pour les motifs fermés, on peut citer A-CLOSE [PBTL99a] qui a été étendu en CLOSE [PBTL99b], CHARM [ZH02] ou encore LCM [UAUA04] qui est réputé comme l'un des algorithmes les plus efficaces pour cette tâche. Les motifs libres vérifient la propriété d'anti-monotonicité et peuvent être extraits en s'appuyant sur les propriétés d'élagage liées à l'anti-monotonicité [BB00]. Il existe plusieurs extensions de la notion de motifs libres tels que les motifs k-libres [CG07] et les motifs delta-libres [BBR03]. Les représentations condensées précédentes s'appuient sur la notion de fermeture de Galois et sont bien adaptées aux contraintes définies sur des mesures fondées sur la fréquence comme par exemple les mesures de contrastes [SCR04, LLW07].

1.4 Motifs émergents et équivalences

Nous définissons dans cette section la notion de motifs émergents. Nous expliquons ensuite les liens entre les motifs émergents et d'autres types motifs définis dans d'autres domaines.

Définition 1.4.1 - *Motif émergent* : Soient un jeu de données partitionnées en deux classes : ($\mathcal{G}_+$, $\mathcal{G}_-$) et un taux de croissance minimum ρ. Un motif p est un motif émergent de $\mathcal{G}_-$ vers $\mathcal{G}_+$ si $\mathcal{GR}_{\mathcal{G}_+}(p) \geq \rho$.

Les motifs émergents présentent un intérêt particulier puisqu'ils correspondent à des motifs dont la fréquence varie fortement d'une classe à une autre.

Sur l'exemple du tableau 1.1, le motif $p = \{a_1,a_2,a_5\}$ a une fréquence dans $\mathcal{G}_+$ égal à $\frac{2}{5} = 40\%$ et une fréquence dans $\mathcal{G}_-$ de $\frac{1}{5} = 20\%$. Son taux de croissance de $\mathcal{G}_-$ vers $\mathcal{G}_+$ est égal à $\frac{40}{20} = 2$. p est un motif émergent si le taux de croissance minimum $\rho \leq 2$

Les *Jumping Emerging Patterns (JEP)* sont un cas particulier de motifs émergents. Leur taux de croissance est égal à ∞. Ce sont donc les motifs qui ne sont présents que dans une seule classe.

Nous précisons maintenant la notion de *JEP minimal* qui correspond à un JEP dont aucun sous motif n'est un JEP.

Exemple 1.4.1 Le motif $p = \{a_3,a_4,a_5\}$ est un JEP puisqu'il n'apparaît que dans la classe $\mathcal{G}_+$. Cependant $p = \{a_3,a_4,a_5\}$ n'est pas minimal car il contient le JEP $p = \{a_3,a_5\}$.

En revanche, le motif $\{a_3,a_5\}$ est un JEP minimal.

Plusieurs classifieurs comme CAEP [DZWL99], JEPC [LDR00], DeEPs [LDRW04], BCEP [FR03] sont fondés sur les motifs émergents car ces motifs permettent d'avoir une bonne prédiction grâce notamment à leur fort pouvoir discriminant.

Les motifs émergents forment un cas particulier de motifs de contraste, ce type de motifs ayant pour but de faire ressortir les différences entre les classes. Nous indiquons maintenant d'autres ensembles de motifs de contraste définis dans la littérature.

Hypothèse et espace des versions en analyse formelle des concepts : Les motifs émergents sont liés à la notion d'*espace des versions* [Mit82] en analyse formelle des concepts (AFC). Un espace des versions regroupe les motifs qui sont supportés par tous les objets d'une classe.

Sebag et al. [Seb96] introduisent la notion d'*espace des versions disjunctif* qui regroupe les motifs qui sont supportés au moins par un objet d'une classe et par aucun autre objet d'une autre classe. Ce sont donc les JEPs.

En AFC un JEP correspond par définition à une *hypothèse* [GK03] qui regroupe les ensembles d'attributs présents que dans une seule classe.

Subgroup Discovery : L'objectif du *subgroup discovery (SD)* [Klo96, Wro97] est : "étant donné un ensemble d'individu avec leurs propriétés, de trouver la sous-population d'individu (avec leurs caractéristiques) qui sont les plus intéressants selon une propriété". Ce problème est similaire à celui de l'extraction des motifs émergents. Il s'agit de découvrir des motifs décrivant un sous-ensemble d'objets plus intéressants selon une mesure dans une classe par rapport aux autres classes. Plusieurs mesures sont utilisées comme la *couverture pondérée* [LKFT04b] ou le *Weighted Relative Accuracy(WRAcc)*. Les approches de subgroup discovery ont été utilisées par exemple en médecine [KLGK07, GL02] ou en marketing [dJGHM07].

Contrast Set Mining : Soit un jeu de données partitionné en groupes (classes) et composé d'objets décrits par des attributs. Bay et Panzani [BP01] ont défini le *contrast set mining* comme étant la découverte de "conjonctions d'attributs qui diffèrent significativement d'un groupe à un autre". Plus généralement, un contrast set correspond à la prémisse d'une règle dont la conclusion est une classe, et dont le support de la prémisse vérifie des seuils de support dans les différents groupes. On peut ainsi constater le lien étroit avec les motifs émergents qui cependant utilise la mesure du taux de croissance dans leur définition.

Plusieurs évolutions ont été proposées dans le cadre du contrast set mining : Wong et al. [WT05] ont proposé l'intégration de la négation (absence) d'attribut et Simon et Hilderman [SH07] définissent une approche qui permet d'obtenir les contrast sets sur des données quantitatives sans pour autant les discrétiser. Les contrast sets ont été utilisés dans plusieurs domaines tels que la médecine [KLGK07] ou la modélisation de programme d'assurance [WT05].

Supervised Descriptive Rule Discovery (SDR) : Le contrast set mining, le subgroup discovery ou encore l'extraction de motifs émergents poursuivent un même objectif mais avec des techniques différentes. Aussi Novak et al. [NLW09] ont défini un cadre unifiant ces trois domaines nommé *Supervised Descriptive Rule Discovery (SDR)*. Les exemples d'applications de ces SDR sont les réseaux sociaux [HCGdJ11, Atz15] ainsi qu'en médecine [FGL12, DB13, chapters 11, 12].

1.5 Evaluation d'un classifieur

Pour une partie de nos contributions, nous utilisons un classifieur afin d'évaluer la qualité des règles extraites. Quatre indicateurs (c.f. tableau 1.3) sont couramment utilisés pour évaluer les performances d'un classifieur [DG06]. Ces indicateurs se basent sur les classes réelles et les classes prédites des objets : l'approche consiste à prédire la classe d'un objet puis de vérifier sur de nouveaux objets si la classe prédite est bien la classe réelle de cet objet.

Tableau 1.3 – Indicateurs de performances d'un classifieur dans un contexte de classification à deux classes

Classe prédite	**Classe réelle**	
	Positive	**Négative**
Positive	tp (vrai positive)	fp (faux positive)
Négative	fn (faux négative)	tn (vrai négative)

Un classifieur idéal (i.e. prédisant correctement toutes les classes réelles des objets) a les taux de tp (true positive) et de tn (true negative) égaux à 1 et ceux de fp (false positive) et de fn (false negative) égaux à 0. Dans ce mémoire, nous évaluons le classifieurs en utilisant la mesure appelée *précision*.

Définition 1.5.1 - *Précision d'un classifieur* : La précision d'un classifieur $\mathcal{C}$ est le ratio du nombre d'objets de classe réelle positive correctement prédites sur le nombre total de transactions prédites positives : elle est notée $prec(C) = \frac{tp}{tp+fp}$

Pour calculer cette précision, nous utilisons dans ce travail deux approches : la validation croisée à n volets et le leave one out [KEN95].

Définition 1.5.2 - *Validation croisée à n volets* : La validation croisée à n volets consiste d'abord à diviser le jeu de données en n sous ensembles. Ensuite, tour à tour pour chaque sous-ensemble $\mathcal{O}$ l'apprentissage est fait sur les $n-1$ autres sous ensembles. La prédiction du classifieur $\mathcal{C}$ est effectuée sur l'ensemble $\mathcal{O}$ pour obtenir une précision $prec_{\mathcal{O}}(C)$. La précision du classifieur correspond à la moyenne des $prec_{\mathcal{O}}(C)$ avec $\mathcal{O}$ étant les différents sous-ensembles du jeu de données.

Définition 1.5.3 - *Leave One Out* : Concernant le *Leave One Out(LOO)*, chaque objet du jeu données est tour à tour exclu. Le reste du jeu de données sert pour l'apprentissage. Et la prédiction est faite sur l'objet exclu. La précision est ensuite calculée selon la formule donnée à la définition 1.5.1

1.6 Conclusion

Dans ce chapitre, nous avons expliqué la notion de motifs et de règles d'association ainsi que les mesures permettant d'évaluer leur qualité. Nous avons aussi expliqué la notion de motifs émergents ainsi que son équivalence dans d'autres domaines. Les motifs émergents sont largement utilisés pour la prédiction de la classe des objets. Nous avons, pour finir, détaillé quelques mesures qui sont employées pour évaluer la qualité d'un classifieur. Nous utilisons ces mesures dans la suite de ce mémoire.

Chapitre 2

Nouvelle méthode d'extraction des JEPs minimaux

Sommaire

Calculer tous les *JEPs minimaux* permet de réduire considérablement le nombre de motifs extraits par rapport à l'ensemble des JEPs et aussi d'obtenir des classifieurs avec

une bonne précision. Cependant extraire tous les JEPs minimaux peut être une tâche très coûteuse en temps d'exécution. Nous proposons dans ce chapitre un nouvel algorithme *AMinJEP* qui permet d'extraire efficacement tous les JEPs minimaux. *AMinJEP* permet aussi d'extraire l'ensemble des JEPs minimaux, en y incluant si l'utilisateur le souhaite une contrainte de fréquence. *AMinJEP* permet aussi d'extraire d'une manière efficace les JEPs minimaux en intégrant l'absence d'attribut. *AMinJEP* définit une structure d'arbre *ToMJEPs (Tree of the Minimal JEPs)* dont la construction se base sur la notion de *différences entre objets*. Dans la première section de ce chapitre, nous définissons les notations et les préliminaires nécessaires pour la présentation de *AMinJEP*. La deuxième section présente la notion de différences entre objets ainsi que la structure d'arbre *ToMJEPs* que nous proposons et dont tout chemin de la racine vers un nœud "valide" correspond à un JEP minimal. Nous terminons ce chapitre avec une partie expérimentale qui montre l'efficacité de la méthode comparé aux autres méthodes d'extraction des JEPs minimaux. Puis, nous évaluons les JEPs minimaux dans le cas où ils sont considérés comme des règles concluant sur une classe ; nous montrerons dans quelle mesure les JEPs minimaux couvrent les objets du jeu de données et quelle est la confiance de ces règles.

Le travail présenté dans ce chapitre fait l'objet d'une publication à PAKDD [KCC15].

2.1 Notations et préliminaires

2.1.1 Notations

Soit $\mathcal{G}$ un jeu de données composé d'un ensemble d'objets et partitionné en deux classes $\mathcal{G}_+$ et $\mathcal{G}_-$. Un objet de $\mathcal{G}_+$ est appelé objet *positif* et un objet de $\mathcal{G}_-$ un objet *négatif*. L'ensemble des attributs de $\mathcal{G}$ permettant de décrire les objets est noté $\mathcal{M}$. Dans ce chapitre, nous ne faisons pas de différence entre un objet et sa description.

Un *motif* désigne un ensemble d'attributs, un élément de $\mathcal{P}(\mathcal{M})$. On définit la relation, I, sur $\mathcal{G} \times \mathcal{P}(\mathcal{M})$ comme suit : pour chaque objet o et pour tout motif p, $oIp \iff p \subseteq o$. Un motif p est *supporté par* un objet o si p est un sous-ensemble de la description de o : $p \subseteq o$. L'*extension* d'un motif p dans $\mathcal{G}$, noté $p'_{\mathcal{G}}$, correspond à l'ensemble des objets qui supportent p : $p'_{\mathcal{G}} = \{o \in \mathcal{G} : oIp\}$. Le support d'un motif p dans $\mathcal{G}$, noté $supp_{\mathcal{G}}(p)$, correspond à la cardinalité de $p'_{\mathcal{G}}$.

Le *complémentaire* d'un objet o, noté $\overline{o}$, correspond à l'ensemble des attributs de $\mathcal{P}(\mathcal{M})$ qui ne sont pas supportés par o : $\{m \in \mathcal{M} : \neg(o\ I\ m)\}$.

Exemple 2.1.1 Sur le tableau 2.1, un jeu de données composé de 10 objets est représenté. Ce jeu de données est composé de cinq objets négatifs et de cinq objets positifs.

Six attributs sont utilisés pour décrire les objets. La présence d'un attribut dans un objet est noté par le signe x.

Comme on peut le voir sur le tableau 2.1, l'objet négatif o_6 est décrit par les attributs a_2,a_3,a_4 et a_6 : $o_6 = \{a_2,a_3,a_4,a_6\}$.

Le motif $p = \{a_1,a_3,a_5\}$ est supporté par les objets $\mathbf{o}_2$ et $\mathbf{o}_3$, $p'_{\mathcal{G}} = \{\mathbf{o}_2,\mathbf{o}_3\}$.

Le complémentaire de l'objet o_6, $\overline{o_6} = \{a_1,a_5\}$ comme indiqué sur le tableau 2.2.

Tableau 2.1 – Jeu de données de 10 objets partitionné en deux classes

	Objets \ Attributs	a_1	a_2	a_3	a_4	a_5	a_6
$\mathcal{G}_+$	$\mathbf{o}_1$	x	x			x	
	$\mathbf{o}_2$	x		x	x	x	
	$\mathbf{o}_3$	x		x		x	
	$\mathbf{o}_4$	x	x			x	
	$\mathbf{o}_5$			x	x	x	
$\mathcal{G}_-$	$\mathbf{o}_6$		x	x	x		x
	$\mathbf{o}_7$	x	x			x	x
	$\mathbf{o}_8$				x		
	$\mathbf{o}_9$		x			x	
	$\mathbf{o}_{10}$	x			x	x	x

2.1.2 Correspondance entre les complémentaires d'objets et les JEPs minimaux

Cette section montre le lien qui existe entre les complémentaires d'objets et les JEPs minimaux. Une remarque importante est de constater qu'un motif p n'est pas supporté par un objet o, si p contient un ou plusieurs attributs de $\overline{o}$.

Remarque 2.1.1 : *Soit o un objet et p un motif. p n'est pas supporté par o si p intersecte avec le complémentaire de o : $\neg(o\ I\ p) \Leftrightarrow \overline{o} \cap p \neq \emptyset$.*

Le fait d'utiliser les complémentaires donne lieu à des ensembles d'attributs qui possèdent un fort pouvoir discriminant. Il s'ensuit de la remarque 2.1.1 la proposition suivante qui définit les JEPs minimaux en utilisant les intersections des complémentaires d'objets négatifs.

Proposition 2.1.1 : *Un motif supporté p est un JEP minimal si les deux conditions suivantes sont vérifiées :*

i) $\forall o^- \in \mathcal{G}_-$, $\overline{o^-} \cap p \neq \emptyset$

ii) $\forall a \in p$, $\exists o^- \in \mathcal{G}_-$ *tel que* $p \cap \overline{o^-} = \{a\}$.

Cette proposition atteste qu'un motif p est un JEP minimal si :

— p possède au moins un attribut dans le complémentaire de chaque objet négatif. En application de la remarque 2.1.1, p n'est donc supporté par aucun objet négatif. La condition i) permet à p d'*exclure* tous les objets négatifs de son extension.

— La condition ii) impose que pour chaque attribut a élément de p, il existe un complémentaire d'un objet négatif dont l'intersection avec p est égale à $\{a\}$. Chacun des attributs de p est indispensable pour exclure les objets négatifs de son extension. Cette condition s'intéresse à la minimalité de p.

Exemple 2.1.2 Considérons le jeu de données du tableau 2.1, l'intersection du motif $p = \{a_1, a_3, a_5\}$ avec chacun des complémentaires des objets négatifs est non vide comme l'indique le tableau 2.2. En application de la remarque 2.1.1, p n'est supporté par aucun objet négatif. p étant supporté par l'objet positif o_1, p est un JEP.

Montrons que $p = \{a_1,a_3,a_5\}$ n'est pas un JEP minimal. En appliquant la condition ii) de proposition 2.1.1, on peut constater qu'il n'existe aucun complémentaire d'objet négatif dont l'intersection avec p est réduite à $\{a_1\}$. L'attribut a_1 n'est jamais nécessaire à p pour exclure un objet négatif de son extension. On peut ainsi en déduire que le motif $p\backslash\{a_1\} = \{a_3,a_5\}$ est un JEP. Ce qui montre que p n'est un JEP minimal.

La condition ii) de la proposition 2.1.1 permet de conclure sur la minimalité du JEP en examinant uniquement ses attributs pour en identifier ceux qui sont indispensables.

Comme indiqué sur la quatrième ligne du tableau 2.2, $p' = \{a_3,a_5\}$ est un JEP minimal car il vérifie les deux conditions de la proposition 2.1.1 :

— son intersection avec chacun des complémentaires des objets négatifs est non vide. p' est supporté par les objets positifs o_2, o_3 et o_5.

— Pour chaque attribut a de p', il existe un complémentaire d'objet négatif dont l'intersection avec p' est égal à a : $p' \cap \overline{o_7} = \{a_3\}$ et $p' \cap \overline{o_6} = \{a_5\}$

Preuve : de la proposition 2.1.1

1. D'abord démontrons que si les conditions i) et ii) sont vérifiées alors un motif supporté est un JEP minimal.

   Si un motif supporté respecte la condition i), il intersecte avec le complémentaire de chaque objet négatif. Par application de la remarque 2.1.1, ce motif est un JEP.

   Pour la minimalité, supposons que p est un JEP non minimal. Par définition, il existe un JEP q, différent de p, tel que $q \subsetneq p$. Considérons un attribut a tel que

Tableau 2.2 – Intersection de motifs avec les complémentaires des objets négatifs

Objet	$\mathbf{o}_6$	$\mathbf{o}_7$	$\mathbf{o}_8$	$\mathbf{o}_9$	$\mathbf{o}_{10}$
Complémentaire	a_1 , a_5	a_3 , a_4	a_1 , a_2 , a_3 , a_5 , a_6	a_1 , a_3 , a_4 , a_6	a_2 , a_3
Intersection avec $p = \{a_1,a_3,a_5\}$	a_1 , a_5	a_3	a_1 , a_3 , a_5	a_1 , a_3	a_3
Intersection avec $p = \{a_3,a_5\}$	a_5	a_3	a_3 , a_5	a_3	a_3

$a \in p \setminus q$. q étant un JEP, grâce à la remarque 2.1.1, nous avons $q \cap \overline{o^-} \neq \emptyset$, $\forall o^- \in \mathcal{G}_-$, il s'ensuit $p \cap \overline{o^-} \neq \{a\}$, $\forall o^- \in \mathcal{G}_-$. Par contraposition, on peut en conclure que, si un JEP p satisfait la condition ii), alors p est un JEP minimal.

2. Maintenant montrons par contraposition que si un motif est un JEP minimal alors les conditions i) et ii) sont vérifiées.

   Concernant la condition i), supposons qu'un motif supporté p n'intersecte pas avec *un* complémentaire d'un objet négatif o^-. Grâce à la remarque 2.1.1, on en conclut que p est supporté par o^-. D'où p n'est pas un JEP.

   Pour la condition ii), supposons que p est un JEP qui ne vérifie pas la condition ii) : il existe un attribut a dans p tel que $\forall o^- \in \mathcal{G}_-$, $p \cap \overline{o^-} \neq \{a\}$. Comme p est un JEP, en application de la remarque 2.1.1, $\overline{o^-} \cap p \neq \emptyset$, $\forall o^- \in \mathcal{G}_-$. Il s'ensuit que, $\forall o^- \in \mathcal{G}_-$, $\overline{o^-} \cap p \setminus \{a\} \neq \emptyset$. p étant supporté par au moins un objet positif, $p \setminus \{a\}$ est donc supporté par au moins un objet positif. Et grâce à la remarque 2.1.1, $p \setminus \{a\}$ est donc un JEP. D'où p n'est pas un JEP minimal.

La proposition 2.1.1 montre qu'un JEP minimal est un motif supporté par au moins un objet positif, qui exclut tous les objets négatifs et dont chaque attribut est nécessaire pour exclure un objet négatif. Il s'ensuit donc la conséquence suivante :

Conséquence 2.1.1 - (de la proposition 2.1.1) : *Soient p un JEP minimal pour le jeu de données $\mathcal{G}_+ \cup \mathcal{G}_-$ et $o^- \in \mathcal{G}_-$. Si p n'est pas un JEP minimal pour le jeu de données $\mathcal{G}_+ \cup \mathcal{G}_- \setminus \{o^-\}$ alors il existe un unique attribut $a \in p$, tel que $p \setminus \{a\}$ est un JEP minimal pour le jeu de données $\mathcal{G}_+ \cup \mathcal{G}_- \setminus \{o^-\}$.*

Cette conséquence découle de la condition ii) de la proposition 2.1.1 qui établit que dans un JEP minimal chaque attribut est nécessaire pour exclure un objet négatif. En enlevant un attribut a d'un JEP minimal p, le motif obtenu est un JEP minimal pour le jeu de données excepté des objets négatifs dont l'intersection entre le complémentaire et

p est égal à a.

2.2 AMinJEP : une nouvelle méthode de calcul des JEPs minimaux

Cette partie présente le cœur de notre contribution méthodologique, à savoir une nouvelle méthode d'extraction des JEPs minimaux fondée sur les différences entre objets.

2.2.1 ToMJEPS (Tree of the minimal JEPs) : un arbre de recherche des JEPs minimaux

Nous proposons une structure dédiée à générer l'ensemble des JEPs minimaux pour un jeu de données : un arbre enraciné pour lequel l'ensemble des feuilles "valides" est en bijection avec l'ensemble des JEPs minimaux. Cet arbre permet aussi d'élaguer une partie de l'espace de recherche en considérant certains nœuds comme des nœuds d'arrêt c'est à dire qu'aucun de leurs descendants ne peut correspondre à un JEP minimal. Dans la suite, nous supposons que $\forall o^- \in \mathcal{G}_-$, $\overline{o^-} \neq \emptyset$, d'après la proposition 2.1.1, cette condition est nécessaire pour qu'il existe au moins un JEP minimal. D'autre part, un ordre strict est donné entre les objets négatifs c'est à dire que pour deux objets négatifs o^- et o'^-, $o^- \prec o'^-$ si o^- est avant o'^-. Cet ordre permet de choisir d'une manière successive les attributs qui vont servir à construire l'arbre de recherche.

2.2.1.1 Vocabulaire concernant les arbres enracinés

Un *arbre enraciné* (T,r) est un arbre pour lequel un nœud en particulier est distingué, à savoir la racine r. Dans un arbre enraciné, un nœud de degré un, excepté la racine est appelé une *feuille*. Le *chemin* de la racine au nœud x, noté $Chem(x)$, correspond à l'ensemble des nœuds entre la racine et y. Si $\{x,y\}$ est une arête de l'arbre telle que le chemin de la racine de l'arbre à y passe par x, alors y est un *fils* de x. Un *ancêtre* de x désigne n'importe quel nœud sur le chemin de la racine au nœud x. Si x est un ancêtre de y, alors y est un *descendant* de x, et on le note par $x \leqslant y$; si $x \neq y$, on écrit $x < y$.

2.2.1.2 Notations et préliminaires

L'arbre des JEPs minimaux (T,r) que nous proposons est un arbre enraciné dont chaque nœud excepté la racine comporte deux étiquettes : un attribut , $l_{attr}(x) \in \mathcal{M}$, et un objet

négatif $l_{obj}(x) \in \mathcal{G}_-$. Pour un nœud x de (T,r), $Br(x)$ est la conjonction des étiquettes attribut sur le chemin de la racine au nœud x : $Br(x) = \{l_{attr}(y), y \leqslant x\}$; $Br(x)$ correspond au motif considéré au nœud x.

Pour tout nœud x de (T,r) et tout attribut a, $a \in Br(x)$, $crit(a,x)$ est constitué de l'ensemble des objets négatifs déjà considérés au nœud x et dont l'exclusion est due *uniquement* à la présence de l'attribut a dans $Br(x)$: $crit(a,x) = \{o^- \preceq l_{obj}(x) : \overline{o^-} \cap Br(x) = \{a\}\}$.

Définition 2.2.1 - Définition d'un arbre : *Un arbre enraciné* (T,r) *est un* arbre des JEPs minimaux *pour* $\mathcal{G}$ *si les conditions suivantes sont vérifiées :*

i) tout nœud x, excepté la racine r, a deux étiquettes : une étiquette attribut, $l_{attr}(x) \in \mathcal{M}$, et une étiquette objet négatif, $l_{obj}(x) \in \mathcal{G}_-$.

ii) si x est un nœud interne alors :

a) tout nœud fils y de x possède comme étiquette objet le même objet négatif : $l_{obj}(y) = min\{o^- \in \mathcal{G}_- : \overline{o^-} \cap Br(x) = \emptyset\}$, $\forall y$ un nœud fils de x,

b) deux nœuds fils de x possèdent des étiquettes attributs différentes,

c) l'union des différentes étiquettes attributs des nœuds fils y de x correspond au complémentaire de l'étiquette objet de y.

iii) x est une feuille *alors satisfait une de ces conditions :*

a) $\exists z \preceq x$ tel que $crit(l_{attr}(z),x) = \emptyset$,

b) $\forall o^- \in \mathcal{G}_-$, $\overline{o^-} \cap Br(x) \neq \emptyset$.

Une feuille qui satisfait le critère iii)a) est appelée *feuille d'arrêt*, sinon elle est appelée *feuille candidate*. Une feuille candidate supportée par le jeu de données est appelée *feuille valide*. L'ensemble des motifs $Br(x)$, avec x étant les feuilles valides, est en bijection avec l'ensemble des JEPs minimaux.

Exemple 2.2.1 S'appuyant sur le jeu de données du tableau 2.1 et sur les complémentaires d'objets négatifs indiqués sur le tableau 2.1, la figure 2.1 montre un ToMJEPs pour le jeu de données du tableau 2.1. Les feuilles entourées par une ligne en tiret correspondent à des feuilles d'arrêt. Celles entourées d'une ligne grasse sont des feuilles valides. Elles correspondent aux JEPs minimaux. Tandis que les feuilles entourées d'une ligne pleine constituent les feuilles candidates qui ne sont pas supportées par le jeu de données.

Par exemple, expliquons la construction du chemin jusqu'au nœud x tel que $Br(x) = \{a_1,a_3\}$:

— premièrement l'attribut a_1 est sélectionné dans $\overline{o_6}$. $Br(x) = \{a_1\}$;

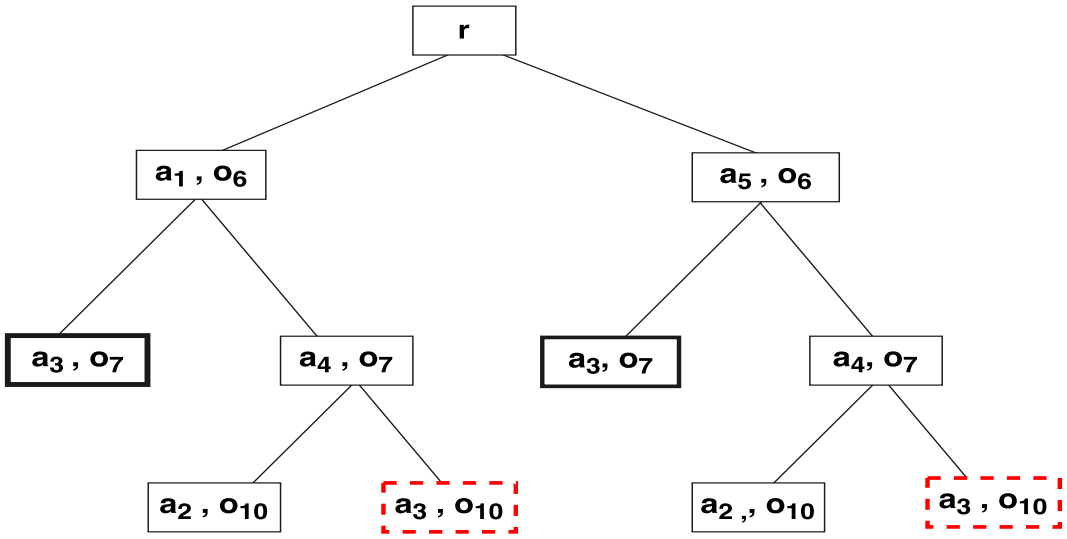

FIGURE 2.1 – Exemple d'arbre des JEPs minimaux

— ensuite comme $Br(x) \cap o_7 = \emptyset$, vient l'attribut a_3 dans $\overline{o_7}$. $Br(x) = \{a_1,a_3\}$, $crit(a_1,a_3) = \{o_6\}$, $crit(a_3,a_3) = \{o_7\}$;
— étant donné que $Br(x) \cap o_8 = \emptyset$, $Br(x) \cap o_9 = \emptyset$ et $Br(x) \cap o_{10} = \emptyset$, aucun attribut n'est sélectionné dans $\overline{o_7}$, $\overline{o_9}$ et $\overline{o_{10}}$
$Br(x) = \{a_1,a_3\}$, $crit(a_1,a_3) = \{o_6\}$, $crit(a_3,a_3) = \{o_7,o_{10}\}$;
— le nœud ayant comme étiquette d'attribut a_3, pour $Br(x) = \{a_1,a_3\}$, est une feuille puisque la condition iii)b) de la définition 2.2.1 est respectée.

La condition iii)a) de la définition 2.2.1 de l'arbre n'étant pas vérifiée, le nœud ayant comme étiquette d'attribut a_3, pour $Br(x) = \{a_1,a_3\}$, est une feuille candidate. Le motif $Br(x) = \{a_1,a_3\}$ étant supporté par les objets positifs $\{o_2,o_3\}$, $Br()$ est un JEP minimal.

Comme autre exemple, $Br(x) = \{a_1,a_4,a_3\}$ est une feuille d'arrêt : car $\forall o^- \in \{o_6,o_7,o_8,o_9,o_{10}\}$, $Br(x) \cap \overline{o^-} \neq \{a_4\}$. La condition iii)a) de la définition 2.2.1 est vérifiée : $crit(a_4,a_3) = \emptyset$. L'attribut a_4 n'est donc pas nécessaire pour exclure un objet négatif de l'extension de $Br(x)$.

A présent, montrons la bijection qui existe entre les feuilles valides et l'ensemble des JEPs minimaux. Pour un jeu de données $\mathcal{G} = \mathcal{G}_+ \cup \mathcal{G}_-$, nous introduisons deux lemmes :

— le premier lemme démontre que chaque feuille valide de l'arbre correspond à un JEP minimal,
— le deuxième lemme montre que pour tout JEP minimal p pour le jeu de données $\mathcal{G}$, il existe un unique chemin de l'arbre, $Br(x)$, égal à p, x étant une feuille candidate.

Lemme 2.2.1 : *Soient (T,r) un ToMJEPs et x un nœud de (T,r), différent d'une*

feuille d'arrêt. S'il existe $o^+ \in \mathcal{G}_+$ *tel que* $o^+ \mathcal{I} Br(x)$ *alors* $Br(x)$ *est un JEP minimal pour le jeu de données* $\mathcal{G}' = \mathcal{G}_+ \cup \{o^- \leq l_{obj}(x)\}$.

Ce lemme permet de conclure, par généralisation sur tout le jeu de données, que chaque feuille valide correspond à un JEP minimal pour le jeu de données $\mathcal{G}$.

Preuve : Par définition d'un ToMJEPs, pour un nœud x, on a $Br(x) \cap \overline{o^-} \neq \emptyset, \forall o^- \leq l \leq l_{obj}(x)$. La condition i) de la proposition 2.1.1 étant vérifiée, $Br(x)$ est un motif ne supportant aucun objet négatif $o^- \leq l_{obj}(x)$.

Si x n'est pas une feuille d'arrêt, par définition d'un ToMJEPs, on a $\forall z \leq x$, $crit(l_{attr}(z),x) \neq \emptyset$, alors $\forall a \in Br(x)$, $\exists o^- \in \{o^- \leq l_{obj}(x)\}$ tel que $Br(x) \cap \overline{o^-} = \{a\}$. La condition ii) de la proposition 2.1.1 est satisfaite, chaque attribut de $Br(x)$ est nécessaire à $Br(x)$ pour exclure l'ensemble des objets négatifs $\{o^- \leq l_{obj}(x)\}$ de son extension.

$Br(x)$ étant supporté par au moins un objet positif $o^+ \in \mathcal{G}_+$, la proposition 2.1.1 établit que $Br(x)$ est un JEP minimal pour le jeu de données $\mathcal{G}_+ \cup \{o^- \leq l_{obj}(x)\}$.

Lemme 2.2.2 : *Soient* (T,r) *un ToMJEPs et* p *un motif. Si* p *est* JEP minimal *pour le jeu de données* $\mathcal{G}_+ \cup \mathcal{G}_-$ *alors il existe une unique feuille candidate* x *tel que* $Br(x) = p$.

Ce lemme montre que tout JEP minimal pour le jeu de données $\mathcal{G}$ correspond à un unique chemin de l'arbre, $Br(x)$, x étant une feuille candidate.

Preuve : La preuve est faite par induction sur $\mathcal{G}_-$. Par souci de simplicité, on note ici l'ensemble des objets négatifs comme $\{1, \ldots, k\}$ avec $k = |\mathcal{G}_-|$. et $1 < 2 < \ldots < k$.

La condition ii)b) de la définition 2.2.1 implique que les nœuds fils du nœud r sont en rapport avec 1 (le premier objet négatif), on a $\overline{1} = \{l_{attr}(x) : x \text{ est un fils de } r\}$. En plus, par définition du ToMJEPs, $crit(l_{attr}(x),x) \neq \emptyset$, donc aucun des fils de r est une feuille d'arrêt. Pour chaque motif p correspondant à un JEP minimal pour le jeu de données $\mathcal{G}_+ \cup \{1\}$, il existe un unique nœud x, différent d'une feuille d'arrêt, tel que $Br(x) = p$.

Supposons maintenant que pour tout JEP minimal p pour le jeu de données $\mathcal{G}_+ \cup \{1, \ldots, l\}$ avec $l < k$, il existe un unique nœud x, différent d'une feuille d'arrêt, tel que $Br(x) = p$. Si on considère un motif q, étant un JEP minimal pour le jeu de données $\mathcal{G}_+ \cup \{1, \ldots, l, l+1\}$, deux cas se présentent :

— Si q est un JEP minimal pour $\mathcal{G}_+ \cup \{1, \ldots, l\}$, alors, grâce à l'hypothèse d'induction, il existe un nœud unique x_q tel que $Br(x_q) = q$.

— Sinon, grâce à la proposition 2.1.1, il existe un attribut a tel que $\overline{(l+1)} \cap q = \{a\}$ et $\overline{o^-} \cap q \neq \{a\}$, $\forall o^- \preceq l$. La proposition 2.1.1 implique que $q \setminus \{a\}$ est un JEP minimal pour $\mathcal{G}_+ \cup \{1, \ldots, l\}$. Grâce à l'hypothèse d'induction, il existe un nœud unique x, différent d'une feuille d'arrêt, tel que $Br(x) = q \setminus \{a\}$. Par définition d'un ToMJEPs, il existe un unique nœud y fils de x, tel que $Br(y) = q$. Comme q est un *minimal* JEP, x n'est pas une feuille d'arrêt.

La proposition suivante est une conséquence des lemmes 2.2.1 et 2.2.2 :

Proposition 2.2.1 : *Soit (T,r) un ToMJEPs. Il existe une bijection entre l'ensemble des feuilles valides et l'ensemble des JEP minimaux.*

La proposition 2.2.1 montre qu'on peut générer les JEPs minimaux en faisant un parcours du ToMJEPs et en extrayant les chemins des feuilles valides.

Il est intéressant de remarquer qu'il n'est pas nécessaire de calculer ni de stocker l'ensemble de l'arbre. Le parcours en profondeur nécessite juste le stockage du chemin de la racine jusqu'au nœud couramment visité. Ce stockage se traduit par l'utilisation d'une pile. Pour générer les fils d'un nœud x, seule la connaissance des nœuds de $Chem(x)$ est nécessaire. Les informations sur les autres nœuds de l'arbre ne sont pas prises en compte.

Lorsque le parcours du ToMJEPs arrive sur une feuille alors un backtrack est utilisé, si possible, pour chercher les autres feuilles de l'arbre.

2.2.2 Différences entre objets

Cette section introduit l'idée de la différence entre objets pour le calcul des JEPS minimaux. Après avoir donné une définition de cette notion, nous montrons le lien qui existe entre les complémentaires et les différences d'objets. Dans un deuxième temps, nous exposons comment simplifier le calcul des JEPs minimaux en utilisant les différences entre objets.

Définition 2.2.2 - Différence entre objets : *Soit $\mathcal{G}$ un jeu de données partitionné en deux classes $\mathcal{G}_+$ et $\mathcal{G}_-$. La* différence entre un objet i et un objet j *regroupe les attributs de o qui ne sont pas supportés par o' : $\mathcal{D}_{o,o'} = o \setminus o' = \{m \in \mathcal{M} : \ o\ I\ m \text{ and } \neg o'\ I\ m\}$.*

Si on se limite à un objet négatif obj^-, la **différence globale** *pour obj^- correspond à l'union des différences entre obj^- et chaque objet $o^+ \in \mathcal{G}_+$: $\mathcal{D}_{\bullet obj^-} = \cup_{o^+ \in \mathcal{G}_+} \mathcal{D}_{o+,obj^-}$.*

Exemple 2.2.2

Avec le jeu données du tableau 2.1, la différence globale de l'objet négatif o_8 est égal à $\{a_1\ ,\ a_2\ ,\ a_3\ ,\ a_5\ \}$. Le calcul suivant illustre ce résultat :
$\mathcal{D}_{\bullet o_8} = \mathcal{D}_{o_1,o_8} \cup \mathcal{D}_{o_2,o_8} \cup \mathcal{D}_{o_3,o_8} \cup \mathcal{D}_{o_4,o_8} \cup \mathcal{D}_{o_5,o_8}$
$\mathcal{D}_{\bullet o_8} = \{a_1\ ,\ a_2\ ,\ a_5\ \} \cup \{a_1\ ,\ a_3\ ,\ a_5\ \} \cup \{a_1\ ,\ a_3\ ,\ a_5\ \} \cup \{a_1\ ,\ a_2\ ,\ a_5\ \} \cup \{\ a_3\ ,\ a_5\ \}$
$\mathcal{D}_{\bullet o_8} = \{a_1\ ,\ a_2\ ,\ a_3\ ,\ a_5\ \}$

Les différences globales des objets négatifs sont données sur le tableau 2.3.

Tableau 2.3 – Différences globales des objets négatifs

Objets	$\mathbf{o}_6$	$\mathbf{o}_7$	$\mathbf{o}_8$	$\mathbf{o}_9$	$\mathbf{o}_{10}$
Différences	$a_1\ ,\ a_5$	$a_3\ ,\ a_4$	$a_1\ ,\ a_2\ ,\ a_3\ ,\ a_5$	$a_1\ ,\ a_3\ ,\ a_4$	$a_2\ ,\ a_3$

2.2.2.1 Lien entre les différences et les complémentaires d'objets :

On peut constater aisément que le complémentaire d'un objet négatif est un sur-ensemble de sa différence globale. En effet la différence globale pour un objet o^- regroupe l'ensemble des attributs qui ne sont pas supportés par o^- et qui sont supportés par les objets positifs. Donc le complémentaire de o^- contient la différence globale de o^-.

Cependant tous les éléments du complémentaire d'un objet négatif ne sont pas forcément présents dans la différence globale. Si un attribut a de $\mathcal{M}$ n'est pas supporté par aucun objet positif tout en étant supporté par des objets négatifs o^-, alors a est inclus dans le complémentaire mais jamais dans la différence globale de $o'^- \neq o^-$. Le complémentaire peut ainsi être un sur-ensemble strict de la différence globale.

La remarque 2.1.1 et la proposition 2.1.1 sont toujours valables en utilisant les différences globales. En effet deux étapes sont nécessaires pour le confirmer :

- Un motif p n'est supporté par aucun objet négatif s'il possède au moins un attribut dans chaque différence globale des objets négatifs. En plus si le motif en question est supporté par au moins un objet positif, il correspond alors à un JEP.
- p est minimal si chaque attribut a d'un JEP p, a une intersection avec une différence globale d'un objet négatif égal à a.

La démonstration est similaire à celle de la proposition 2.1.1 en remplaçant cette fois-ci le complémentaire par la différence globale.

Exemple 2.2.3 Si on compare les tableaux 2.2 et 2.3, on constate que les différences globales sont un sous ensemble des complémentaires. Par exemple pour l'objet négatif o_8

et o_9, l'attribut a_6 n'est jamais présent dans la différence globale car il n'est supporté par aucun objet positif du jeu de données.

2.2.2.2 Comment calculer les différences globales

Pour un objet négatif o^-, la différence globale se calcule à partir du complémentaire avec l'algorithme suivant :

Algorithme 1 : Obtention des différences globales en utilisant les complémentaires

```
1 D_{•o⁻} = ∅;
2 pour tous les a ∈ ō⁻ faire
3 |   pour tous les o⁺ ∈ G₊ faire
4 |   |   si a ∈ o⁺ alors
5 |   |   |   D_{•o⁻} = D_{•o⁻} + {a};
6 |   |   |   break;
7 |   |   fin
8 |   fin
9 fin
```

Le calcul des différences est généralement plus rapide que celui des complémentaires. Même si les complémentaires et les différences globales peuvent être équivalents, ces deux ensembles sont différents s'il existe des attributs qui sont présents que dans une seule classe et si de plus leur nombre est important. En effet pour calculer un complémentaire d'un objet négatif, on ne vérifie pas si les attributs du complémentaire sont supportés par les objets positifs. Certains attributs du complémentaire peuvent ne pas être supportés par aucun objet positif et ne seront donc pas dans la différence globale. Dans ce cas, plus ces attributs sont nombreux, plus la cardinalité de la différence globale est moins importante comparée à celle du complémentaire et l'espace de recherche pour la construction du ToMJEPs est beaucoup moins important.

2.2.3 Essential JEPs et top$k$ JEPs minimaux

L'algorithme d'extraction que nous avons avons proposé, effectue un parcours en profondeur afin d'extraire les feuilles valides du ToMJEPs. Nous appelons cet algorithme *AMinJEP*. Deux méthodes d'extraction sont étudiées dans cette partie expérimentale : la première appelée *post-traitement de l'extension* extrait d'abord les feuilles candidates et ensuite vérifie si elles sont des feuilles valides. La deuxième méthode appelée *maintien de l'extension* calcule l'extension au moment de générer chaque nœud du ToMJEPs. La méthode *maintien de l'extension* permet de calculer d'une manière directe les top-k JEPs

minimaux qui sont les k JEPs minimaux les plus supportés et les essential JEPs qui sont les JEPs minimaux ayant un support supérieur à un seuil minimum fixé.

Pour les essential JEPs, à chaque ajout d'un nœud du ToMJEPs on vérifie si le support est supérieur ou égal au seuil de support fixé s. Un nœud x du ToMJEPs est, dans ce contexte, une feuille d'arrêt si $supp_{\mathcal{G}}(Br(x)) < s$. Concernant le calcul des top-k JEPs minimaux, supposons qu'on ait une liste, noté L, qui contient les top-k JEPs minimaux. Il est à noter que L peut contenir plus de k JEPs minimaux du moment où plusieurs JEPs minimaux de même support peuvent y figurer.

Dans le cas de l'extraction des top-k JEPs minimaux, un nœud x du ToMJEPs est une feuille d'arrêt si $supp_{\mathcal{G}}(Br(x))$ est inférieur au support minimum des JEPs minimaux de L, avec $|L| \geq k$. La méthode d'extraction des top-k JEPs minimaux se déroule, pour chaque JEP minimal extrait, comme décrit à l'algorithme 2.

Algorithme 2 : Mise à jour de la liste des top-k JEPs minimaux

Données : L : liste des top-k JEPs minimaux, *minjep* : un JEP minimal, k : seuil de la liste fixé au départ

```
si |L| ≥ k alors
    si support(minjep) == support-minimum(L) alors
        Ajouter minjep à L ;
    fin
    sinon
        si supp(minjep) ≥ support-minimum(L) alors
            Nb-elt-a-remplacer = {p ∈ L : support(motif) < supp(minjep) } ;
            si (|L| - (Nb-elt-a-remplacer) <  k alors
                Ajouter minjep à L ;
            fin
            sinon
                Enlever tous p ∈ L ;
                Ajouter minjep à L ;
            fin
        fin
    fin
fin
sinon
    Ajouter minjep à L ;
fin
```

2.2.4 Apport de la négation d'attribut

Walczak et *al.* [TW07] ont montré d'une manière expérimentale que prendre en compte l'absence d'un attribut apporte une connaissance qui améliore la précision des classifieurs. Nous incluons cette possibilité dans *AMinJEP*. Pour cela, un nouvel attribut représentant la négation est ajouté, pour chaque attribut, au jeu de données.

Exemple 2.2.4

En intégrant la négation d'attribut sur le jeu données du tableau 2.1, on obtient le jeu de données du tableau 2.4. Le tableau 2.5 indique les différences globales avec la prise en compte de la négation d'attribut.

Tableau 2.4 – Jeu de données de 10 objets partitionné en deux classes avec la négation d'attribut

	Objets \ Attributs	a_1	$\neg a_1$	a_2	$\neg a_2$	a_3	$\neg a_3$	a_4	$\neg a_4$	a_5	$\neg a_5$	a_6	$\neg a_6$
$\mathcal{G}_+$	$\mathbf{o}_1$	x		x			x		x	x			x
	$\mathbf{o}_2$	x			x	x		x		x			x
	$\mathbf{o}_3$	x			x	x			x	x			x
	$\mathbf{o}_4$	x		x			x		x	x			x
	$\mathbf{o}_5$		x		x	x		x		x			x
$\mathcal{G}_-$	$\mathbf{o}_6$		x	x		x		x			x	x	
	$\mathbf{o}_7$	x		x			x		x	x		x	
	$\mathbf{o}_8$		x		x		x	x			x		x
	$\mathbf{o}_9$		x	x			x		x	x			x
	$\mathbf{o}_{10}$	x			x		x	x		x		x	

Tenant compte des différences globales avec la négation d'attribut, le motif $p = \{a_3, \neg a_4,\}$ est un JEP minimal car les deux conditions de la proposition 2.1.1 sont vérifiées :

Tableau 2.5 – Tableau des différences globales des objets négatifs en utilisant la négation d'attribut

Objets	$\mathbf{o_6}$	$\mathbf{o_7}$	$\mathbf{o_8}$	$\mathbf{o_9}$	$\mathbf{o_{10}}$
Différences globales	a_1,$\neg a_2$,$\neg a_3$,$\neg a_4$,a_5	$\neg a_1$,$\neg a_2$,a_3,a_4,$\neg a_5$,	a_1,a_2,a_3,$\neg a_4$,a_5	a_1,$\neg a_2$,a_3,a_4,$\neg a_5$	$\neg a_1$,a_2,a_3,$\neg a4$, $\neg a_5$

i) l'intersection de p avec chacun des différences globales des objets négatifs est non vide. p est supporté par l'objet positif o_3.

ii) Pour chaque attribut a de p, il existe une différence globale d'objet négatif dont l'intersection avec p est égal à a : $p' \cap \overline{o_7} = \{a_3\}$ et $p' \cap \overline{o_6} = \{\neg a_4\}$

2.3 Lien avec les traverses minimales d'hypergraphes

Étant donné un hypergraphe $\mathcal{G}$ bien défini, nous montrons, dans cette section, que les JEPs minimaux sont en bijection avec les traverses minimales de $\mathcal{G}$. Nous détaillons aussi l'apport algorithmique du ToMJEPs pour extraire efficacement les JEPs minimaux en comparaison avec les algorithmes de calcul des traverses minimales.

Hypergraphe, traverse et traverse minimale.

Définition 2.3.1 - *Hypergraphe [Ber73]* : Un hypergraphe $\mathcal{G}$ est défini comme étant un couple ($\mathcal{V}$,$\mathcal{E}$), $\mathcal{V}$ étant l'ensemble des sommets et $\mathcal{E}$ un ensemble de parties de $\mathcal{V}$ appelé hyperarêtes.

Définition 2.3.2 - *Traverse minimale d'hypergraphes* : Soit un hypergraphe $\mathcal{G}$ = ($\mathcal{V}$,$\mathcal{E}$). Un ensemble de sommets p est une *traverse* de $\mathcal{G}$ si p intersecte avec toutes les hyperarêtes de $\mathcal{G}$: $\forall e \in \mathcal{E}$, $p \cap e \neq \emptyset$.

Une traverse p de $\mathcal{G}$ est *minimale* si elle n'est incluse dans aucune autre traverse : $\forall p' \subsetneq p$, $\exists e \in \mathcal{E}$ tel que $p' \cap e = \emptyset$

Exemple 2.3.1 Soit l'hypergraphe $\mathcal{G}$ avec $\mathcal{V} = \{a,b,c,d,e\}$ et $\mathcal{E} = \{e1 = \{a,b\}$, $e2 = \{c,d\}$, $e3 = \{b,d\}$, $e4 = \{c,d,e\}\}$.

$p = \{a,d,e\}$ est une traverse car $\{a,d,e\} \cap e1 = \{a\}$, $\{a,d,e\} \cap e2 = \{d\}$, $\{a,d,e\} \cap e3 = \{d\}$, $\{a,d,e\} \cap e4 = \{d,e\}$. Cependant p n'est pas une traverse minimale car il contient $p' = \{a,d\}$ qui est aussi une traverse.

Méthodes de calcul des traverses minimales D'une manière chronologique, le premier algorithme de calcul des traverses minimales est celui de Berge [Ber73]. Cet algorithme construit d'abord les traverses d'une seule hyperarête (i.e. chacun des sommets de l'hyperarête), puis ajoute successivement les autres hyperarêtes, tout en mettant à jours les traverses minimales de l'hypergraphe partiel obtenu à chaque ajout. Cependant cette méthode n'est pas utilisable sur de gros hypergraphes. Fredman et *al.* [FK96], améliorent l'algorithme pour avoir une meilleure complexité. L'algorithme présenté résout le problème en $\mathcal{O}(nm) + m^{logm}$ où n est le nombre de variables et m est la taille combinée de l'entrée et de la sortie. Hébert et *al.* [HBC07] ont défini une approche de calcul des traverses minimales d'hypergraphes qui s'inspire de la représentation condensée des motifs et définit une contrainte anti-monotone qui facilite l'élagage. Cette approche fonctionne bien sur de gros hypergraphes et accélère le temps de calcul.

Pour tous ces algorithmes, la minimalité est vérifiée en post-traitement, ce qui augmente significativement les temps de calculs. Uno et *al.* [MU13] introduisent la notion *d'hyperarête critique* qui permet d'anticiper pendant le calcul sur la minimalité d'une traverse d'hypergraphe. Cette notion d'hyper-arête critique a inspiré le concept d'objets critiques qui est utilisé par Soulet et *al.* [SR14] pour la conception d'un algorithme efficace en profondeur d'extraction des motifs minimaux. En effet, on montre que les objets critiques caractérisent les motifs minimaux et peuvent être calculés à partir de l'information disponible uniquement sur une seule branche du parcours, ce qui est une situation idéale pour la conception d'un algorithme en profondeur.

Relation entre les traverses minimales et les JEPs minimaux : Étant donné un jeu de données, considérons l'hypergraphe $\mathcal{G} = (\mathcal{V},\mathcal{E})$, avec $\mathcal{V}$ correspondant aux attributs du jeu de données et $\mathcal{E}$ aux différences globales (voir section 2.2.2). La première condition de la proposition 2.1.1 affirme que si un motif supporté intersecte avec toutes les différences globales alors c'est un JEP. On calcule ainsi des traverses de $\mathcal{G}$. La deuxième condition de la proposition 2.1.1 assure la minimalité de ces traverses.

Les traverses minimales ainsi obtenues sont en bijection avec l'ensemble des JEPs minimaux car la proposition 2.2.1 montre, en s'appuyant sur la proposition 2.1.1, que l'ensemble de motifs obtenu est en bijection avec l'ensemble composé de tous les JEPs minimaux. Cela implique qu'on peut obtenir les JEPs minimaux en calculant les traverses minimales et exclure celles qui ne sont supportées par aucun objet.

Apport algorithmique du ToMJEPS Dans ce paragraphe, nous montrons l'apport de l'arbre de recherche ToMJEPS, qui en plus de générer tous les JEPs minimaux permet d'anticiper, pendant le calcul, si un motif peut donner lieu à un JEP minimal et d'arrêter le calcul si ce n'est pas le cas. Pour cela nous avons introduit dans l'arbre de recherche la notion de *crit* qui correspond à la notion d'*hyperarête critique* sur les travaux Uno *al.* [MU13] et qui permet d'élaguer l'arbre de recherche en n'explorant que les branches qui correspondent à un JEP minimal. A chaque ajout d'un nœud dans l'arbre, le *crit* de tous les attributs du motif est mis à jour. Nous avons aussi introduit la notion de mise à jour de l'extension qui permet d'améliorer l'élagage de l'arbre de recherche ToMJEPs. Cela consiste pour chaque nœud y à calculer l'extension de $Br(y)$ en se basant sur l'extension de $Br(x)$, x étant le nœud père de y. Soient x et y deux nœuds du ToMJEPs avec $x \prec y$. L'extension de $Br(y)$ est égale à l'ensemble des objets de l'extension de $Br(x)$ qui supportent $Br(y)$. Ainsi à chaque ajout d'un nœud, l'extension est calculée sur un nombre d'objets beaucoup plus réduit se basant uniquement sur l'extension du nœud père. Si l'extension pour un nœud x est vide alors x est une feuille d'arrêt. Le processus de mise à jour de l'extension et du *crit* se déroule donc selon ces étapes :

- si l'extension pour un nœud x n'est pas vide, alors la mise à jour du *crit* est effectuée,
- si *crit(x)* est vide pour $x \in Br(x)$ alors x devient une feuille d'arrêt, sinon on continue la génération,
- si l'intersection de $Br(x)$ avec toutes les différences globales est vide avec *crit(x)* non vide pour tout $x \in Br(x)$ et l'extension de $Br(x)$ non vide, alors $Br(x)$ est un JEP minimal.

Par rapport aux travaux de Uno *al.* [MU13] qui utilisent une pile pour extraire à la suite les traverses minimales, nous proposons ainsi un cadre, le ToMJEPs, qui permet d'exprimer les JEPs minimaux et ainsi associer plusieurs méthodes de parcours tout en donnant des informations sur l'extension des motifs.

Dans la partie expérimentale, deux approches sont comparées : le calcul de l'extension une fois que l'intersection de $Br(x)$ avec toutes les différences globales est vide (post-traitement), et la mise de l'extension à chaque ajout d'un nœud.

2.4 Résultats expérimentaux

L'objectif de cette section est de fournir et de commenter les résultats expérimentaux de la méthode d'extraction des JEPs minimaux. Premièrement nous étudions l'efficacité de la méthode d'extraction sur des jeux de données de l'UCI [Lic13] notamment en évaluant les temps d'extraction des JEPs minimaux comparés aux méthodes existantes. Nous mon-

trons aussi comment la méthode d'extraction proposée permet de calculer d'une manière directe les essentials JEPs ou les top-k JEPs minimaux. Les JEPs minimaux étant un sous ensemble des motifs libres, cette section montre aussi la comparaison de *AMinJEP* avec l'algorithme d'extraction des motifs libres de Boulicaut et al. [BBR03]. Nous effectuons une comparaison avec l'algorithme des *motifs optimaux (MO)* d'Ugarte et al. [UBLC15] qui propose une méthode d'extraction des JEPs minimaux basé sur la programmation par contrainte. En dernière partie, cette section montre l'intérêt d'utiliser les JEPs minimaux comme des règles exprimant l'implication entre un motif et une modalité de la classe.

2.4.1 Matériels et méthodes

2.4.1.1 Jeux de données

Les jeux de données, utilisés dans cette partie expérimentale, sont disponibles sur le répertoire UCI [1] [Lic13]. Ils ont été choisis parce qu'ils sont généralement utilisés dans les travaux concernant l'extraction des JEPs [DL05, FR02, TW08]. Ces jeux de données sont décrits au tableau 2.6. La colonne *Répartition (%)* désigne le pourcentage d'objets pour chaque classe du jeu de données.

Préparation des jeux de données Comme l'algorithme d'extraction fonctionne avec des jeux de données binaires et certains jeux de données n'étant pas sous cette forme, nous les avons binarisé en utilisant l'algorithme de discrétisation proposé par Frans Coenen [2]. Cette technique de discrétisation est une méthode dite *supervisée*, *bottom-up* et *directe*. Cette méthode est supervisée car les attributs concernant la classe sont supposés connus et ainsi ne sont pas discrétisés contrairement aux méthodes *non supervisées*. Elle est aussi directe parce qu'elle spécifie un nombre maximum d'intervalles de valeurs pour les attributs, à contrario des méthodes *incrémentales* où un nombre maximum n'est pas fixé. Cette méthode est dite bottom-up car, pour un attribut, elle donne au départ une valeur pour chaque intervalle et fait un réajustement jusqu'à ce que tous les intervalles soient couverts par au moins un objet. Tandis que une méthode de discrétisation de type *Top-down*, on part d'une seule valeur d'intervalle pour un attribut et divise itérativement cet intervalle jusqu'à ce que tous les intervalles soient couverts par au moins un objet.

On peut citer, parmi les principaux algorithmes de discrétisation supervisées, bottom-up et incrémentales, la méthode de Kleber [Ker92] et celle de Haun et Setiono [LS97]. Deux méthodes reconnues de discrétisation supervisées, top-down et incrémentales sont Kurgan et Cios [KC04] et Tsai et al. [TLY08].

1. `http://archive.ics.uci.edu/ml/`
2. `http://cgi.csc.liv.ac.uk/~frans/KDD/Software/LUCS-KDD-DN/exmpleDNnotes.html`

La technique de discrétisation de Frans Coenen permet, contrairement aux autres méthodes de discrétisation, d'avoir une meilleure répartition des intervalles de valeur pour un attribut grâce notamment au fait que, pour un attribut, plusieurs intervalles de valeurs de valeur sont définis au départ et qu'ensuite un rééquilibrage est effectué pour couvrir au mieux les objets du jeu de données. La méthode de Coenen est aussi plus efficace car le nombre maximum d'intervalles étant fixé au départ, les temps d'exécution sont plus réduits comparées aux méthodes incrémentales.

La description détaillée de la méthode de Coenen est fournie à cette adresse web `http://cgi.csc.liv.ac.uk/~frans/KDD/Software/LUCS-KDD-DN/lucs-kdd_DN.html`. On peut récupérer à partir de ce lien un logiciel implémenté sous JAVA et une documentation très détaillée sur son mode d'emploi.

2.4.2 Extraction des JEPs minimaux : analyse des résultats expérimentaux

Nous utilisons, dans cette partie expérimentale, les différences globales. D'autre part, dans l'objectif de produire tous les JEPs minimaux pour chaque valeur de classe, nous considérons d'une manière itérative chaque classe du jeu de données comme étant la classe positive et l'union des autres classes comme la classe négative. Nous avons calculé les JEPs minimaux sur 13 jeux de données choisis en utilisant les méthodes en *post-traitement* et du *maintien de l'extension*. Nous extrayons aussi les essential JEPs avec deux seuils de support, 1% et 5%. Les top-k JEPs minimaux sont aussi calculés pour des seuils de $k = 10$ et $k = 20$. Les résultats sont fournis en utilisant la négation d'attribut ou pas.

Le tableau 2.7 donne, sans tenir compte de la négation d'attribut, le nombre de JEPs minimaux extraits ainsi que les temps d'exécution si on maintient l'extension durant le parcours du ToMJEPs ou si on effectue le calcul de l'extension en *post-traitement*. Le tableau 2.8 montre les mêmes caractéristiques si la négation d'attribut est utilisée. Sur les tableaux 2.7 et 2.8, la colonne *Nb. JEPs min.* désigne le nombre de JEPs minimaux extraits. La colonne *Post-trait.* fait référence à la méthode d'extraction en *post-traitement*. La colonne *Compl* mentionne le cas où les complémentaires d'objets négatifs sont utilisés pour le calcul des JEPs minimaux plutôt que les différences globales.

Les temps d'extraction avec la négation d'attributs sont plus importants que sans l'usage de la négation. En moyenne sur les 13 jeux de données et en extrayant tous les JEPs minimaux, ces temps d'extraction sont 2 fois plus importants avec la négation d'attribut. Une augmentation du temps d'extraction était prévisible vu que l'usage de la négation d'attribut revient à ajouter des attributs.

Cette différence de temps d'extraction est aussi observée entre les méthodes de *post-*

Tableau 2.6 – Jeu de données UCI

Jeu de donnée	**Objets**	**Attributs**	**Classes**	**Répartition (%)**
breast	699	20	2	$C_1 = 65.52$ $C_2 = 34.48$
congres	435	34	2	$C_1 = 61.38$ $C_2 = 38.62$
ecoli	336	34	8	$C_1 = 45.56$ $C_2 = 22.92$ $C_3 = 15.48$ $C_4 = 10.42$ $C_5 = 5.95$ $C_6 = 1.49$ $C_7 = 0.60$ $C_8 = 0.60$
glass	214	48	7	$C_1 = 35.51$ $C_2 = 32.71$ $C_3 = 12.05$ $C_4 = 7.94$ $C_5 = 6.07$ $C_6 = 4.21$ $C_7 = 1.50$
heart	303	52	5	$C_1 = 54.13$ $C_2 = 18.15$ $C_3 = 11.88$ $C_4 = 11.55$ $C_5 = 4.29$
hepatitus	155	56	2	$C_1 = 79.35$ $C_2 = 20.65$
iris	150	19	3	$C_1 = 33.33$ $C_2 = 33.33$ $C_3 = 33.33$
mushroom	8124	90	2	$C_1 = 51.80$ $C_2 = 48.20$
pima	768	38	2	$C_1 = 65.10$ $C_2 = 34.90$
tic-tac-toe	958	29	2	$C_1 = 65.34$ $C_2 = 34.66$
waveform	5000	101	3	$C_1 = 33.92$ $C_2 = 33.14$ $C_3 = 32.94$
wine	178	68	3	$C_1 = 39.89$ $C_2 = 33.15$ $C_3 = 26.97$
zoo	101	42	7	$C_1 = 40.59$ $C_2 = 19.80$ $C_3 = 12.87$ $C_4 = 9.90$ $C_5 = 7.92$ $C_6 = 4.95$ $C_7 = 3.96$

Tableau 2.7 – Extraction des JEPs minimaux sans la négation d'attribut

	Tous les JEPs minimaux				Essential JEPs		Top-k JEPs minimaux	
		Post-trait.	*Maint. Ext.*	*Compl.*	1%	5%	10	20
Jeu de données	**Nb. JEPs min.**	**Temps**	**Temps**	**Temps**	**Temps**	**Temps**	**Temps**	**Temps**
iris	46	94,415 ms	49,362 ms	68,287 ms	32,452 ms	20,411 ms	14,281 ms	24,392 ms
breast	28	217,216 ms	89,467 ms	121,241 ms	70,23 ms	53,212 ms	39,312 ms	48,439 ms
ecoli	124	745,281 ms	328,173 ms	504,216 ms	210,356 ms	170,214 ms	105,315 ms	143,319 ms
zoo	946	1908,376 ms	715,271 ms	941,228 ms	401,351 ms	300,391 ms	173,457 ms	251,334 ms
pima	169	383,385 ms	171,673 ms	241,287 ms	80,431 ms	53,287 ms	45,186 ms	61,386 ms
glass	1552	1629,219 ms	755,213 ms	985,213 ms	417,272 ms	310,286 ms	154,376 ms	227,197 ms
congres	2805	15,242 s	6,261 s	6,992 s	3,054 s	1,997 s	0,756 s	1,121 s
hepatitus	4134	5,897 s	2,452 s	3,102 s	1,003 s	0,788 s	0,452 s	0,772 s
heart	4802	2781,340 ms	1204,376 ms	1506,724 ms	698,314 ms	439,365 ms	314,345 ms	429,217 ms
tic-tac-toe	3287	2318,345 ms	925,211 ms	1421,271 ms	532,454 ms	403,452 ms	276,276 ms	397,420 ms
wine	4604	2845,42 ms	1287,288 ms	1721,916 ms	789,345 ms	503,340 ms	401,451 ms	604,437 ms
mushroom	4438	13,256 s	5,013 mn	5,995 mn	3,376 mn	2,089 mn	1,234 mn	2,004 mn
waveform	1934992	178,387 mn	59,342 mn	62,654 mn	34,350 mn	25,345 mn	16,450 mn	22,451 mn

Tableau 2.8 – Extraction des JEPs minimaux avec la négation d'attribut

	Tous les JEPs minimaux				Essential JEPs		Top-k JEPs minimaux	
		Post trait.	*Maint. Ext.*	*Compl.*	1%	5%	10	20
Jeu de données	**Nb. JEPs min.**	**Temps**	**Temps**	**Temps**	**Temps**	**Temps**	**Temps**	**Temps**
iris	126	210,271 ms	79,418 ms	110,310 ms	52,178 ms	30,378 ms	47,271 ms	55,198 ms
breast	82	2147,38 ms	721,003 ms	1021,218 ms	399,45 ms	173,43 ms	218,471 ms	252,998 ms
ecoli	1576	7,100 s	2,935 s	4,045 s	1,564 s	0,801 s	1,341 s	1,678 s
zoo	20453	10,451 s	5,341 s	6,812 s	1,897 s	0,894 s	0,672 s	0,845 s
pima	2899	16,652 s	7,190 s	8,819 s	1,953 s	1,210 s	2,512 s	4,012 s
glass	358584	159,032 s	70,508 s	74,783 s	43,561 s	20,352 s	9,241	11,988 s
congres	113826	183,790 s	82,154 s	85,378 s	44,561 s	20,451 s	7,894	11,004 s
hepatitus	840318	264,041 s	117,318 s	122,217 s	55,187 s	30,810 s	7,951 s	9,193 s
heart	626321	17,052 mn	7,017 mn	7,976 mn	2,560 mn	1,451 mn	49,161 s	57,600 s
tic-tac-toe	219898	13,172 mn	7,421 mn	8,162 mn	118,718 s	30,783 s	11,225 s	15,573 s
wine	4601976	510,152 mn	200,451 mn	203,319 mn	145,150 mn	96,919 mn	20,562 mn	31,452 mn
mushroom	33690281	1210,217 mn	801,201 mn	808,767 mn	369,467 mn	189,534 mn	50,345 mn	76,281 mn
waveform	73686219	5189,367 mn	2679,297 mn	2699,451 mn	1100,452 mn	689,874 mn	134,671 mn	168,245 mn

traitement et de *maintien de l'extension*. La dernière est de 1,6 à 3 fois plus rapide que la première. Cet écart est plus significatif si le nombre de JEPs minimaux est très important. On peut observer ce phénomène sur les jeux de données volumineux ayant beaucoup d'attributs comme waveform ou wine.

Concernant le calcul des essential JEPs ou des top-k JEPs minimaux, on peut voir sur les tableaux 2.7 et 2.8 que plus le seuil est élevé plus le temps d'extraction et le nombre de motifs extraits diminue en comparaison avec l'ensemble de tous les JEPs minimaux.

D'autre part, nous avons comparé les temps d'exécution de la méthode d'extraction AMinJEP en utilisant les complémentaires d'objets (voir section 2.1) ou les différences globales (voir section 2.2.2). Ces temps d'exécution sont très proches. En effet les complémentaires d'objets et les différences globales sont différents uniquement s'il existe des attributs qui sont définis que dans une seule classe ; ce qui n'est pas généralement pas le cas dans les jeux de données que nous avons utilisés.

2.4.3 Comparaison de AMinJEP avec d'autres méthodes d'extraction des JEPs minimaux

Nous comparons expérimentalement notre méthode avec deux méthodes d'extraction des JEPs minimaux. La première est un algorithme d'extraction des motifs libres proposé par Boulicaut et al. [BBR03] appelé *ACMiner*. Comme les JEPs minimaux sont des motifs libres, mais que tous les libres ne sont pas des JEPs minimaux, nous sélectionnons parmi les motifs libres ceux qui correspondent à des JEPs minimaux via un post traitement de la sortie de *ACMiner*.

L'autre méthode d'extraction des JEPs minimaux est une méthode fondée sur la programmation par contraintes proposée par Ugarte et al. [UBLC15] et qui introduit la notion de *Motifs Optimaux (MO)*. Les *MO* sont les meilleurs motifs par rapport à une préférence définie par l'utilisateur. On peut citer à titre d'exemple les motifs libres, maximaux, fermés ou les skypatterns. Dans cette partie, nous nous intéressons aux JEPs minimaux et faisons la comparaison avec la méthode d'extraction des JEPs minimaux que nous proposons, *AminJEP*, avec l'algorithme ACMiner et la méthode des *MO*. Concernant l'algorithme *AminJEP*, la méthode avec *maintien de l'extension* est utilisée ainsi que les différences globales.

Le résultat de la comparaison entre ces trois méthodes est donné au tableau 2.9, sans la négation d'attribut, et au tableau 2.10 avec la négation d'attribut.

On remarque, d'après les tableaux 2.9 et 2.10, que notre méthode AMinJEP est plus rapide en temps d'exécution. Cet écart devient plus significatif si la négation d'attribut est utilisée. Dans le cas où la négation d'attribut n'est pas utilisée, la méthode AMinJEP est

Tableau 2.9 – Comparaison entre AminJEP, ACMiner et MO (sans la négation d'attribut)

		AMinJEP	ACMiner	MO
Jeu de données	**Nb.Min.JEPs**	**Temps**	**Temps**	**Temps**
iris	46	49,362 ms	90,426 ms	121,044 ms
ecoli	124	328,173 ms	381,128 ms	497,368 ms
breast	28	89,467 ms	134,866 ms	161,078 ms
pima	169	171,673 ms	253,307 ms	587,941 ms
zoo	946	715,271 ms	880,504 ms	1552,838 ms
glass	1552	755,213 ms	923,447 ms	2031,526 ms
heart	4802	1204,376 ms	1855,935 ms	3277,162 ms
tic-tac-toe	3287	925,211 ms	1221,271 ms	2310,196 ms
wine	4604	1287,288 ms	1782,363 ms	5717,270 ms
hepatitus	4134	2,452 s	5,021 s	9,210 s
congres	2805	6,261 s	9,012 s	22,101 s
mushroom	4438	5,013 mn	7,408 mn	17,593 mn
waveform	1934992	59,342 mn	65,289 mn	213,818 mn

en moyenne sur les 13 jeu de données presque 1,5 fois plus rapide, en temps d'exécution, que ACMiner et presque 3 fois plus rapide que la méthode des *MO*. Si la négation d'attribut est intégrée dans l'extraction des JEPs minimaux, cet écart monte à 3 pour ACMiner et 5 pour les *MO*.

Cette différence de temps d'exécution s'explique par le fait que ACMiner et MO extraient des motifs qui ne sont pas des JEPs (et donc pas des JEPs minimaux). Ces deux algorithmes génèrent souvent beaucoup de motifs inutiles pour notre cas d'étude. Grâce aux conditions de la proposition 2.1.1 de la section 2.1 qui permettent de définir la notion de feuille d'arrêt dans le ToMJEPs, AMinJEP détecte efficacement les motifs qui ne sont pas des JEPs minimaux.

Tableau 2.10 – Comparaison entre AminJEP, ACMiner et MO (avec la négation d'attribut)

		AMinJEP	ACMiner	MO
Jeu de données	**Nb.Min.JEPs**	**Temps**	**Temps**	**Temps**
iris	126	79,418 ms	214,487 ms	332,320 ms
ecoli	1576	2,935 s	9,214 ms	16,217 s
breast	82	721,003 ms	1429,378 ms	3510,418 ms
pima	2899	7,190 s	15,101 s	24,056 s
zoo	20453	5,341 s	17,281 s	25,192 s
glass	358584	1,175 mn	5,361 mn	6,715 mn
heart	626321	7,017 mn	17,324 mn	31,210 mn
tic-tac-toe	219898	7,421 mn	13,165 mn	20,248 mn
wine	4601976	200,451 mn	670,218 mn	981,127 mn
hepatitus	840318	1,955 mn	3,697 mn	6,703 mn
congres	113826	1,369 mn	2,120 mn	3,669 mn
mushroom	33690281	801,201 mn	2143,219 mn	3711,201 mn
waveform	73686219	2679,297 mn	9101,101 mn	17297,205 mn

2.4.4 Les JEPs minimaux comme règles exprimant des corrélations

Un JEP exprime une forte corrélation entre les attributs qui le composent et une classe. Dans cette partie nous étudions l'intérêt des JEPs minimaux s'ils sont considérés comme des règles exprimant des corrélations. Pour cela on vérifie dans quelle proportion les JEPs minimaux couvrent les objets du jeu de données et si les JEPs minimaux sont des règles ayant une bonne confiance. Nous faisons aussi une étude comparative ayant pour but d'étudier l'apport de la négation d'attribut dans ce contexte.

Pour évaluer les valeurs de couverture et de confiance, nous avons procédé à un leave one out (LOO). Le LOO consiste à exclure successivement chaque objet o d'un jeu de

données $\mathcal{G}$. Les JEPs minimaux (essential JEP ou top-k JEPs minimaux) sont extraits sur le jeu de données $\mathcal{G} \setminus \{o\}$. Le *LOO* permet d'être au plus proche de la réalité et d'avoir des résultats beaucoup plus fiables même si les temps d'exécution sont plus importants.

La *couverture* d'une règle correspond à la proportion d'objets qui supportent cette règle. La *confiance* dénote la probabilité que la conclusion de la règle soit vérifiée lorsque la prémisse de la règle est supportée. La *couverture moyenne* pour un ensemble de règles correspond à la moyenne des couvertures des règles de l'ensemble. La *confiance moyenne* pour un ensemble de règles correspond à la moyenne des confiances des règles de l'ensemble. Le tableau 2.11 montre la couverture moyenne et la confiance moyenne en considérant les JEPs minimaux, les essential JEPs ou les top-k JEPs minimaux comme des règles d'association. La colonne *Pas de négation* fait référence au cas où les négations d'attributs ne sont pas prises en compte et la colonne *Avec la négation* si les négations d'attributs sont utilisées. La colonne *Cov* indique la couverture et *Conf* la confiance de l'ensemble de règles considéré. La colonne *eJEP* indique la couverture et la confiance si les essential JEPs sont employés. Quant à la colonne *top-k*, elle correspond au cas des top-k JEPs minimaux.

Par exemple, si on considère le jeu de données *tic-tac-toe* en ne considérant pas la négation d'attribut, 82,42% des objets sont couverts au moins par un JEP minimal. Cette couverture atteint 91,03% dans le cas où on intègre la négation d'attribut. Si on reprend toujours le même jeu de données et sans la négation d'attribut, quand les JEPs minimaux couvrent un objet alors dans 78,39% des cas en moyenne ils prédisent la classe de l'objet couvert. Cette confiance chute faiblement et passe à 75,38% si la négation d'attribut n'est pas prise en compte. D'une manière générale, les valeurs de confiance, dans les cas où on intègre la négation d'attribut ou pas, sont assez comparables.

Le tableau 2.11 montre aussi que les JEPs minimaux couvrent une large proportion des objets : pour 7 jeu de données sur 13, plus de 80% des objets sont couverts par au moins un JEP minimal. Cette couverture devient plus importante si la négation d'attribut est utilisée. Pour *hepatitus* un écart de couverture de presque 8% est constaté. Les valeurs de confiance calculées montrent que les JEPs minimaux possèdent généralement une bonne confiance et mettent bien en exergue la corrélation entre un motif et une classe.

Plus le seuil minimum de support est élevé, plus la couverture baisse. Par exemple pour le jeu de données *hepatitus*, la couverture passe de 82,42% pour tous les JEPs minimaux à 69,32% avec les essential JEPs de support 5%. Cette couverture est égale à 69,91% pour les top-k JEPs minimaux avec $k = 10$. Cependant en utilisant un seuil de support minimum les JEPs minimaux extraits ont une plus grande confiance comparée au cas où tous les JEPs minimaux sont extraits. Toujours avec le même jeu de données *hepatitus*, la confiance augmente de 78,39% pour tous les JEPs minimaux à 84,01% avec les essential

JEPs de support 5%. Et la confiance s'élève à 85,90% avec les top-k JEPs minimaux, k étant égal à 10.

Les deux descriptions, avec et sans négation d'attribut, produisent des JEPs minimaux qui ont des valeurs de confiance assez similaires. Cependant les JEPs minimaux extraits avec la négation d'attribut couvrent un plus grand nombre d'objets que les JEPs minimaux calculés sans la négation d'attribut. Le temps de calcul augmente en intégrant la négation d'attribut. En utilisant un seuil minimum de support élevé, on couvre moins d'objets du jeu de données comparé au cas où aucun seuil n'est fixé (support minimum égal à 1). Cependant les JEPs minimaux extraits possèdent une plus grande confiance.

2.5 Conclusion

Dans ce chapitre, nous avons proposé un algorithme *AminJEP* qui permet d'extraire d'une manière efficace tous les JEPs minimaux avec ou sans l'absence d'attribut. Nous avons proposé une structure d'arbre *ToMJEPs* se basant sur les différences entre objets pour extraire les JEPs minimaux. Cet arbre permet de réduire significativement l'espace de recherche et contribue fortement à l'efficacité de la méthode. Nous avons montré expérimentalement que *AminJEP* est beaucoup plus rapide en temps d'exécution que les méthodes existantes pour calculer les JEPs minimaux. Les tests expérimentaux ont aussi montré que les JEPs minimaux, considérés comme des règles concluant sur une classe, ont une bonne confiance avec cependant une couverture qui dépend fortement du support des JEPs minimaux. En effet plus le support des JEPs minimaux est élevé, plus leur confiance est importante avec une moindre couverture des objets. En intégrant l'absence d'attribut, on a constaté d'après les résultats expérimentaux, que la couverture est plus importante comparée au cas où l'absence d'attribut n'est pas prise en compte, avec une confiance assez similaire dans les deux cas.

Partant du constat que les motifs avec un support élevé possèdent une bonne confiance mais ne couvrent pas une bonne partie des objets, nous proposons dans le chapitre 3, une nouvelle méthode qui montre la possibilité d'améliorer la couverture des motifs avec un support élevé sans pour autant dégrader leur confiance.

Tableau 2.11 – Évaluation des JEPs minimaux comme des règles exprimant des corrélations

	Pas de négation										Avec la négation									
	Min. JEPs		eJEP				top-k				All Min. JEPs		eJEP				top-k			
			1%		5%		10		20				1%		5%		10		20	
Datasets	**Cov**	**Conf**	**Cov**	**Conf**	**Cov**	**Conf**	**Cov**	**Conf**	**Cov**	**Conf**	**Cov**	**Conf**	**Cov**	**Conf**	**Cov**	**Conf**	**Cov**	**Conf**	**Cov**	**Conf**
breast	47,78	98,19	45,2	98,32	44,81	99,10	39,16	98,5	43,29	97,01	49,33	96,13	47,42	97,49	45,25	98,51	38,21	97,92	40,21	95,85
congres	99,08	89,7	91,29	91,98	80,37	92,31	71,32	93,72	78,53	91,67	99,00	88,10	92,28	90,31	85,42	92,13	60,43	95,23	70,14	94,23
ecoli	34,52	63,79	30,71	66,38	25,74	71,63	20,32	75,73	25,92	68,97	37,32	71,23	32,12	73,98	26,43	74,21	25,92	75,43	27,13	74,01
glass	90,18	62,17	80,32	70,38	77,90	78,12	70,45	70,32	75,27	68,43	94,13	64,15	87,12	66,53	80,23	68,43	75,43	70,29	78,58	65,23
heart	97,02	58,5	83,32	61,90	80,32	64,87	70,32	71,43	74,23	69,25	98,02	56,42	86,32	57,82	82,55	60,43	61,21	80,32	67,01	78,43
hepatitus	82,42	78,39	72,63	81,30	69,32	84,01	69,91	85,90	72,72	83,94	91,03	75,38	73,23	79,17	70,31	81,83	62,62	80,72	70,39	77,38
iris	88,67	90,98	79,64	92,23	75,43	94,61	71,60	96,10	74,38	94,67	91,33	89,40	85,32	95,39	83,21	96,37	78,62	97,23	82,76	96,37
mushroom	70,15	77,12	52,20	79,44	47,53	88,43	58,18	81,38	60,06	79,92	74,76	78,43	57,92	80,01	53,91	82,54	63,74	80,32	66,34	77,98
pima	17,05	54,2	14,47	58,12	13,54	60,01	13,01	62,43	14,43	58,91	18,54	61,20	16,03	63,01	14,92	65,53	10,74	62,91	13,59	59,54
tic-tac-toe	81,62	86,57	65,32	88,10	61,93	89,21	60,43	80,21	62,32	78,32	82,12	82,34	70,64	83,21	65,47	84,24	72,31	69,43	75,32	67,30
waveform	45,28	72,19	32,93	74,91	29,99	75,10	33,19	73,10	35,92	72,01	48,09	76,98	40,67	78,01	39,52	82,94	37,91	79,63	40,47	74,78
wine	72,91	67,43	56,72	71,93	41,73	83,01	51,84	70,84	53,42	62,81	73,54	65,89	61,92	67,81	59,02	69,67	55,81	69,51	61,90	67,39
zoo	89,11	92,22	72,19	94,44	60,91	95,10	68,92	93,78	73,24	87,32	93,24	91,43	79,32	92,47	77,19	93,93	60,33	92,48	67,39	90,03

Chapitre 3

Un nouveau cadre de sélection de règles supervisées

Sommaire

Dans la section 2.4.4 du chapitre 2, nous avons montré que les JEPs avec un support élevé ne couvrent pas une part importante des objets mais ont cependant une confiance élevée. Dans le cas d'une utilisation prédictive, ce défaut de couverture implique que la règle par défaut d'un classifieur à base de règles, s'appuyant sur les JEPs fréquents, est souvent utilisée et dégrade ainsi la précision du classifieur. Nous considérons les motifs utilisés comme des règles concluant sur une classe. Dans ce chapitre, nous proposons une nouvelle approche de sélection de règles ; cette dernière utilise un ensemble de règles prototypes afin d'obtenir un ensemble de règles ayant une couverture significativement améliorée, tout en conservant un niveau de confiance similaire aux règles prototypes. L'idée principale de la

méthode est de considérer un ensemble de règles avec un support élevé comme les règles prototypes et de sélectionner parmi un ensemble de règles candidates, grâce à une méthode des plus proches voisins, celles qui sont les plus similaires aux règles prototypes. Nous espérons ainsi que les règles sélectionnées aient presque la même confiance que les règles prototypes tout en permettant de couvrir des objets qui ne l'étaient pas.

Après avoir expliqué son fonctionnement, nous évaluerons dans une partie expérimentale la méthode de sélection afin de savoir dans quelles proportions la couverture ou la confiance sont améliorées. Nous montrerons aussi l'intérêt de la méthode de sélection dans un cadre de classification, notamment avec l'utilisation du classifieur CPAR. Nous verrons si la sélection permet d'améliorer la précision du classifieur et dans quelle mesure. Nous terminons ce chapitre par une variante itérative de la méthode de sélection qui considère que toutes les règles sélectionnées deviennent à leur tour des règles prototypes lors d'une sélection de nouvelles règles.

3.1 Compromis entre deux mesures de qualité : couverture et confiance

3.1.1 Motivation

Soit un jeu de données composé d'objets et partitionné en plusieurs classes, chaque objet étant décrit par des attributs et un motif correspondant à une conjonction d'attributs. Dans ce contexte, l'objectif du Rule Learning [CN89] est de trouver les relations intéressantes entre les objets et leurs descriptions. En fouille de données, on peut citer ainsi plusieurs travaux qui entrent dans ce cadre, comme par exemple, les règles d'associations [AMS+96], les motifs fréquents [HCXY07] ou les motifs clos [BHPW10]. Dans ce chapitre, nous nous intéressons aux *Supervised Descriptive Rules (SDRs)* qui décrivent le relation entre les descriptions des objets et une classe du jeu de données. Des travaux importants ont été menés dans le domaine des *SDR* à savoir le *subgroup discovery* [Wro97, Atz15], *les motifs émergents* [DL99] et le *contrast set mining* [BP01]. Un motif émergent est défini comme une conjonction d'attributs dont le support varie significativement d'une classe à une autre. En subgroup discovery, l'objectif est de découvrir les groupes d'objets dont les descriptions qui varient le plus d'une classe à une autre. Et en contrast set mining, ce sont les conjonctions d'attributs qui diffèrent significativement d'un groupe à un autre. Ces travaux ayant tous pour objectif d'extraire les conjonctions d'attributs fortement corrélées à une classe, Novak et al. [NLW09] ont donc défini un cadre commun nommé *supervised descriptive rule discovery*. Ce cadre permet d'unifier la terminologie, les définitions et les méthodes de calcul. Les motifs qui concluent

fortement sur une classe présentent un intérêt particulier en matière de découverte de connaissances [CC11, KW10, LPHL$^+$10] ou de prédiction [GTC14].

Pour évaluer la qualité des motifs extraits, plusieurs mesures sont utilisées. Une liste très complète de ces mesures est donnée dans [HCGdJ11, FGL12]. Francisco Herrera et *al* [HCGdJ11] ont partitionné les mesures en quatre catégories comme la complexité, la généralité, la précision et l'intérêt. On peut citer plusieurs mesures qui sont largement exploitées en fouille de données comme les mesures de couverture [LKFT04a], de support [LKFT04a], de fréquence [HCXY07] ou de taux d'émergence [DL99].

Dans ce travail, nous nous concentrons sur les mesures de support, de taux d'émergence. Nous utilisons la notion de règle qui correspond, dans notre cas, à un conjonction d'attributs (motif) comme prémisse et dont la conclusion est une classe. A la section 2.4.4 du chapitre 2, nous avons montré expérimentalement que les règles (JEPs) avec un support élevé ne couvrent généralement pas beaucoup d'objets du jeu données mais ont une confiance importante. Cette affirmation peut se vérifier dans la littérature [KCC15, FR02, TW08].

Après avoir accepté les règles avec un fort pouvoir discriminant et beaucoup supportées par le jeu de données, la question qui se pose est : comment pourrait-on intégrer à cet ensemble d'autres règles afin de couvrir une plus grande partie des objets sans pour autant dégrader la confiance obtenues sur les règles acceptées. Cela permettrait dans un contexte de classification de trouver une explication pour la prédiction de la classe des objets qui n'étaient pas couverts, et éventuellement, d'améliorer la précision du classifieur. Pour réaliser cet objectif, nous utilisons une méthode des plus proches voisins pour valider les règles candidates à l'intégration.

Un classifieur k-NN (k plus proches voisins) [CH67] prédit la classe d'un objet comme étant la classe majoritaire parmi ses k voisins (généralement la distance euclidienne est utilisée pour déterminer les voisins). Cependant ces classifieurs posent le problème du nombre de voisins à consulter qui peut être très important et ainsi des temps de calcul très longs sans oublier la tolérance au bruit. La sélection à base de prototypes [PDP06] a été appliquée sur les classifieurs k-NN pour résoudre ce problème. Les méthodes de sélection à base de prototypes permettent ainsi de sélectionner un ensemble d'objets, qui sont les plus représentatifs, et qui vont être utilisés pour calculer le voisinage d'un objet dont la classe doit être prédite. Pour appliquer la sélection à base de prototypes sur les classifieurs k-NN, plusieurs méthodes ont été utilisées à l'image de *Kcentres* [DJdR$^+$04]. Cette approche construit chaque prototype comme étant l'objet central (selon la distance par exemple) de tous les objets ayant le même voisinage. Pekalska et al. [PPD01] ont aussi montré qu'un choix aléatoire des prototypes pouvait donner de bons résultats en classification. Pekalska et al. [PDP06] ont aussi montré dans leurs résultats expérimentaux que la sélection de 3

à 10% des objets permet d'avoir de meilleurs résultats que le classifieur $k - NN$ sans la sélection. Un état de l'art des différentes méthodes de sélection à base de prototypes est donné dans [GDCH12]. Notre approche rentre bien dans ce cadre : nous utilisons des règles au lieu d'objets et nous validons les règles candidates à l'intégration par une méthode des plus proches voisins. Pour valider une règle, nous calculons ses voisins parmi les règles avec un fort pouvoir discriminant et bien supportées (règles prototypes).

Dans ce travail, nous utilisons le support et le taux de croissance pour la sélection des règles prototypes. En effet ces deux mesures sont de bons gages de fiabilité pour une règle [AS94, DL99]. Nous considérons ainsi les règles les plus supportées comme étant les règles prototypes. Et nous sélectionnons parmi les autres règles (règles candidates) celles qui sont les plus proches (similaires) aux règles prototypes. L'ensemble de règles ainsi obtenu permet d'obtenir une plus grande couverture que celui des règles prototypes avec une confiance assez similaire.

3.1.2 Méthode de sélection

3.1.2.1 Principe de la sélection

La figure 3.1 résume le principe de la méthode de sélection comme décrit dans cette section.

Supposons que l'on dispose d'un jeu de données partitionné en plusieurs classes, sur lequel on souhaite extraire des règles dont la conclusion est une classe. Certaines de ces règles apparaissent suffisamment fréquemment dans le jeu de données pour que l'on puisse évaluer leur pouvoir discriminant. Parmi ces règles, les règles suffisamment discriminantes sont acceptées et considérées comme des règles *prototypes*.

Les autres règles (non prototypes) sont considérées comme les règles *candidates* à être validées. Les gages de fiabilité d'une règle candidate proviennent non seulement de son support et de son taux de croissance mais aussi de sa ressemblance avec les règles prototypes. La méthode de sélection proposée se base sur les règles prototypes pour évaluer les règles candidates : une règle candidate est *validée* si elle est suffisamment similaire aux règles prototypes.

La validation d'une règle candidate s'appuie sur une méthode des plus proches voisins [CH67, KGG85, CD07]. Étant donnée une mesure de similarité, on peut déterminer les règles prototypes qui sont les plus proches d'une règle candidate. Si les plus proches règles prototypes concluent généralement sur la même classe que la règle candidate alors cette règle candidate est validée. L'ensemble des règles *sélectionnées* correspond à l'union des règles prototypes et des règles validées. La méthode de sélection repose donc sur l'intuition que si une règle candidate ressemble fortement à des règles prototypes qui concluent

majoritairement sur la même classe, alors cette règles candidate sera fiable et devrait atteindre un niveau de confiance similaire à celle des règles prototypes.

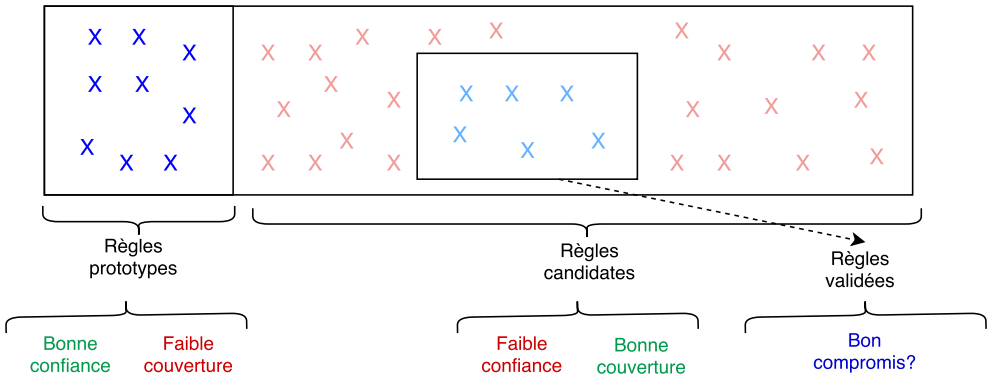

FIGURE 3.1 – Principe de la méthode de sélection

3.1.2.2 Paramètres de la méthode de sélection

Cette méthode de sélection est mise en œuvre grâce à une méthode des plus proches voisins ; cette dernière nécessite de fixer trois paramètres. Le premier paramètre consiste à adopter une mesure de similarité entre deux règles : cette mesure permet d'évaluer les règles prototypes les plus proches d'une règle candidate. Le deuxième paramètre concerne le voisinage d'une règle candidate : il faut établir un seuil de similarité en dessous duquel les règles prototypes ne font font pas partie du voisinage d'une règle candidate. Ce paramètre permet d'identifier les règles prototypes qui seront les voisins d'une règle candidate. Nous notons la similarité minimale pour qu'une règle prototype soit voisine d'une règle candidate par k. Le dernier paramètre porte sur la proportion minimale de règles prototypes voisines qui doivent conclure sur la même classe que la règle candidate afin de la valider ; ce paramètre est noté r.

Illustration de la sélection d'une règle. La figure 3.2 montre un ensemble de règles prototypes et candidates. Chaque règle conclut sur la classe positive (+) ou négative (-). Étant donnée une mesure de similarité, les cas (1), (2) et (3) représentent chacun une règle candidate et ses voisins parmi les règles prototypes.

Fixons à 0,7 la proportion minimale r de règles prototypes voisines qui doivent conclure sur la même classe que la règle candidate afin de la valider. Pour le cas représenté par (1) la règle candidate qui conclut sur la classe positive est sélectionnée car sur les sept règles prototypes voisines, cinq concluent sur la même classe : $r = 5/7 = 0{,}71 > 0{,}7$. Pour le cas (2), la règle candidate est sélectionnée car $r = 3/4 = 0{,}75 > 0{,}7$. Quant à la règle candidate représentée par le cas (3), elle n'est pas sélectionnée car $r = 1/3 = 0{,}33 < 0{,}7$.

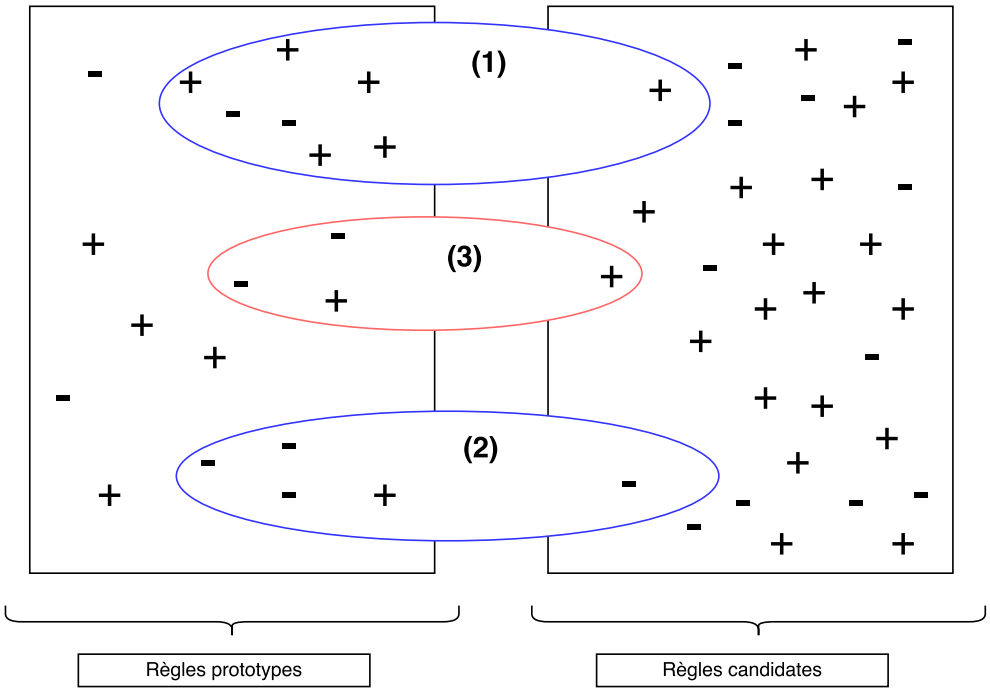

FIGURE 3.2 – Validation de règles avec $r = 0{,}7$ et une similarité définie

3.1.2.3 Déterminer les valeurs de k et r

Pour réaliser la méthode de sélection, après avoir choisi une mesure de similarité, il reste à affecter une valeur aux deux autres paramètres k,r pour chaque modalité de la classe du jeu d'apprentissage. La figure 3.3 montre le processus de calcul de ces couples de valeurs k et r. Le principe consiste à utiliser un leave one out (LOO) sur le jeu d'apprentissage afin de déterminer le meilleur couple de valeur k et r qui permettra de sélectionner les meilleures règles candidates.

Tour à tour chaque objet du jeu d'apprentissage est exclu et considéré comme un objet de validation, $Objet_m$. Le reste des objets constitue le jeu d'entraînement sur lequel les règles prototypes ainsi que les règles candidates sont extraites. Pour chaque modalité de la classe c, nous notons par $rc(m)$, l'ensemble des règles candidates extraites supportant l'objet $Objet_m$ et qui concluent sur la classe c. Et pour chaque couple de valeurs k et r envisagé, une partie des règles candidates concluant sur la classe c est validée. Grâce à l'objet de validation $Objet_m$, on évalue la qualité des règle candidates de $rc(m)$. Si une règle de $rc(m)$ est supportée par l'objet de validation et conclut sur la même classe que l'objet $Objet_m$, alors cette règle est comptée comme un *succès*. Sinon elle est considérée comme un *échec*.

Une fois que chaque objet est considéré comme un jeu de validation, pour chaque modalité de la classe, nous retenons le couple (k, r) qui maximise le taux de succès. La

formule suivante est utilisée pour calculer le taux de succès pour un couple de valeurs (k_i, r_j) si on dispose de p objets dans le jeu d'apprentissage :

$$TS_c(k_i,r_j) = \frac{\sum_{m=1}^{p} |rcv(m)|}{\sum_{m=1}^{p} |rc(m)|}$$

avec $rcv(m)$ étant l'ensemble des règles candidates validées avec le couple de valeurs (k_i, r_j), qui sont supportées par l'objet de validation m et qui concluent sur la même classe que l'objet de validation m. Ces règles candidates correspondent aux succès et sont appelées les règles *validées*. On remarque que $rcv(m) \subseteq rc(m)$.

Cette méthode de validation permet de déterminer le couple de valeurs k et r qui valide "au mieux" les règles candidates les plus proches des règles prototypes.

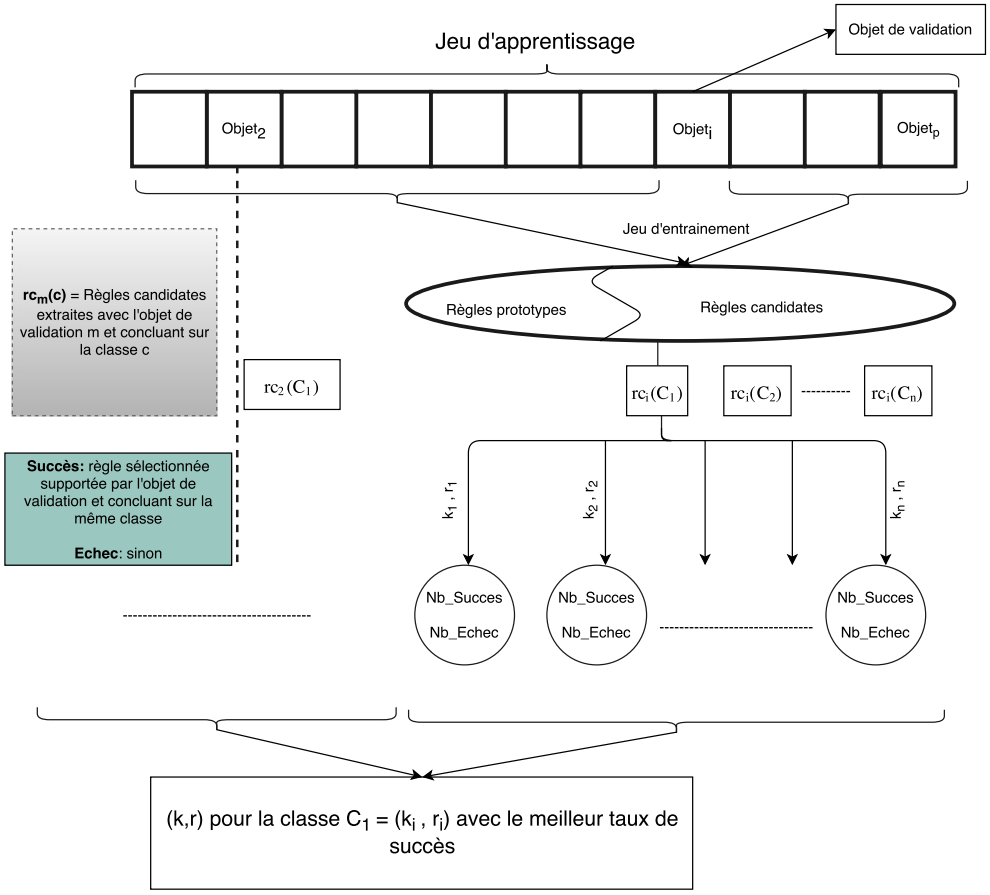

FIGURE 3.3 – Processus de calcul du couple de valeurs k et r

Un couple de valeur k et r est défini pour chaque modalité de la classe car le nombre de règles peut différer largement d'une classe à une autre. Le choix d'un couple de valeurs unique sur toutes les classes pourrait impliquer la non sélection de certaines règles qui

seraient intéressantes. Comme par exemple, les règles candidates qui concluent sur une classe faiblement représentée dans le jeu de données.

Soient $supmin_{pr}$ le support minimum pour les règles prototypes et $supmin_{cand}$ le support minimum des règles candidates avec $supmin_{pr} > supmin_{cand}$ Pour améliorer l'efficacité de la méthode de calcul du couple de valeurs k et r, nous simulons un leave-one-out sur le jeu d'apprentissage, comme expliqué dans la sous-section 3.1.2.3. Afin de réduire les temps de calcul, nous extrayons une seule fois toutes les règles prototypes et candidates. Pour chaque objet *obj* du jeu d'apprentissage, soit $Cov(obj)$, l'ensemble des règles candidates qui couvrent *obj* et dont le support est compris entre $supmin_{pr}$ et $supmin_{cand} + 1$. Les règles candidates à l'origine étant celles avec un support compris entre $supmin_{pr} - 1$ et $supmin_{cand}$, le support des règles de $Cov(obj)$ est pris entre $supmin_{pr}$ et $supmin_{cand}+1$ pour prendre en compte le fait que si on exclut *obj* de leur support (principe du leave one out) alors ce support est compris entre $supmin_{pr} - 1$ et $supmin_{cand}$ qui correspond au support des règles candidates.

Pour les différents valeurs du couple de valeurs (k,r) envisagées dans le calcul du couple de valeurs (k,r), nous avons fait varier k et r de 0 à 1 avec des pas de 0,02.

3.2 Résultats expérimentaux

Cette section présente et évalue les résultats expérimentaux de la méthode de sélection des règles. Nous introduisons les jeux de données qui seront utilisés pour l'évaluation expérimentale, ainsi que les différents paramètres définis pour l'implémentation de la méthode. Cette section montre la fiabilité de la méthode sélection en étudiant la couverture et la confiance des règles.

La deuxième partie de cette section évalue la méthode de sélection dans le contexte où un classifieur à base de règles est utilisée. Cela permet de mesurer qualitativement les différents ensembles de règles obtenus avec la méthode de sélection en évaluant leur précision.

3.2.1 Matériels et méthodes

3.2.1.1 Jeux de données

Les tests ont été effectués sur 13 jeux de données de l'UCI Machine Learning repository [Lic13]. Ce sont les mêmes jeux de données de la section 2.4 du chapitre 2 qui sont utilisés. Ces jeux de données n'étant pas binaires, nous les avons binarisés avec l'algorithme de discrétisation proposé par Frans Coenen [3].

3. `http://cgi.csc.liv.ac.uk/~frans/KDD/Software/LUCS-KDD-DN/exmpleDNnotes.html`

3.2.1.2 Les paramètres de la méthode de sélection

Notre méthode de sélection repose sur l'utilisation d'une mesure de similarité et d'un choix pour le couple k et r ainsi que du type de règles à utiliser. Le calcul du couple de valeurs (k, r) est expliqué à la section 3.1.2.3.

Mesure de similarité. Pour mesurer la similarité entre deux règles, nous utilisons le coefficient de Jaccard. Nous notons la prémisse d'une règle r par $pr(r)$. En ne considérant que les prémisses des règles, le coefficient de Jaccard pour deux règles r_1 et r_2, est calculé avec la formule suivante :

$$J(pr(r_1),pr(r_2)) = \frac{|pr(r_1) \cap pr(r_2)|}{|pr(r_1) \cup pr(r_2)|} \tag{3.1}$$

.

Illustration. Si on suppose un jeu de données composé d'objets décrits par les attributs binaires de a_1 à a_6. La similarité entre deux règles $pr(r_1) = a_1,a_2,a_6$ et $pr(r_2) = a_2,a_3,a_5,a_6$ est égale à $\frac{|a_2,a_6|}{|a_1,a_2,a_3,a_5,a_6|} = \frac{2}{5} = 0{,}4$

3.2.1.3 Règles étudiées

Dans la continuité du chapitre 2, nous avons choisi d'étudier les règles associées aux motifs émergents (EPs) [DL99]. Dans un contexte où les jeux de données sont partitionnés en plusieurs classes, les règles associées aux EPs présentent un intérêt particulier grâce à leur fort pouvoir discriminant. Une règle associée à un EP avec un fort taux de croissance et un support élevé est un bon gage pour obtenir une bonne confiance lors de son usage (en classification par exemple). Cependant l'ensemble composé de ces motifs ne couvre pas une bonne partie des objets. En choisissant un taux de croissance ou un support minimum faible pour extraire les EPs, ces derniers couvrent une grande partie des objets, mais au prix d'une baisse de la confiance moyenne des règles associées.

Nous avons ici choisi comme règles prototypes les règles associées aux JEPs avec un support minimum égal à 10. Ce choix s'explique par le fait que sur les 13 jeux de données que nous avons utilisés, ces règles possèdent en moyenne une bonne confiance mais ne couvrent pas une grande partie des objets, comme indiqué sur le tableau 3.1.

Les règles candidates sont des règles associées aux EPs. Nous avons obtenu plusieurs ensembles de règles candidates en fixant successivement le taux de croissance minimum à l'infini, 20, 10, 5 et 2, et le support minimum à 8 à 6, 4, 3 et 2.

Enfin, nous définissons les ensembles de règles suivants et les notations associées qui sont notamment utilisées pour la présentation des résultats :

— *règles prototypes ($\mathcal{RP}$)* : règles associées aux JEPs avec un support minimum égal à $supmin_{pr}$. Ces règles sont jugées fiables.
— *règles candidates ($\mathcal{RC}$)* : règles associées aux EPs dont le support *supp* est compris entre $supmin_{pr} - 1$ et $supmin_{cand}$: $(supmin_{pr} - 1) \geq supp \geq supmin_{cand}$.
— *règles fréquentes ($\mathcal{RF}$)* : règles associées aux EPs dont le support est supérieur ou égal à $supmin_{cand}$. L'ensemble des règles fréquentes correspond à l'union des règles prototypes et des règles candidates.
— *règles validées ($\mathcal{RV}$)* : ce sont les règles validées parmi les règles candidates.
— *règles sélectionnées ($\mathcal{RS}$)* : ensemble composé des règles prototypes et des règles validées.

3.2.1.4 Mesures utilisées pour l'évaluation de la méthode de sélection

Couverture et confiance. Les mesures calculées sur les différents ensembles de règles sont la *couverture* et la *confiance*. La *couverture* d'une règle correspond à la proportion d'objets qui supportent cette règle. La *confiance* dénote la probabilité que la conclusion de la règle est vérifiée lorsque la prémisse de la règle est supportée. La *couverture moyenne* pour un ensemble de règles correspond à la moyenne des couvertures des règles de l'ensemble. La *confiance moyenne* pour un ensemble de règles correspond à la moyenne des confiances des règles de l'ensemble. Dans cette partie expérimentale, toutes les mesures de couverture et de confiance sont empiriques : elles sont mesurées sur un échantillon du jeu de données. Les mesures indiquées correspondent à des valeurs de couverture et de confiance moyennes.

Illustration. Le tableau 3.2 indique les valeurs de couverture et de confiance moyenne des règles prototypes, des règles fréquentes et des règles sélectionnées. La colonne *Cov* indique la couverture et *Conf* la confiance de l'ensemble de règles considéré. *TC* désigne le taux de croissance. La colonne $\mathcal{RS}$ représente les règles sélectionnées et la colonne $\mathcal{RF}$ les règles fréquentes. Les valeurs en gras indique la couverture et la confiance des règles prototypes. Sur ce tableau 3.2, on peut voir que la couverture des règles prototypes (en gras) est égale à 57,93% car seuls 58 objets sur les 101 objets que compte ce jeu de données supportent au moins une règle prototype. Pour ces objets qui supportent au moins une règle prototype, la confiance moyenne des règles prototypes est de 80,29% : si la prémisse d'une règle prototype est supportée par un objet, alors, dans 80,29% des cas, la classe de cet objet correspond à la conclusion la règle.

Variation de couverture et de confiance

Pour tout ensemble de règles, les variations de couverture et de confiance sont toujours mesurées par rapport à la couverture et à la confiance des règles prototypes. Nous parlerons dans cette partie expérimentale de gain en couverture et de perte en confiance. Ces mesures sont calculées avec les formules suivantes :

$$\begin{aligned} Gain\_Couverture(ensemble\_regles) &= \frac{Couverture(ensemble\_regles) - Couverture(regles\_prototypes)}{Couverture(regles\_prototypes)} \\ Perte\_Confiance(ensemble\_regles) &= \frac{Confiance(reglesprototypes) - Confiance(ensemble\_regles)}{Confiance(regles\_prototypes)} \end{aligned} \tag{3.2}$$

Nous illustrons maintenant ces variations avec le tableau 3.2 qui montre, par exemple le gain en couverture des règles sélectionnées avec un taux de croissance infini et un support minimum de 4. Ce gain est égal à $46\% = 0{,}46 = \frac{85{,}02-57{,}93}{57{,}93}$, avec 85,02% étant la couverture des règles sélectionnées avec un taux de croissance infini et un support minimum de 4 et 57,93% la couverture des prototypes. La perte en confiance correspondante est égale à $3{,}9\% = 0{,}39 = \frac{80{,}29-77{,}14}{80{,}29}$, avec 80,29% étant la confiance des règles sélectionnées avec un taux de croissance infini et un support minimum de 4 et 77,14% la confiance des prototypes.

Méthode de calcul de la couverture et de la confiance. Pour chaque jeu de données, nous évaluons la fiabilité de la méthode de sélection en utilisant une validation croisée à 5 volets. Pour chaque volet de la validation croisée, le jeu de données est découpé en un jeu de test et un jeu d'apprentissage. Nous faisons une étude comparative entre la couverture et la confiance des règles sélectionnées et des règles fréquentes grâce au processus défini comme suit.

— Au moyen d'un leave-one-out sur le jeu d'apprentissage, pour chaque modalité de la classe, on détermine le couple de valeurs k et r comme expliqué à la sous-section 3.1.2.3.
— On applique le principe de la sélection énoncé à la sous-section 3.1.2.1. On examine chaque règle candidate rc qui est sélectionnée si la proportion de règles prototypes voisines qui concluent sur la même classe que rc est supérieure ou égale à r.
— Finalement, on mesure sur le jeu de test la couverture moyenne et la confiance moyenne des règles sélectionnées et des règles fréquentes.

Les calculs ont été faits sur un serveur utilisant Ubuntu 14.04 avec 2 processeurs Intel Xeon 2.80 GHz et 512 gigabits de RAM.

3.2.2 Analyse des résultats expérimentaux

3.2.2.1 Analyse des règles prototypes

Le tableau 3.1 montre la couverture moyenne et la confiance moyenne, sur l'ensemble des règles prototypes pour les 13 jeux de données utilisés. Sur ces résultats, on peut observer que la couverture des règles prototypes varie de 33,81% pour le jeu de données *ecoli* à 79,46% pour le jeu de données *wine*. La confiance moyenne, quant à elle, varie de 65,79% pour le jeu de données *waveform* à 91,04% pour le jeu de données *breast*.

D'une manière générale, les règles prototypes ne couvrent pas une bonne partie des objets du jeu de données mais elles possèdent cependant une bonne confiance. Sur les résultats expérimentaux montrés sur le tableau 3.4, on observe que les règles prototypes (valeurs marqués en gras) possèdent une bonne confiance, 78,43% en moyenne sur les 13 jeux de données, avec une faible couverture équivalent à 60.53%. On peut donc raisonnablement les considérer comme des règles fiables mais qui cependant ne couvrent pas une bonne partie de données. Ce phénomène a aussi été observé dans les travaux étudiant les JEPs ayant un support élevé [KCC15, DB13].

Tableau 3.1 – Couverture et confiance moyennes des règles prototypes sur les 13 jeux de données

Dataset	*Cov*	*Conf*	Dataset	*Cov*	*Conf*
breast	50,51	91,04	mushroom	74,98	69,89
congres	64,45	76,32	pima	41,83	84,75
ecoli	33,81	76,07	tic-tac-toe	50,94	84,69
glass	40,45	70,15	waveform	82,98	65,79
heart	69,98	78,05	wine	79,46	67,01
hepatitus	60,45	81,32	zoo	57,93	80,29
iris	67,45	92,37			

3.2.2.2 Analyse des règles sélectionnées

La couverture et la confiance des règles prototypes, fréquentes et sélectionnées pour les jeux de données *Zoo* (respectivement *Mushroom*) sont représentées sur les tableaux 3.2 (respectivement 3.3). Les valeurs en gras représente les mesures pour les règles prototypes.

La colonne $\mathcal{RS}$ représente les règles sélectionnées et la colonne $\mathcal{RF}$ les règles fréquentes. *TC* désigne le taux de croissance minimum pour extraire les EPs associés à l'ensemble de règles considéré, de même *Support_ min* désigne le support minimum utilisé. La colonne *Cov* indique la couverture moyenne et la colonne *Conf* la confiance moyenne. Les résultats de ces deux colonnes sont mesurés sur l'ensemble de règles considéré.

Le tableau 3.4 représente le gain moyen en couverture ou la perte en confiance des règles sélectionnées en comparaison avec la couverture et la confiance des règles prototypes, ces résultats sont mesurés en moyenne sur les 13 jeux de données. La colonne *Gain Cov* montre le gain moyen en couverture et la colonne *Perte Conf* la perte moyenne en confiance.

Tableau 3.2 – Zoo : performances des ensembles de règles

	Support_ min	10		8		6		4		3		2	
TC		$\mathcal{RF}$	$\mathcal{RS}$	$\mathcal{RF}$	$\mathcal{RS}$	$\mathcal{RF}$	$\mathcal{RS}$	$\mathcal{RF}$	$\mathcal{RS}$	$\mathcal{RF}$	$\mathcal{RS}$	$\mathcal{RF}$	$\mathcal{RS}$
∞	*Cov*	**57.93**	**57.93**	64.37	59.38	78.27	72.37	87.54	85.02	90.26	88.37	93.26	89.38
	Conf	**80.29**	**80.29**	74.24	79.27	62.18	78.36	54.30	77.14	52.10	75.97	49.08	75.20
	Nb	122	122	201	165	458	196	628	209	1107	310	2256	529
20	*Cov*	58.94	58.29	67.27	61.98	80.28	75.38	89.22	86.85	92.81	90.04	96.36	92.10
	Conf	78.45	79.81	70.35	79.10	60.22	77.37	51.06	75.76	49.97	74.98	47.98	71.08
	Nb	179	156	349	201	601	250	803	296	1706	398	3125	618
10	*Cov*	63.18	60.01	71.38	63.87	85.28	81.21	94.67	90.02	96.38	92.17	98.37	93.27
	Conf	71.84	75.02	64.38	74.78	55.37	74.01	46.74	73.05	43.98	71.07	41.19	68.29
	Nb	259	189	678	301	998	402	1394	501	2106	631	4197	801
5	*Cov*	70.52	64.83	76.21	66.35	90.12	87.29	97.41	93.61	97.78	94.88	99.23	95.37
	Conf	65.93	72.98	63.27	71.87	52.18	71.04	40.58	70.17	38.29	68.38	37.38	66.30
	Nb	413	214	1101	378	2005	509	3951	689	7674	710	9174	798
2	*Cov*	86.72	74.41	82.17	71.34	92.14	89.95	99.46	95.48	98.21	96.03	99.95	97.38
	Conf	58.68	70.76	55.38	69.34	42.17	68.11	33.92	67.55	32.81	66.20	31.18	65.87
	Nb	893	381	2438	421	4387	601	6042	873	10987	920	14392	1021

Tableau 3.3 – Mushroom : performances des ensembles de règles

	Support_ min	10		8		6		4		3		2	
TC		$\mathcal{RF}$	$\mathcal{RS}$	$\mathcal{RF}$	$\mathcal{RS}$	$\mathcal{RF}$	$\mathcal{RS}$	$\mathcal{RF}$	$\mathcal{RS}$	$\mathcal{RF}$	$\mathcal{RS}$	$\mathcal{RF}$	$\mathcal{RS}$
∞	*Cov*	**74.98**	**74.98**	80.24	76.38	90.38	87.48	93.56	92.02	95.19	93.12	96.88	94.65
	Conf	**69.89**	**69.89**	61.89	69.01	57.38	68.10	46.96	67.12	44.89	65.89	43.06	64.87
	Nb	4371	4371	6569	4987	9861	5307	13573	6010	24287	6854	32453	7210
20	*Cov*	78.93	76.00	83.29	78.39	91.56	89.66	95.01	92.88	96.37	93.89	97.34	95.28
	Conf	63.65	66.69	57.29	65.96	53.17	65.26	41.45	64.96	39.32	63.83	38.30	62.80
	Nb	4995	4604	7687	5431	10586	5806	16943	6985	28956	7328	40167	7995
10	*Cov*	83.56	80.05	87.37	81.29	93.18	91.35	97.49	94.08	98.08	94.94	98.94	96.65
	Conf	58.92	60.61	50.54	60.00	46.89	59.27	37.78	58.93	35.38	56.93	34.10	55.90
	Nb	5991	5210	10864	5978	18586	6321	27156	7439	40163	8222	51095	8967
5	*Cov*	88.62	83.67	90.34	84.10	97.27	93.87	99.03	96.00	99.49	96.60	99.12	97.17
	Conf	50.73	56.87	44.67	56.05	40.56	55.38	35.93	54.73	33.19	53.08	31.98	51.87
	Nb	7831	5967	19750	6745	35278	7387	45118	8176	70671	8994	99675	9223
2	*Cov*	91.78	86.94	92.16	87.34	98.05	94.94	99.06	97.86	99.87	98.06	99.98	98.74
	Conf	44.47	52.69	40.99	51.78	37.28	51.01	32.88	49.79	30.27	47.19	29.00	46.44
	Nb	9032	6494	22786	7045	42187	8301	58632	9656	94897	10241	138965	11008

Tableau 3.4 – Gain en couverture et perte en confiance en moyenne sur les 13 jeux de données comparé aux règles prototypes

	Support_ min	10		8		6		4		3		2	
GR		$\mathcal{RF}$	$\mathcal{RS}$	$\mathcal{RF}$	$\mathcal{RS}$	$\mathcal{RF}$	$\mathcal{RS}$	$\mathcal{RF}$	$\mathcal{RS}$	$\mathcal{RF}$	$\mathcal{RS}$	$\mathcal{RF}$	$\mathcal{RS}$
	Gain Cov	**60.53**	**60.53**	+11.31	+3.93	+37.66	+28.95	+47.95	+43.86	+54.54	+46.60	+61.62	+50.06
∞	*Perte Conf*	**78.43**	**78.43**	-9.52	-1.16	-20.71	-2.13	-35.77	-3.52	-38.33	-4.86	-40.23	-6.33
	Gain Cov	+4.21	+1.91	+16.42	+7.45	+43.05	+35.45	+51.03	+46.19	+59.34	+49.38	+65.12	+53.17
20	*Perte Conf*	-3.72	-2.14	-14.73	-3.75	-24.99	-4.73	-39.47	-5.89	-42.09	-7.60	-43.58	-9.55
	Gain Cov	+12.47	+7.79	+25.05	+13.56	+48.27	+39.21	+58.63	+49.73	+64.17	+52.54	+68.80	+56.84
10	*Perte Conf*	-11.78	-7.79	-21.32	-8.59	-31.81	-9.52	-44.70	-10.29	-47.37	-12.31	-49.50	-13.96
	Gain Cov	+24.71	+17.52	+33.64	+22.59	+58.60	+48.21	+69.48	+58.78	+70.75	+61.64	+73.32	+63.93
5	*Perte Conf*	-19.00	-12.93	-27.42	-14.31	-38.00	-15.27	-51.35	-16.34	-53.93	-18.36	-55.56	-19.88
	Gain Cov	+38.48	+28.19	+43.49	+31.48	+64.56	+54.75	+75.14	+66.10	+75.46	+66.77	+77.32	+69.03
2	*Perte Conf*	-27.48	-19.25	-35.21	-20.48	-45.93	-21.73	-58.31	-22.71	-60.77	-24.70	-62.31	-25.99

La méthode de sélection proposée a pour but de sélectionner les règles qui sont les plus similaires aux règles prototypes afin d'améliorer la couverture sans pour autant dégrader la confiance obtenue avec les règles prototypes. Son principe est de déterminer les règles candidates les plus similaires aux règles prototypes (car celles-ci sont estimées fiables) permettant ainsi de couvrir plus d'objets avec une confiance similaire à celle des règles prototypes.

Comparaison entre les règles sélectionnées et les règles prototypes. Sur le jeu de données *Zoo* représenté au tableau 3.2, on peut observer globalement que les ensembles règles sélectionnées améliorent la couverture des règles prototypes avec une confiance assez comparable. Par exemple la couverture de l'ensemble des règles sélectionnées avec un support minimum égal à 4 et un taux de croissance infini passe de 57,93% pour l'ensemble règles prototypes à 85,02%. Dans le même temps, la confiance baisse seulement de 80,29% pour l'ensemble règles prototypes à 77,14% pour l'ensemble règles sélectionnées considéré. Cette perte de confiance et ce gain en couverture de l'ensemble règles sélectionnées s'accentuent si on baisse le taux de croissance minimum ou le support minimum. Comme exemple, si on considère l'ensemble des règles sélectionnées avec un support minimum égal à 3 et un taux de croissance minimum égal à 5, on couvre la quasi-totalité des objets avec une couverture à 94,88% mais la confiance chute fortement et équivaut à 68,38%.

Ces mêmes observations peuvent être faites sur le tableau 3.3 représentant le jeu de données *Mushroom*.

Ces mêmes tendances se confirment aussi en moyenne sur les 13 jeux de données utilisés. Sur le tableau 3.4, on observe le fait que l'ensemble des règles sélectionnées permet d'avoir un gain moyen en couverture très important comparé à l'ensemble règles prototypes avec une perte moyenne en confiance très faible. Par exemple pour l'ensemble des règles sélectionnées avec un support minimum égal à 4 et un taux de croissance infini, le gain moyen en couverture, sur les 13 jeux de données, est de 43,86% avec seulement une perte en confiance de 3,52%.

Comparaison entre les règles sélectionnées et les règles fréquentes. Une autre particularité intéressante de l'ensemble des règles sélectionnées est qu'il possède un gain en couverture, comparé à l'ensemble règles prototypes, presque équivalent à celui de l'ensemble des règles fréquentes correspondant. Les règles sélectionnées ont aussi un niveau de confiance assez comparable à celui de l'ensemble des règles prototypes. Toujours sur le jeu de données *Zoo* du tableau 3.2 représentant le jeu de données *Zoo*, avec la couverture de l'ensemble des règles prototypes à 57,93% et la confiance à 80,29%, la couverture de l'ensemble des règles sélectionnées avec un support minimum égal à 4 et un taux de

croissance infini est égal 85,02% avec une confiance de 77,14%. Tandis que la couverture l'ensemble des règles fréquentes aux mêmes seuils de support et de taux de croissance est de 87,54% mais avec une confiance de seulement 54,30%.

Ces mêmes observations peuvent être remarquées sur le tableau 3.3 représentant le jeu de données *Mushroom*.

La figure 3.4 synthétise cette comparaison entre les règles sélectionnées et les règles fréquentes en montrant, pour le jeu de données Mushroom, la couverture et la confiance, respectivement, des règles fréquentes et sélectionnées, associées aux JEPs de support 10, 8, 6, 4, 3 et 2. *cov($\mathcal{RF}$)* et *conf($\mathcal{RF}$)* désignent respectivement la couverture et la confiance des règles fréquentes. *cov($\mathcal{RS}$)* et *conf($\mathcal{RS}$)* représentent la couverture et la confiance des règles sélectionnées

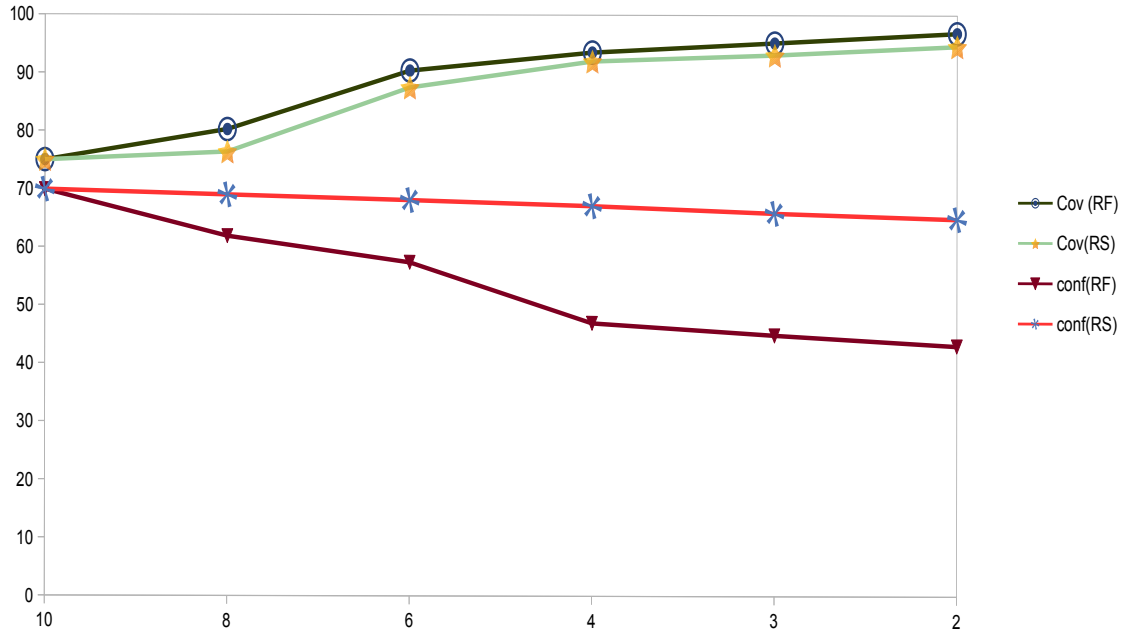

FIGURE 3.4 – Mushroom : couverture et confiance des règles **fréquentes** et **sélectionnées** associées aux JEPs

Sur la figure 3.4, on peut observer que plus le support est faible plus la couverture des règles fréquentes est élevée mais avec aussi une confiance qui chute fortement. On note aussi sur cette figure que les règles sélectionnées permettent d'atteindre sensiblement les mêmes couvertures avec une perte en confiance qui n'est pas très significatif. On observe aussi le même phénomène en étudiant l'évolution de la couverture et de la confiance en fonction du taux de croissance. Cela signifie que plus le taux de croissance est faible, plus l'ensemble des règles fréquentes couvre les objets mais avec une perte en confiance très importante. Alors que l'ensemble des règles sélectionnées atteint presque les mêmes

valeurs de couvertures sans pour autant beaucoup perdre en précision comparé aux règles prototypes. Cette assertion se vérifie sur le tableau 3.3.

En faisant une moyenne sur les 13 jeux de données et en comparaison avec l'ensemble règles prototypes, le tableau 3.4 montre aussi que le gain moyen en couverture de l'ensemble des règles sélectionnées est très proche de celui l'ensemble des règles fréquentes correspondant, avec une perte moyenne en confiance beaucoup moins importante. Par exemple, si on considère le tableau 3.4 et l'ensemble règles sélectionnées avec un support minimum égal à 4 et un taux de croissance infini, le gain moyen en couverture est de 43,86% et une perte moyenne en confiance de 3,52%. Alors que pour l'ensemble règles fréquentes avec un support minimum égal à 4 et un taux de croissance infini, le gain moyen en couverture est de 47,95% mais avec une grosse perte moyenne en confiance de 35,77%.

Proposition de classement des différents ensembles de règles. La figure 3.5 montre le compromis entre le gain en couverture et la perte en confiance des ensembles de règles sur les jeux de données *Zoo* et *Mushroom*. Seuls les ensembles de règles qui tolèrent une perte de confiance de 5% sont considérés. La colonne de couleur blanche montre le nombre de jeux de données où l'ensemble de règles correspondant tolère une perte de 5% de la confiance. La colonne de couleur grise montre le gain en couverture en moyenne sur les 13 jeux de données et la colonne en noir la perte en confiance moyenne. Pour chaque ensemble de règles, la notation utilisée dans cette figure est la suivante : a_b_c : a désigne le taux de croissance, b le type de règle à savoir les règles prototypes, fréquentes ou sélectionnées. Et en dernier c représente le support minimum fixé. La notation $\mathcal{RS}$ désigne les règles sélectionnées et $\mathcal{RF}$ les règles fréquentes. Par exemple sur le jeu de données *Zoo*, l'ensemble des règles fréquentes avec un taux de croissance infini et un support minimum de 4 ($inf\_\mathcal{RS}\_4$) permet d'avoir un gain en couverture de 46,76% avec seulement une perte en confiance de 3,92%. Sur le jeu de données *Mushroom*, la perte en confiance pour $inf\_\mathcal{RS}\_4$ est de 3,96% avec un gain en couverture de 22,73%.

La figure 3.6 montre les ensembles de règles qui, en moyenne sur les 13 jeux de données utilisés, tolèrent une perte de 5% de confiance comparé aux prototypes . Dans 10 jeux de données sur 13, il n'y a que les règles sélectionnées qui vérifient cette contrainte : ce qui montre que les règles sélectionnées possèdent de bonnes caractéristiques si un bon compromis entre la couverture et la confiance est recherchée. Il est aussi intéressant de noter que pour les ensembles règles fréquentes, seuls ceux avec un taux de croissance minimum de 20 apparaissent sur cette figure. Cela est dû au fait qu'ils sont très proches des règles prototypes en terme de couverture et de confiance.

Les règles sélectionnées avec un taux de croissance ∞ et un support minimum de 4 présente un réel intérêt parce qu'ils apparaissent dans presque tous les jeux de données et

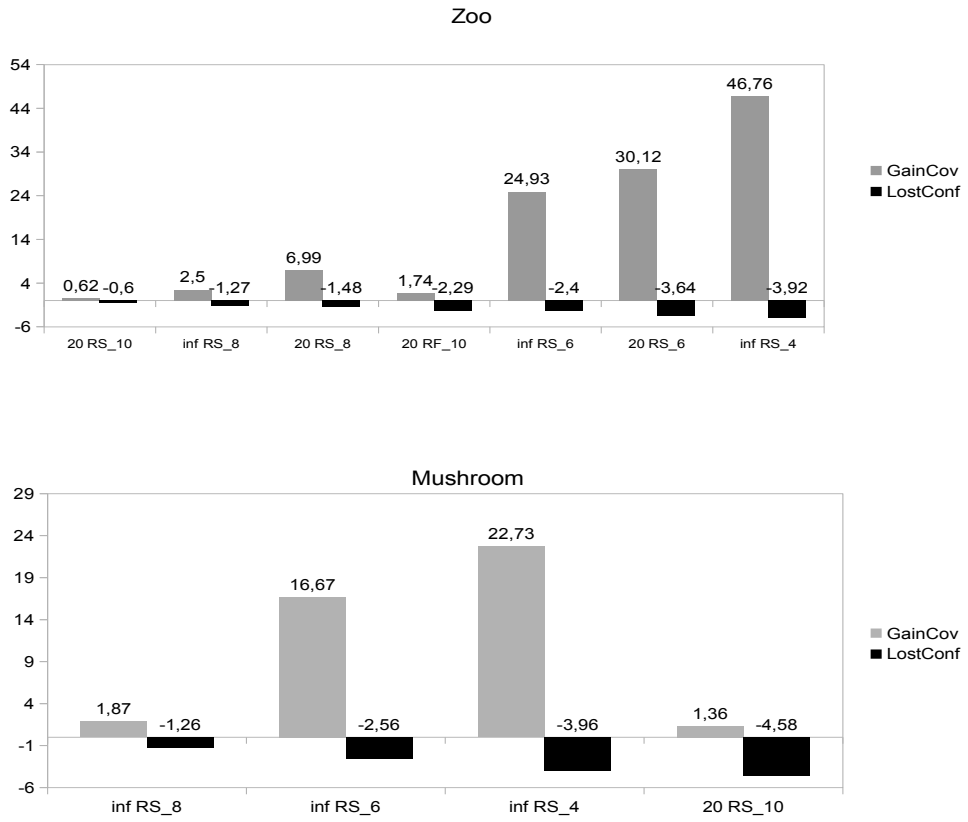

FIGURE 3.5 – Compromis entre le gain en couverture et la perte en confiance de Zoo et Mushroom comparé aux règles prototypes

ont le gain en couverture le plus important (43,9%) avec seulement une perte en confiance de 3,5% comparé aux règles prototypes.

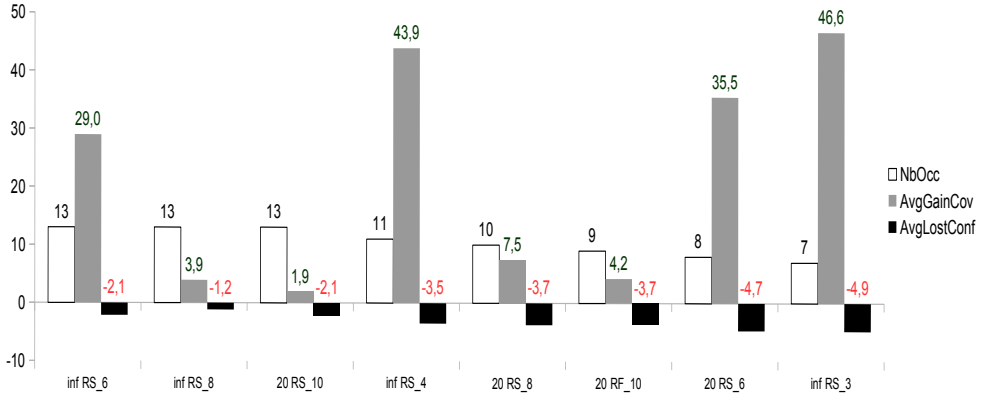

FIGURE 3.6 – Compromis entre le gain en couverture et la perte en confiance comparé aux règles prototypes sur les 13 jeux de données

3.2.2.3 Analyse des règles fréquentes

A la différence des règles prototypes, les règles fréquentes couvrent une très grande partie des objets particulièrement quand le taux de croissance minimum ou le support minimum est faible. Cependant leur confiance est significativement plus basse que celle des règles prototypes. Par exemple, en considérant les résultats expérimentaux sur le jeu de données *zoo*, représentés sur le tableau 3.2, et l'ensemble des règles fréquentes avec un support minimum égal à 4 et un taux de croissance infini, la couverture passe de 57,93% pour les règles prototypes à 87.54% pour les règles fréquentes. Dans le même temps, la confiance chute de 80.29% à 54.30%. On observe que ce gain de couverture et cette baisse de confiance s'amplifie si le support ou le taux de croissance devient plus faible. Par exemple, la couverture de l'ensemble des règles fréquentes avec un support minimum égal à 3 et un taux de croissance infini est de 90,26% mais leur confiance chute encore plus pour atteindre 52,10%. Si on considère l'ensemble des règles fréquentes avec un support minimum égal à 4 et un taux de croissance minimum égal à 20, la couverture augmente pour atteindre 89,22% mais la confiance chute à 51,06%.

Ces mêmes tendances peuvent aussi être observées sur le tableau 3.3 représentant le jeu de données *mushroom* et sont vérifiées sur les 13 jeux de données employés dans cette partie expérimentale.

En moyenne sur les 13 jeux de données, on a quantifié le gain moyen en couverture et la perte moyenne en confiance des ensembles règles fréquentes sur le tableau 3.4. On peut voir sur ce tableau 3.4 que le gain moyen en couverture, comparé aux règles prototypes, de l'ensemble règles fréquentes avec un support minimum égal à 4 et un taux de croissance infini est de 47.95% avec une perte moyenne en confiance de 35.77%. Ce gain en couverture s'accroît si le support minimum passe de 4 à 3 pour égaler 54,54%. Cependant la perte en confiance est plus importante et équivaut à 38,33%. Si c'est le taux de croissance minimum qui baisse à 20, alors la couverture augmente aussi de 51,03% avec une chute de la confiance de 39,47%.

3.2.3 Utilisation d'un classifieur à base de règles : CPAR

En deuxième partie de l'expérimentation, nous évaluons la méthode de sélection si les règles sont utilisées dans un contexte de classification à base de règles. Pour cela nous avons utilisé une méthode classique de classification à base de règles : CPAR [YH03]. Ce classifieur a été choisi parce qu'il a une meilleure précision comparée aux autres méthodes de classification à base de règles et avec une plus grande efficacité concernant les temps d'exécution [YH03]. La règle par défaut utilisée ici correspond à la classe majoritaire. Une validation croisée à 5 volets est effectuée pour obtenir les valeurs de couverture et

de précision. La ligne *Cov* correspond au pourcentage d'objets pour lesquels la règle par défaut n'est pas appliquée. La couverture et la précision des différents ensembles de règles sont montrées sur les tableaux 3.6 et 3.7. Les colonnes *PerfSelected* et *PerfCovered* donne la couverture précision que sur les objets qui sont couverts par les règles. La colonne *PerfSelected* désigne la couverture et la précision quand seules les règles validées (section 3.2.1.3) sont utilisées. La colonne *PerfCovered* correspond à la couverture et à la précision quand seules les règles sélectionnées (section 3.2.1.3) sont employées. Le tableau 3.5 montre le gain ou la perte moyenne en précision ou en couverture comparé aux règles prototypes en moyenne sur les 13 jeux de données. Les valeurs en gras représente les valeurs de couverture et de confiance pour les règles prototypes. La colonne *Gain Cov* montre le gain en couverture et la colonne *(Gain|Perte) Conf* le gain ou la perte en confiance. La formule du gain ou de la perte en couverture est le même que celle définie avec la formule 3.2. La perte ou le gain en précision est calculé comme suit :

$$Perte\_ou\_Gain\_Precision(ensemble\_regles) = \frac{Precision(regles\_prototypes) - Precision(ensemble\_regles)}{Precision(regles\_prototypes)} \tag{3.3}$$

3.2.3.1 Fonctionnement de CPAR(Classification based on Predictive Association Rules) :

Nous avons utilisé le principe de classification de CPAR afin de classer les objets. Pour cela, la précision de chaque règle est estimée selon la mesure de *Laplace Accuracy* [CB91] correspondant à la formule 3.4.

$$LaplaceAccuracy = (n_c + 1)/(n_{tot} + k) \tag{3.4}$$

avec k étant le nombre de classes, n_{tot} le nombre total d'exemples satisfaisant la prémisse de la règle, et n_c le nombre d'objets parmi n_{tot} ayant la même conclusion que la règle.

Supposons qu'on dispose des règles pour chaque classe. Pour classer un objet, CPAR procède comme suit :

- sélectionne l'ensemble des règles dont la prémisse est satisfaite par l'objet,
- choisit les k meilleures règles pour chaque classe. Le choix des meilleures règles se fait grâce à la mesure du *Laplace Accuracy*. k est généralement fixé à 5 (que nous avons aussi choisi pour notre cas).
- une moyenne, avec la mesure du *Laplace Accuracy*, est calculée sur les k meilleures règles de chaque classe. L'objet est classé dans la classe avec la plus grande moyenne.

Tableau 3.5 – Perte et gain sur la précision des règles prototypes avec le classifieur CPAR et en moyenne sur les 13 jeux de données

	Support_ min	10		8		6		4		3		2	
TC		$\mathcal{RF}$	$\mathcal{RS}$	$\mathcal{RF}$	$\mathcal{RS}$	$\mathcal{RF}$	$\mathcal{RS}$	$\mathcal{RF}$	$\mathcal{RS}$	$\mathcal{RF}$	$\mathcal{RS}$	$\mathcal{RF}$	$\mathcal{RS}$
	Gain Cov	**60.53**	**60.53**	+11.31	+3.93	+37.66	+28.95	+47.95	+43.86	+54.54	+46.71	+61.73	+50.28
∞	*(Gain|Perte) Conf*	**83.46**	**83.46**	-1.30	+0.69	-2.84	+1.86	-4.28	+2.91	-8.26	-0.90	-11.19	-2.94
	Gain Cov	+4.21	+1.91	+16.42	+7.45	+43.05	+35.45	+51.03	+46.19	+59.34	+49.39	+65.18	+53.30
20	*(Gain|Perte) Conf*	-3.91	-2.07	-4.93	-1.30	-6.38	-0.45	-7.77	+0.37	-10.80	-3.10	-14.16	-5.64
	Gain Cov	+12.47	+7.79	+25.05	+13.56	+48.27	+39.21	+58.63	+49.84	+64.17	+52.54	+68.80	+56.95
10	*(Gain|Perte) Conf*	-9.36	-6.01	-10.60	-5.25	-12.12	-4.20	-13.49	-3.19	-16.53	-6.83	-18.97	-9.07
	Gain Cov	+24.71	+17.52	+33.64	+22.59	+58.60	+48.21	+69.48	+59.11	+70.75	+61.53	+73.32	+64.15
5	*(Gain|Perte) Conf*	-14.18	-8.73	-16.19	-8.35	-18.56	-7.47	-19.83	-6.71	-22.68	-9.83	-24.74	-12.19
	Gain Cov	+38.48	+28.19	+43.49	+31.48	+64.56	+54.75	+75.25	+66.53	+75.46	+66.77	+77.32	+69.25
2	*(Gain|Perte) Conf*	-19.82	-12.11	-21.64	-11.72	-24.22	-10.81	-26.08	-10.15	-28.24	-13.47	-30.92	-15.74

3.2.3.2 Règles prototypes

Les règles prototypes ne couvrent pas une bonne partie des objets, ce qui entraîne utilisation fréquente de la règle par défaut dans un contexte de classification. Ce qui déprécie au final la qualité du classifieur. Sur l'exemple du tableau 3.6, l'ensemble des règles prototypes couvre 57,93% des objets et possède une précision de 92,10% sur ces objets couverts. Mais à cause des 42,03% des objets où la règle par défaut est appliquée, la précision globale du classifieur chute à 86,91%.

Le tableau 3.5 montre les mêmes observations sur l'ensemble des 13 jeux de données.

3.2.3.3 Règles fréquentes

Ces règles étant très nombreuses avec un seil de fréquence ou de taux de croissance faible, elles permettent de moins utiliser la règle par défaut avec une augmentation du pourcentage d'objets couverts mais elles n'améliorent pas la précision du classifieur. Par

exemple le tableau 3.6 indique que pour les règles fréquentes associées aux JEPs de support minimum 4, la couverture varie de 57,93% à 87,54%. Cependant la précision chute de 86,91% à 83,91%. On peut remarquer que plus le support est faible, plus les règles fréquentes permettent de moins utiliser la règle par défaut mais la précision du classifieur n'en est pas pour autant améliorée. Cela s'explique par le fait que plus le support est faible plus les règles sélectionnées ont une confiance moins fiables comparées aux prototypes et tendront quasiment vers la même confiance que la règle par défaut.

Le tableau 3.5 montre les mêmes observations sur l'ensemble des 13 jeux de données.

3.2.3.4 Règles sélectionnées

En plus d'avoir quasiment la même couverture que les règles fréquentes à taux de croissance et support équivalent, les règles sélectionnées améliorent la précision du classifieur. L'explication est que ces règles ont généralement une bonne confiance, similaire à celle des règles prototypes. On peut d'ailleurs remarquer sur le tableau 3.6 que si on utilise seulement les règles validées (*PerfSelected*), la précision sur les objets qu'elles couvrent est sensiblement égale à celle des règles sélectionnées (*PerfCovered*). Donc si la règle par défaut est moins appliquée grâce aux règles sélectionnées, la précision du classifieur est améliorée car ces règles ont une confiance généralement bien meilleure que la règle par défaut. Par exemple sur le tableau 3.6, on observe que les règles sélectionnées associées aux JEPs de support minimum 4, permettent d'améliorer la précision du classifieur passe de 86,91% pour les prototypes à 88,67%. On peut expliquer cela par le fait que cet ensemble de règles permet, premièrement, de moins utiliser la règle par défaut avec une couverture qui varie de 57,93% à 85,02. Deuxièmement, la précision de cet ensemble de règles est très bonne sur les objets couverts, soit 90,76% ; ce qui au final fait que la précision du classifieur est améliorée.

3.2.3.5 Classement des ensembles de règles

La figure 3.7 montre tous les ensembles de règles qui permettent d'obtenir un gain sur la précision comparée à celle des règles prototypes.

La colonne de couleur blanche désigne le nombre de jeux de données où ce gain a été obtenu avec l'ensemble de règles correspondant. La colonne de couleur grise indique le gain en couverture et la colonne de couleur noire la perte en précision. Pour les ensembles de règles, nous utilisons la notation *a_b_c* : *a* désigne le taux de croissance, *b* le type de règle à savoir les règles prototypes, fréquentes ou sélectionnées, *c* représente le support minimum fixé. Par exemple, l'ensemble des règles sélectionnées avec un taux de croissance égal à ∞ et un support minimum de 4 ($inf\_\mathcal{RS}\_4$) augmente en moyenne la précision à

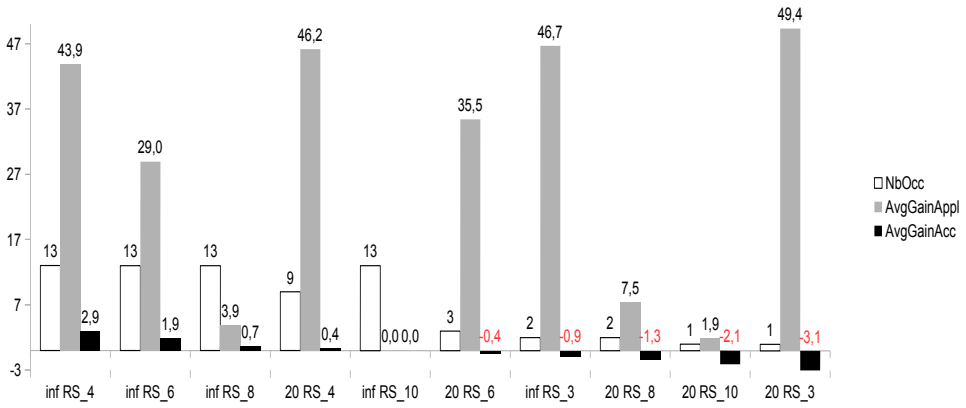

FIGURE 3.7 – Comparaison de la précision et de la couverture des règles sélectionnées avec les règles prototypes

2.9% comparé aux prototypes et avec 43.9% en plus d'objets qui sont couverts par des règles sur lesquels la règle par défaut ne s'applique pas.

Dans ce contexte de classification, l'amélioration de la précision n'est obtenue qu'avec les règles sélectionnées. En particulier, les règles sélectionnées avec un taux de croissance égal à ∞ et un support minimum de 4 conduisent à une amélioration sur tous les jeux de données avec en moyenne 43,9% des objets en plus où la règle par défaut n'est pas appliquée et 2,9% de gain en précision. Ce gain en précision s'explique par le fait qu'il y a 43,9% d'objets en moins pour lesquels la règle par défaut du classifieur n'est plus appliquée.

3.3 Sélection de règles prototypes en plusieurs passes

3.3.1 Principe de la méthode

Quelles que soient les règles candidates à la sélection, la méthode de sélection proposée à la section 3.1.2.1 utilise un seul ensemble de règles fiables appelé des règles prototypes. Nous appelons cette technique la méthode de sélection *simple*.

Nous présentons dans cette section, une extension de cette méthode de sélection. L'idée est d'effectuer plusieurs passes pour sélectionner les règles. Pour une première passe, cette méthode choisit les règles suffisamment supportées et donc considérées comme fiables. Ces règles représentent les règles prototypes de départ. Ensuite à chaque passe, les règles prototypes sont composées des règles prototypes de la passe précédente auxquelles sont ajoutées les règles sélectionnées à la passe précédente.

La figure 3.8 détaille ce processus de sélection en plusieurs passes en l'illustrant avec

l'utilisation du taux de croissance et du support. Les passes sont effectuées en faisant varier le support ou le taux de croissance pour les ensembles de règles considérés. Considérons l'étape noté A sur cette figure. Les règles candidates à la sélection sont désignées par $\mathcal{RC}_{1\_3}$ qui correspond aux règles candidates associées aux motifs émergents avec la première valeur pour le taux de croissance et la troisième valeur de support. Pour sélectionner des règles prototypes parmi l'ensemble de règles candidates $\mathcal{RC}_{1\_3}$, les règles prototypes ne seront plus seulement $\mathcal{RP}_{1\_1}$, mais les règles sélectionnées avec la première valeur du taux de croissance et la deuxième valeur de support.

L'étape noté B sur la figure 3.8 (dernière colonne) montre le cas où les règles candidates sont celles associées aux motifs émergents avec les dernières valeurs de taux de croissance et de support. Pour appliquer la sélection sur ces règles candidates $\mathcal{RC}_{m\_n}$, l'ensemble des règles prototypes correspond à l'union des règles sélectionnées avec l'avant dernière valeur du taux de croissance et celles sélectionnées avec l'avant dernière valeur du support.

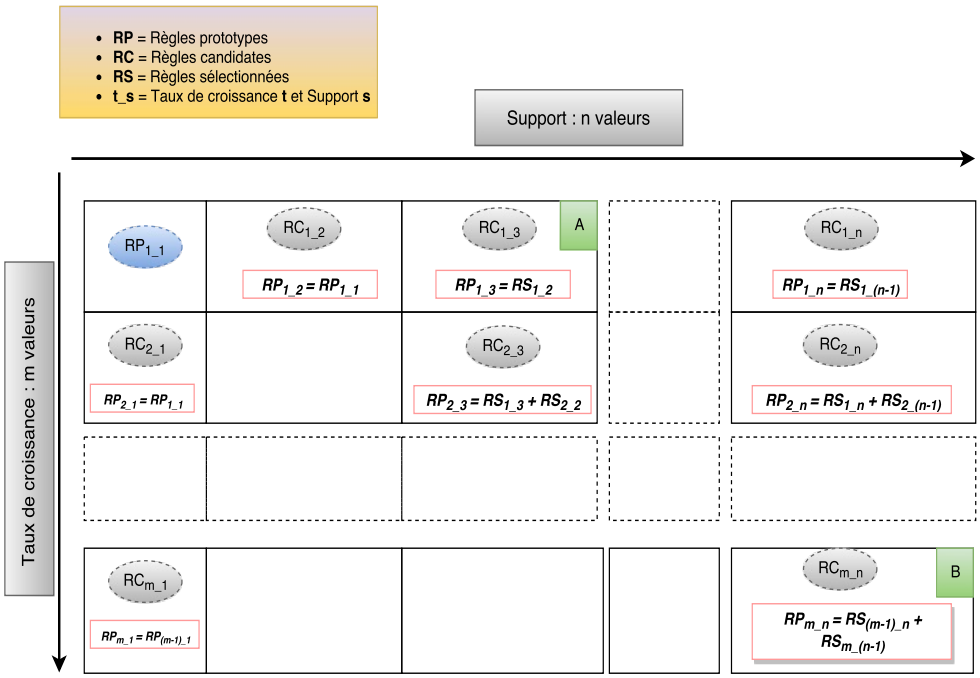

FIGURE 3.8 – Principe de la méthode de sélection en plusieurs passes

3.3.2 Matériels et méthodes

Les jeux de données utilisés sont les mêmes que ceux de la sous-section 3.2.1. Nous avons choisi les règles prototypes comme étant les règles associées aux JEP de support minimum 10. A chaque passe, nous avons défini les ensembles de règles candidates en

fixant successivement un taux de croissance minimum de infini, 20, 10, 5 et 2, et le support minimum de 8 à 6, 4, 3 et 2.

3.3.3 Analyse des résultats expérimentaux

Les tableaux 3.8 et 3.9 montrent la couverture et la confiance des ensembles de règles sélectionnées sans et avec la méthode de sélection à plusieurs passes sur les jeux de données *Zoo* et *Mushromm*. La colonne $\mathcal{RS}$ représente les règles sélectionnées avec la méthode de sélection simple et la colonne $\mathcal{RS}$ *(PP)* celles sélectionnées avec la méthode en plusieurs passes.

Pour un même ensemble de règle, la méthode de sélection en plusieurs passes permet de gagner en couverture avec une faible perte en confiance comparée à la méthode sélection simple.

Par exemple, pour le jeu de données em Zoo représenté sur le tableau 3.8, en moyenne sur tous les ensembles de règles du jeu de données, le gain en couverture est de 1,9% et la perte de confiance de 3,1% pour la méthode sélection à plusieurs passes comparée à la méthode de sélection simple. Si on considère les résultats sur les 13 jeux de données, la méthode sélection à plusieurs passes permet en moyenne de gagner en couverture 2,15% avec seulement une perte en confiance de 2,34%, comparée à la méthode de sélection simple. Ainsi, avec la méthode de sélection à plusieurs passes, on arrive encore un peu plus à améliorer la couverture avec une faible perte en confiance comparé à la méthode de sélection simple, mais avec un temps de calcul plus important.

3.4 Conclusion

Dans ce chapitre nous avons défini une méthode de sélection de règles qui permet d'améliorer la couverture des règles prototypes sans pour autant dégrader leur précision. Pour la sélection des règles, nous avons proposé une approche se basant sur un LOO (Leave One Out) et une méthode des plus proches voisins. Dans un contexte de classification la méthode de sélection conduit à une amélioration de la précision du classifieur pour 4 ensembles de règles ainsi que de la proportion d'objets couverts. Notre méthode de sélection permet donc de moins utiliser la règle par défaut et de pouvoir ainsi classifier un plus grand nombre d'objets. Ainsi, on peut donner une explication à la prédiction d'un nombre important d'objets qui étaient classifiés en utilisant la règle par défaut. Les résultats expérimentaux ont montré un cas intéressant que sont les règles associées aux JEPs avec un support minimum de 4 qui améliorent, en moyenne sur les 13 jeux de données utilisés, la couverture de 43,9% avec une perte en confiance de 3,5% comparé aux

règles prototypes. Et en utilisant le classifieur CPAR, cet ensemble de règles améliore la précision du classifieur de 2,9% comparé aux règles prototypes.

Compte tenu de ces résultats expérimentaux encourageants, nous appliquons dans le chapitre 4, la méthode de sélection sur un jeu de données chimique. Nous voulons ainsi savoir si les mêmes performances sont obtenues dans ce cas d'application. Cela permettrait aux chimistes dans un contexte de classification, de mieux expliquer la classification de plusieurs molécules, qui sans la méthode de sélection serait prédites uniquement grâce à la règle par défaut du classifieur.

Tableau 3.6 – Zoo : Performances avec CPAR

		Support_ min	10		8		6		4		3		2	
TC			$\mathcal{RF}$	$\mathcal{RS}$	$\mathcal{RF}$	$\mathcal{RS}$	$\mathcal{RF}$	$\mathcal{RS}$	$\mathcal{RF}$	$\mathcal{RS}$	$\mathcal{RF}$	$\mathcal{RS}$	$\mathcal{RF}$	$\mathcal{RS}$
∞	*Accuracy*	*Conf*	86.91	86.91	86.31	87.02	84.83	87.99	83.91	88.67	79.34	84.87	77.18	83.19
	PerfCovered	*Cov*	57.93	57.93	64.37	59.38	78.27	72.37	87.54	85.02	90.26	88.37	93.26	89.38
		Conf	92.10	92.10	90.34	91.97	88.78	91.06	87.78	90.76	83.25	87.37	80.36	85.77
	PerfSelected	*Cov*	57.93	57.93	20.29	24.19	30.78	41.89	38.67	45.70	41.56	50.15	45.38	54.91
		Conf	92.10	92.10	90.17	91.78	87.10	89.76	85.66	88.97	80.25	86.37	78.19	83.23
20	*Accuracy*	*Conf*	84.91	85.30	84.01	85.61	81.26	85.84	80.04	86.06	77.37	83.97	75.28	80.76
	PerfCovered	*Cov*	58.94	58.29	67.27	61.98	80.28	75.38	89.22	86.85	92.81	90.04	96.36	92.10
		Conf	90.87	91.71	88.93	91.06	87.11	89.35	85.88	88.56	81.23	85.10	77.05	83.44
	PerSelected	*Cov*	11.67	16.89	24.19	28.84	35.98	47.35	48.76	55.89	43.27	54.20	49.99	59.28
		Conf	87.51	88.67	86.38	88.04	82.56	87.62	81.67	86.95	79.03	84.33	74.32	81.89
10	*Accuracy*	*Conf*	79.22	80.97	78.91	81.32	77.34	82.85	76.29	83.19	74.39	79.37	70.65	76.43
	PerfCovered	*Cov*	63.18	60.01	71.38	63.87	85.28	81.21	94.67	90.02	96.38	92.17	98.37	93.27
		Conf	85.78	88.55	84.39	87.67	81.86	87.01	80.51	85.91	76.26	81.56	72.10	78.19
	PerfSelected	*Cov*	15.60	21.85	29.19	33.20	47.92	53.39	51.75	58.20	48.38	59.26	54.33	63.90
		Conf	83.81	85.19	82.97	84.77	79.36	82.99	77.19	82.07	74.00	79.90	69.88	77.04
5	*Accuracy*	*Conf*	77.10	79.55	75.39	80.12	72.17	80.56	70.67	80.96	70.35	78.56	68.89	75.18
	PerfCovered	*Cov*	70.52	64.83	76.21	66.35	90.12	87.29	97.41	93.61	97.78	94.88	99.23	95.37
		Conf	81.59	85.93	80.85	85.01	76.18	84.10	75.29	83.47	72.10	80.23	70.11	76.78
	PerfSelected	*Cov*	20.41	26.78	35.19	40.67	52.19	54.97	56.78	63.96	50.33	64.18	56.29	67.98
		Conf	79.56	84.01	78.35	83.78	73.29	82.17	72.19	81.78	69.38	79.31	67.45	75.44
2	*Accuracy*	*Conf*	75.04	78.95	73.28	79.15	70.41	79.98	69.01	80.97	68.36	77.27	64.27	73.19
	PerfCovered	*Cov*	86.72	74.41	82.17	71.34	92.14	89.95	99.46	95.48	98.21	96.03	99.95	97.38
		Conf	77.39	83.89	76.74	83.02	73.06	82.93	72.03	82.11	69.35	78.10	65.00	74.95
	PerfSelected	*Cov*	25.89	32.81	42.17	47.28	64.25	71.38	67.16	72.81	58.29	72.19	60.18	77.74
		Conf	75.74	81.75	74.19	81.65	71.67	81.24	70.78	81.05	67.08	77.00	62.89	73.44

Tableau 3.7 – Mushroom : Performances avec CPAR

		Support_ min	10		8		6		4		3		2	
TC			$\mathcal{RF}$	$\mathcal{RS}$	$\mathcal{RF}$	$\mathcal{RS}$	$\mathcal{RF}$	$\mathcal{RS}$	$\mathcal{RF}$	$\mathcal{RS}$	$\mathcal{RF}$	$\mathcal{RS}$	$\mathcal{RF}$	$\mathcal{RS}$
∞	*Accuracy*	*Conf*	82.33	82.33	82.00	82.56	80.44	82.94	79.80	83.12	77.08	80.89	75.18	79.27
	PerfCovered	*Cov*	74.98	74.98	80.24	76.38	90.38	87.48	93.56	92.02	95.19	93.12	96.88	94.65
		Conf	87.95	87.95	86.34	87.08	82.88	85.84	81.89	84.39	79.28	82.89	76.99	81.86
	PerfSelected	*Cov*	74.98	74.98	20.17	23.17	26.98	33.78	28.76	36.91	35.25	47.35	38.19	55.29
		Conf	87.95	87.95	85.39	86.30	81.99	85.08	80.71	83.91	76.21	80.94	73.02	79.38
20	*Accuracy*	*Conf*	77.41	79.58	76.81	80.12	74.10	80.89	73.06	81.16	72.98	78.28	69.21	76.19
	PerfCovered	*Cov*	78.93	76.00	83.29	78.39	91.56	89.66	95.01	92.88	96.37	93.89	97.34	95.28
		Conf	81.72	83.61	80.42	83.12	77.11	82.99	75.95	82.78	74.38	79.36	71.12	77.96
	PerSelected	*Cov*	13.78	20.65	23.19	27.28	28.37	32.22	29.70	34.81	38.35	50.34	42.55	59.26
		Conf	79.75	82.67	78.76	82.41	75.10	82.11	73.89	81.93	70.36	78.07	69.12	76.97
10	*Accuracy*	*Conf*	72.39	74.62	72.03	75.16	70.85	76.79	69.29	77.83	67.36	74.01	64.21	72.88
	PerfCovered	*Cov*	83.56	80.05	87.37	81.29	93.18	91.35	97.49	94.08	98.08	94.94	98.94	96.65
		Conf	77.44	81.06	75.38	80.92	72.56	79.89	70.61	79.06	68.93	76.21	65.89	74.81
	PerfSelected	*Cov*	17.44	25.92	28.31	31.78	33.98	35.76	33.71	40.01	41.25	54.21	45.10	64.21
		Conf	75.99	80.61	72.87	79.82	70.11	79.00	68.95	78.19	65.00	74.28	63.02	73.44
5	*Accuracy*	*Conf*	70.17	72.89	68.45	73.01	65.10	73.42	63.78	73.83	62.00	71.98	60.01	69.55
	PerfCovered	*Cov*	88.62	83.67	90.34	84.10	97.27	93.87	99.03	96.00	99.49	96.60	99.12	97.17
		Conf	72.01	78.93	70.56	78.11	66.14	77.32	64.29	76.70	63.99	73.42	61.87	71.42
	PerfSelected	*Cov*	21.67	33.89	32.19	34.78	38.34	41.56	38.73	45.78	44.32	57.38	48.22	67.29
		Conf	70.97	76.06	68.25	75.55	65.13	75.05	63.98	74.80	60.44	72.10	59.38	69.98
2	*Accuracy*	*Conf*	65.73	71.81	64.21	72.22	62.07	72.84	60.15	73.26	57.25	70.97	54.26	68.47
	PerfCovered	*Cov*	91.78	86.94	92.16	87.34	98.05	94.94	99.06	97.86	99.87	98.06	99.98	98.74
		Conf	67.83	76.54	65.06	76.01	62.78	75.65	61.56	74.94	58.22	71.78	55.87	69.56
	PerfSelected	*Cov*	25.89	32.81	34.96	39.45	40.15	46.36	40.93	50.64	47.38	61.87	52.10	70.55
		Conf	65.93	74.98	62.19	74.50	61.45	74.04	60.58	73.87	54.23	70.09	53.26	68.44

Tableau 3.8 – Zoo : Performances des règles sélectionnées en plusieurs passes

	Support_ min	10		8		6		4		3		2	
GR		$\mathcal{RS}$	$\mathcal{RS}$(PP)	$\mathcal{RS}$	$\mathcal{RS}$(PP)	$\mathcal{RS}$	$\mathcal{RS}$(PP)	$\mathcal{RS}$	$\mathcal{RS}$(PP)	$\mathcal{RS}$	$\mathcal{RS}$(PP)	$\mathcal{RS}$	$\mathcal{RS}$(PP)
∞	*Cov*	57,93	57,93	59,38	59,38	72,37	74,06	85,02	86,71	88,37	90,04	89,38	90,76
	Conf	80,29	80,29	79,27	79,27	78,36	74,56	77,14	73,41	75,97	72,41	75,20	72,08
20	*Cov*	58,29	58,29	61,98	63,38	75,38	76,57	86,85	88,76	90,04	91,19	92,10	93,35
	Conf	79,81	79,81	79,10	76,64	77,37	73,84	75,76	72,14	74,98	72,43	71,08	68,39
10	*Cov*	60,01	61,27	63,87	65,78	81,21	82,43	90,02	91,41	92,17	93,54	93,27	95,99
	Conf	75,02	72,47	74,78	71,39	74,01	71,66	73,05	70,30	71,07	67,71	68,29	64,43
5	*Cov*	64,83	66,19	66,35	67,46	87,29	88,32	93,61	95,32	94,88	96,24	95,37	97,96
	Conf	72,98	69,66	71,87	68,86	71,04	67,36	70,17	67,31	68,38	66,32	66,30	63,35
2	*Cov*	74,41	76,16	71,34	72,82	89,95	91,29	95,48	97,34	96,03	97,91	97,38	98,91
	Conf	70,76	68,47	69,34	66,93	68,11	65,90	67,55	64,47	66,20	62,55	65,87	61,39

Tableau 3.9 – Mushroom : Performances des règles sélectionnées en plusieurs passes

	Support_ min	10		8		6		4		3		2	
GR		$\mathcal{RS}$	$\mathcal{RS}$(PP)	$\mathcal{RS}$	$\mathcal{RS}$(PP)	$\mathcal{RS}$	$\mathcal{RS}$(PP)	$\mathcal{RS}$	$\mathcal{RS}$(PP)	$\mathcal{RS}$	$\mathcal{RS}$(PP)	$\mathcal{RS}$	$\mathcal{RS}$(PP)
∞	*Cov*	74,98	74,98	76,38	76,38	87,48	89,47	92,02	93,34	93,12	94,22	94,65	96,26
	Conf	69,89	69,89	69,01	69,01	68,10	65,18	67,12	64,85	65,89	63,57	64,87	62,28
20	*Cov*	76,00	76,00	78,39	80,34	89,66	91,02	92,88	94,63	93,89	95,15	95,28	96,48
	Conf	66,69	66,69	65,96	63,01	65,26	62,89	64,96	62,37	63,83	60,99	62,80	59,94
10	*Cov*	80,05	81,41	81,29	83,12	91,35	92,38	94,08	95,74	94,94	96,59	96,65	98,07
	Conf	60,61	57,85	60,00	57,50	59,27	57,15	58,93	56,82	56,93	54,03	55,90	53,06
5	*Cov*	83,67	84,78	84,10	85,11	93,87	95,28	96,00	97,47	96,60	97,71	97,17	98,17
	Conf	56,87	54,44	56,05	54,56	55,38	53,26	54,73	51,80	53,08	50,45	51,87	49,77
2	*Cov*	86,94	88,81	87,34	89,28	94,94	96,42	97,86	99,19	98,06	99,58	98,74	100,00
	Conf	52,69	49,98	51,78	48,80	51,01	49,99	49,79	46,93	47,19	45,18	46,44	44,54

Chapitre 4

Détection de la toxicité aïgue avec la méthode de sélection de règles supervisées

Sommaire

Dans la section 3.2 du chapitre 3, nous avons montré expérimentalement que notre méthode de sélection de règles permet d'améliorer la couverture des règles avec un support élevé sans pour autant dégrader la confiance. En chémoinformatique, cette méthode de sélection apporte une contribution importante. En effet, pour classer des molécules, toxiques ou non toxiques par exemple, on peut ne pas avoir d'explication sur la raison pour laquelle une molécule est classée comme toxique ou pas, car c'est la règle par défaut du classifieur qui est utilisée. En augmentant la couverture grâce à notre méthode de sélection, la règle par défaut serait moins utilisée et on pourrait donner une raison à la classification des molécules. Comme les règles que nous sélectionnons ont une confiance assez similaire aux règles prototypes, la précision du classifieur ne serait alors pas dégradée. La sélection de règles permet aussi de découvrir de nouvelles molécules qui sont importantes pour classer une molécule dont la classe doit être prédite.

Dans ce chapitre, après avoir expliqué le contexte du travail, nous donnons les détails sur le jeu de données Aquatox qui est utilisé pour l'application de la méthode de sélection. Ensuite, nous définissons les ensembles de règles prototypes, candidates et sélectionnées qui sont utilisés, ainsi que les paramètres de la méthode de sélection. Nous terminons par une évaluation expérimentale de la méthode de sélection appliquée au jeu de données Aquatox en terme de couverture et de confiance mais aussi en terme de classification. Nous terminons ce chapitre par une évaluation qualitative de la méthode de sélection avec pour objectif de découvrir quelles sont les molécules intéressantes qui sont découvertes ou reconnues dans la littérature.

4.1 Contexte

La prévention des risques chimiques passe par une connaissance suffisante des propriétés et des utilisations des substances ainsi que par une approche cohérente et systématique d'évaluation de leurs risques. C'est principalement pour répondre à cet objectif que la Commission Européenne a proposé en octobre 2003 un nouveau dispositif réglementaire, baptisé REACH (Registration, Evaluation, Autorisation and restriction of CHemicals) (Figure 4.1), qui introduit un système global de contrôle des substances chimiques. Ce règlement met en œuvre plusieurs procédures : l'enregistrement, l'évaluation et l'autorisation (ou dans certains cas la restriction) d'usage de ces substances.

Si le système mis en place pour les substances nouvelles est considéré comme plutôt efficace, pour les substances anciennes (estimées à plus ou moins 30 000 substances produites en quantité supérieure à 1 tonne/an) le processus est lent et les moyens disponibles insuffisants pour faire face à la méconnaissance de leurs risques. Les exigences

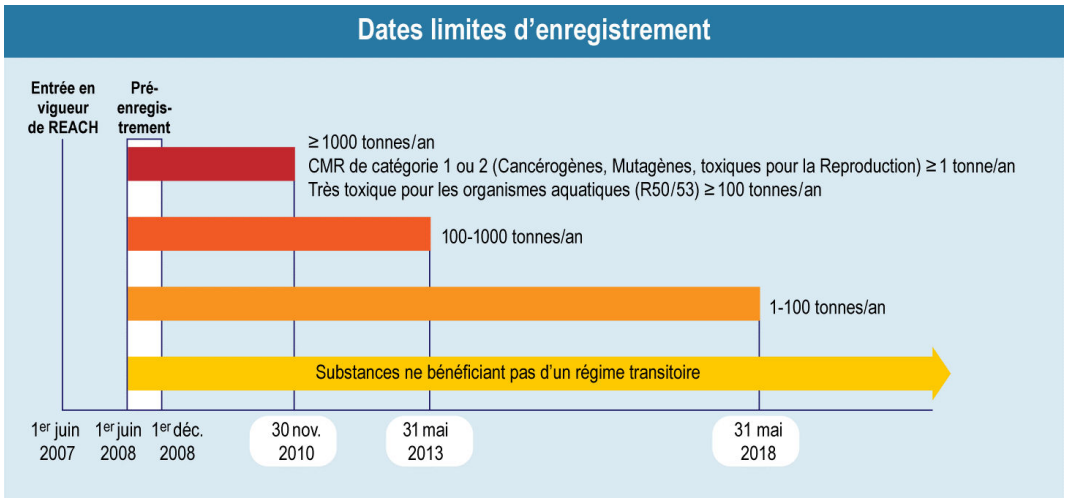

FIGURE 4.1 – Dates limites d'enregistrement pour les substances incluses dans le champ d'application de REACH

de REACH en matière d'informations pour les dossiers d'enregistrement (endpoints) sont listées dans la figure 4.2, en fonction du tonnage de la substance. C'est pour répondre à ces moyens insuffisants que le recours aux méthodes alternatives telles que la toxicologie computationnelle est encouragé [GMB+12].

Dans le cadre de cette étude, nous nous sommes particulièrement intéressés à la toxicité aiguë des substances chimiques vis-à-vis de trois espèces aquatiques : daphnies (endpoint 9.1.1), algues (endpoint 9.1.2) et poissons (endpoint 9.1.3). Pour être considérée comme toxique pour l'environnement aquatique, une substance chimique doit causer la mort ou l'inhibition de croissance d'au moins 50% de de l'une de ces population dans des conditions expérimentales précises.

L'objectif de ce chapitre est d'appliquer la méthode de sélection développée dans le cadre de ces travaux de thèse pour mettre en évidence des fragments chimiques responsables d'un excès de toxicité pour des organismes aquatiques. De tels fragments sont également appelés toxicophores et leur découverte est bien souvent à la base des modèles de prédiction en écotoxicologie [VvLH92].

Dans la suite de ce chapitre, nous explicitons le jeu de données utilisé, ainsi que les ensembles de règles définis. Nous ferons ensuite une évaluation quantitative et qualitative de l'application de la méthode de sélectionsur le jeu de données chimique.

EXIGENCES DE REACH EN MATIERE D'INFORMATIONS STANDARD	TONNAGE
INFORMATIONS SUR LES PROPRIÉTÉS PHYSICOCHIMIQUES DE LA SUBSTANCE	
7.1. État de la substance à 20 °C et 101,3 kPa	≥ 1t
7.2. Point de fusion/congélation	≥ 1t
7.3. Point d'ébullition	≥ 1t
7.4. Densité relative	≥ 1t
7.5. Pression de vapeur	≥ 1t
7.6. Tension superficielle	≥ 1t
7.7. Hydrosolubilité	≥ 1t
7.8. Coefficient de partage n-octanol/eau	≥ 1t
7.9. Point d'éclair	≥ 1t
7.10. Inflammabilité	≥ 1t
7.11. Propriétés explosives	≥ 1t
7.12. Température d'auto-inflammation	≥ 1t
7.13. Propriétés comburantes	≥ 1t
7.14. Granulométrie	≥ 1t
7.15. Stabilité dans les solvants organiques et identité des produits de dégradation à prendre en considération	≥ 100t
7.16. Constante de dissociation	≥ 100t
7.17. Viscosité	≥ 100t
INFORMATIONS TOXICOLOGIQUES	
Irritation	
8.1. Irritation ou corrosion cutanée *in vitro*	≥ 1t
8.1.1. Irritation cutanée *in vivo*	≥ 10t
8.2. Irritation oculaire *in vitro*	≥ 1t
8.2.1. Irritation oculaire *in vivo*	≥ 10t
Sensibilisation	
8.3. Sensibilisation cutanée	≥ 1t
Mutagénicité	
8.4.1. Étude *in vitro* de mutations géniques sur des bactéries	≥ 1t
8.4.2. Étude *in vitro* de cytogénicité sur cellules de mammifères ou étude in vitro du micronoyau	≥ 10t
8.4.3. Étude *in vitro* de mutation génique sur cellules de mammifères	≥ 10t
Toxicité aigüe	
8.5.1. Par voie orale	≥ 1t
8.5.2. Par inhalation	≥ 10t
8.5.3. Par voie cutanée	≥ 10t
Toxicité par administration répétée	
8.6.1. Étude de toxicité à court terme par administration répétée (28 jours)	≥ 10t
8.6.2. Étude de toxicité subchronique (90 jours)	≥ 100t
Toxicité pour la reproduction	
8.7.1. Dépistage de la toxicité pour la reproduction/le développement,	≥ 10t
8.7.2. Étude de toxicité sur le développement prénatal	≥ 100t
8.7.3. Étude de toxicité pour la reproduction sur deux générations	≥ 100t
Toxicocinétique	
8.8.1. Évaluation du comportement toxicocinétique de la substance	≥ 10t
Carcinogénicité	
8.9.1. Étude de carcinogénicité	≥ 1000t
INFORMATIONS ÉCOTOXICOLOGIQUES	
Toxicité aquatique	
9.1.1. Essais de toxicité aquatique à court terme sur invertébrés	≥ 1t
9.1.2. Étude d'inhibition de croissance sur plantes aquatiques	≥ 1t
9.1.3. Essais de toxicité aquatique à court terme sur des poissons	≥ 10t
9.1.4. Étude de l'inhibition respiration sur boue activée	≥ 10t
9.1.5. Essais de toxicité aquatique à long terme sur invertébrés	≥ 100t
9.1.6. Essais de toxicité aquatique à long terme sur des poissons	≥ 100t
9.1.6.1. Essais de toxicité aquatique sur des poissons aux premiers stades de leur vie (FELS)	≥ 100t
9.1.6.2. Essai de toxicité aquatique à court terme sur des poissons aux stades de l'embryon et de l'alevin	≥ 100t
9.1.6.3. Poissons, essai sur la croissance des juvéniles	≥ 100t
Dégradation	
9.2.1.1. Biodégradabilité facile (biotique)	≥ 1t
9.2.1.2. Essais de simulation sur la dégradation ultime dans les eaux de surface (biotique)	≥ 100t
9.2.1.3. Essais de simulation dans le sol (biotique)	≥ 100t
9.2.1.4. Essais de simulation dans les sédiments (biotique)	≥ 100t
9.2.2.1. Hydrolyse en tant que fonction du pH (abiotique)	≥ 10t
9.2.3. Identification des produits de dégradation	≥ 100t
Devenir et comportement dans l'environnement	
9.3.1. Dépistage de l'adsorption/désorption	≥ 10t
9.3.2. Bioaccumulation dans une espèce aquatique	≥ 100t
9.3.3. Informations supplémentaires sur l'adsorption/désorption	≥ 100t
9.3.4. Informations supplémentaires sur le devenir et le comportement dans l'environnement de la substance et/ou des produits de dégradation	≥ 1000t
Effets sur les organismes terrestres	
9.4.1. Toxicité à court terme pour les invertébrés	≥ 100t
9.4.2. Effets sur les micro-organismes du sol	≥ 100t
9.4.3. Toxicité à court terme pour les plantes	≥ 100t
9.4.4. Essais de toxicité à long terme sur des invertébrés	≥ 100t
9.4.6. Essais de toxicité à long terme sur des plantes	≥ 1000t
Effets sur les organismes des sédiments	
9.5.1. Toxicité à long terme pour les organismes vivant dans des sédiments	≥ 1000t
Effets sur les oiseaux	
9.6.1. Toxicité à long terme ou toxicité pour la reproduction chez les oiseaux	≥ 1000t

4.2 Matériels et méthodes

4.2.1 Jeu de données Aquatox

Nous appelons *Aquatox* la base de données nous ayant permis de réaliser ces études de recherche de toxicophores. Ce jeu de données a été construit au laboratoire CERMN en regroupant des données internes et des informations extraites à partir du site de l'ECHA (European Chemicals Agency). Aquatox est un jeu de de données chimique sur l'environnement aquatique et il est partitionné en 362 substances très toxiques et 187 faiblement toxiques. Pour appliquer la méthode de sélection définie au chapitre 3, le jeu de données de molécules a été redécrit à l'aide de descripteurs moléculaires. La première étape a donc consisté à générer des fragments chimiques de référence dans le domaine de la chémo-informatique : les descripteurs ECFP6 (Extended-connectivity fingerprints) (voir Figure 4.3) dont le diamètre maximal est inférieur ou égal à 6 liaisons dans une molécule [RH10]. En utilisant les descripteurs ECFP6, nous obtenus 523 descripteurs différents. Ainsi en résumé, Aquatox constitue un jeu de données de 549 objets et 523 descripteurs avec 66% de molécules très toxiques et 34% faiblement toxiques.

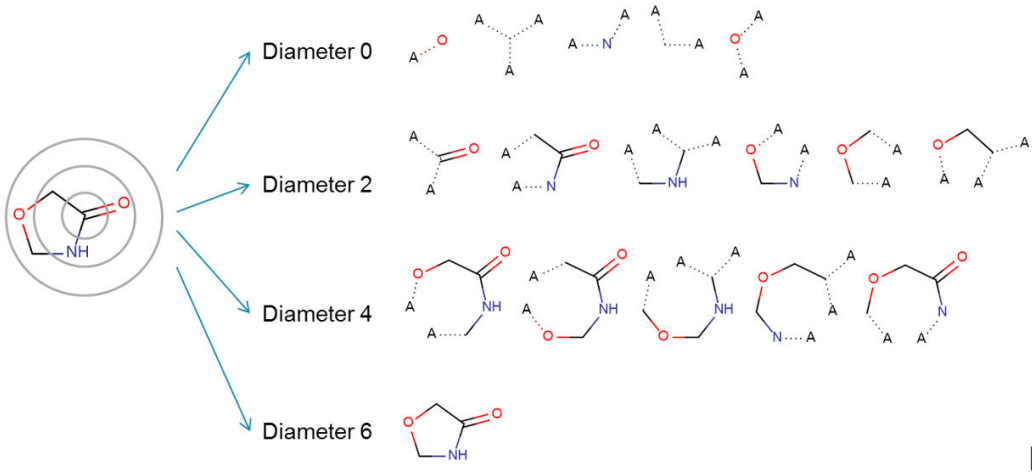

FIGURE 4.3 – Exemple de génération des descripteurs ECFP 0, 2, 4 et 6 (illustration inspirée de https ://docs.chemaxon.com/)

4.2.2 Ensembles de règles utilisés :

Afin d'implémenter la méthode de sélection des règles sur le jeu de données Aquatox, nous avons besoin de choisir les ensembles de règles prototypes et candidates. Pour cette expérience, les ensembles de règles suivants ont été définis :

Règles prototypes correspondent aux règles associées aux motifs émergents de taux de croissance infini et de support minimum égal à 7. Le choix de ces règles prototypes s'explique par le fait qu'elles ont les mêmes caractéristiques que les règles prototypes du chapitre 3, comme expliqué à la section 3.2.2.1 du chapitre 3. Cela veut dire dire qu'elles possèdent une bonne confiance sur les objets qui sont couverts mais malheureusement ne couvrent pas une bonne partie des objets. Les règles associées aux JEPs de support minimum 10, choisies comme règles prototypes dans la section 3.2.1.3 du chapitre 3, sont très peu présentes dans le jeu de données Aquatox. Elles n'ont donc pas été choisies comme règles prototypes dans cette partie expérimentale.

Règles candidates. Nous avons défini 10 ensembles de règles candidates correspondant respectivement aux motifs émergents de taux de croissance 20, 10, ou 5 et de support minimum 10, 7, ou 3. Le support minimum ne varie pas jusqu'à 1 car au vu de la taille du jeu de données les règles associées aux motifs émergents de support minimum égal à 1 couvrent une grande partie des molécules mais ont une confiance faible.

Les règles fréquentes seront ainsi l'union des règles prototypes et des règles candidates.

4.2.3 Paramètres pour la sélection :

Similarité minimum et proportion minimum de règles prototypes voisines. Le calcul de ces paramètres reste le même qu'à la section 3.1.2.2 du chapitre 3. Ces deux paramètres permettent de sélectionner les règles candidates qui sont les plus similaires aux règles prototypes.

4.2.4 Validation croisée :

Les valeurs de couverture et de confiance sont mesurées en utilisant une validation croisée à 5 volets. Pour chaque volet, nous avons 80% (440) molécules d'Aquatox pour le jeu d'apprentissage et 20% (138) molécules d'Aquatox pour le jeu de test.

4.2.5 Variation de couverture et de confiance

Comme dans le chapitre 3, pour un ensemble de règles considéré, nous définissons le gain de couverture et la perte de confiance en comparaison avec les règles prototypes correspondantes en utilisant la formule 4.1.

$$\begin{aligned} Gain\_Couverture(ensemble\_regles) &= \frac{Couverture(ensemble\_regles) - Couverture(regles\_prototypes)}{Couverture(regles\_prototypes)} \\ Perte\_Confiance(ensemble\_regles) &= \frac{Confiance(reglesprototypes) - Confiance(ensemble\_regles)}{Confiance(regles\_prototypes)} \end{aligned} \tag{4.1}$$

Ces deux mesures permettent d'évaluer la qualité de la méthode de sélection, c'est à dire dans quelle mesure on gagne en couverture et on perd en confiance.

4.2.6 Classification avec CPAR

Après avoir évalué la qualité de la méthode de sélection des règles en mesurant la couverture et la confiance des règles sélectionnées, nous voulons aussi étudier la cohérence de l'ensemble des règles sélectionnées dans un contexte de classification ayant pour objectif la prédiction de la toxicité des molécules. Pour la méthode de classification, nous avons utilisé *Classification based on Predictive Association Rules (CPAR)* [YH03] qui montre de bons résultats en terme de précision comparé aux méthodes de classification à base de règles d'associations comme expliqué à la section 3.2.3 du chapitre 3. Nous voulons savoir grâce à cette expérience, si les règles sélectionnées permettent de moins utiliser la règle par défaut du classifieur et si elles permettent d'améliorer la précision du classifieur par rapport aux règles prototypes.

4.3 Application de la méthode de sélection des règles

4.3.1 Évaluation de la couverture et de la confiance des ensembles de règles

Dans cette partie, nous évaluons la qualité de la méthode de sélection. Tout d'abord la couverture et la confiance des règles prototypes et des règles fréquentes sont étudiées. Ensuite, la couverture et la confiance des règles sélectionnées sont évaluées. Nous montrerons l'intérêt de la méthode sélection en comparant les valeurs de couverture et de confiance des règles sélectionnées par rapport à celles des règles prototypes et fréquentes. Les résultats que nous présentons ici sont des moyennes sur une validation croisée à 5 volets. Le tableau 4.1 montre la couverture et la confiance moyenne des différents ensembles de règles pour le jeu de données Aquatox. La colonne *Cov* représente la couverture moyenne et la colonne *Conf* la confiance moyenne. *TC* fait référence au taux de croissance et *Support_ min* le support minimum. La colonne *RS* représente les règles sélectionnées et la colonne *RF* les règles fréquentes. Le tableau 4.2 montre le gain en couverture et la perte en confiance des différents ensembles de règles grâce à la formule 4.1. La valeurs en gras sont les valeurs réelles de couverture et de confiances des règles prototypes. La colonne *GainCov* représente le gain en couverture et *PerteConf* la perte en confiance.

Tableau 4.1 – Couverture et confiance des ensembles de règles sur le jeu de données Aquatox

	Support_ min	10		7		3	
TC		RF	RS	RF	RS	RF	RS
∞	*Cov*	x	x	**83,85**	**83,85**	95,46	93,94
	Conf	x	x	**79,54**	**79,54**	63,92	77,03
20	*Cov*	85,71	85,42	94,2	90,15	96,97	94,28
	Conf	76,39	78,99	70,65	76,95	60,73	75,45
10	*Cov*	91,29	88,56	97,35	92,77	98,76	96,77
	Conf	74,04	76,38	68,07	73,52	57,81	72,48
5	*Cov*	93,64	90,63	98,15	96,87	100,00	98,72
	Conf	68,29	70,53	63,37	68,99	56,41	65,81

Tableau 4.2 – Gain en couverture et perte en confiance des différents ensembles de règles

	Support_ min	10		7		3	
TC		RF	RS	RF	RS	RF	RS
∞	*GainCov*	x	x	**83,85**	**83,85**	13,85	12,03
	PerteConf	x	x	**79,54**	**79,54**	-19,64	-3,16
20	*GainCov*	2,22	1,87	12,34	5,13	15,65	12,44
	PerteConf	-3,96	-0,69	-11,18	-3,26	-23,65	-5,14
10	*GainCov*	8,87	5,62	16,29	10,64	17,78	15,41
	PerteConf	-6,91	-3,97	-14,42	-7,57	-27,32	-8,88
5	*GainCov*	11,68	8,09	17,05	15,53	19,26	17,5
	PerteConf	-14,14	-11,33	-20,33	-13,26	-29,08	-17,26

4.3.1.1 Analyse des règles prototypes

Les règles prototypes, sur le jeu de données Aquatox, possèdent généralement une bonne confiance, mais ne couvrent pas une bonne partie des molécules. En effet, les règles prototypes ont une couverture de 83,85%. Cela veut dire donc qu'une bonne partie (16,15%) des molécules ne sont pas couvertes par une règle prototype. Cependant la confiance de ces règles s'élève à 79,54%, ce qui laisse espérer un contexte favorable pour appliquer la méthode de sélection. Cela veut dire que que la méthode de sélection permettrait d'augmenter le nombre de molécules couvertes avec une bonne valeur de confiance.

4.3.1.2 Analyse des règles fréquentes

Ces règles couvrent une bonne partie des molécules mais ont une confiance faible. Sur le tableau 4.1 par exemple, les règles fréquentes associées aux JEPs de support minimum 3 couvrent 95,46% des molécules mais ont seulement une confiance de 63,92%. Plus le taux de croissance ou le support est faible, plus ces règles couvrent les molécules mais leur confiance chute fortement. Par exemple, sur le tableau 4.1, la couverture passe de 95,46% pour les règles fréquentes associées aux JEPs de support minimum 3 à 98,76% pour les règles associées aux motifs émergents de taux de croissance 10 de support minimum 3. Cependant la confiance chute fortement de 63,92% à 57,81%. En s'appuyant sur le tableau 4.2, on peut observer ces mêmes tendances en terme de pourcentage. Pendant que les règles fréquentes associées aux JEPs de support minimum 3 permettent de gagner en moyenne 13,85% en couverture avec une perte en confiance de 19,64%, les règles fréquentes associées aux motifs émergents de taux de croissance 10 de support minimum 3 montrent un gain en couverture plus important équivalent à 17,78% avec cependant une perte en confiance beaucoup plus importante de 27,32%.

Le tableau 4.2 permet de voir dans quelle mesure un relâchement du support ou du taux de croissance permet de gagner en terme de couverture au prix d'une perte de confiance. Naturellement l'assouplissement des contraintes de support ou de taux de croissance conduit à retenir un sur-ensemble des règles, entraînant une augmentation de la couverture des molécules mais avec une confiance plus faible. Le tableau 4.2 permet de mesurer ces variations.

4.3.1.3 Analyse des règles sélectionnées

Comparaison avec les règles fréquentes. Grâce à l'application de la méthode de sélection, les règles sélectionnées permettent d'avoir un gain en couverture très similaire à celle des règles fréquentes avec en plus une perte en confiance beaucoup moins importante comparée aux règles prototypes. Par exemple sur le tableau 4.1, la couverture des règles

sélectionnées associées aux JEPs de support minimum 3 étant égal à 93,94% est sensiblement égale à celle des règles fréquentes correspondantes qui s'élève à 95,46%. Cependant la confiance des règles sélectionnées associées aux JEPs de support minimum 3 est de 77,03% tandis que les règles fréquentes correspondantes ont une confiance beaucoup plus faible égal à 63,92%.

En terme de gain en couverture et de perte en précision, on observe sur le tableau 4.2 que les règles sélectionnées ont un gain en couverture assez similaire aux règles fréquentes avec une perte en confiance beaucoup moins importante. Par exemple, les règles sélectionnées associées aux JEPs avec un support minimum de 3 ont un gain en couverture de 12,03% alors que les règles fréquentes correspondantes ont un gain en couverture de 13,85%. Pour ces dernières la perte en confiance est de 19,64% alors que les règles sélectionnées correspondantes ont une perte en confiance de 3,16%.

Comparaison avec les règles prototypes. Les règles sélectionnées permettent d'améliorer significativement la couverture des règles prototypes avec une perte en confiance très faible. Par exemple sur le tableau 4.1, on peut voir que les règles sélectionnées associées aux JEPs de support minimum 3 permettent à la couverture de passer de 83,85% pour les règles prototypes à 93,94%. Alors que la confiance varie de 79,54% pour les règles prototypes à 77,03%.

En terme de gain en couverture et de perte en précision, on voit sur le tableau 4.2 que les règles sélectionnées associées aux JEPs de support minimum 3 ont un gain en couverture de 12,03% avec seulement une perte en confiance de 3,16%. Cet ensemble de règles possède une particularité intéressante dû au fait qu'il améliore significativement la couverture des règles prototypes sans pour autant dégrader leur confiance.

4.3.1.4 Ensembles de règles particuliers

Il est intéressant de pointer des ensembles de règles sélectionnées ou fréquentes qui permettent d'obtenir un bon compromis entre le gain en couverture et la perte en confiance. En considérant la confiance et la couverture, nous proposons la figure 4.4 qui montre les ensembles de règles pour lesquels la perte en confiance est d'au maximum 5% comparée à celle des prototypes. La colonne de couleur grise représente le gain en couverture et la colonne de couleur noire la perte en confiance. La notation utilisée pour les ensembles de règles est *a_b_c*, avec *a* étant le taux de croissance, *b* le type de règles à savoir les règles prototypes, fréquentes ou sélectionnées et *c* le support minimum fixé. Parmi les cinq ensembles de règles qui vérifient la perte d'au maximum 5% de la confiance, on observe que quatre parmi eux correspondent à des règles sélectionnées, ce qui montre que les règles sélectionnées non seulement permettent d'améliorer la couverture des règles mais aussi ont

une faible perte en confiance comparé aux règles prototypes. Une remarque intéressante est que l'ensemble des règles sélectionnées associées aux JEPs de support minimum 3 (inf_ RS_ 3) possède un très bon compromis entre le gain en couverture et la perte en confiance avec un gain en couverture en moyenne de 12,03% avec seulement une perte en confiance de 3,16%.

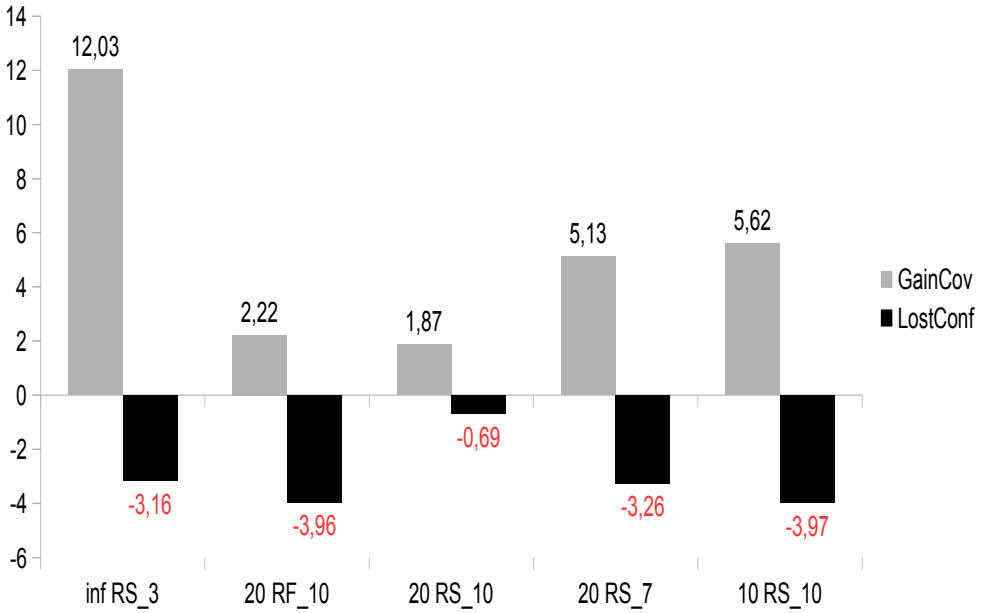

FIGURE 4.4 – Compromis entre le gain en couverture et la perte en confiance comparé aux règles prototypes

4.4 Classification avec CPAR

Dans cette deuxième partie de l'expérimentation, nous utilisons l'algorithme de classification CPAR [YH03] afin d'évaluer la qualité de la méthode de sélection des règles appliquée au jeu de données Aquatox. La règle par défaut conclut sur la classe majoritaire. Nous mesurons la couverture et la précision pour chaque ensemble de règles. Ces valeurs sont montrées sur le tableau 4.3 pour le jeu de données *Aquatox*. La colonne *RF* désigne les règles fréquentes et *RS* les règles sélectionnées. La colonne *Accuracy* montre le taux de bonne prédiction du classifieur, c'est à dire la précision. La colonne *Cov* représente le pourcentage d'objets pour lesquels la règle par défaut n'est pas utilisée, que nous appelons les objets couverts.

Le tableau 4.4 montre, quant à lui, le gain en couverture et la perte ou le gain en

précision en moyenne sur le jeu de données Aquatox. Les valeurs en gras correspondent aux valeurs de couverture et de précision sur les règles prototypes. La colonne *GainCov* représente le gain en couverture et la colonne *(Gain|Perte) Accuracy* le gain ou la perte en précision.

4.4.1 Règles prototypes

Le tableau 4.4 montre que les règles prototypes ne couvrent que 83,85% des molécules. Cela implique donc que la règle par défaut est souvent utilisée afin de faire une prédiction. Ce qui affecte significativement la précision du classifieur qui est en moyenne de 80,94%. Les règles prototypes ne permettent donc pas d'atteindre une précision optimale pour le classifieur.

4.4.2 Règles fréquentes

Les règles fréquentes permettent de trouver une explication à beaucoup de molécules avec une couverture qui est significativement améliorée comparé aux règles prototypes. Cependant la précision du classifieur est faible si on utilise les règles fréquentes. Par exemple sur le tableau 4.3, les règles fréquentes associées aux JEPs de support minimum 3 améliorent la couverture qui passe de 83,85% pour les prototypes à 95,46%. Néanmoins, la précision du classifieur chute de 80,94% pour les prototypes à 76,10%.

Cette même tendance se confirme si on étudie le tableau 4.4. Par exemple, les règles associées aux JEPs de support minimum 3 permettent de trouver une explication concernant la prédiction, à 13,85% plus de molécules comparé aux règles prototypes. Cependant la perte de précision du classifieur est significative et équivaut à 5,98%.

4.4.3 Règles sélectionnées

Les règles sélectionnées possèdent le double avantage de donner une explication à plus de molécules lors de la phase de prédiction et aussi d'améliorer significativement la précision du classifieur. Par exemple sur le tableau 4.3, les règles associées aux JEPs de support minimum 3 améliore la couverture qui passe de 83,85% pour les prototypes à 93,94%, et augmente aussi la précision du classifieur qui varie de 80,94% pour les prototypes à 84%.

Sur le tableau 4.4, on constate, par exemple, que les règles associées aux JEPs de support minimum 3 permettent d'améliorer la précision du classifieur de 3,78% comparée à la précision des règles prototypes et donne une explication à 12,03% de molécules en plus.

Tableau 4.3 – Précision des ensembles de règles avec le classifieur CPAR

	Support_ min	10		7		3	
TC		RF	RS	RF	RS	RF	RS
∞	*Accuracy*	x	x	**80,94**	**80,94**	76,10	84,00
	Cov	x	x	**83,85**	**83,85**	95,46	93,94
20	*Accuracy*	80,91	81,69	76,04	82,99	73,87	83,03
	Cov	85,71	85,42	94,20	88,15	96,97	94,28
10	*Accuracy*	74,85	76,65	71,99	73,93	68,36	71,10
	Cov	91,29	88,56	97,35	92,77	98,76	96,77
5	*Accuracy*	70,45	71,17	67,21	70,89	62,17	68,45
	Cov	93,64	90,63	98,15	96,87	100,00	98,72

Tableau 4.4 – Gain en couverture et perte en précision des différents ensembles de règles

	Support_ min	10		7		3	
TC		RF	RS	RF	RS	RF	RS
∞	*GainCov*	x	x	**83,85**	**83,85**	13,85	12,03
	(Gain|Perte) Accuracy	x	x	**80,94**	**80,94**	-5,98	3,78
20	*GainCov*	2,22	1,87	12,34	5,13	15,65	12,44
	(Gain|Perte) Accuracy	-0,04	0,93	-6,05	2,53	-8,73	2,58
10	*GainCov*	8,87	5,62	16,1	10,64	17,78	15,41
	(Gain|Perte) Accuracy	-7,52	-5,3	-11,06	-8,66	-15,54	-12,16
5	*GainCov*	11,68	8,09	17,05	15,53	19,26	17,5
	(Gain|Perte) Accuracy	-12,96	-12,07	-16,96	-12,42	-23,19	-15,43

La figure 4.5 montre les ensembles de règles qui permettent d'améliorer la couverture et la précision du classifieur pour le jeu de données Aquatox. La colonne de couleur noire représente le gain en couverture et la colonne de couleur grise le gain en précision avec à

chaque fois les valeurs des gains. Pour les ensembles de règles, nous utilisons la notation a b_c, avec a étant le taux de croissance, b le type de règle à savoir les règles prototypes, fréquentes ou sélectionnées et c le support minimum fixé. On peut observer sur la figure 4.5 que les seuls ensembles de règles qui permettent d'améliorer la précision du classifieur sont les règles sélectionnées associées aux JEPs de support minimum 7 et 3, ainsi que les règles associées aux motifs émergents de taux de croissance minimum égal à 20 et un support minimum, respectivement, de 10, 7 et 3.

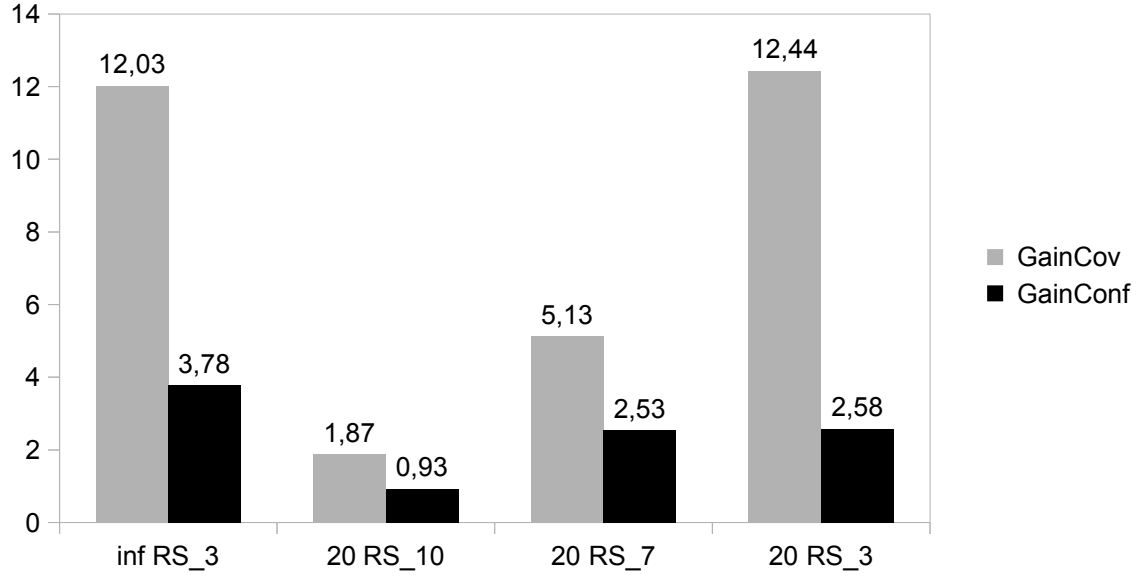

FIGURE 4.5 – Comparaison de la précision et de la couverture des règles sélectionnées avec les règles prototypes

4.4.4 Règles sélectionnées associées aux JEPs de support minimum 3

Ces règles présentent un intérêt particulier parce qu'elles améliorent significativement la couverture et la précision du classifieur . En s'appuyant sur le tableau 4.4, on observe que cet ensemble de règles permet d'obtenir un gain en couverture 12,03% avec un gain en précision valant 3,16%. Cela veut dire que cet ensemble de règles permet de donner une explication à beaucoup de molécules en plus, tout en améliorant significativement la précision du classifieur fournissant ainsi une meilleure prédiction.

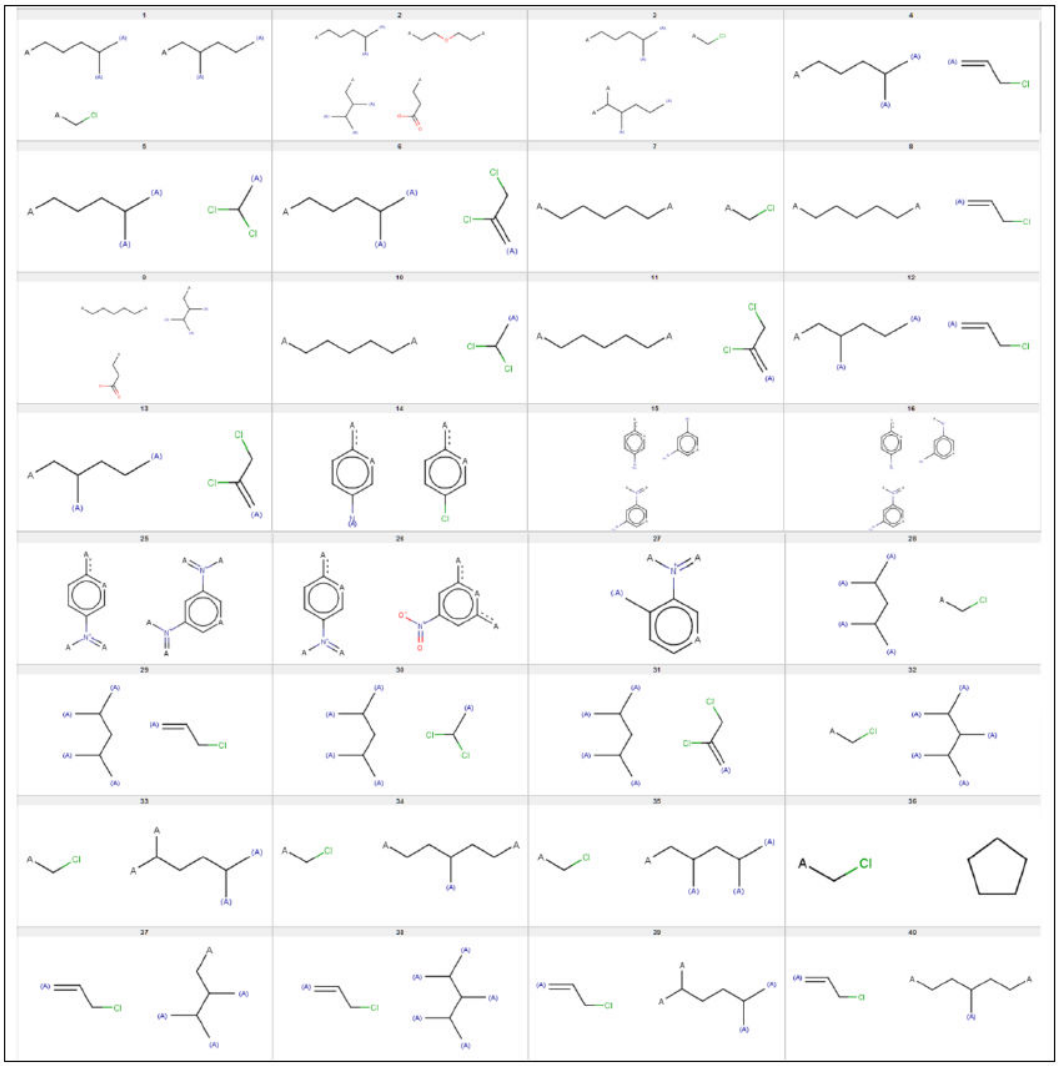

FIGURE 4.6 – Règles prototypes du jeu de données Aquatox

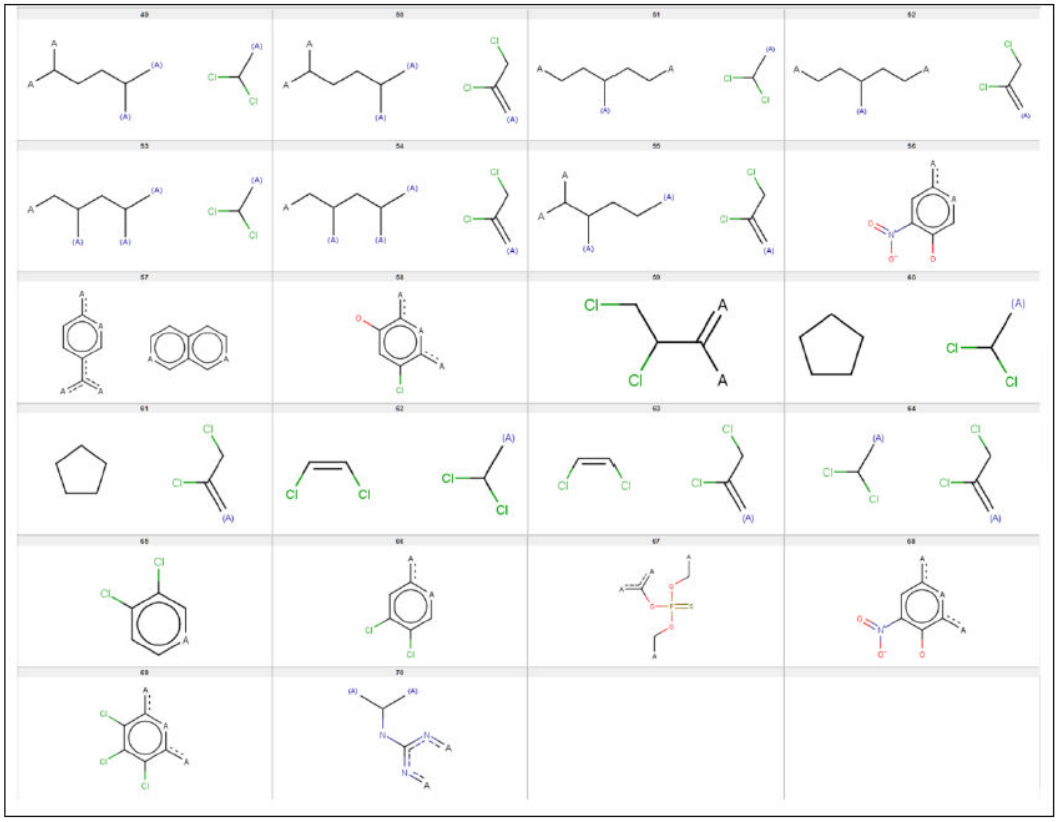

FIGURE 4.7 – Règles prototypes du jeu de données Aquatox (suite)

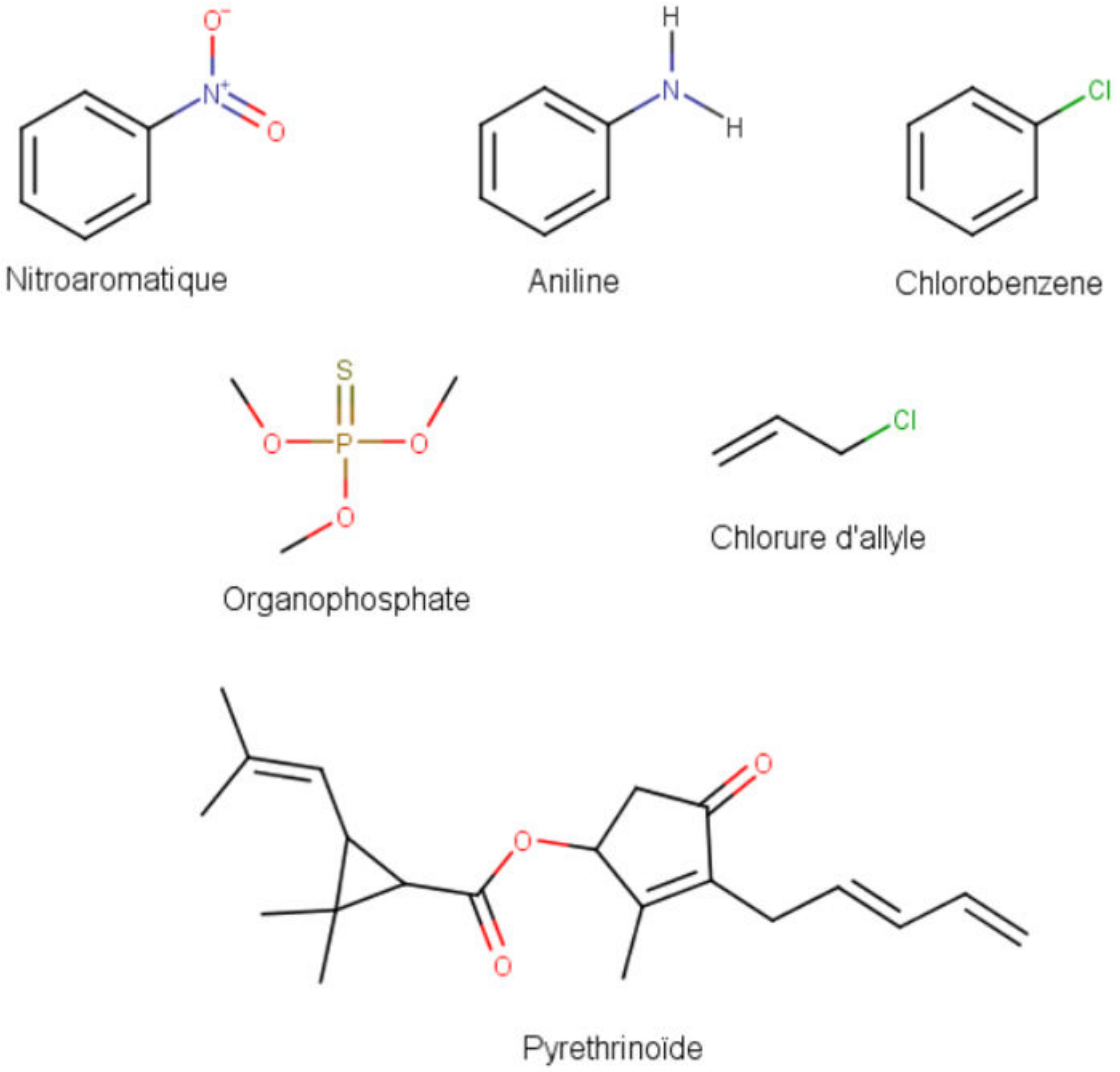

FIGURE 4.8 – Exemples de toxicophores en accord avec les règles prototypes

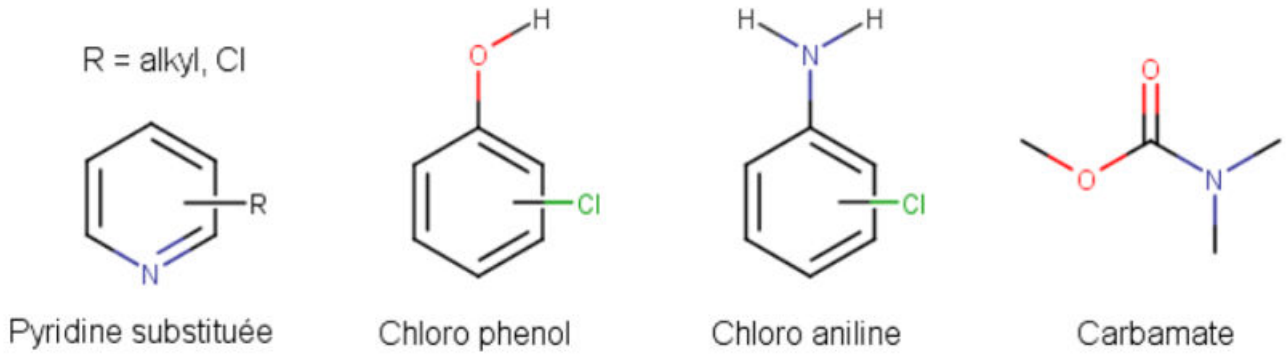

FIGURE 4.9 – Exemples de toxicophores en accord avec des règles sélectionnées

4.5 Analyse qualitative de la méthode de sélection appliquée au jeu de données Aquatox

En considérant comme prototypes les JEPs ayant un support minimum de 7, plusieurs toxicophores bien connus ont été retrouvés (Figure 4.8). Un listing complet des fragments chimiques décrits comme toxiques pour les organismes aquatiques est accessible sur le site internet de ToxAlert [SSP+12]. Trois d'entre eux sont des composés aromatiques : les nitroaromatiques, les anilines et les chlorobenzènes. La contamination de l'eau et des sols par les produits chimiques aromatiques est en effet très répandue puisqu'on les retrouve dans de nombreux produits industriels tels que des pesticides, des solvants, des colorants et même des explosifs. Beaucoup de ces produits peuvent s'accumuler dans la chaine alimentaire et ont un effet néfaste aussi bien sur les plantes que sur les animaux et l'homme. Le quatrième toxicophore mis en évidence correspond au motif organophosphate qui est également retrouvé dans de nombreux pesticides. Pour compléter cette liste, on retrouve également de nombreux groupement halogénés (essentiellement chlorés, comme par exemple les chlorures d'allyle) et des fragments moléculaires rappelant les pyrethrinoïdes, des composés chimiques utilisés comme insecticides. L'analyse des règles candidates sélectionnées montre que la souplesse apportée au niveau de la sélection permet de récupérer des toxicophores également bien décrits dans la littérature (Figure 4.9). En particulier, cette extraction permet d'identifier de façon beaucoup plus exhaustive des cycles aromatiques de différentes natures. En effet, cette nature peut varier en fonction i) de la présence/absence d'hétéroatomes (N, S, O), ii) du nombre de cycles, iii) de la présence/absence de substituants. On retrouve en particulier des pyridines substituées par des groupements alkyles ainsi que des phénols ou des anilines substituées par des chlores. Il est également important de noter que parmi les règles sélectionnées on retrouve un autre type de toxicophore très néfaste pour les organismes aquatiques, à savoir le groupement carbamate qui est un composant majeur d'une catégorie de pesticides.

4.6 Conclusion

Dans ce chapitre, nous avons appliqué la méthode de sélection sur un jeu de données chimique nommé Aquatox. En prenant les prototypes comme étant les règles associées aux JEPs de support minimum 7, les résultats expérimentaux ont montré que la méthode de sélection permet d'améliorer la couverture avec une perte en confiance assez faible. Par exemple, l'ensemble des règles sélectionnées associées aux JEPs de support minimum 3 permet d'améliorer la précision du classifieur de 3,78% comparée à la précision des règles prototypes et donne une explication à 12,03% de molécules en plus. Dans un contexte

de classification la précision est en plus améliorée de 3,16% pour l'ensemble des règles sélectionnées associées aux JEPs de support minimum 3, comparée aux règles prototypes. Cela montre donc que la méthode est fiable et permet de donner une explication à beaucoup plus de molécules dans un contexte de classification sans pour autant dégrader la précision du classifieur.

L'analyse qualitative de la méthode sélection a aussi montré que la sélection de règles conduit à la découverte de molécules ayant des propriétés intéressantes et qui permettent de prédire la classe des molécules.

Conclusion et perspectives

Bilan

Dans ce travail, nous avons proposé une nouvelle méthode efficace d'extraction des JEPs minimaux à partir des différences globales. Dans la pratique, les JEPs les plus supportés ne couvrant pas une bonne partie des objets, nous avons défini un cadre pour la sélection de règles supervisées qui apporte une amélioration significative en terme de couverture et de précision dans un contexte de classification. L'application de cette méthode de sélection en chémoinformatique a aussi permis la découverte de nouvelles connaissances intéressantes sur les données et une meilleure prédiction de la toxicité des molécules.

Extraction des JEPs minimaux

Dans un contexte de classification, les JEPs (Jumping Emerging Patterns) permettent d'avoir une bonne prédiction grâce à leur fort pouvoir discriminant. Cependant leur extraction reste une tâche difficile. Plusieurs travaux utilisent une contrainte sur le seuil de support afin d'extraire les JEPs minimaux les plus significatifs au risque de passer à côté de JEPs très intéressants mais de fréquence faible.

Nous avons proposé une nouvelle méthode efficace qui permet d'extraire tous les JEPs avec ou sans contrainte sur le seuil de support [KCC15]. Nous avons aussi inclus dans notre approche la prise en compte de l'absence d'attribut qui apporte une nouvelle connaissance sur les motifs extraits. Notre méthode se base sur les différences globales afin de créer un arbre de recherche (ToMJEPs) pour générer efficacement les JEPs minimaux. Nous avons montré expérimentalement que les temps d'exécution sont trois fois meilleurs, en moyenne sur 13 jeux de données, comparés aux autres méthodes d'extraction des JEPs minimaux. Les tests expérimentaux ont aussi prouvé que plus les JEPs minimaux sont supportés plus leur couverture est faible et plus leur confiance est grande. Pour améliorer la relation entre le support, la couverture et la confiance, nous avons proposé une méthode de sélection à base de prototypes.

Sélection de règles à base de prototypes

Un JEP minimal peut être considéré comme une règle, donc une conjonction d'attributs qui conclut sur une classe. Nous avons montré, d'une manière expérimentale, que plus le support des JEPs minimaux est élevé plus la couverture de l'ensemble de règles associées à ces JEPs minimaux est faible et plus la confiance de cet ensemble de règles est grande. Cela constitue un inconvénient majeur quant à l'utilisation des JEPs minimaux avec un seuil élevé de support en classification car, cet ensemble de JEPs ne couvre pas une partie conséquente des objets.

Nous avons ainsi proposé une méthode de sélection à base de prototypes qui s'appuie sur les règles associées aux JEPs avec un support élevé (règles prototypes). La sélection est alors faite sur les autres règles qui sont les plus similaires à ces règles prototypes. Pour les ensembles de règles ainsi obtenus, les tests expérimentaux ont montré que la couverture est beaucoup plus importante (jusqu'à 20% de plus) que celle des règles prototypes avec une faible perte en confiance comparée aux règles prototypes (pas plus de 5% pour certains seuils de support ou de taux de croissance). En utilisant le classifieur CPAR, nous avons aussi montré que plusieurs ensembles de règles, en plus d'augmenter la couverture, permettent d'améliorer jusqu'à 2,9% la précision comparée aux règles prototypes.

Nous avons aussi proposé une amélioration de la méthode de sélection appelée méthode de sélection à plusieurs passes. Les règles sélectionnées deviennent à leur tour des règles prototypes lors de la sélection d'un nouvel ensemble de règles candidates. Cette amélioration permet aussi d'augmenter la couverture des règles sélectionnées sans pour autant dégrader leur confiance.

Application de la méthode de sélection sur le jeu de données chimique Aquatox

Nous avons appliqué notre méthode de sélection en chémoinformatique. Pour cette partie, nous avons travaillé en collaboration avec le Centre d'Études et de Recherche sur le Médicament de Normandie (CERMN, UPRES EA 4258, Université de Caen Normandie).

Le jeu de de données chimique utilisé se nomme Aquatox. C'est un jeu données ayant rapport à l'environnement aquatique et partitionné en 362 substances très toxiques et 187 faiblement toxiques. L'application de la méthode de sélection a permis de donner plus d'explication quant à la classification de certaines molécules en minimisant l'utilisation de la règle par défaut, cet apport est lié à l'augmentation de la couverture. La méthode de sélection a aussi permis d'extraire des molécules toxiques et d'autres molécules qui sont bien reconnues dans la littérature, ce qui montre sa fiabilité à découvrir des connaissances

intéressantes.

Perspectives

Nous donnons maintenant des perspectives de recherche de ce travail que nous structurons en 3 axes.

Efficacité des méthodes d'extraction des motifs émergents

La tâche d'extraction de motifs en fouille de données est souvent très coûteuse et même infaisable si les jeux de données sont très volumineux avec beaucoup d'attributs. Avec la méthode d'extraction des JEPs minimaux que nous avons proposée, une perspective intéressante consiste à étudier l'ordonnancement des différences entre objets. Un ordre bien choisi pourrait donner lieu à un arbre de recherche avec moins de nœuds qui ne concluent pas sur des JEPs minimaux et ainsi améliorer considérablement les temps d'extraction.

Pondération

L'extraction de motifs en fouille de données génère souvent un très grand nombre de motifs qui peuvent avoir une grande taille. Il est donc difficile pour l'expert d'en faire une interprétation. Dans un cadre d'extraction de motifs émergents, l'objectif serait donc de mettre en relief la partie des motifs qui est (presque) responsable de la corrélation à une classe. Cela permettra à l'expert, en chémoinformatique par exemple, de se concentrer uniquement sur des sous motifs intéressants et non sur des motifs de très grande taille. Le but final est de proposer des méthodes de pondération adéquates pour cette mise en relief des motifs. Nous pourrons ainsi appliquer cette pondération dans la méthode de sélection à base de règles, plus précisément avec la similarité entre les règles. Comme base pour cette mise en relief des motifs, nous pourrons nous appuyer sur les travaux de Fang et *al.* [FWO+11] qui classe les motifs en quatre catégories selon une relation de dépendance entre les différents attributs. Tout motif émergent dont un sous-motif a presque le même pouvoir discriminant est classé dans le premier groupe. Jusqu'au groupe de motifs où, pour chaque motif, chaque attribut a approximativement la même importance pour l'émergence du motif. Par exemple, sur le tableau 1.1 du chapitre 1, le motif $\{a_3, a_5\}$ est un motif émergent de $\mathcal{G}_-$ vers $\mathcal{G}_+$ avec un taux de croissance infini. Ce motif serait classé dans le premier groupe car l'attribut a_3 est presque seul responsable de l'émergence (a_3 n'est présent que dans un seul objet négatif sur cinq alors que a_5 est présent dans trois objets négatifs sur cinq).

Préférences

Une autre perspective est de définir des préférences entre motifs afin de présenter à un expert uniquement les plus pertinents selon ses critères. Cette perspective consiste donc à extraire les ”meilleurs” motifs tout en se gardant la possibilité de présenter les motifs ”un peu” moins intéressants qui peuvent cependant montrer un réel intérêt. La sélection des meilleurs motifs pourrait être effectué en combinant plusieurs mesures comme le poids, le fréquence, ou le taux de croissance. Sur ce plan, les travaux effectués sur les motifs skylines [SRPC11] représentent une base intéressante sur laquelle s'appuyer pour calculer des motifs ayant un bon compromis entre plusieurs mesures.

Bibliographie

[AIS93] Rakesh Agrawal, Tomasz Imieliński, and Arun Swami. Mining association rules between sets of items in large databases. *SIGMOD Rec.*, 22(2) :207–216, June 1993.

[AMS+96] Rakesh Agrawal, Heikki Mannila, Ramakrishnan Srikant, Hannu Toivonen, and A. Inkeri Verkamo. Fast discovery of association rules. In *Advances in Knowledge Discovery and Data Mining*, pages 307–328. AAAI/MIT Press, 1996.

[AS94] Rakesh Agrawal and Ramakrishnan Srikant. Fast algorithms for mining association rules in large databases. In *VLDB*, pages 487–499, 1994.

[Atz15] Martin Atzmueller. Subgroup discovery. *Wiley Interdisc. Rew. : Data Mining and Knowledge Discovery*, 5(1) :35–49, 2015.

[BB00] Jean-François Boulicaut and Artur Bykowski. Frequent closures as a concise representation for binary data mining. In Takao Terano, Huan Liu, and Arbee L. P. Chen, editors, *Knowledge Discovery and Data Mining, Current Issues and New Applications, 4th Pacific-Asia Conference, PADKK 2000, Kyoto, Japan, April 18-20, 2000, Proceedings*, volume 1805 of *Lecture Notes in Computer Science*, pages 62–73. Springer, 2000.

[BBR03] Jean-François Boulicaut, Artur Bykowski, and Christophe Rigotti. Free-sets : A condensed representation of boolean data for the approximation of frequency queries. *Data Min. Knowl. Discov.*, 7(1) :5–22, 2003.

[Ber73] C. Berge. *Graphs and hypergraphs.* North-Holland mathematical library. North-Holland Pub. Co., 1973.

[BHPW10] Mario Boley, Tamas Horvath, Axel Poigne, and Stefan Wrobel. Listing closed sets of strongly accessible set systems with applications to data mining. *THEORETICAL COMPUTER SCIENCE*, 411(3) :691–700, JAN 6 2010.

[BP01] Stephen D. Bay and Michael J. Pazzani. Detecting group differences : Mining contrast sets. *Data Min. Knowl. Discov.*, 5(3) :213–246, 2001.

[CB91] Peter Clark and Robin Boswell. Rule induction with cn2 : Some recent improvements. In Yves Kodratoff, editor, *EWSL*, volume 482 of *Lecture Notes in Computer Science*, pages 151–163. Springer, 1991.

[CC11] Xiaoyun Chen and Jinhua Chen. Emerging patterns and classification algorithms for dna sequence. *JSW*, 6(6) :985–992, 2011.

[CD07] Pádraig Cunningham and Sarah Jane Delany. k-nearest neighbour classifiers, 2007.

[CG07] Toon Calders and Bart Goethals. Non-derivable itemset mining. *Data Min. Knowl. Discov.*, 14(1) :171–206, 2007.

[CH67] Thomas M. Cover and Peter E. Hart. Nearest neighbor pattern classification. *IEEE Trans. Information Theory*, 13(1) :21–27, 1967.

[CN89] Peter Clark and Tim Niblett. The CN2 induction algorithm. *Machine Learning*, 3 :261–283, 1989.

[DB13] Guozhu Dong and James Bailey, editors. *Contrast Data Mining : Concepts, Algorithms, and Applications*. CRC Press, 2013.

[DG06] J. Davis and M. Goadrich. The relationship between precision-recall and roc curves. In *ICML*, 2006.

[DJdR+04] Robert P. W. Duin, Piotr Juszczak, Dick de Ridder, Pavel Paclik, Elzbieta Pekalska, and David M.J. Tax. Prtools, a matlab toolbox for pattern recognition, 2004.

[dJGHM07] María José del Jesús, Pedro González, Francisco Herrera, and Mikel Mesonero. Evolutionary fuzzy rule induction process for subgroup discovery : A case study in marketing. *IEEE Trans. Fuzzy Systems*, 15(4) :578–592, 2007.

[DL99] Guozhu Dong and Jinyan Li. Efficient mining of emerging patterns : Discovering trends and differences. In *KDD*, pages 43–52, 1999.

[DL05] Guozhu Dong and Jinyan Li. Mining border descriptions of emerging patterns from dataset pairs. *Knowl. Inf. Syst.*, 8(2) :178–202, 2005.

[DZWL99] Guozhu Dong, Xiuzhen Zhang, Limsoon Wong, and Jinyan Li. CAEP : classification by aggregating emerging patterns. In *Discovery Science, Second International Conference, DS '99, Tokyo, Japan, December, 1999, Proceedings*, pages 30–42, 1999.

[FGL12] Johannes Fürnkranz, Dragan Gamberger, and Nada Lavrac. *Foundations of Rule Learning*. Cognitive Technologies. Springer, 2012.

[FK96] Michael L. Fredman and Leonid Khachiyan. On the complexity of dualization of monotone disjunctive normal forms. *Journal of Algorithms*, 21(3) :618 – 628, 1996.

[FPSSU96] Usama M. Fayyad, Gregory Piatetsky-Shapiro, Padhraic Smyth, and Ramasamy Uthurusamy, editors. *Advances in Knowledge Discovery and Data Mining*. American Association for Artificial Intelligence, Menlo Park, CA, USA, 1996.

[FR02] Hongjian Fan and Kotagiri Ramamohanarao. An efficient single-scan algorithm for mining essential jumping emerging patterns for classification. In *PAKDD*, pages 456–462, 2002.

[FR03] Hongjian Fan and Kotagiri Ramamohanarao. A bayesian approach to use emerging patterns for classification. In Klaus-Dieter Schewe and Xiaofang Zhou, editors, *ADC*, volume 17 of *CRPIT*, pages 39–48. Australian Computer Society, 2003.

[FR06] Hongjian Fan and Kotagiri Ramamohanarao. Fast discovery and the generalization of strong jumping emerging patterns for building compact and accurate classifiers. *IEEE Trans. Knowl. Data Eng.*, 18(6) :721–737, 2006.

[FWO+11] Gang Fang, Wen Wang, Benjamin Oatley, Brian Van Ness, Michael Steinbach, and Vipin Kumar. Characterizing discriminative patterns. *CoRR*, abs/1102.4104, 2011.

[GDCH12] Salvador Garcia, Joaquin Derrac, José Ramon Cano, and Francisco Herrera. Prototype selection for nearest neighbor classification : Taxonomy and empirical study. *IEEE Trans. Pattern Anal. Mach. Intell.*, 34(3) :417–435, 2012.

[GH06] Liqiang Geng and Howard J. Hamilton. Interestingness measures for data mining : A survey. *ACM Comput. Surv.*, 38(3), September 2006.

[GK03] Bernhard Ganter and Sergei O. Kuznetsov. Hypotheses and version spaces. In *ICCS*, pages 83–95, 2003.

[GL02] Dragan Gamberger and Nada Lavrac. Expert-guided subgroup discovery : Methodology and application. *J. Artif. Intell. Res. (JAIR)*, 17 :501–527, 2002.

[GMB+12] Paul Gleeson, Sandeep Modi, Andreas Bender, Johannes Kirchmair, Malinee Promkatkaew, Supa Hannongbua, and Robert C. Glen. The challenges involved in modeling toxicity data in silico : A review. *Current Pharmaceutical Design*, 18(9) :1266–1291, 2012.

[GSW04] Bernhard Ganter, Gerd Stumme, and Rudolf Wille. Formal concept analysis : Theory and applications. *J. UCS*, 10(8) :926, 2004.

[GTC14] Milton García-Borroto, José Francisco Martínez Trinidad, and Jesús Ariel Carrasco-Ochoa. A survey of emerging patterns for supervised classification. *Artif. Intell. Rev.*, 42(4) :705–721, 2014.

[HBC07] Céline Hébert, Alain Bretto, and Bruno Crémilleux. A data mining formalization to improve hypergraph minimal transversal computation. *Fundam. Inform.*, 80(4) :415–433, 2007.

[HCGdJ11] Francisco Herrera, Cristóbal J. Carmona, Pedro González, and María José del Jesús. An overview on subgroup discovery : foundations and applications. *Knowl. Inf. Syst.*, 29(3) :495–525, 2011.

[HCXY07] Jiawei Han, Hong Cheng, Dong Xin, and Xifeng Yan. Frequent pattern mining : current status and future directions. *Data Min. Knowl. Discov.*, 15(1) :55–86, 2007.

[KC04] Lukasz A. Kurgan and Krzysztof J. Cios. CAIM discretization algorithm. *IEEE Trans. Knowl. Data Eng.*, 16(2) :145–153, 2004.

[KCC15] Bamba Kane, Bertrand Cuissart, and Bruno Crémilleux. Minimal jumping emerging patterns : Computation and practical assessment. In *Advances in Knowledge Discovery and Data Mining - 19th Pacific-Asia Conference, PAKDD 2015, Ho Chi Minh City, Vietnam, May 19-22, 2015, Proceedings, Part I*, pages 722–733, 2015.

[KEN95] Reinhard Kneser, Ute Essen, and Hermann Ney. *On the Use of the Leaving-One-Out Method in Statistical Language Modelling*, pages 256–259. Springer Berlin Heidelberg, Berlin, Heidelberg, 1995.

[Ker92] Randy Kerber. Chimerge : Discretization of numeric attributes. In William R. Swartout, editor, *AAAI*, pages 123–128. AAAI Press / The MIT Press, 1992.

[KGG85] James M. Keller, Michael R. Gray, and James A. Givens. A fuzzy k-nearest neighbor algorithm. *IEEE Transactions on Systems, Man, and Cybernetics*, 15(4) :580–585, 1985.

[KLGK07] Petra Kralj, Nada Lavrac, Dragan Gamberger, and Antonija Krstacic. Contrast set mining through subgroup discovery applied to brain ischaemina data. In *Advances in Knowledge Discovery and Data Mining, 11th Pacific-Asia Conference, PAKDD 2007, Nanjing, China, May 22-25, 2007, Proceedings*, pages 579–586, 2007.

[Klo96] W. Kloesgen. Explora : A multipattern and multistrategy discovery assistant. In *Advances in Knowledge Discovery and Data Mining*, pages 249–271. AAAI, 1996.

[KW10] Lukasz Kobylinski and Krzysztof Walczak. Spatial emerging patterns for scene classification. In *ICAISC (1)*, pages 515–522, 2010.

[LDR00] Jinyan Li, Guozhu Dong, and Kotagiri Ramamohanarao. Making use of the most expressive jumping emerging patterns for classification. In *PAKDD*, pages 220–232, 2000.

[LDRW04] Jinyan Li, Guozhu Dong, Kotagiri Ramamohanarao, and Limsoon Wong. Deeps : A new instance-based lazy discovery and classification system. *Machine Learning*, 54(2) :99–124, 2004.

[Lic13] M. Lichman. Uci machine learning repository, 2013.

[LKFT04a] Nada Lavrac, Branko Kavsek, Peter A. Flach, and Ljupco Todorovski. Subgroup discovery with CN2-SD. *Journal of Machine Learning Research*, 5 :153–188, 2004.

[LKFT04b] Nada Lavrač, Branko Kavšek, Peter A. Flach, and Ljupco Todorovski. Subgroup Discovery with CN2-SD. *Journal of Machine Learning Research*, 5 :153–188, 2004.

[LLW07] Jinyan Li, Guimei Liu, and Limsoon Wong. Mining statistically important equivalence classes and delta-discriminative emerging patterns. In Pavel Berkhin, Rich Caruana, and Xindong Wu, editors, *Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Jose, California, USA, August 12-15, 2007*, pages 430–439. ACM, 2007.

[LPHL+10] Sylvain Lozano, Guillaume Poezevara, Marie-Pierre Halm-Lemeille, Elodie Lescot-Fontaine, Alban Lepailleur, Ryan Bissell-Siders, Bruno Cremilleux, Sylvain Rault, Bertrand Cuissart, and Ronan Bureau. Introduction of Jumping Fragments in Combination with QSARs for the Assessment of Classification in Ecotoxicology. *J. Chem. Inf. Model.*, 50(8) :1330–1339, 2010.

[LS97] Huan Liu and Rudy Setiono. Feature selection via discretization. *IEEE Trans. Knowl. Data Eng.*, 9(4) :642–645, 1997.

[Mit82] Tom M. Mitchell. Generalization as search. *Artif. Intell.*, 18(2) :203–226, 1982.

[MLB+15] Jean-Philippe Métivier, Alban Lepailleur, Aleksey Buzmakov, Guillaume Poezevara, Bruno Crémilleux, Sergei O. Kuznetsov, Jérémie Le Goff, Amedeo Napoli, Ronan Bureau, and Bertrand Cuissart. Discovering structural alerts for mutagenicity using stable emerging molecular patterns. *Journal of Chemical Information and Modeling*, 55(5) :925–940, 2015. PMID : 25871768.

[MT96a] H. Mannila and H. Toivonen. Multiple uses of frequent sets and condensed representations. In *Proceedings of the 2nd international conference on Know-*

ledge Discovery and Data mining (KDD'96), pages 189–194. AAAI Press, August 1996.

[MT96b] Heikki Mannila and Hannu Toivonen. Multiple uses of frequent sets and condensed representations (extended abstract). In Evangelos Simoudis, Jiawei Han, and Usama M. Fayyad, editors, *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (KDD-96), Portland, Oregon, USA*, pages 189–194. AAAI Press, 1996.

[MT97a] Heikki Mannila and Hannu Toivonen. Levelwise search and borders of theories in knowledge discovery. *Data Min. Knowl. Discov.*, 1(3) :241–258, 1997.

[MT97b] Heikki Mannila and Hannu Toivonen. Levelwise search and borders of theories in knowledge discovery. *Data Min. Knowl. Discov.*, 1(3) :241–258, 1997.

[MU13] Keisuke Murakami and Takeaki Uno. Efficient algorithms for dualizing large-scale hypergraphs. In *ALENEX*, pages 1–13, 2013.

[NLW09] Petra Kralj Novak, Nada Lavrac, and Geoffrey I. Webb. Supervised descriptive rule discovery : A unifying survey of contrast set, emerging pattern and subgroup mining. *Journal of Machine Learning Research*, 10 :377–403, 2009.

[PBTL99a] Nicolas Pasquier, Yves Bastide, Rafik Taouil, and Lotfi Lakhal. Discovering frequent closed itemsets for association rules. In Catriel Beeri and Peter Buneman, editors, *Database Theory - ICDT '99, 7th International Conference, Jerusalem, Israel, January 10-12, 1999, Proceedings.*, volume 1540 of *Lecture Notes in Computer Science*, pages 398–416. Springer, 1999.

[PBTL99b] Nicolas Pasquier, Yves Bastide, Rafik Taouil, and Lotfi Lakhal. Efficient mining of association rules using closed itemset lattices. *Inf. Syst.*, 24(1) :25–46, 1999.

[PDP06] Elzbieta Pekalska, Robert P. W. Duin, and Pavel Paclík. Prototype selection for dissimilarity-based classifiers. *Pattern Recognition*, 39(2) :189–208, 2006.

[PPD01] Elzbieta Pekalska, Pavel Paclik, and Robert P. W. Duin. A generalized kernel approach to dissimilarity-based classification. *Journal of Machine Learning Research*, 2 :175–211, 2001.

[RH10] David Rogers and Mathew Hahn. Extended-connectivity fingerprints. *Journal of Chemical Information and Modeling*, 50(5) :742–754, 2010. PMID : 20426451.

[SCR04] Arnaud Soulet, Bruno Crémilleux, and François Rioult. Condensed representation of emerging patterns. In Honghua Dai, Ramakrishnan Srikant, and Chengqi Zhang, editors, *Advances in Knowledge Discovery and Data Mining,*

8th Pacific-Asia Conference, PAKDD 2004, Sydney, Australia, May 26-28, 2004, Proceedings, volume 3056 of *Lecture Notes in Computer Science*, pages 127–132. Springer, 2004.

[Seb96] Michèle Sebag. Delaying the choice of bias : A disjunctive version space approach. In *ICML*, pages 444–452, 1996.

[SGJV12] Richard Sherhod, Valerie J. Gillet, Philip N. Judson, and Jonathan D. Vessey. Automating knowledge discovery for toxicity prediction using jumping emerging pattern mining. *Journal of Chemical Information and Modeling*, 52(11) :3074–3087, 2012. PMID : 23092382.

[SH07] Mondelle Simeon and Robert J. Hilderman. Exploratory quantitative contrast set mining : A discretization approach. In *ICTAI (2)*, pages 124–131. IEEE Computer Society, 2007. 0-7695-3015-X.

[SJH+14] Richard Sherhod, Philip N. Judson, Thierry Hanser, Jonathan D. Vessey, Samuel J. Webb, and Valerie J. Gillet. Emerging pattern mining to aid toxicological knowledge discovery. *Journal of Chemical Information and Modeling*, 54(7) :1864–1879, 2014. PMID : 24873983.

[SR14] Arnaud Soulet and François Rioult. Efficiently depth-first minimal pattern mining. In Vincent S. Tseng, Tu Bao Ho, Zhi-Hua Zhou, Arbee L. P. Chen, and Hung-Yu Kao, editors, *Advances in Knowledge Discovery and Data Mining - 18th Pacific-Asia Conference, PAKDD 2014, Tainan, Taiwan, May 13-16, 2014. Proceedings, Part I*, volume 8443 of *Lecture Notes in Computer Science*, pages 28–39. Springer, 2014.

[SRPC11] Arnaud Soulet, Chedy Raïssi, Marc Plantevit, and Bruno Crémilleux. Mining dominant patterns in the sky. In *11th IEEE International Conference on Data Mining, ICDM 2011, Vancouver, BC, Canada, December 11-14, 2011*, pages 655–664, 2011.

[SSP+12] Iurii Sushko, Elena Salmina, Vladimir A. Potemkin, Gennadiy Poda, and Igor V. Tetko. Toxalerts : A web server of structural alerts for toxic chemicals and compounds with potential adverse reactions. *Journal of Chemical Information and Modeling*, 52(8) :2310–2316, 2012. PMID : 22876798.

[TLY08] Cheng-Jung Tsai, Chien-I Lee, and Wei-Pang Yang. A discretization algorithm based on class-attribute contingency coefficient. *Inf. Sci.*, 178(3) :714–731, 2008.

[TW07] Pawel Terlecki and Krzysztof Walczak. Jumping emerging patterns with negation in transaction databases classification and discovery. *Inf. Sci.*, 177(24) :5675–5690, 2007.

[TW08] Pawel Terlecki and Krzysztof Walczak. Efficient discovery of top-k minimal jumping emerging patterns. In *RSCTC*, pages 438–447, 2008.

[UAUA04] Takeaki Uno, Tatsuya Asai, Yuzo Uchida, and Hiroki Arimura. An efficient algorithm for enumerating closed patterns in transaction databases. In Einoshin Suzuki and Setsuo Arikawa, editors, *Discovery Science, 7th International Conference, DS 2004, Padova, Italy, October 2-5, 2004, Proceedings*, volume 3245 of *Lecture Notes in Computer Science*, pages 16–31. Springer, 2004.

[UBLC15] Willy Ugarte, Patrice Boizumault, Samir Loudni, and Bruno Crémilleux. Modeling and mining optimal patterns using dynamic CSP. In *27th IEEE International Conference on Tools with Artificial Intelligence, ICTAI 2015, Vietri sul Mare, Italy, November 9-11, 2015*, pages 33–40, 2015.

[VvLH92] Henk J.M. Verhaar, Cees J. van Leeuwen, and Joop L.M. Hermens. Classifying environmental pollutants. *Chemosphere*, 25(4) :471 – 491, 1992.

[Wro97] Stefan Wrobel. An algorithm for multi-relational discovery of subgroups. In *PKDD*, pages 78–87, 1997.

[WT05] Tzu-Tsung Wong and Kuo-Lung Tseng. Mining negative contrast sets from data with discrete attributes. *Expert Syst. Appl.*, 29(2) :401–407, 2005.

[YH03] Xiaoxin Yin and Jiawei Han. CPAR : classification based on predictive association rules. In *Proceedings of the Third SIAM International Conference on Data Mining, San Francisco, CA, USA, May 1-3, 2003*, pages 331–335, 2003.

[ZH02] Mohammed Javeed Zaki and Ching-Jiu Hsiao. CHARM : an efficient algorithm for closed itemset mining. In Robert L. Grossman, Jiawei Han, Vipin Kumar, Heikki Mannila, and Rajeev Motwani, editors, *Proceedings of the Second SIAM International Conference on Data Mining, Arlington, VA, USA, April 11-13, 2002*, pages 457–473. SIAM, 2002.

Extraction et sélection de motifs émergents minimaux : application à la chémo-informatique

Résumé : La découverte de motifs est une tâche importante en fouille de données. Ce mémoire traite de l'extraction des motifs émergents minimaux. Nous proposons une nouvelle méthode efficace qui permet d'extraire les motifs émergents minimaux sans ou avec contrainte de support ; contrairement aux méthodes existantes qui extraient généralement les motifs émergents minimaux les plus supportés, au risque de passer à côté de motifs très intéressants mais peu supportés par les objets. De plus, notre méthode prend en compte l'absence d'attribut qui apporte une nouvelle connaissance intéressante.

En considérant les règles associées aux motifs émergents avec un support élevé comme des règles prototypes, on a montré expérimentalement que cet ensemble de règles possède une bonne confiance sur les objets couverts, mais malheureusement ne couvre pas une bonne partie des objets ; ce qui constitue un frein pour leur usage en classification. Nous proposons une méthode de sélection à base de prototypes qui améliore la couverture de l'ensemble des règles prototypes sans pour autant dégrader leur confiance. Au vu des résultats encourageants obtenus, nous appliquons cette méthode de sélection sur un jeu de données chimique ayant rapport à l'environnement aquatique : Aquatox. Cela permet ainsi aux chimistes, dans un contexte de classification, de mieux expliquer la classification des molécules, qui sans cette méthode de sélection serait prédites par l'usage d'une règle par défaut.

Mots-clés : fouille de données ; motifs émergents minimaux ; classification à base de règles ; chimie-informatique ; sélection à base de prototypes ; règles supervisées ; toxicologie prédictive.

Extraction and selection of minimal emerging patterns : application to chemo-informatics

Abstract : Pattern discovery is an important field of Knowledge Discovery in Databases. This work deals with the extraction of minimal emerging patterns. We propose a new efficient method which extracts the minimal emerging patterns with or without constraint of support ; unlike existing methods that typically extract the most supported minimal emerging patterns, at the risk of missing interesting but less supported patterns. Moreover, our method takes into account the absence of attribute that brings a new interesting knowledge.

Considering the rules associated with highly supported emerging patterns as prototype rules, we have experimentally shown that this set of rules has good confidence on the covered objects but unfortunately does not cover a significant part of the objects ; which is a disadvantage for their use in classification. We propose a prototype-based selection method that improves the coverage of the set of the prototype rules without a significative loss on their confidence. We apply our prototype-based selection method to a chemical data relating to the aquatic environment : Aquatox. In a classification context, it allows chemists to better explain the classification of molecules, which, without this method of selection, would be predicted by the use of a default rule.

Keywords : pattern mining ; minimal emerging patterns ; rule-based classification ; prototype-based selection ; supervised rules ; chemoinformatics ; computational toxicology.

Discipline : Informatique

Laboratoire : Laboratoire GREYC - CNRS-UMR 6072, Campus Côte de Nacre, Boulevard du Maréchal Juin, BP 5186, 14032 Caen Cedex