Fouille de Motifs

Notes de cours

Nadjib Lazaar
Université Paris-Saclay
lazaar@lisn.fr

Introduction

La fouille de motifs ensemblistes (ou itemset mining) est une technique essentielle dans le domaine de la fouille de données, visant à découvrir des ensembles d'éléments fréquents ou intéressants dans de grandes bases de données transactionnelles. En analysant les transactions, cette approche permet d'extraire des motifs, c'est-à-dire des combinaisons d'éléments qui apparaissent souvent ensemble, afin de révéler des relations cachées et des tendances. Cette méthode est utilisée dans divers domaines tels que l'analyse de paniers d'achat, la détection de fraude, la recherche de corrélations entre produits ou encore dans des systèmes de recommandation.

1 Exploration de Données (ED) et Découverte de Connaissances dans les Bases de Données (KDD)

L'Exploration de Données (ED), souvent appelée *Découverte de Connaissances dans les Bases de Données (KDD)*, est un domaine multidisciplinaire visant à extraire des motifs, relations et connaissances utiles à partir de grands ensembles de données complexes, pour la prise de décision, la prédiction et l'exploration.

1.1 Définition et Portée

L'ED et le KDD incluent :

— **Exploration des connaissances :** Identifier des structures cachées, tendances et relations dans les données.

— **Développement de processus :** Créer des méthodologies pour extraire des informations utiles.

— **Conception d'algorithmes :** Développer des modèles pour traiter et analyser efficacement les données.
— **Mécanismes de récupération des connaissances :** Créer des outils permettant de découvrir des connaissances dans de grandes bases de données.

1.2 Étapes du processus KDD

Le processus KDD comporte :

1. **Sélection des données :** Identifier les ensembles de données pertinents.
2. **Prétraitement des données :** Nettoyer et transformer les données pour garantir leur qualité.
3. **Transformation des données :** Adapter les données pour l'analyse.
4. **Exploration des données :** Appliquer des algorithmes pour extraire des modèles.
5. **Évaluation et interprétation :** Vérifier la validité des connaissances découvertes.
6. **Présentation des connaissances :** Communiquer les résultats de manière compréhensible.

1.3 Applications de l'ED

L'ED est utilisée dans de nombreux domaines :
— **Intelligence d'affaires :** Segmentation, détection de fraudes, analyse de paniers d'achat.
— **Santé :** Prédiction de maladies, découverte de médicaments.
— **Recherche scientifique :** Analyse de données en génomique, astronomie, sciences des matériaux.
— **Réseaux sociaux :** Analyse des sentiments, détection de tendances.
— **Cybersécurité :** Détection des intrusions et anomalies.

1.4 Défis de l'ED et du KDD

Les principaux défis sont :
— **Complexité des données :** Gestion des données bruyantes ou incomplètes.
— **Passage à l'échelle :** Analyse de grands ensembles de données.
— **Interprétabilité :** Rendre les modèles compréhensibles pour les non-experts.
— **Éthique :** Respect de la confidentialité et réduction des biais dans les décisions.
— **Interaction humaine :** Intégrer les retours des utilisateurs dans le processus de découverte.

L'ED et le KDD sont essentiels dans la science des données, permettant de transformer les données brutes en connaissances exploitables pour prendre des décisions éclairées.

2 Extraction de Motifs Fréquents

L'Extraction de Motifs Fréquents (FIM : Frequent Itemset Mining), introduite par Agrawal et al., vise à identifier des motifs récurrents dans les données, révélant des structures utiles en analyse de marché, bioinformatique et sécurité des réseaux.

Un concept clé est la découverte des **itemsets fréquents**, ensembles d'éléments coapparaissant fréquemment dans des transactions. Ce domaine a évolué avec plusieurs extensions majeures :

- **Itemsets fermés :** Motifs fréquents clos ou maximaux évitant la redondance.
- **Règles d'association :** Relations de type "si-alors" entre éléments.
- **Itemsets rares :** Combinations peu fréquentes mais pertinentes (fraude, maladies rares).
- **Motifs séquentiels :** Capturent l'ordre des événements (comportement client, génomique).
- **Motifs émergents :** Identifient les tendances évolutives des données.

FIM reste un pilier de l'exploration de données, avec des avancées continues en efficacité algorithmique et en applications pratiques.

3 Notations et Définitions

Dans cette section, nous introduisons les concepts fondamentaux liés à la fouille d'itemsets fréquents, en définissant les principales notions et notations utilisées.

3.1 Jeu de données transactionnel et Itemsets

Définition 1 (Itemset et ensemble d'items)**.** *Soit $\mathcal{I} = \{p_1, \ldots, p_n\}$ un ensemble de n objets distincts, appelés* items*. Un* itemset *P est défini comme un sous-ensemble non vide de $\mathcal{I}$:*

$$P \subseteq \mathcal{I}.$$

Définition 2 (Jeu de données transactionnel)**.** *Un* jeu de données transactionnel *$\mathcal{D}$ est une collection de m transactions $t_1, \ldots, t_m$, où chaque transaction est un sous-ensemble d'items, c'est-à-dire :*

$$t_i \subseteq \mathcal{I}, \quad \forall i \in \{1, \ldots, m\}.$$

Example 1. *Considérons un jeu de données transactionnel $\mathcal{D}_1$ contenant 5 transactions et 5 items : Par exemple, $\{A, B\}$ et $\{B, C, E\}$ sont des itemsets dans $\mathcal{D}_1$.* ♢

TABLE 1 – Exemple de jeu de données transactionnel ($\mathcal{D}_1$).

Transaction	Items
t_1	A B D E
t_2	A C
t_3	A B C E
t_4	B C E
t_5	A B C E

3.2 Couverture et Fréquence d'un Itemset

Définition 3 (Couverture d'un itemset)**.** *La* couverture *d'un itemset P dans $\mathcal{D}$, notée* $\texttt{cover}(P)$*, est l'ensemble des transactions contenant P :*

$$\texttt{cover}(P) = \{t_i \in \mathcal{D} \mid P \subseteq t_i\}.$$

Définition 4 (Fréquence d'un itemset)**.** *La* fréquence *d'un itemset P dans $\mathcal{D}$, notée* $\texttt{freq}(P)$*, est le nombre de transactions dans lesquelles P apparaît :*

$$\texttt{freq}(P) = |\texttt{cover}(P)|.$$

Example 2. *Dans $\mathcal{D}_1$, considérons l'itemset $P = \{A, B\}$:*
- *Il apparaît dans les transactions t_1, t_3 et t_5.*
- *Donc, sa couverture est* $\texttt{cover}(P) = \{t_1, t_3, t_5\}$.
- *Sa fréquence est* $\texttt{freq}(P) = 3$.

♢

3.3 Itemsets Fréquents et Rares

Définition 5 (Itemset Fréquent et Rare)**.** *Étant donné un seuil de fréquence minimale α :*
- *Un itemset P est* fréquent *si* $\texttt{freq}(P) \geq \alpha$.
- *Un itemset P est* rare *(ou peu fréquent) si* $\texttt{freq}(P) < \alpha$.

Nous notons $\texttt{FIs}_\alpha$ et $\texttt{RIs}_\alpha$ respectivement les ensembles des itemsets fréquents et rares par rapport à α :

$$\texttt{FIs}_\alpha = \{P \mid \texttt{freq}(P) \geq \alpha\}, \quad \texttt{RIs}_\alpha = \{P \mid \texttt{freq}(P) < \alpha\}.$$

Example 3. *Si $\alpha = 3$, alors dans $\mathcal{D}_1$:*
- *L'itemset $\{A, B\}$ est fréquent car* $\texttt{freq}(\{A, B\}) = 3$.
- *L'itemset $\{D\}$ est rare car* $\texttt{freq}(\{D\}) = 1$.

♢

3.4 Itemsets Maximaux et Minimaux

Définition 6 (Itemset Maximal et Minimal)**.** *Un itemset P est :*

— Maximal *s'il est fréquent et qu'aucun de ses supersets n'est fréquent :*

$$P \in \texttt{MaxIs}_\alpha \iff \texttt{freq}(P) \geq \alpha \text{ et } \forall Q \supset P, \texttt{freq}(Q) < \alpha.$$

— Minimal *s'il est rare et qu'aucun de ses sous-ensembles n'est rare :*

$$P \in \texttt{MinIs}_\alpha \iff \texttt{freq}(P) < \alpha \text{ et } \forall Q \subset P, \texttt{freq}(Q) \geq \alpha.$$

Example 4. *Si $\alpha = 3$, dans $\mathcal{D}_1$:*
— *$\{A, B, E\}$ est un itemset maximal car il est fréquent, mais aucun de ses supersets n'est fréquent.*
— *$\{A, C, E\}$ est un itemset minimal car il est rare et aucun sous-ensemble plus petit n'est rare.*

$\diamondsuit$

3.5 Itemsets Fermés et Générateurs

Définition 7 (Itemset Fermé et Générateur)**.** *Un itemset P est :*
— Fermé *si aucun de ses supersets n'a la même fréquence :*

$$P \in \texttt{CIs}_\alpha \iff \nexists Q \supset P, \texttt{freq}(Q) = \texttt{freq}(P).$$

— Générateur *si aucun de ses sous-ensembles n'a la même fréquence :*

$$P \in \texttt{GIs}_\alpha \iff \nexists Q \subset P, \texttt{freq}(Q) = \texttt{freq}(P).$$

Example 5. *Dans $\mathcal{D}_1$ avec $\alpha = 3$:*
— *$\{B, E\}$ est fermé car aucun de ses supersets n'a la même fréquence.*
— *$\{B, C\}$ est générateur car aucun de ses sous-ensembles n'a la même fréquence.*

$\diamondsuit$

3.6 Résumé des Relations entre Itemsets

— Tout itemset maximal est un itemset fermé, mais l'inverse n'est pas vrai.
— Tout itemset minimal est un itemset générateur, mais l'inverse n'est pas vrai.
— Chaque itemset fréquent possède un superset fermé avec la même couverture :

$$\forall P \in \texttt{FIs}_\alpha, \quad \texttt{cover}(P) = \max_{Q \in \texttt{CIs}_\alpha, Q \subseteq P} \texttt{cover}(Q).$$

— Chaque itemset rare possède un sous-ensemble générateur avec la même couverture :

$$\forall P \in \texttt{RIs}_\alpha, \quad \texttt{cover}(P) = \min_{Q \in \texttt{GIs}_\alpha, Q \supseteq P} \texttt{cover}(Q).$$

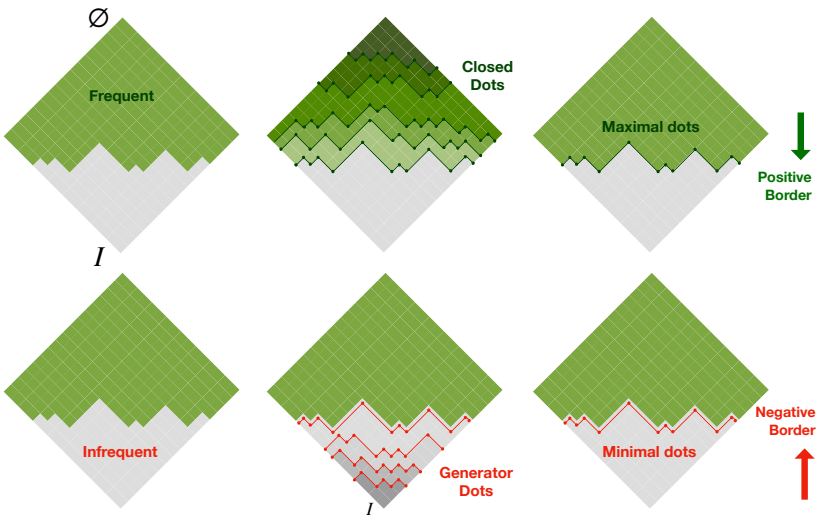

FIGURE 1 – Itemsets Fréquents/Peu Fréquents, Fermés/Générateurs et Maximaux/Minimaux.

4 Anti-monotonie et Propriété Apriori

La fouille des itemsets fréquents repose sur deux principes clés : l'anti-monotonie et la propriété Apriori. Ces propriétés permettent de réduire efficacement l'espace de recherche en éliminant rapidement les itemsets non pertinents.

4.1 Anti-monotonie

L'anti-monotonie est une propriété essentielle qui facilite l'exploration des itemsets fréquents.

Définition 8 (Anti-monotonie)**.** *Un itemset P est* anti-monotone *si, pour tout itemset $Q \supseteq P$, on a :*

$$\texttt{freq}(Q) \leq \texttt{freq}(P).$$

Cela signifie qu'un itemset ne peut jamais avoir une fréquence supérieure à celle de ses sous-ensembles. Une conséquence importante est que si un itemset est rare, alors tous ses supersets le seront aussi. Cette propriété est utilisée pour éliminer rapidement les candidats non fréquents et éviter des calculs inutiles.

Example 6. *Dans un jeu de données transactionnel, supposons que* $\texttt{freq}(\{A, B\}) = 2$*. Alors, tout itemset contenant $\{A, B\}$ (comme $\{A, B, C\}$ ou $\{A, B, D\}$) aura une fréquence inférieure ou égale à 2.* ♢

4.2 Propriété Apriori

La propriété Apriori découle directement de l'anti-monotonie : un itemset fréquent ne peut contenir de sous-ensembles rares.

Définition 9 (Propriété Apriori)**.** *Si un itemset P est fréquent, alors tous ses sous-ensembles le sont aussi :*

$$\forall S \subseteq P, \quad \texttt{freq}(P) \geq \alpha \Rightarrow \texttt{freq}(S) \geq \alpha.$$

Cette propriété permet d'explorer les itemsets de manière progressive : on commence par identifier les itemsets fréquents de petite taille, puis on construit des candidats plus grands uniquement à partir d'itemsets déjà validés.

Example 7. *Si $\{A, B, C\}$ est fréquent, alors $\{A, B\}$ et $\{B, C\}$ doivent aussi être fréquents. Inversement, si $\{A, B\}$ est rare, alors $\{A, B, C\}$ ne peut pas être fréquent.* ♢

4.3 L'algorithme Apriori

L'algorithme Apriori est l'un des algorithmes les plus connus utilisés pour l'exploration des itemsets fréquents. Il utilise la propriété Apriori pour générer efficacement les itemsets candidats et élaguer ceux qui sont peu fréquents. L'idée de base de l'algorithme Apriori est une recherche en largeur où les itemsets candidats sont générés en joignant les itemsets fréquents du niveau précédent, et les itemsets peu fréquents sont élagués à l'aide de la propriété d'anti-monotonie.

L'algorithme fonctionne comme suit :

1. **Initialisation :** Commencez avec l'ensemble de tous les itemsets fréquents de taille 1, qui sont les items apparaissant dans au moins le seuil minimal de transactions.
2. **Processus itératif :**
   — Générez les itemsets candidats de taille k en joignant les itemsets fréquents de taille $k-1$.
   — Pour chaque itemset candidat, scannez le jeu de données et calculez sa fréquence.
   — Élaguez les candidats peu fréquents (c'est-à-dire, ceux dont la fréquence est inférieure au seuil minimal α).
3. **Répétez le processus :** Répétez le processus jusqu'à ce qu'aucun autre itemset fréquent ne puisse être trouvé.
4. **Générer des règles :** Une fois tous les itemsets fréquents trouvés, générez les règles d'association à partir des itemsets fréquents.

Algorithme 1 : Algorithme Apriori

Input : Base de données transactionnelles $\mathcal{D}$, seuil minimum de support α

Output : Itemsets fréquents

$k \leftarrow 1$;

$L_k \leftarrow \{p_i \mid p_i \in \mathcal{I} \wedge \texttt{freq}(p_i) \geq \alpha\}$;

while $L_k \neq \emptyset$ **do**

  $C \leftarrow \texttt{aprioriGen}(L_k)$;

  $k \leftarrow k + 1$;

  $L_k \leftarrow \{c \mid c \in C \wedge \texttt{freq}(c) \geq \alpha\}$;

return $\bigcup_i L_i$;

Fonction `aprioriGen`(L_k) **:**

  $E \leftarrow \emptyset$;

  for *chaque paire d'itemsets* $P', P'' \in L_k$ *tels que* $P' = \{p_{i_1}, \ldots, p_{i_{k-1}}, p_{i_k}\}$ *et* $P'' = \{p_{i_1}, \ldots, p_{i_{k-1}}, p_{i'_k}\}$ **do**

    if $p_{i_k} \neq p_{i'_k}$ **then**

      $P \leftarrow P' \cup P''$;

      if $\forall p_i \in P,\ P \setminus \{p_i\} \in L_k$ **then**

        $E \leftarrow E \cup \{P\}$;

  return E ;

Les étapes clés de l'algorithme Apriori consistent à générer itérativement des itemsets candidats, scanner le jeu de données pour déterminer leurs fréquences, et élaguer les itemsets peu fréquents en fonction de la propriété Apriori. Ce processus est répété jusqu'à ce qu'aucun autre itemset fréquent ne puisse être trouvé, après quoi les règles d'association peuvent être générées à partir des itemsets fréquents.