

EMD de Data Mining

Exercice1. (8 pts)

Une base de données a quatre transactions. Supposer que $\min \text{sup} = 60\%$ et $\min \text{conf} = 80\%$.

client ID	TID	Articles vendus (de la forme of marque-article-catégorie)
01	T100	{Pêcherie-Poisson, Hodna-lait, Soumam-fromage, Meilleur-Pain}
02	T200	{Meilleur-fromage, ONA-lait, Ferme-pomme, Cherchell-biscuit, Bon-pain}
01	T300	{Ouest-pomme, ONA-lait, Bon-pain, Cherchell-biscuit}
03	T400	{Bon-pain, Hodna-lait, ONA-fromage}

- 1) En considérant la granularité de la catégorie de l'article où un exemple d'item peut être lait, pour la règle suivante :

$$\forall X \in \text{transaction}, \text{achète}(X, \text{item1}) \wedge \text{achète}(X, \text{item2}) \Rightarrow \text{achète}(X, \text{item3}) [s, c]$$

Enumérer les k -itemset fréquents pour la plus grande valeur de k , et toutes les règles d'association (avec leur support s et confiance c) contenant le k -itemset fréquent pour le plus grand k (5 pts)

- 2) En considérant la granularité de la catégorie de la marque de l'article où un exemple d'item peut être Hodna-lait, pour la règle suivante :

$$\forall X \in \text{client}, \text{achète}(X, \text{item1}) \wedge \text{achète}(X, \text{item2}) \Rightarrow \text{achète}(X, \text{item3}) [s, c]$$

Enumérer les k -itemset fréquents pour la plus grande valeur de k . (3 pts)

Exercice2. (12 pts)

La plupart des algorithmes de fouille de motifs fréquents ne considèrent que des articles distincts dans une transaction. Cependant en réalité, des occurrences multiples d'un même article dans un panier, telles que quatre bouteilles de jus et trois cartons de laits peuvent être importantes pour l'analyse des données transactionnelles.

- Comment peut-on extraire des motifs fréquents en considérant plusieurs occurrences des articles ? (4 pts)
- Rappeler l'algorithme Apriori (4 pts)
- Proposer des modifications sur l'algorithme Apriori pour répondre à la première question (4 pts)

Exercice 1:

1) 4 transactions supports 60% donc $60\% \times 4 = 2,4 \Rightarrow 3$ trans

{Poisson}	1
{lait}	4
{fromage}	3
{pain}	4
{fromage}	2
{biscuit}	2



{lait}	4
{fromage}	3
{pain}	4

{lait, fromage}	3
{lait, pain}	4
{fromage, pain}	3



{lait, fromage, pain}	3
-----------------------	---

Confiance = 80% $80\% \times 4 = 3,2$

4 transactions / 4

Achète (lait) \wedge Achète (x, fromage) \Rightarrow Achète (x, pain) [75%, 100%]

Achète (x, fromage) \wedge Achète (x, pain) \Rightarrow Achète (x, lait) [75%, 100%]

9)

sup = 60%

nb client = 3. \Rightarrow sup = $60\% \times 3 = 1,8$
 $\lfloor 1,8 \rfloor = 1$ sur 3 clients
 0,5

Item	Support
hoda-lait	1
ova-lait	2
chardell-biscuit	1
bon-pain	1
hoda-lait	1
ova-lait	1
chardell-biscuit	1
bon-pain	1
hoda-lait	1
ova-lait	1
chardell-biscuit	1
bon-pain	1

{hoda-lait}	2
{ova-lait}	2
{chardell-biscuit}	2
{bon-pain}	3

0,5

{hoda-lait, ova-lait}	1
{hoda-lait, chardell-biscuit}	1
{hoda-lait, bon-pain}	2
{ova-lait, chardell-biscuit}	2
{ova-lait, bon-pain}	2
{chardell-biscuit, bon-pain}	2

0,5

{hoda-lait, bon-pain}	2
{ova-lait, chardell-biscuit}	2
{ova-lait, bon-pain}	2
{chardell-biscuit, bon-pain}	2

0,5

{ova-lait, bon-pain, chardell-biscuit}	2
--	---

0,5

Réponse

k = 3

3-itemset = {ova-lait, bon-pain, chardell-biscuit}

0,5

Exercice 2

- 1) $C_1 \rightarrow$ associer à chaque 1-item son nombre d'occurrences
- 2) $C_{k+1} \rightarrow$ joindre de L_k
 - associer à chaque $k+1$ -item, le minimum des occurrences des k -items joints.
- 3) Arrêter le processus jusqu'à la fin.

Ex-pte : Table des transactions : $T = \{t_1, t_2, \dots, t_n\}$

$$T = \{(\text{item}_1, \text{occurrence}_1), (\text{item}_2, \text{occurrence}_2), \dots, (\text{item}_m, \text{occurrence}_m)\}$$

2) Algorithme Apriori

Algorithme Apriori T

Notations : C_k : itemsets candidats de taille k

L_k : itemsets fréquents de taille k

Input : $T = \{t_1, \dots, t_n\}$ $t_i = \{\text{item}_1, \dots, \text{item}_m\}$

Output : $L = L_1 \cup L_2 \cup \dots \cup L_k$

Notations : $C_1 = \{ \text{candidats de taille 1} \}$
 $L_1 = \{ \text{items fréquents} \}$

Pour $(k=1; C_k \neq \emptyset, k++)$ faire

Notations : $C_{k+1} =$ candidats engendrés à partir de L_k

Pour chaque transaction t faire

incrémenter le compteur de tous les candidats

de C_{k+1} qui sont contenus dans t

Notations : $L_{k+1} =$ candidats de C_{k+1} avec support minimum

Retourner $\bigcup_k L_k$