Exercice 4
Extraction d'Items Fréquents et règles d'associations et de corrélations

Format des données

On dit que D est un ensemble de données pertinentes pour la tâche d'extraction d'Items Fréquents, lorsqu'il regroupe un ensemble de transactions $\boldsymbol{T}$ = {$\boldsymbol{T}_1$, $\boldsymbol{T}_2$, … $\boldsymbol{T}_n$} de base de données où chaque transaction $\boldsymbol{T}_i$ est un ensemble d'éléments non vide comportant un certain nombre d'Item de $\boldsymbol{I}$, avec $\boldsymbol{I}$ = {$\boldsymbol{I}1$, $\boldsymbol{I}2$, $\boldsymbol{I}3$, … $\boldsymbol{I}m$}.
Les données d'un dataset ne sont pas toujours formaté de manière adéquate pour l'extraction d'Items Fréquents. C'est pourquoi, il est parfois nécessaire de détecter ce qui représente une Transaction et ce que représente une Item dans notre Dataset. Ainsi, le dataset devra subir des changements afin de regrouper ensemble les Items de chaque transaction avant l'extraction d'une quelconque information du dataset.

Questions :
1- Étudier le jeu de données "Dataset2", et en déduire ce que représente une Transaction et ce que représente un Item.
2- Donnez le nombre de Transaction et le nombre d'Item de ce dataset.
3- Effectuer les prétraitements nécessaires (traitement des valeurs manquantes, aberrantes…)
3- Construire un dataset "Dataset2_bis" à partir du dataset "Dataset2" ayant le bon format transactionnel nécessaire à l'extraction d'Items Fréquents.

Support: *supp_min*

Le support du 1-itemset $\boldsymbol{I}1$ = Pourcentage de transactions contenant l'item $\boldsymbol{I}1$.
= nombre de Transaction dans lesquelles apparaît l'item $\boldsymbol{I}1$ / nombre de transaction de *D*.
Exemple : $\boldsymbol{I}1$ apparaît dans $\boldsymbol{T}1$, $\boldsymbol{T}5$, $\boldsymbol{T}12$, et $\boldsymbol{T}51$ Alors Support($\boldsymbol{I}1$) = 4.

Le support du 2-itemset {$\boldsymbol{I}1$, $\boldsymbol{I}2$} = Pourcentage de transactions contenant l'item $\boldsymbol{I}1$ et $\boldsymbol{I}2$.
= nombre de Transaction dans lesquelles apparaissent les items $\boldsymbol{I}1$ et $\boldsymbol{I}2$ (en même temps) / nombre de transaction de *D*.
Exemple : $\boldsymbol{I}1$ et $\boldsymbol{I}2$ apparaissent dans $\boldsymbol{T}1$, $\boldsymbol{T}12$, et $\boldsymbol{T}51$ Alors Support({$\boldsymbol{I}1$, $\boldsymbol{I}2$}) = 3.

… et ainsi de suite.

Belkadi Widad Hassina (G1-G2): belkadi.wh.usthb@gmail.com
Khelfa Celia (G3-G4): khelfacelia99@gmail.com

A chaque Itération de l'algorithme Apriori, une liste de k-itemset candidats C_k est construite. Et à partir de chaque C_k une liste des k-itemsets fréquents L_k est créée, en ne gardant que les k-itemset de C_k ayant un support >= *supp_min (variable à fixer)*.

- **C_1 est la liste des 1-itemsets candidats.**
- Génération de C1 à travers le listing de tous les Items.
- **L_1 est la liste des 1-itemsets fréquents de C_1.**
- Calcule du support des tous les éléments de C_1 à partir du dataset de base *D*.
- Copier uniquement les éléments ayant un support >= *supp_min*.

- **C_2 est la liste des 2-itemsets candidats.**
- Génération de C2 à travers une opération de jointure:

  Pour chaque item I_i de L1
  Pour chaque item I_j de L1 ($I_i <> I_j$)
  Ajouter { I_i , I_j } à C2 ;
  Fait;
  Fait;
- **L_2 est la liste des 2-itemsets fréquents de C_2.**
- Calcule du support des tous les éléments de C_2 à partir du dataset de base *D*.
- Copier uniquement les éléments ayant un support >= *supp_min*.

… et ainsi de suite.

Questions :

1- Écrire une fonction python permettant de générer les k-itemsets candidats C_k.
2- Écrire une fonction python permettant de calculer le support des k-itemsets C_k.
3- Écrire une fonction python permettant de générer les k-itemsets fréquents L_k.

Confiance: *conf_min*

Soient A et B des k-itemsets et A ⇒ B une règle d'association.
Exemple: A = {$I1$,$I2$} et B = {$I3$}. La règle : {$I1$, $I2$} ⇒ {$I3$}.
Confiance(A ⇒ B) = Le pourcentage de transactions dans *D* contenant A qui contiennent également B. Il s'agit de la probabilité conditionnelle, P(B/A).

$$\text{Confiance } (A \Rightarrow B) = P(B/A) = \frac{Support\ (A\ U\ B)}{Support\ (A)}$$

Belkadi Widad Hassina (G1-G2): belkadi.wh.usthb@gmail.com
Khelfa Celia (G3-G4): khelfacelia99@gmail.com

Questions :

1- Écrire une fonction python permettant de générer toutes les règles d'association d'un L_k.

2- Écrire une fonction python permettant de calculer la confiance d'une règle d'association.

Lien vers dataset 2 :

https://drive.google.com/file/d/1ES8iL_ujc7FB4wMGZHD6GImH5tPCDuP2/view?usp=drive_link

Bon courage.

Belkadi Widad Hassina (G1-G2): belkadi.wh.usthb@gmail.com
Khelfa Celia (G3-G4): khelfacelia99@gmail.com