Examen final

Apprentissage statistique - clustering, classification, règles d'association

le 8 décembre 2020, durée 1h30

L'ordre de résolution des sujets n'est pas imposé. Les sujets sont indépendants.

1. Pour quoi deux algorithmes de clustering peuvent fournir des résultats différents sur un ensemble ? Est-ce donc possible qu'un algorithme qui s'exécute deux fois sur un même ensemble donne des résultats très différents ? que pouvez vous dire sur l'algorithme en question ?
2. L'algorithme DBCAN a besoin de deux paramètres :
   ε - le distance maximale pour considérer que deux éléments sont proches
   $MinPoint$ - le nombre minimum de voisins pour considérer qu'un élément n'est pas isolé.
   Pour un choix des valeurs ε_0 et $MinPoint_0$ l'algorithme détermine lors d'une première exécution sur un ensemble S :
   — un nombre de j points isolés
   — un nombre de k clusters ; pour fixer les idées on peut nommer les clusters C_1, C_2, .. C_k avec $dimension(C_1) > dimension(C_2) > ... > dimension(C_K)$.
   On exécute ensuite trois fois cet algorithme sur l'ensemble S et on se pose la question si le résultat fourni est différent de la solution calculée la première fois :
   (a) on ne change pas les paramètres ε_0 et $MinPoint_0$
   (b) on garde inchangé ε_0 et on prend $MinPoint_1 = MinPoint_0 - 1$
   (c) on garde inchangé $MinPoint_0$ et on augmente ε : $\varepsilon_2 > \varepsilon_0$
3. Certaines librairies qui implémentent le calcul de règles d'association (RA) affichent aussi un autre indicateur : *coverage* (la couverture)

qui est le support de la partie gauche de la règle d'association. Deux RA avec la même partie gauche, chacune un support supérieur un $min\_support$ et une confiance dépassant $min\_confiance$ ont, naturellement, une même valeur de *coverage*.

— si on a deux RA $X \rightarrow Y$ et $X' \rightarrow Y$ avec $X \subset X'$, que peut-on affirmer de la relation entre $coverage(X->Y)$ et $coverage(X'->Y)$; entre $LIFT(X->Y)$ et $LIFT(X'->Y)$?

— voyez vous une utilité à cet indicateur ?

4. Soit le petit ensemble de transactions suivant :

	items	transactionID
[1]	{item2,item4}	trans1
[2]	{item1,item4}	trans2
[3]	{item1,item2,item5}	trans3
[4]	{item1,item2,item3,item4,item5}	trans4
[5]	{item1,item5}	trans5
[6]	{item2,item3,item5}	trans6
[7]	{item3}	trans7

(a) à l'aide de l'algorithme Apriori ou de l'algorithme ECLAT calculez tous les itemsets fréquents pour un $min\_support = \frac{2}{7}$ (2/7 inclus).

(b) déterminez ensuite les règles d'association qui font apparaître `item1` dans la partie gauche ou droite et qui ont un support supérieur ou égal à 2/7 et une confiance plus grande que 8/10.