Université de Sherbrooke
Département d'informatique

IFT870/BIN710
Forage de données / Forage de données pour la bio-informatique

Hiver 2020

Examen final

Professeure :
Aïda Ouangraoua

À remettre **avant le jeudi 16 avril à 23h59**
sur **opus.dinf.usherbrooke.ca**

Cet examen est à faire de façon individuelle. Lors de la correction, la note zéro (0) sera attribuée à tout travail pour lequel une preuve de plagiat est attestée. Pour la soumission du travail, se connecter dans un fureteur au serveur http://opus.dinf.usherbrooke.ca, puis choisir le cours IFT870 (BIN710) et le projet ExamenFinal. Charger le fichier examenfinal.ipynb et le soumettre. Le nom du fichier de remise doit être exactement examenfinal.ipynb.

Cet examen comporte 5 questions (25 points, 20 points, 25 points, 20 points, 10 points) sur 3 pages.

DÉBUT

Nous avons vu en cours quatre principaux critères de qualité pour l'évaluation de clustering en utilisant des méthodes intrinsèques (qui ne nécessitent pas de connaître les clusters réels) : cohésion, séparation, connectivité, et robustesse. Plusieurs méthodes d'évaluation intrinsèques basées sur les critères de cohésion, séparation, et connectivité existent dans la bibliothèque scikit-learn, mais aucune méthode basée sur le critère de robustesse.

Dans la stratégie d'évaluation basée sur la robustesse, on distingue deux approches pour évaluer le résultat d'un modèle de clustering :

1. Faire varier la valeur des paramètres de l'algorithme et évaluer la similarité entre les résultats.
2. Ajouter du bruit aux données et évaluer la similarité entre les résultats.

Vous devez implémenter des méthodes d'évaluation intrinsèques basées sur ces deux approches pour les modèles KMeans, AgglomerativeClustering et DBSCAN. Pour tester vos méthodes d'évaluation, vous utiliserez l'ensemble de données utilisé pour le TP#4 : ensemble de données d'images de visages de personnes connues (Labeled Faces in the Wild), limité à 40 images maximum par personne (1916 données de dimension 5655)

Étant donnés N clusterings $C_1,\ldots,C_N$ obtenus en faisant varier les paramètres d'un modèle, ou en ajoutant du bruit aux données, le score de robustesse *R* est calculé comme suit :

$$R = \frac{\sum_{(i,j)\in P} \#(i,j)}{|P| * N}$$

tel que *P* est l'ensemble des paires d'objets appartenant au même cluster dans au moins un des clustering $C_1,\ldots,C_N$; | *P* | est la taille de *P ;* Pour une paire (*i,j*) dans *P,* #(*i,j*) est le nombre de clusterings dans lesquels *i,j* appartiennent au même cluster.

Question 1 : Robustesse aux changement de paramètres d'un modèle KMeans ou AgglomerativeCLustering (25 pts)

Écrivez **une fonction prenant en paramètre une instance de la classe KMeans ou de la classe AgglomerativeClustering**, et retournant la robustesse de cette instance, calculée comme suit :
Faire varier uniquement le paramètre n_clusters de l'instance en lui additionnant les valeurs [-5,-4,-3,-2,-1,0,1,2,3,4,5]. Pour chaque valeur du paramètre n_clusters, entraîner le modèle et prédire un clustering. Calculer le score de robustesse *R* correspondant aux 11 clusterings obtenus.

Calculer la robustesse des modèles : KMeans(n_clusters=k, random_state=0) et AgglomerativeClustering(n_clusters=k) pour k = 40, 60 ou 80. Quel est le modèle le plus robuste suivant le score *R ?*

Question 2 : Robustesse aux changement de paramètres d'un modèle DBSCAN (20 pts)

Écrivez **une fonction prenant en paramètre une instance du modèle DBSCAN**, et retournant la robustesse de cette instance, calculée comme suit :
Faire varier uniquement le paramètre eps de l'instance en lui additionnant les valeurs [-0.5,-0.4,-0.3,-0.2,-0.1,0,0.1,0.2,0.3,0.4,0.5]. Pour chaque valeur du paramètre eps, entraîner le modèle et prédire un clustering. Calculer le score de robustesse *R* correspondant aux 11 clusterings obtenus.

Calculer la robustesse des modèles : DBSCAN(min_samples =3, eps=e) pour k = 7, 8 ou 9. Quel est le modèle le plus robuste suivant le score *R ?*

Question 3 : Robustesse à l'ajout de bruit d'un modèle KMeans ou AgglomerativeCLustering (25 pts)

Écrivez **une fonction prenant en paramètres une instance de la classe KMeans ou de la classe AgglomerativeClustering et un entier X de valeur comprise entre 0 et 100 représentant un pourcentage**, et retournant la robustesse de cette instance, calculée comme suit :
Générer aléatoirement 10 ensembles contenant chacun X*1960/100 données (bruit) de la même forme que les données utilisées (5655 dimensions) suivant la loi normale $N(\mu, \sigma^2)$ pour chaque dimension telle que μ est la moyenne de la dimension et σ^2 sa variance (utiliser numpy.random.randn par exemple). Le 11^e ensemble de bruit est vide. Faire varier les données en leur ajoutant à chaque itération un des ensembles de bruit générés. Pour chaque itération, entraîner le modèle et prédire un clustering. Calculer le score de robustesse *R* correspondant aux 11 clusterings obtenus.

Calculer la robustesse des modèles : KMeans(n_clusters=k, random_state=0) et AgglomerativeClustering(n_clusters=k) pour k = 40, 60 ou 80, pour une valeur X = 5. Quel est le modèle le plus robuste suivant le score *R ?*

Question 4 : Robustesse aux changement de paramètres d'un modèle DBSCAN (20 pts)

Écrivez **une fonction prenant en paramètre une instance du modèle DBSCAN et un entier X de valeur comprise entre 0 et 100 représentant un pourcentage**, et retournant la robustesse de cette instance, calculée comme suit :
Générer aléatoirement 11 ensembles de bruit (dont 1 vide) comme indiqué à la Question 3. Faire varier les données en leur ajoutant à chaque itération un des ensembles de bruit. Pour chaque itération, entraîner le modèle et prédire un clustering. Calculer le score de robustesse *R* correspondant aux 11 clusterings obtenus.

Calculer la robustesse des modèles : DBSCAN(min_samples =3, eps=e) pour k = 7, 8 ou 9, pour une valeur X = 5. Quel est le modèle le plus robuste suivant le score *R ?*

Question 5 : Modèle pour la génération du bruit (10 pts)

Critiquez le modèle utilisé pour générer le bruit dans les Questions 3 et 4. Proposez un autre modèle de bruit avec une justification du modèle.

FIN