TP Clustering
5e ISS

MJ. HUGUET
homepages.laas.fr/huguet

Objectifs

Le but de ces TP est mettre en oeuvre et de comparer différents algorithmes de clustering tout d'abord à partir de quelques méthodes fournies par `scikit-learn`.

- k-Means
- clustering hiérarchique (agglomératif)
- `DBSCAN`

Encadrants

Marie-José Huguet, Laure Feuillet, Hieu Trong Tran

1 Environnement de travail

Durant ces TP, nous allons utiliser le langage Python et la distribution Anaconda (installée à l'INSA, **uniquement sous Linux**). Cette distribution fournit la librairie Python `scikit-learn` contenant un certain nombre de méthodes d'analyse de données et d'apprentissage.
Documentation : `https://scikit-learn.org/stable/`

2 Jeux de données

Nous allons utiliser des jeux de données disponibles et visualisables sur le site `https://github.com/deric/clustering-benchmark`. Seuls les jeux de données "artificiels" sont considérés dans ces TP. Ces jeux de données sont principalement en 2 dimensions pour des raisons pédagogiques. Vous trouverez une archive des jeux de données artificiels sur la page moodle de cet enseignement.

Note : ne prenez pas en compte les jeux de données en gris.

La lecture de ces jeux de données (format `arff`), s'appuie sur le package python `arf` : `from scipy.io import arff`. Le jeu de données est ensuite mémorisé sous la forme d'un tableau à d dimensions ($d = 2$ pour la plupart des jeux de données proposés) puis affiché sous la forme d'un nuage de points.

Note : dans ces jeux de données, pour chaque exemple, la dernière colonne fournit un numéro de cluster (sans précision sur la méthode utilisée pour l'obtenir). En pratique, **vous ne devez pas utiliser cette colonne** car on suppose que les clusters ne sont pas connus.

A faire

- Récupérez le code fourni pour lire un jeu de données `arf` et l'afficher. Profitez en pour lire et afficher un jeu de données en dimension 3, des jeux de données avec des nombres variables de points, et pour tester des paramètres de la visualisation.
  - Objectif : Découvrir quelques outils de visualisation (consultez la libairie `matplotlib`)
- Appliquez des pré-traitements pour standardiser les données et comparez les visualisations des données initiales et des données standardisées. Choisissez pour cela des jeux de données avec des caractéristiques différentes.
  - Objectif : Apprendre des méthodes de pré-traitement. (consultez la documentation de `scikit-learn` sur le pre-processing)
- Identifiez les méthodes de clustering disponibles sur `scikit-learn` et leur caractéristiques générales. Consultez la comparaison des méthodes de clustering de `scikit-learn`.
  - Objectif : Avoir un premier aperçu (par l'exemple) des avantages et limites des méthodes disponibles.

.

3 Clustering k-Means - TP1

Documentation : `https://scikit-learn.org/stable/modules/clustering.html#k-means`.

A faire

- Récupérez le code fourni qui applique la méthode k-means pour un nombre fixé de clusters sur le jeu de données `xclara.arff`.
- Testez des paramètres de la méthode.
- Evaluez le résultat avec d'autres métriques de `scikit-learn`.

.

3.1 Evaluation de la méthode

On va chercher à déterminer "automatiquement" le bon nombre de clusters en appliquant la méthode k-means pour plusieurs valeurs de k et en évaluant les résultats obtenus.

A faire

- Codez des procédures d'évaluation de la méthode k-means (méthode du "coude" basée sur les valeurs d'inertie, valeurs du coefficient de silhouette et valeurs des autres métriques disponibles).
- Présentez des graphiques pour les différentes métriques mesurées.
- Sélectionnez 2 ou 3 jeux de données pour lesquels il vous semble que la méthode k-means devrait bien fonctionner et 2 ou 3 exemples pour lesquels il vous semble que cette méthode ne fonctionnera pas correctement.

.

- Inspiration possible sur le web ... (attention à la qualité du code ...)
- N'oubliez pas de relever les temps de calcul.
- Arrivez-vous à déterminer correctement le bon nombre de clusters à l'aide des différentes métriques utilisées ?
- Avez-vous identifié les avantages et limites de la méthode k-means ?

4 Clustering agglomératif - TP2 (1/2 séance)

Documentation : `https://scikit-learn.org/stable/modules/clustering.html#hierarchical-clustering`

A faire

- Récupérez le code fourni qui permet de visualiser un dendrogramme sur du jeu de données `xclara.arff` et qui applique la méthode de clustering agglomératif pour un nombre fixé de clusters.
- Dans le code le jeu de données est standardisé.
- Testez des paramètres de la méthode de calcul du dendrogramme et de la méthode de clustering agglomératif.
- Evaluez le résultat avec différentes métriques de `scikit-learn`.

.

4.1 Evaluation de la méthode

On va chercher à déterminer "automatiquement" le bon nombre de clusters en appliquant la méthode de clustering hiérarchique pour plusieurs valeurs de k et en évaluant les résultats obtenus.

A faire

- Codez des procédures d'évaluation de la méthode hiérarchique (par exemple en utilisant les valeurs du coefficient de silhouette et valeurs des autres métriques disponibles).
- Présentez des graphiques pour les différentes métriques mesurées.
- Sélectionnez 2 ou 3 jeux de données pour lesquels il vous semble que la méthode hiérarchique devrait bien fonctionner et 2 ou 3 exemples pour lesquels il vous semble que cette méthode ne fonctionnera pas correctement.

.

- Inspiration possible sur le web ... (attention à la qualité du code ...)
- N'oubliez pas de relever les temps de calcul.
- Arrivez-vous à déterminer correctement le bon nombre de clusters à l'aide des différentes métriques utilisées ?
- Avez-vous identifié les avantages et limites de la méthode hiérarchique ?

5 Clustering `DBSCAN` - TP2 et TP3 (1/2 séance)

Documentation : `https://scikit-learn.org/stable/modules/clustering.html#dbscan`

A faire

- Récupérez le code fourni qui permet de lancer le clustering `DBSCAN` pour des valeurs données des paramètres `epsilon` et `min_samples`.
- Evaluez le résultat avec différentes métriques de `scikit-learn`.

.

5.1 Evaluation de la méthode

On va chercher à déterminer "automatiquement" le bon nombre de clusters en appliquant la méthode `DBSCAN`. Pour cela il faut déterminer un intervalle de valeurs possibles pour les paramètres `epsilon` et `min_samples`.

A faire

- Appliquez une méthode basée sur le calcul des distances des k plus proches voisins pour déterminer des intervalles de variation pertinents de ces paramètres.
- Codez des procédures d'évaluation de la méthode `DBSCAN`.
- Présentez des graphiques pour les différentes métriques mesurées.
- Sélectionnez 2 ou 3 jeux de données pour lesquels il vous semble que la méthode `DBSCAN` devrait bien fonctionner et 2 ou 3 exemples pour lesquels il vous semble que cette méthode ne fonctionnera pas correctement.

.

- Inspiration possible sur le web ... (attention à la qualité du code ...)
- N'oubliez pas de relever les temps de calcul.
- Arrivez-vous à déterminer correctement le bon nombre de clusters à l'aide des différentes métriques utilisées ?
- Avez-vous identifié les avantages et limites de la méthode `DBSCAN` ?

6 Synthèse - TP3 (1/2 séance + travail personnel)

Dans cette partie, vous allez appliquer les différentes méthodes de clustering sur de nouveaux jeux de données. Ces jeux de données sont à récupérer sur la page du cours (disponibles à la séance TP3) et correspondent à des données artificielles de petites dimensions (2 ou 3) et à un jeu de données "réel" ayant un nombre d'attributs plus important.

A faire

- Appliquez les méthodes de k-means, de clustering hiérarchique ou `DBSCAN` sur les jeux de données artificiels et comparez les résultats obtenus (qualité des solutions obtenues, performances des méthodes, ...).
- Sélectionnez une autre méthode de clustering (par exemple parmi celles disponibles dans `scikit-learn`). Expliquez le principe de cette méthode et complétez les comparaisons précédentes avec cette nouvelle méthode
- Pour le jeu de données réel, appliquez une réduction des dimensions (analyse en composantes principales) en complément des méthodes de clustering. Evaluez et comparez les résultats obtenus. (Note : utilisez le package `pandas` pour lire un fichier `csv`).

.

Pour lancer votre analyse expérimentale vous pouvez utiliser différents serveurs de calcul (normalement accessible à distance via le vpn insa) : srv-gei-gpu1 et srv-gei-gpu2.

7 Evaluation

L'évaluation est à déposer sur moodle (un rapport par binôme). La date limite est : **xxx janvier 2022**.

Consignes pour le rapport :

- fichier pdf (maximum 15 pages)
- fournir dans le rapport un lien vers votre code (un dépot git, un jupyter notebook).
- Les différentes visualisations des jeux de données ou des résultats peuvent être jointes en annexe.
- Plan :
  - Partie 1 : Points forts et points faibles identifiés pour les différentes méthodes de clustering étudiées (5 à 6 pages)
  - Partie 2 : Analyse comparative sur les nouvelles données fournies (6 à 8 pages)
  - Conclusion