Formation: M2 SDTS

Module: Deep Learning

TP1: Apprentissages non supervisé, Méthodes de Clustering et de Réduction de Dimensionnalité

Introduction

Dans ce TP, vous allez explorer et comparer plusieurs méthodes de clustering non supervisé et de réduction de dimensionnalité, notamment **KMeans**, le **clustering hiérarchique**, **DBSCAN** et l'**Analyse en Composantes Principales (ACP)**. Vous appliquerez chacune de ces méthodes sur trois jeux de données différents: **Iris**, **Wine** et **Wholesale Customers**. L'objectif est de comprendre comment chaque algorithme se comporte sur des données variées et d'analyser leurs performances respectives. Vous utiliserez des bibliothèques Python telles que scikit-learn, pandas, numpy et matplotlib ... etc, pour manipuler et visualiser les données.

Jeux de données utilisés:

1. **Iris**: Un jeu de données classique en apprentissage automatique qui contient des mesures de fleurs de trois espèces d'iris.
2. **Wine**: Ce jeu de données contient des informations chimiques sur différents vins cultivés dans la même région en Italie mais provenant de trois cultivars différents.
3. **Wholesale Customers**: Ce jeu de données comprend les dépenses annuelles en unités monétaires (m.u.) de clients d'un grossiste, réparties en différentes catégories de produits.

Objectifs du TP:

- Comprendre le fonctionnement des méthodes de clustering KMeans, hiérarchique et DBSCAN.
- Appliquer ces méthodes sur des jeux de données réels.
- Comparer les performances et les résultats de chaque méthode.
- Utiliser l'ACP pour réduire la dimensionnalité des données et faciliter la visualisation.
- Interpréter les résultats obtenus.

Partie 1: Préparation des Données

1.1 Importation des bibliothèques nécessaires

Commencez par importer les bibliothèques Python nécessaires pour ce TP.

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
%matplotlib inline
from sklearn import datasets
from sklearn.preprocessing import StandardScaler
from sklearn.cluster import Kmeans
from scipy.cluster.hierarchy import dendrogram, linkage from sklearn.cluster import
AgglomerativeClustering
from sklearn.cluster import DBSCAN
from sklearn.decomposition import PCA
```

1.2 Chargement des jeux de données

- Chargez les jeux de données Iris, Wine et Wholesale Customers.
- Examinez les premières lignes de chaque jeu de données.
- Vérifiez les dimensions et les types de données.

Questions:

1. Quelles sont les caractéristiques (features) présentes dans chaque jeu de données?
2. Y a-t-il des données manquantes dans ces jeux de données?
3. Les données nécessitent-elles une normalisation ou une mise à l'échelle? Pourquoi?

Partie 2: Clustering KMeans

2.1 Application de KMeans sur les jeux de données

- Appliquez l'algorithme KMeans sur les trois jeux de données.
- Choisissez un nombre de clusters approprié pour chaque jeu de données.
- Visualisez les clusters obtenus.

Questions:

1. Comment choisir le nombre optimal de clusters pour KMeans?
2. Comparez les clusters obtenus avec les vraies étiquettes (targets). Quelle est la performance du clustering (utilisez rand index score)?
3. Quels sont les avantages et les inconvénients de KMeans?

Partie 3: Clustering Hiérarchique

3.1 Application du clustering hiérarchique sur les jeux de données

- Utilisez le clustering hiérarchique (agglomératif) sur les trois jeu de données.
- Visualisez le dendrogramme.
- Découpez l'arbre pour obtenir un nombre de clusters approprié.

Questions:

1. Quels sont les différents types de liens (linkage) utilisés dans le clustering hiérarchique? Essayez-en plusieurs et comparez les résultats.
2. Comment interpréter le dendrogramme?
3. Quels sont les avantages du clustering hiérarchique par rapport à KMeans?

Partie 4: DBSCAN

4.1 Application de DBSCAN sur les jeux de données Wholesale Customers

- Appliquez l'algorithme DBSCAN sur les trois jeux de données.
- Trouvez des valeurs appropriées pour les paramètres epsilon (ε) et min_samples.
- Identifiez les points bruyants (outliers).

Questions:

1. Comment choisir les valeurs de ε et min_samples?
2. Quels types de structures de données DBSCAN peut-il détecter que KMeans et le clustering hiérarchique ne peuvent pas?
3. Quels sont les inconvénients de DBSCAN?

Partie 5: Analyse en Composantes Principales (PCA)

5.1 Réduction de dimensionnalité avec PCA

- Appliquez la PCA sur le jeu de données Wine.
- Réduisez les données à deux composantes principales.
- Visualisez les données dans l'espace 2D des composantes principales.

Questions:

1. Quelle proportion de la variance totale est expliquée par les deux premières composantes principales?
2. Pourquoi la PCA est-elle utile avant d'appliquer des algorithmes de clustering?
3. Pouvez-vous améliorer les résultats des méthodes de clustering en utilisant les données réduites par PCA?

Partie 6: Comparaison des Méthodes

- Comparez les résultats obtenus avec les différentes méthodes de clustering sur les jeux de données utilisés.
- Discutez des performances, avantages et inconvénients de chaque méthode.

Questions:

1. Quelle méthode de clustering a donné les meilleurs résultats pour chaque jeu de données? Pourquoi?
2. Dans quels cas utiliseriez-vous l'une des méthodes plutôt qu'une autre?
3. Comment l'échelle des données affecte-t-elle les résultats du clustering?

Partie 7: Conclusion

- Résumez ce que vous avez appris dans ce TP.
- Proposez des améliorations possibles pour les méthodes appliquées.
- Comment ces méthodes peuvent-elles être utilisées dans des situations réelles?

Instructions supplémentaires:

- Commentez votre code pour expliquer ce que vous faites à chaque étape.
- N'hésitez pas à expérimenter avec les paramètres des algorithmes pour voir comment ils affectent les résultats.
- Sauvegardez vos figures et incluez-les dans votre rapport.