UNIVERSIDAD DE ANTIOQUIA
1803

Laboratorio II - Clustering

Aprendizaje automático II

Requisitos previos

Python

Python

- Estructuras de datos (propiedades de listas, tuplas, dicts, módulos incorporados...)
- Clases
- Paquetes y módulos

NumPy

- Matrices
- Producto interior
- Producto vectorial - matricial
- Distancias

Álgebra lineal

Conceptos de clase

- Vectores y matrices
- Propiedades de las matrices
- Eigendecomposition
- Agrupación

Taller II

1. Investigue sobre el método **Spectral Clustering** y responda a las siguientes preguntas:
    a. ¿En qué casos puede ser más útil aplicarlo?
    b. ¿Cuáles son sus fundamentos matemáticos?
    c. ¿Cuál es el algoritmo para calcularlo?
    d. ¿Tiene alguna relación con algunos de los conceptos mencionados anteriormente en clase? ¿Cuáles y cómo?

2. Investiga sobre el método **DBSCAN** y responde a las siguientes preguntas:
    a. ¿En qué casos puede ser más útil aplicarlo?
    b. ¿Cuáles son sus fundamentos matemáticos?
    c. ¿Existe alguna relación entre DBSCAN y el Clustering Espectral? En caso afirmativo, ¿cuál es?

3. ¿Cuál es el método del codo en la agrupación? ¿Y qué fallos presenta para evaluar la calidad?

4. ¿Recuerdas el paquete Python *no supervisado* que creaste en la unidad anterior? □ Es hora de actualizarlo.
    a. Implementar el módulo **k-means** usando Python y Numpy
    b. Implementar el módulo **k-medoids** usando Python y Numpy
    c. Recuerde mantener la máxima coherencia posible con la API de Scikit-Learn

5. Utilicemos los módulos recién creados en *unsupervised* para agrupar algunos datos de juguete.
    a. Utilice el siguiente fragmento de código para crear datos dispersos X
    ```
    from sklearn.datasets import make_blobs
    X, y = make_blobs(
        n_muestras=500,
        n_características=2,
        centros=4,
        cluster_std=1,
        center_box=(-10,0, 10,0),
        shuffle=True,
        random_state=1,
    )
    ```
    b. Represente gráficamente el conjunto de datos resultante. ¿Cuántos conglomerados hay? ¿A qué distancia están unos de otros?
    c. Tanto para k-means como para k-medoids (sus implementaciones), calcule los gráficos de silueta y los coeficientes para cada ejecución, iterando K de 1 a 5 clusters.
    d. ¿Qué número de K obtuvo la mejor puntuación en siluetas? ¿Qué se puede decir de las cifras? ¿Es éste el resultado esperado?

6. Utilice el siguiente fragmento de código para crear diferentes tipos de datos dispersos:
    ```
    import numpy as np
    from sklearn import cluster, datasets, mixture

    # ============
    # Generar conjuntos de datos. Elegimos el tamaño suficiente para ver la
    escalabilidad # de los algoritmos, pero no demasiado grande para evitar
    tiempos de ejecución demasiado largos.
    # ============
    n_muestras = 500
    noisy_circles = datasets.make_circles(n_muestras=n_muestras, factor=0.5, ruido=0.05)
    noisy_moons = datasets.make_moons(n_muestras=n_muestras, noise=0.05)
    blobs = datasets.make_blobs(n_muestras=n_muestras, random_state=8)
    no_structure = np.random.rand(n_muestras, 2), None
    ```

```
# Datos distribuidos anisotrópicamente
random_state = 170
X, y = datasets.make_blobs(n_muestras=n_muestras, random_state=estado_aleatorio)
transformation = [[0.6, -0.6], [-0.4, 0.8]]
X_aniso = np.dot(X, transformación)
aniso = (X_aniso, y)

# blobs with varied variances
variado = datasets.make_blobs(
    n_muestras=n_muestras, cluster_std=[1.0, 2.5, 0.5], random_state=estado_aleatorio
)
```

a. Representa los distintos conjuntos de datos en figuras separadas. ¿Qué puedes decir sobre ellos?
b. Aplicar k-means, k-medoids, DBSCAN y Spectral Clustering de Scikit-Learn sobre cada conjunto de datos y comparar los resultados de cada algoritmo con respecto a cada conjunto de datos.

Recursos útiles

https://github.com/rushter/MLAlgorithms/blob/035e489a879d01a84fffff74885dc6b1bca3c96f/mla/kmeans.py
https://github.com/patchy631/machine-learning/blob/main/ml_from_scratch/KMeans_from_scratch.ipynb
https://scikit-learn.org/stable/auto_examples/cluster/plot_kmeans_silhouette_analysis.html