Facultad de Matemática, Astronomía y Física y Computación
Universidad Nacional de Córdoba

Ciencia de Datos

Práctico N°10: Clustering

Scikit-learn provee varios algoritmos de clustering en el módulo sklearn.cluster. La propuesta de este práctico es estudiar las implementaciones de K-means, para separar ejemplos no etiquetados de k-grupos disjuntos de varianzas similares y AgglomerativeClustering como ejemplo de clustering jerárquico.

Problema 1: K-means

a) Usar una visualización interactiva online en 2D para ganar intuición sobre le algoritmo de Lloyd.
b) Estudiar los parámetros de la función `KMeans()`; en particular, cuál permite eligir los algoritmos: `lloyd`, `elkan`, las opciones de inicialización para los centroides y las opciones de detención. ¿Qué definición de distancia tiene implementada? Identificar la cantidad que se minimiza.
c) Para la evaluación de performance de la tarea de clustering se dispone de varias métricas. Estudiar las definiciones de los siguientes scores: `Rand index`, `mutual information`, `homogeneity`, `completeness`, `V measure`, `silhouette coefficient`.

Problema 2: Utilizando la información disponible en el ejemplo de scikit-learn, aplicar K-means sobre el digits dataset.
a) Utilizar como inicialización de los centroides: `k-means++, random` y las 10 componentes de PCA sobre el dataset.
b) Dado que se dispone de *ground truth* (la clase), implementar las métricas de evaluación estudiadas y reproducir la tabla del ejemplo. Discutir los resultados.
c) Proyectar sobre el plano definido por las 2 primeras componentes de PCA para visualizar los datos y el resultado de clusterizar los datos proyectados con k-means.

Problema 3: Utilizar el coefiente silhouette para seleccionar el número de clusters adecuado, usando como ejemplo datos generados con la función make_blobs().

Problema 4: Clustering jerárquico

a) Estudiar las nociones de enlace: `ward`, `complete average` y `single`, y las posibles métricas de distancia disponibles para cada opción de enlace en la función AgglomerativeClustering()
b) Visualizar el efecto de los distintos *linkages* sobre los datos, usando un embedding 2D de los clusters resultante sobre el digits dataset, usando la función provista en el ejemplo de scikit-learn.

Problema 5: Aplicar AgglomerativeClustering() sobre el iris dataset y visualizar el dendograma usando el método disponible en scipy.