

Ciencia de Datos

Práctico N°8: 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. Compare con el método de Mezcla de Gaussianas y visualice en PCA y tSNE.

Problema 1: K-means

- a) Usar una visualización interactiva online en 2D para ganar intuición sobre el algoritmo de Lloyd.
- b) Estudiar los parámetros de la función `KMeans()`; en particular, cuál permite elegir 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 agrupar con k-means.
- d) Proyectar sobre el plano definido por t-SNE para visualizar los datos y el resultado de agrupar con k-means. Utilice varios valores de perplexity.

Problema 3: Clustering aglomerativo

- 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.
- c) Proyectar sobre el plano definido por t-SNE para visualizar el efecto de los distintos *linkages* sobre los datos. Utilice varios valores de perplexity.

Problema 4:

- a) Aplicar `AgglomerativeClustering()` sobre el `iris` dataset y visualizar el dendograma usando el método disponible en `scipy`.
- b) Proyectar sobre el plano definido por t-SNE para visualizar el efecto de los distintos *linkages* sobre los datos. Utilice varios valores de perplexity.

Problema 5: Mezcla de Gaussianas

- a) Aplicar `GaussianMixture()` sobre el `iris` dataset cambiando el tipo de Covarianza supuesta.

- b) visualizar con t-SNE.
- c) visualizar sobre las dos primeras componentes principales.

Problema 6:

- a) Utilizar el **coeficiente silhouette** para seleccionar el número de clusters adecuado en kmedias, usando como ejemplo datos generados con la función `make_blobs()`.
- b) Utilizar el BIC para seleccionar el número de clusters en una mezcla de gaussianas aplicada al mismo grupo de datos del punto anterior (Selección de Modelo)
- c) Visualice la selección final de agrupamiento para los dos métodos
- d) Dado que se conoce la Ground Truth, realice una tabla de métricas para cada modelo. (utilice Pritty Table)



FaMAF 2024