Pontificia Universidad Católica de Chile
Escuela de Ingeniería
Departamento de Ciencia de la Computación

Actividad (evaluada): *k-means*

Aplicaciones en Ciencia de Datos e Inteligencia Artificial

Profesor : Francisco Pérez Galarce
Ayudante : Yesenia Salinas

Fecha : 03 de diciembre de 2024

1 Introducción

Dentro del aprendizaje no supervisado, los algoritmos de clustering juegan un papel crucial debido a su amplia gama de aplicaciones prácticas. Entre estas aplicaciones se destacan la segmentación de clientes, el procesamiento de imágenes, la detección de anomalías y muchas otras áreas de interés [Jain, 2010].

En esta actividad, exploraremos el algoritmo de k-means mediante ejemplos prácticos. Usted implementará este método utilizando librerías de Python, ajustará, experimentará y analizará los resultados obtenidos para comprender su comportamiento y aplicaciones en diversos contextos.

2 Instrucciones de la actividad

2.1 *Implementación y visualización de k-means en 2D* (30 ptos)

0 ptos Abrir entorno de programación, de preferencia utilizar `Visual studio` o `Jupyter notebook`.

5 ptos Cargar la base de datos *kmeans1.csv* utilizando pandas.

5 ptos Por medio de matplotlib (o seaborn) genere un gráfico de dispersión (scatter plot) de las variables A y B.

5 ptos Aplique el algoritmo k-means (*from sklearn.cluster import KMeans*), observe y analice los outputs disponibles (centroides, clusters asignados y distancia dentro de las clases).

5 ptos Ajuste k-means considerando 1,2,...10 clusters, guarde la distancia intra clases en un diccionario. Estudie la documentación de la clase y modifique los argumentos `init`, `n_init` y `max_it`.

5 ptos Utilizando matplotlib (o seaborn) genere un gráfico que presente la distancia intra clases para cada k utilizado. A través del criterio del codo defina el número de clusters óptimo.

2.2 *Análisis de clusters* (30 ptos)

5 ptos Aplicar el algoritmo *k-means* a la base de datos *k-means2.csv*. Aplique el procesamiento necesario a los datos para una correcta asignación de clusters.

10 ptos Determine el número de clusters adecuados con alguno de los tres métodos vistos en clases (regla del codo, Davies-Bouldin o Silhouette plot). Genere una función (o un conjunto de funciones) que le permita obtener el reporte de las métricas y visualizaciones.

5 ptos Interprete los resultados de los centroides. Implemente una función que le permita analizar los resultados de los centroides de forma automática. No olvide considerar la transformaciones inversas del escalamiento o transformaciión z-score.

2.3 Clase `k-means` (10 ptos bonus)

5 ptos Implemente una clase para ajustar el algoritmo k-means a una base de datos. Para el preprocesamiento de la base de datos puede usar clases desarrolladas en actividades anteriores. La clase debe tener un método *fit(df, k)* que recibe los datos y entrega el una lista con la etiqueta de cluster para cada fila en la base de datos original. Dentro de la clase se recomienda incluir los métodos *initialize_centroids*, *assign_clusters*, *update_centroids* y *has_converged*. Como atributos se deben considerar la distancia intra clusters y entre clusters, los centroides y la etiqueta asignada a cada dato.

5 ptos Considere que se le ha pedido agrupar a los clientes del año 2024 para definir los principales segmentos de clientes, luego se desea utilizar los centroides de este modelo para asignar un segmento a los nuevos clientes que ingresarán a la compañía durante el año 2025. ¿Cómo podemos disponibilizar este modelo en base al requerimiento?, implemente las funciones que sean necesarias. Recuerde que en el despligue no se puede entrenar el modelo durante cada nuevo ingreso de clientes.

2.4 Entrega

- La actividad deberá entregarse en un archivo comprimido donde incluya archivos `Jupyter notebook` (.ipynb), `Python` (.py) y otros archivos para gestionar parámetros (ejemplo: `.yaml, .json`). El archivo comprimido debe subirse a la plataforma del curso y subirse a su repositorio del curso en `Github`[a]. No subir el archivo con los datos originales.
- La actividad debe realizarse de forma individual.
- La actividad debe ser subida a la plataforma antes del miércoles 04 a las 23:59 P.M.

References

Anil K Jain. Data clustering: 50 years beyond k-means. *Pattern recognition letters*, 31(8):651–666, 2010.

[a] https://github.com/