Distancia euclidia: Distancia mas típica. La distancia entre dos puntos.

Clasificador no es lo mismo que cluster

Inercia: Es la distancia total de todos los puntos respecto a su centroide. Mas clases (k), la inercia cae.

Idea: La inercia NO sirve para medir cual es la mejor K. Sirve para medir para una misma K cual es la mejor configuración.

* Podemos ver una grafica con la inercia. (Método del codo). Alli buscamos el punto donde las K’s dejan de mejorar la inercia.

BUSCAR QUE ES SILUETA

DBSCAN: Ve correlaciones. No se le da la K. Se le dan los parámetros: saltos (épsilon), y el mínimo de miembros para el club.

¿Qué es silueta?

¡Claro, Luis! La **silueta** es una métrica que se utiliza para evaluar la calidad de los clusters generados por un algoritmo de clustering. A diferencia de la inercia (que mide la compacidad de los clusters), la silueta combina dos factores: la cohesión interna de los clusters y la separación entre clusters. Esto la hace muy útil para evaluar si los clusters formados son significativos.

**Definición de la Silueta**

La silueta mide qué tan bien se encuentra un punto dentro de su cluster en comparación con otros clusters. Para cada punto iii, el **coeficiente de silueta** s(i)s(i)s(i) se calcula como:

s(i)=b(i)−a(i)max⁡(a(i),b(i))s(i) = \frac{b(i) - a(i)}{\max(a(i), b(i))}s(i)=max(a(i),b(i))b(i)−a(i)​

Donde:

1. **a(i)a(i)a(i): Cohesión interna**
   * Es la distancia media entre el punto iii y todos los demás puntos en el mismo cluster.
   * Representa qué tan cerca está iii de los puntos de su propio cluster.
2. **b(i)b(i)b(i): Separación**
   * Es la distancia media entre el punto iii y todos los puntos del cluster más cercano (al que iii no pertenece).
   * Representa qué tan lejos está iii de otros clusters.

**Interpretación del coeficiente de silueta**

El valor de s(i)s(i)s(i) está en el rango de [−1,1][-1, 1][−1,1]:

* **s(i)s(i)s(i) cercano a 1**:
  + El punto está bien agrupado dentro de su cluster, y su cluster está bien separado de otros.
  + Excelente calidad del clustering.
* **s(i)s(i)s(i) cercano a 0**:
  + El punto está en el límite entre su cluster y el más cercano.
  + Puede que los clusters estén solapados o mal definidos.
* **s(i)s(i)s(i) cercano a -1**:
  + El punto estaría mejor en otro cluster.
  + Indica una mala asignación de clusters.

**Índice de silueta global**

En lugar de analizar s(i)s(i)s(i) para cada punto, podemos calcular la **silueta media** de todos los puntos:

S=1N∑i=1Ns(i)S = \frac{1}{N} \sum\_{i=1}^{N} s(i)S=N1​i=1∑N​s(i)

* **SSS cercano a 1**: Los clusters están bien definidos y separados.
* **SSS cercano a 0**: Los clusters se solapan o son ambiguos.
* **SSS negativo**: Mala asignación de clusters.

**Modelos de Clustering**

**1. K-means**

* **¿Qué hace?** K-means busca dividir los datos en kkk grupos, donde kkk es un número predefinido de clusters. El objetivo es minimizar la suma de las distancias cuadráticas entre los puntos y el centroide de su grupo.
* **Cómo funciona:**
  1. Elige kkk (número de clusters).
  2. Inicializa kkk centroides de manera aleatoria.
  3. Asigna cada punto al centroide más cercano.
  4. Actualiza la posición de los centroides como el promedio de los puntos asignados.
  5. Repite los pasos 3 y 4 hasta que los centroides no cambien (o se alcance un número máximo de iteraciones).
* **Ventajas:**
  1. Rápido y fácil de entender.
  2. Funciona bien con datos de forma esférica y tamaños similares.
* **Limitaciones:**
  1. Sensible a los valores iniciales de los centroides.
  2. No maneja bien clusters de formas irregulares o de tamaños desiguales.
  3. Requiere predefinir kkk, que no siempre es conocido.

**2. DBSCAN (Density-Based Spatial Clustering of Applications with Noise)**

* **¿Qué hace?** DBSCAN agrupa puntos densamente conectados y detecta ruido o puntos atípicos (*outliers*). Es ideal para clusters de forma arbitraria.
* **Cómo funciona:**
  1. Define dos parámetros importantes:
     + ε\varepsilonε (*epsilon*): El radio dentro del cual un punto se considera vecino de otro.
     + min\_samples: El número mínimo de puntos necesarios para formar un cluster denso.
  2. Comienza en un punto aleatorio:
     + Si tiene al menos min\_samples puntos en su radio ε\varepsilonε, forma un nuevo cluster.
     + Si no, se clasifica como ruido.
  3. Expande el cluster explorando sus vecinos hasta que no pueda crecer más.
  4. Repite hasta procesar todos los puntos.
* **Ventajas:**
  1. Detecta clusters de formas arbitrarias (no es limitado a formas esféricas como k-means).
  2. Identifica ruido/outliers automáticamente.
* **Limitaciones:**
  1. No funciona bien con datos de densidades muy variables.
  2. La elección de ε\varepsilonε y min\_samples puede ser sensible.

**3. Clustering Jerárquico**

* **¿Qué hace?** Crea una jerarquía de clusters que pueden representarse como un árbol o dendrograma. Se puede hacer de dos maneras:
  + **Aglomerativo**: Empieza con cada punto como un cluster y fusiona los más cercanos.
  + **Divisivo**: Empieza con todos los puntos en un único cluster y los divide progresivamente.
* **Cómo funciona (aglomerativo):**
  + Calcula una matriz de distancias entre todos los puntos.
  + Une los dos puntos o clusters más cercanos para formar un nuevo cluster.
  + Actualiza la matriz de distancias considerando el nuevo cluster.
  + Repite hasta que todos los puntos estén en un único cluster.
* **Ventajas:**
  + No requiere predefinir el número de clusters.
  + El dendrograma permite explorar diferentes niveles de agrupación.
* **Limitaciones:**
  + Es computacionalmente costoso con conjuntos de datos grandes.
  + Las decisiones iniciales de fusión/división no se pueden deshacer (no es flexible).

**Comparativa Rápida**

| **Modelo** | **Forma de Clusters** | **Necesita kkk** | **Maneja Ruido** | **Escalabilidad** |
| --- | --- | --- | --- | --- |
| K-means | Esféricos | Sí | No | Alta |
| DBSCAN | Arbitraria | No | Sí | Media |
| Jerárquico | Arbitraria | No | No | Baja |