3.4.2 Clustering jerárquico

December 27, 2020

1 Clustering jerárquico

Es un algoritmo de aprendizaje automático no supervisado que se utiliza para agrupar puntos de datos sin etiquetar. Al igual que K-means, la agrupación jerárquica también agrupa los puntos de datos con características similares. En algunos casos, el resultado de la agrupación jerárquica y K-Means puede resultar similar.

La técnica de agrupamiento puede ser muy útil cuando se trata de datos sin etiquetar. Dado que la mayoría de los datos en el mundo real no están etiquetados y la anotación de los datos tiene costos más altos.

1.1 Tipos de clustering jerárquico

Hay dos tipos de agrupación jerárquica: Aglomerativa y Divisiva.

Aglomerativa : los puntos de datos se agrupan utilizando un enfoque de abajo hacia arriba que comienza con puntos de datos individuales.

Dividiba: se sigue un enfoque de arriba hacia abajo donde todos los puntos de datos se tratan como un gran conjunto y el proceso de agrupación implica dividir el único.

1.2 Pasos del algoritmo

- 1. Tratar cada punto de datos como un grupo. Por lo tanto, el número de grupos al comienzo será K, mientras que K es un número entero que representa el número de puntos de datos.
- 2. Formar un clúster uniendo los dos puntos de datos más cercanos que dan como resultado los clusters K-1.
- 3. Formar más grupos al unir los dos grupos más cercanos, lo que da como resultado grupos K-2.
- 4. Repetir los tres pasos anteriores hasta que se forme un grupo grande.
- 5. Una vez que se forma un solo clúster, los dendrogramas se utilizan para dividirse en varios clústeres, según el problema.

Hay diferentes maneras de encontrar la distancia entre los grupos. La distancia en sí puede ser euclidiana o Manhattan.

1.3 Cómo medir la distancia entre dos grupos:

- Medir la distancia entre los puntos de cierre de dos grupos.
- Medir la distancia entre los puntos más lejanos de dos grupos.
- Medir la distancia entre los centroides de dos grupos.
- Medir la distancia entre todas las combinaciones posibles de puntos entre los dos grupos y considerar la media.

1.4 Uso de dendograma

Anteriormente se mencionó que una vez que se forma un grupo grande por la combinación de grupos pequeños, se usan los dendrogramas del grupo para dividir el grupo en múltiples grupos de puntos de datos relacionados.

1.5 Explicación de dendograma

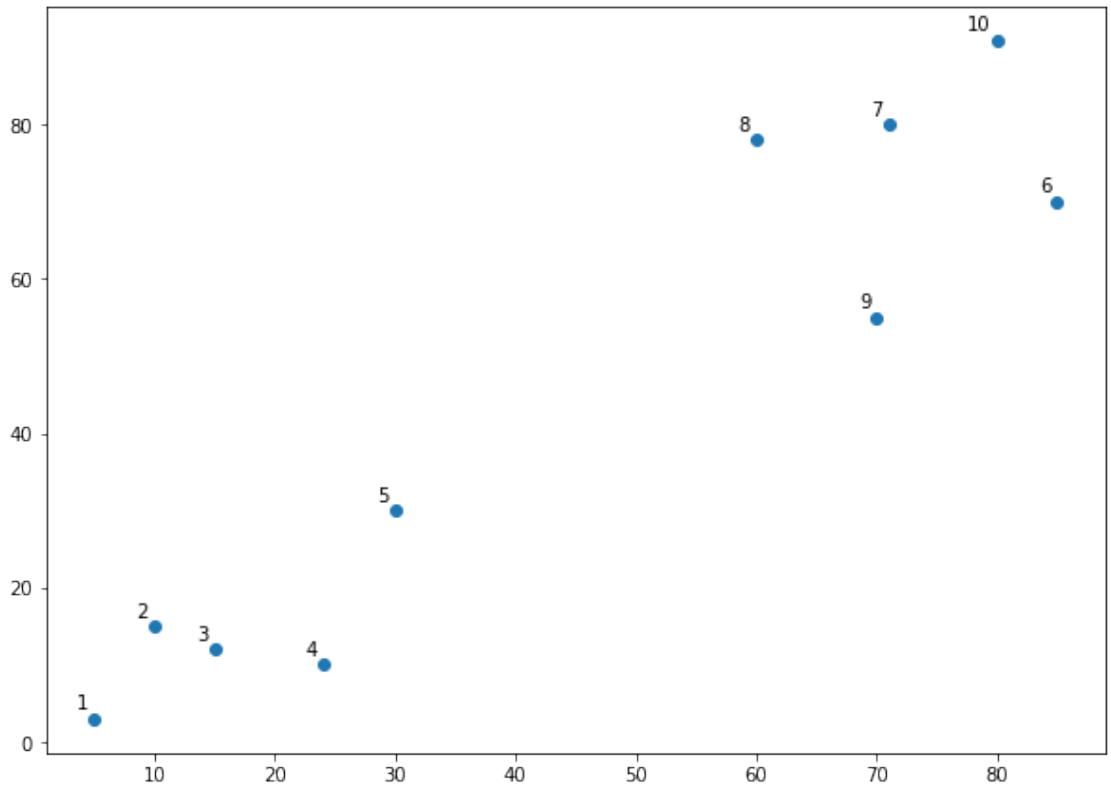
[10]:
```
import numpy as np

X = np.array([[5,3],
    [10,15],
    [15,12],
    [24,10],
    [30,30],
    [85,70],
    [71,80],
    [60,78],
    [70,55],
    [80,91],])
```

[11]:
```
# Se trazan los puntos anteriores
import matplotlib.pyplot as plt

labels = range(1, 11)
plt.figure(figsize=(10, 7))
plt.subplots_adjust(bottom=0.1)
plt.scatter(X[:,0],X[:,1], label='True Position')

for label, x, y in zip(labels, X[:, 0], X[:, 1]):
    plt.annotate(
        label,
        xy=(x, y), xytext=(-3, 3),
        textcoords='offset points', ha='right', va='bottom')
plt.show()
```



1.5.1 Análisis del gráfico

A simple vista que los puntos de datos forman dos grupos: primero en la parte inferior izquierda que consiste en los puntos 1-5, mientras que el segundo en la parte superior derecha consiste en los puntos 6-10.

Sin embargo, en el mundo real, es posible tener miles de puntos de datos en más de 2 dimensiones. En ese caso, no sería posible detectar grupos a simple vista. Esa la razón por la cual existen los algoritmos de clustering.

Ahora se van a graficar los dendrogramas para el set de datos. La biblioteca scipy sirve para cumplir con el objetivo.

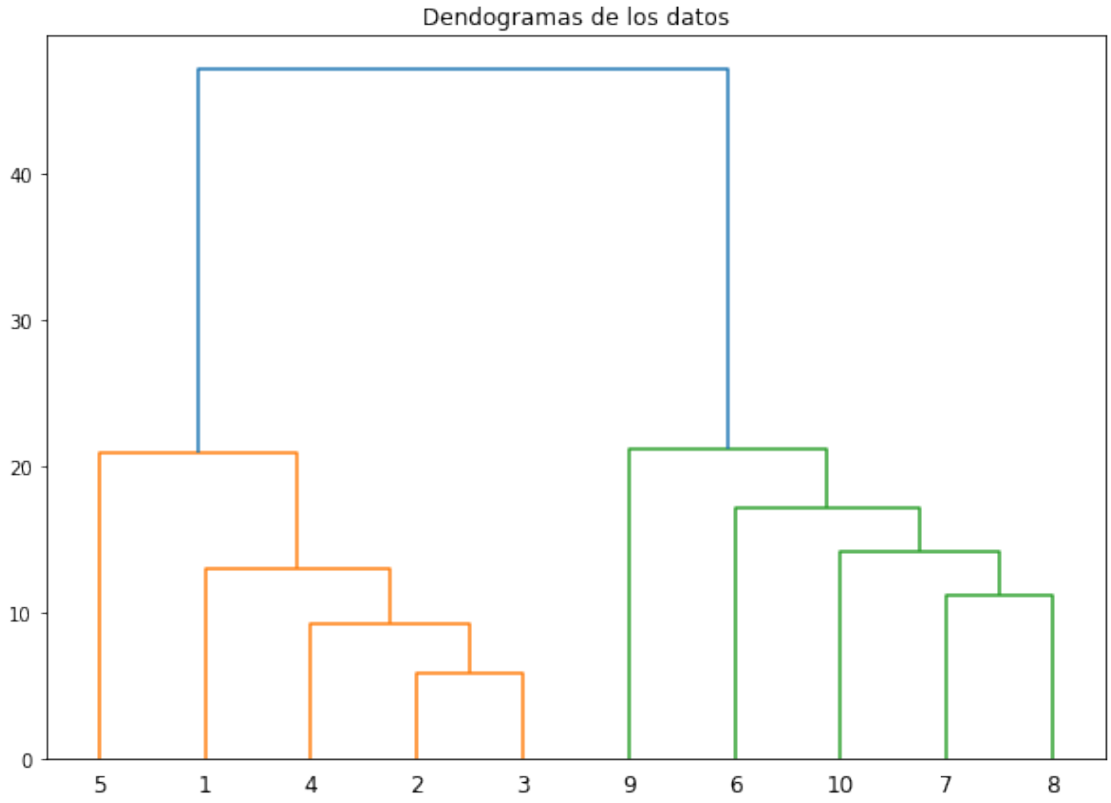
[13]:
```
from scipy.cluster.hierarchy import dendrogram, linkage
from matplotlib import pyplot as plt

linked = linkage(X, 'single')

labelList = range(1, 11)

plt.figure(figsize=(10, 7))
dendrogram(linked,
```

```
            orientation='top',
            labels=labelList,
            distance_sort='descending',
            show_leaf_counts=True)
plt.title("Dendogramas de los datos")
plt.show()
```



1.5.2 Análisis del gráfico

El algoritmo comienza encontrando los dos puntos que están más cerca uno del otro sobre la base de la distancia euclidiana.

Si se mira el primer gráfico, es posible ver que los puntos 2 y 3 están más cerca el uno del otro, mientras que los puntos 7 y 8 están cerca uno del otro. Por lo tanto, primero se formará un grupo entre estos dos puntos.

En el último gráfico, es posible ver que los dendogramas se crearon uniendo los puntos 2 con 3 y 7 con 8. La altura vertical del dendograma muestra las distancias euclidianas entre los puntos. En el último gráfico, se puede ver que la distancia euclidiana entre los puntos 7 y 8 es mayor que la distancia entre los puntos 2 y 3.

El siguiente paso es unirse al clúster formado uniendo dos puntos al siguiente clúster más cercano o

al punto que a su vez se traduce en otro clúster. En el primer gráfico, el punto 4 es el más cercano al grupo de puntos 2 y 3, por lo tanto, en el último gráfico, el dendrograma se genera uniendo el punto 4 con el dendrograma del punto 2 y 3. Este proceso se mantiene hasta que todos los puntos se unen para formar un solo gran racimo.

Una vez que se forma un gran grupo, se selecciona la distancia vertical más larga sin que ninguna línea horizontal pase a través de él y se dibuja una línea horizontal a través de él. El número de líneas verticales por las que pasa esta línea horizontal recién creada es igual al número de grupos.

Podemos ver que la distancia vertical más grande sin ninguna línea horizontal que pase a través de ella está representada por una línea azul. Considere una línea roja horizontal que pasa a través de la línea azul. Dado que cruza la línea azul en dos puntos, el número de agrupaciones será 2.

En esencia, la línea horizontal es un **umbral**, que define la distancia mínima requerida para ser un grupo separado. Si se traza una línea más abajo, el umbral requerido para ser un nuevo grupo se reducirá y se formarán más grupos.

1.6 Aplicación del algoritmo

El set de datos contiene información de clientes considerando: género, edad, sueldo y puntuación de gastos.

El objetivo es segmentar a los clientes en diferentes grupos según sus tendencias de compra.

La columna **Spending Score** (puntuación de gasto) indica la frecuencia con la que el cliente gasta dinero en un centro comercial en una escala de 1 a 100, siendo 100 el gasto más alto.

[1]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sb
```

[20]:
```
data_frame = pd.read_csv("3.4.4 hierarchical-clustering-data.csv")
data_frame.head()
```

[20]:

	CustomerID	Genre	Age	Annual Income (k$)	Spending Score (1-100)
0	1	Male	19	15	39
1	2	Male	21	15	81
2	3	Female	20	16	6
3	4	Female	23	16	77
4	5	Female	31	17	40

[21]:
```
data_frame.shape
```

[21]:
```
(200, 5)
```

[23]:
```
# Se seleccionan las columnas de interes (sueldo y puntuación de gastos)
data_selected = data_frame.iloc[:, 3:5].values
pd.DataFrame(data_selected)
```

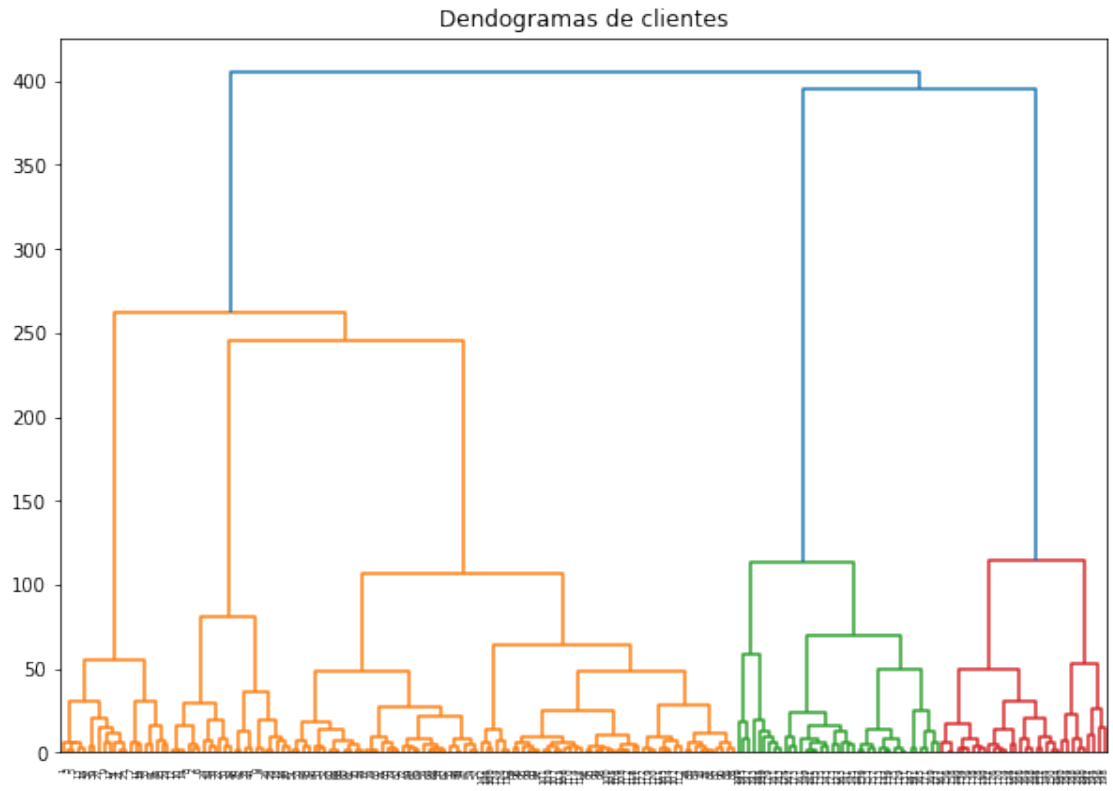
[23]:

	0	1
0	15	39
1	15	81
2	16	6
3	16	77
4	17	40
..	…	..
195	120	79
196	126	28
197	126	74
198	137	18
199	137	83

[200 rows x 2 columns]

[24]:
```
import scipy.cluster.hierarchy as shc

plt.figure(figsize=(10, 7))
plt.title("Dendogramas de clientes")
dend = shc.dendrogram(shc.linkage(data_selected, method='ward'))
```

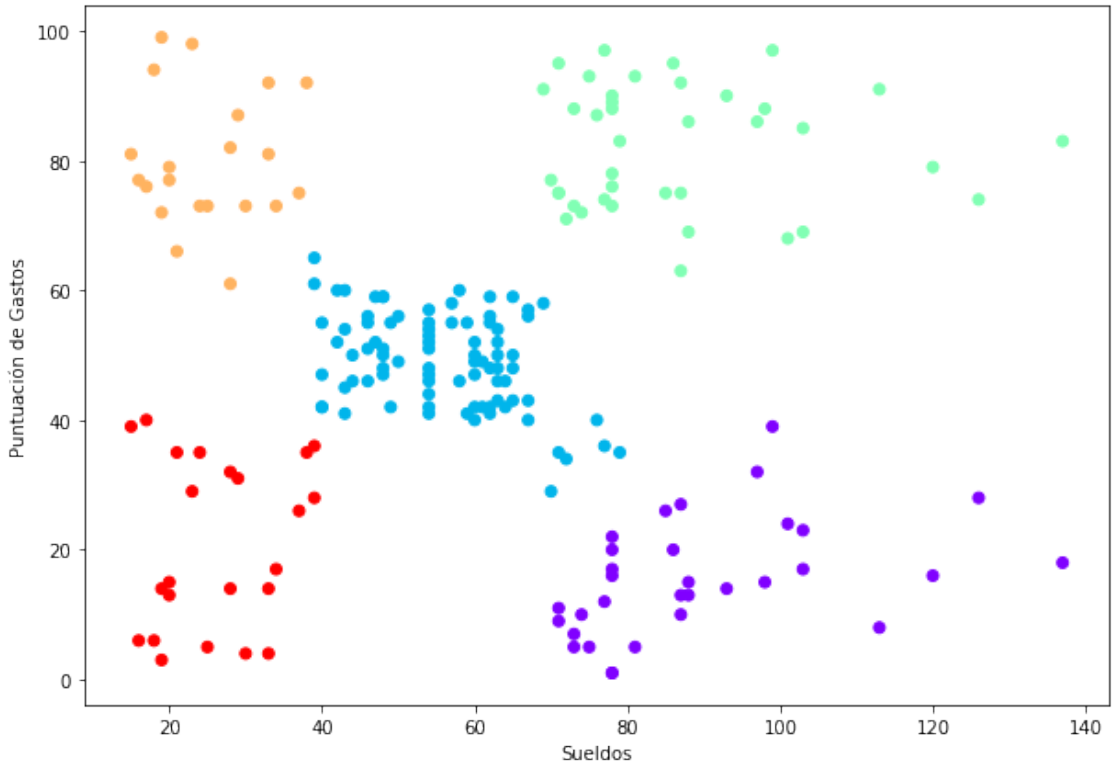


[25]:
```
from sklearn.cluster import AgglomerativeClustering

cluster = AgglomerativeClustering(n_clusters=5, affinity='euclidean',␣
↪linkage='ward')
cluster.fit_predict(data_selected)
```

[25]:
```
array([4, 3, 4, 3, 4, 3, 4, 3, 4, 3, 4, 3, 4, 3, 4, 3, 4, 3, 4, 3,
       4, 3, 4, 3, 4, 3, 4, 3, 4, 3, 4, 3, 4, 3, 4, 3, 4, 3, 4, 1,
       4, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
       1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
       1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
       1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2, 1, 2, 1, 2, 0, 2, 0, 2,
       1, 2, 0, 2, 0, 2, 0, 2, 0, 2, 1, 2, 0, 2, 1, 2, 0, 2, 0, 2, 0, 2,
       0, 2, 0, 2, 0, 2, 1, 2, 0, 2, 0, 2, 0, 2, 0, 2, 0, 2, 0, 2, 0, 2,
       0, 2, 0, 2, 0, 2, 0, 2, 0, 2, 0, 2, 0, 2, 0, 2, 0, 2, 0, 2, 0, 2,
       0, 2])
```

[28]:
```
plt.figure(figsize=(10, 7))
plt.scatter(data_selected[:,0], data_selected[:,1], c=cluster.labels_,␣
↪cmap='rainbow')
plt.xlabel("Sueldos")
plt.ylabel("Puntuación de Gastos")
plt.show()
```



1.7 Análisis del gráfico

Es posible ver los puntos de datos en forma de cinco grupos.

Los puntos de datos en la parte inferior derecha pertenecen a los clientes con sueldos altos y gastos bajos: clientes que gastan su dinero con cuidado.

De forma análoga, los clientes en la parte superior derecha (puntos de datos verdes), son los clientes con sueldos altos y altos gastos. Estos son el tipo de clientes a los que las empresas apuntan.

Los clientes de en medio (puntos de datos azules) son los que tienen gastos promedio y sueldos promedio. Es la categoría con mayor número de clientes. Para las empresas puede resultar interesante considerar a estos clientes dado la cantidad que existe.