TICS-411 Minería de Datos

Clase 4: Clustering Jerárquico

Alfonso Tobar-Arancibia
alfonso.tobar.a@edu.uai.cl

Clustering Jerárquico

Definiciones

Clustering Jerárquico

> Es un tipo de aprendizaje que no requiere de etiquetas (las respuestas correctas) para poder aprender. Se basa en la construcción de Jerarquías para ir construyendo clusters.

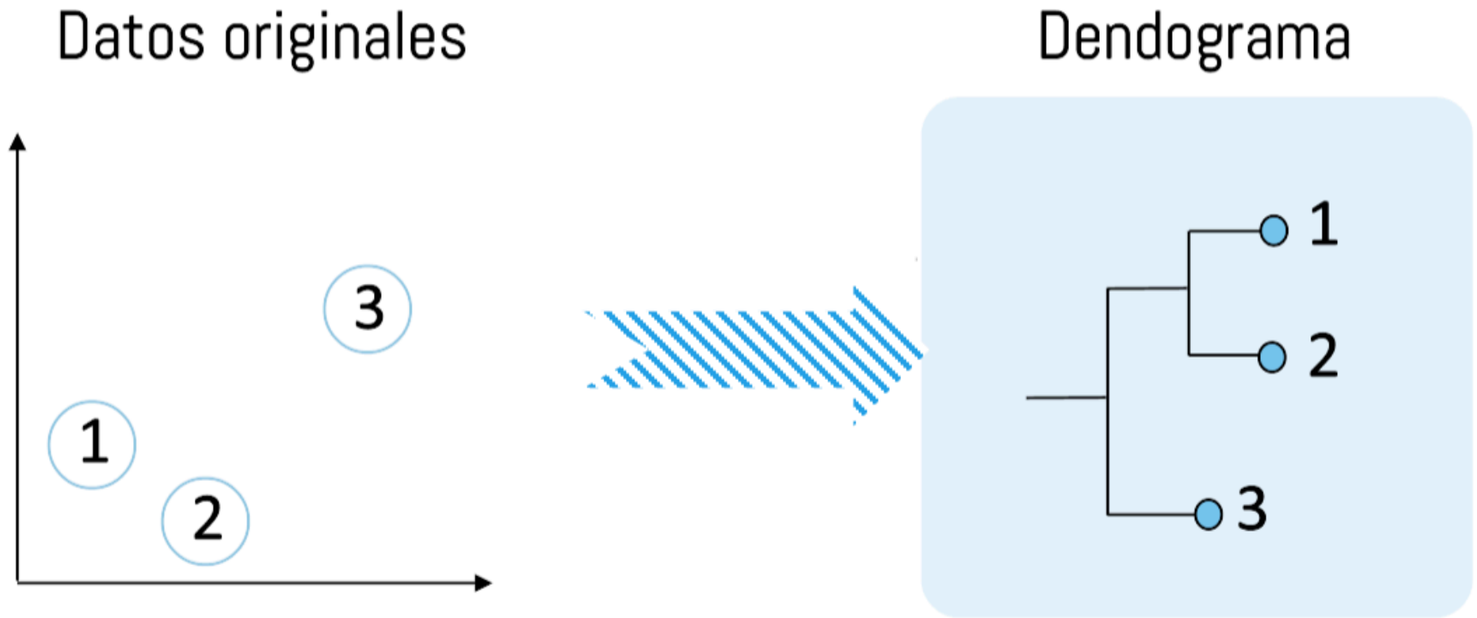
Dendograma

> Corresponde a un diagrama en el que se muestran las distancias de atributos entre clases que son parte de un mismo cluster.

Clustering: Jerarquía

> Los algoritmos basados en jerarquía pueden seguir 2 estrategias:

- **Aglomerativos**: Comienzan con cada objeto como un grupo (bottom-up). Estos grupos se van combinando sucesivamente a través de una métrica de similaridad. `Para n objetos se realizan n-1 uniones`.
- **Divisionales**: Comienzan con un solo gran cluster (bottom-down). Posteriormente este mega-cluster es dividido sucesivamente de acuerdo a una métrica de similaridad.



Clustering Aglomerativo

Clustering Aglomerativo: Algoritmo

Algoritmo

1. Inicialmente se considera **cada punto como un cluster**.
2. Calcula la matriz de *`proximidad/distancia`* entre cada cluster.
3. Repetir `(hasta que exista un solo cluster)`:
    - Unir los cluster más cercanos.
    - Actualizar la matriz de *`proximidad/distancia`*.

ⓘ Lo más importante de este proceso es el cálculo de la matriz de proximidad/distancia entre clusters

ⓘ Distintos enfoques de distancia entre clusters, segmentan los datos en forma distinta.

Clustering Aglomerativo: Ejemplo

Supongamos que tenemos cinco tipos de genes cuya expresión ha sido determinada por 3 caracteríticas. Las siguientes expresiones pueden ser vistas como la expresión dados los genes en tres experimentos.

> Apliquemos un Clustering Jerárquico Aglomerativo utilizando como medida de similaridad la `Distancia Euclideana`.

Otros tipos de distancia también son aplicables siguiendo un procedimiento análogo.

Características

Datos

Gen	Alfa	Beta	Gamma
p53	9	3	7
mdm2	10	2	9
bcl2	1	9	4
CylinE	6	5	5
Caspade	1	10	3

	p53	mdm2	bcl2	CulynE	Caspade
p53	0	2.45	10.44	4.12	11.36
mdm2	2.45	0	12.45	6.40	13.45
bcl2	10.44	12.45	0	6.48	1.41
CulynE	4.12	6.40	6.48	0	7.35
Caspade	11.36	13.45	1.41	7.35	0

Algoritmo: 1era Iteración

El algoritmo considerará que todos los puntos inicialmente son un cluster. Por lo tanto, tratará de encontrar los 2 puntos más cercanos e intentará unirnos en un sólo cluster.

	p53	mdm2	bcl2	CulynE	Caspade
p53	0	2.45	10.44	4.12	11.36
mdm2	2.45	0	12.45	6.40	13.45
bcl2	10.44	12.45	0	6.48	1.41
CulynE	4.12	6.40	6.48	0	7.35
Caspade	11.36	13.45	1.41	7.35	0

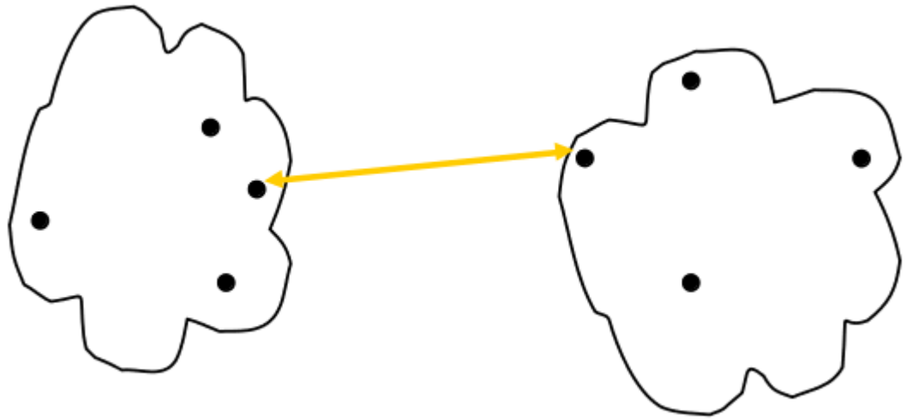
Problema: ¿Cómo actualizamos la matriz de Distancias?

Entonces crearemos un nuevo cluster: `bcl2-Caspade`.

	p53	mdm2	CulynE	bcl2–Caspade
p53	0	2.45	4.12	?
mdm2	2.45	0	6.40	?
CulynE	4.12	6.40	0	?
bcl2–Caspade	?	?	?	0

Clustering Aglomerativo: Single Linkage

- Distancia entre clusters determinada por los puntos más `similares` entre los clusters.



$$D(C_i, C_j) = min\{d(x, y) | x \in C_i, y \in C_j\}$$

Ventajas

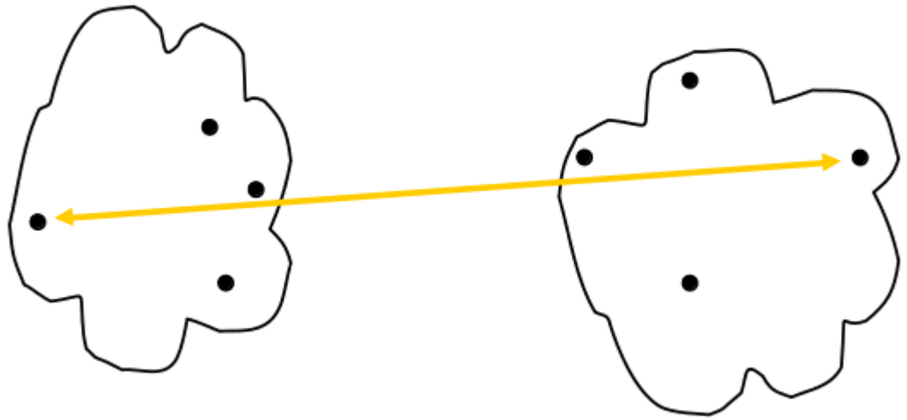
- Genera Clusters largos y delgados.

Limitaciones

- Afectado por Outliers

Clustering Aglomerativo: Complete Linkage

- Distancia determinada por la distancia ente los puntos más *disímiles* entre los clusters.



$$D(C_i, C_j) = max\{d(x, y) | x \in C_i, y \in C_j\}$$

Ventajas

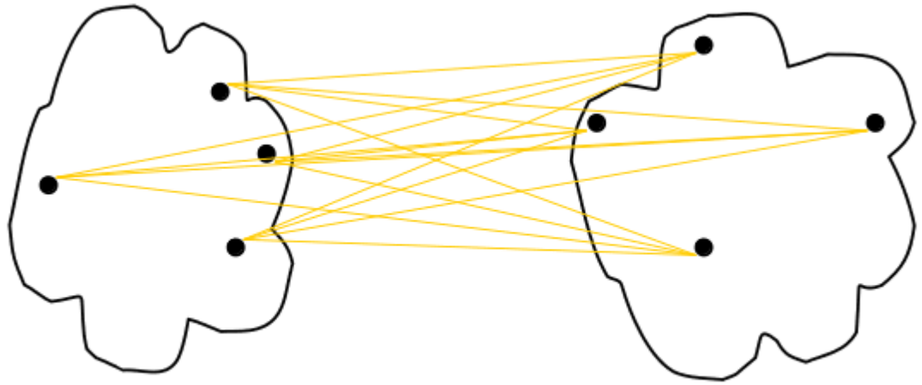
- Menos suceptible a dato atípicos.

Limitaciones

- Tiende a quebrar Clusters Grandes.
- Tiene tendencia a generar Clusters circulares.

Clustering Aglomerativo: Average Linkage

- Distancia determinada por el promedio de las distancias que componen los clusters.
- Punto intermedio entre *Single* y *Complete*.



$$D(C_i, C_j) = avg\{d(x, y) | x \in C_i, y \in C_j\}$$

Ventajas

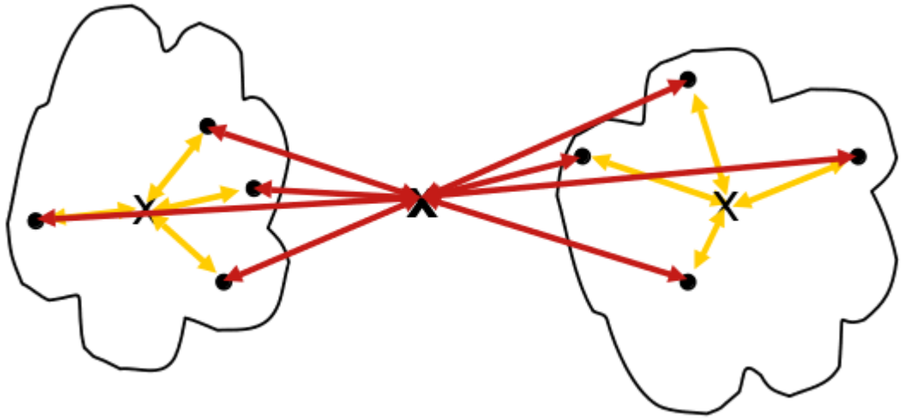
- Menos suceptible a datos atípicos.

Limitaciones

- Tiende a generar clusters circulares.

Clustering Aglomerativo: Ward Linkage

- Distancia determinada por el incremento del `Within cluster distance`.
- Minimiza la distancia intra cluster y maximiza la distancia entre clusters.



$$D(C_i, C_j) = wc(Cij) - wc(C_i) - wc(C_j) = \frac{n_i \cdot n_j}{n_i + n_j} ||\bar{C}_i - \bar{C}_j||^2$$

Ventajas

- Menos suceptible a dato atípicos.

Limitaciones

- Tiende a generar clusters circulares.

Hiperparámetros

Los Hiperparámetros de este modelo serán:

> **Note**
>
> - `linkage`: La forma de calcular la distancia entre clusters.
> - `distancia`: La distancia utilizada como similaridad entre los clusters.

⚠ A diferencia de K-Means, este método no requiere definir el número de Clusters a priori.

Volvamos a la Iteración 1

> Supongamos que por simplicidad utilizaremos `Average Linkage`. (El proceso para utilizar otro linkage es análogo).

	p53	mdm2	CulynE	bcl2–Caspade
p53	0	2.45	4.12	?
mdm2	2.45	0	6.40	?
CulynE	4.12	6.40	0	?
bcl2–Caspade	?	?	?	0

Mean	10.44

	bcl2	Caspade
p53	10.44	11.36

Mean	6.92

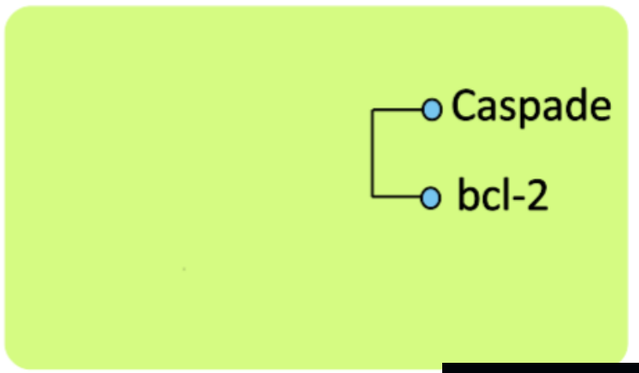
	bcl2	Caspade
CulynE	6.48	7.35

Mean	12.95

	bcl2	Caspade
mdm2	12.45	13.45

Vamos a extraer una Matriz entre los puntos a fusionar y los puntos de los clusters restantes.

Dendograma: 1era Iteración



Iteración 2

	p53	mdm2	CulynE	bcl2-Caspade
p53	0	2.45	4.12	10.90
mdm2	2.45	0	6.40	12.95
CulynE	4.12	6.40	0	6.92
bcl2-Caspade	10.90	12.95	6.92	0

Mean	5.26

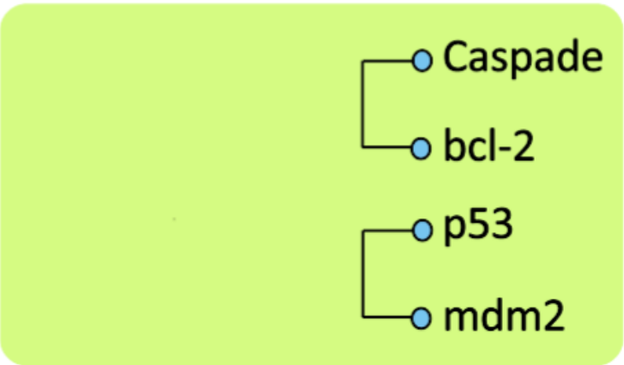
	p53	mdm2
CulynE	4.12	6.40

Mean	11.93

	p53	mdm2
bcl2	10.44	12.45
Caspade	11.36	13.45

	p53-mdm2	CulynE	bcl2-Caspade
p53-mdm2	0	?	?
CulynE	?	0	6.92
bcl2-Caspade	?	6.92	0

Dendograma: 2da Iteración



Iteración 3

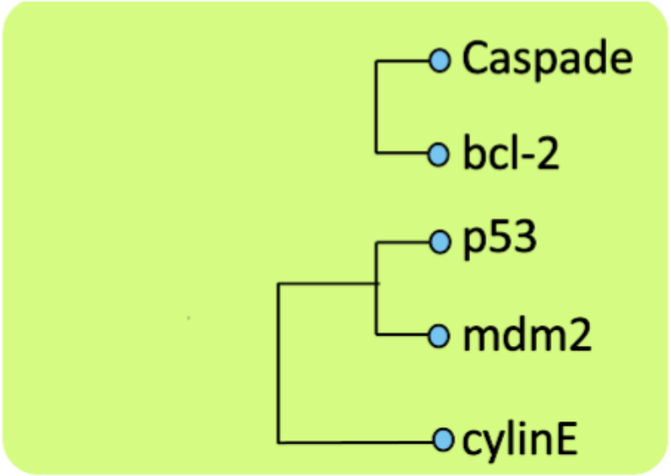
	p53-mdm2	CulynE	bcl2-Caspade
p53-mdm2	0	5.26	11.93
CulynE	5.26	0	6.92
bcl2-Caspade	11.93	6.92	0

Mean	10.30

	p53	mdm2	CulynE
bcl2	10.44	12.45	6.48
Caspade	11.36	13.45	7.35

Dendograma: 3ra Iteración

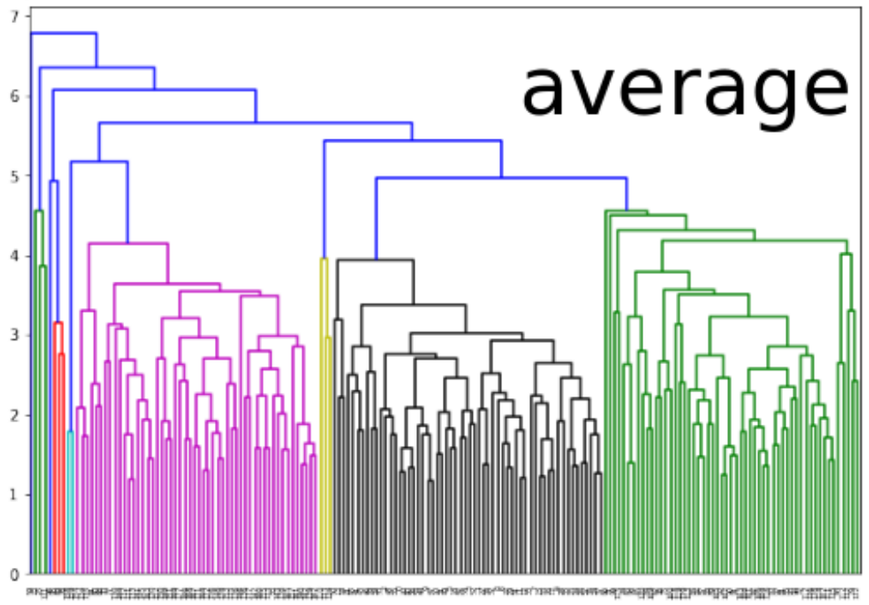
	p53-mdm2-CulynE	bcl2-Caspade
p53-mdm2-CulynE	0	?
bcl2-Caspade	?	0

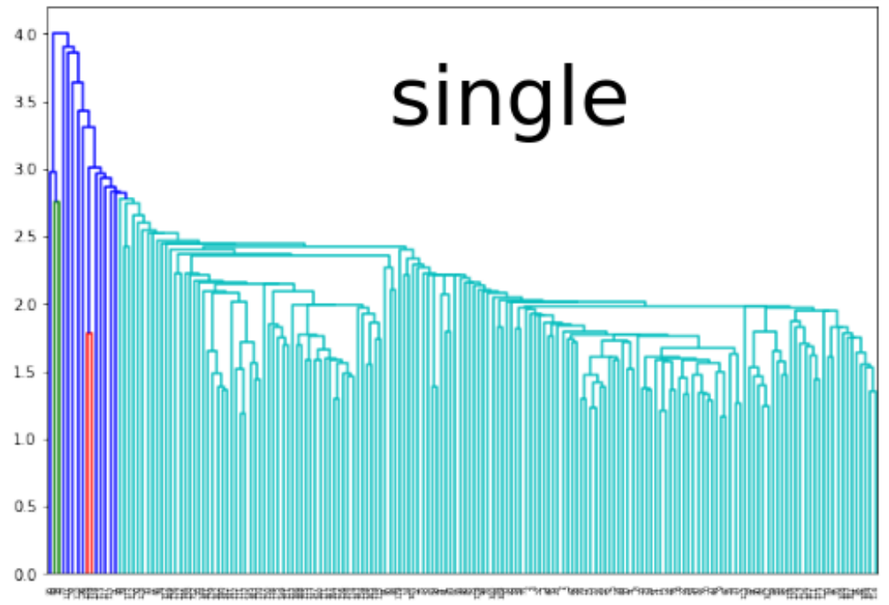


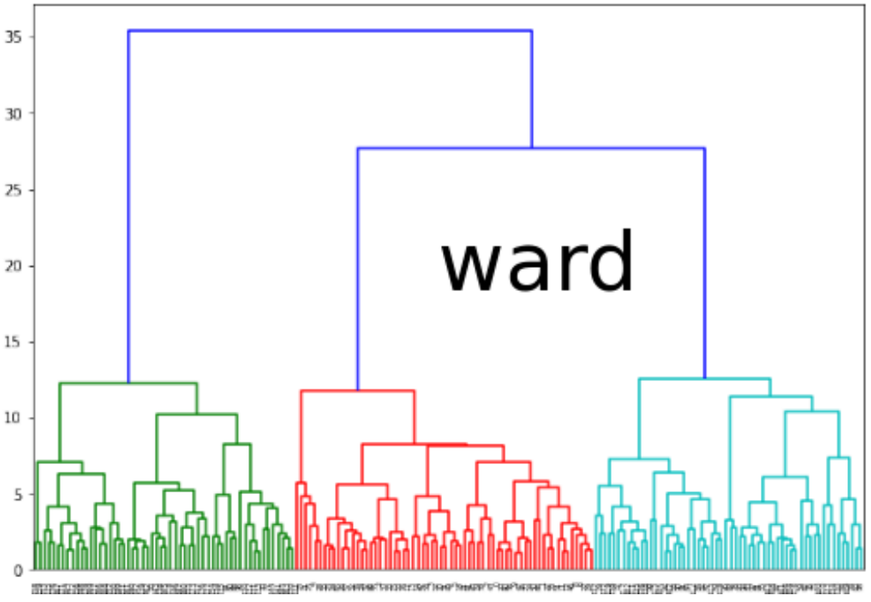
Dendograma Resultante

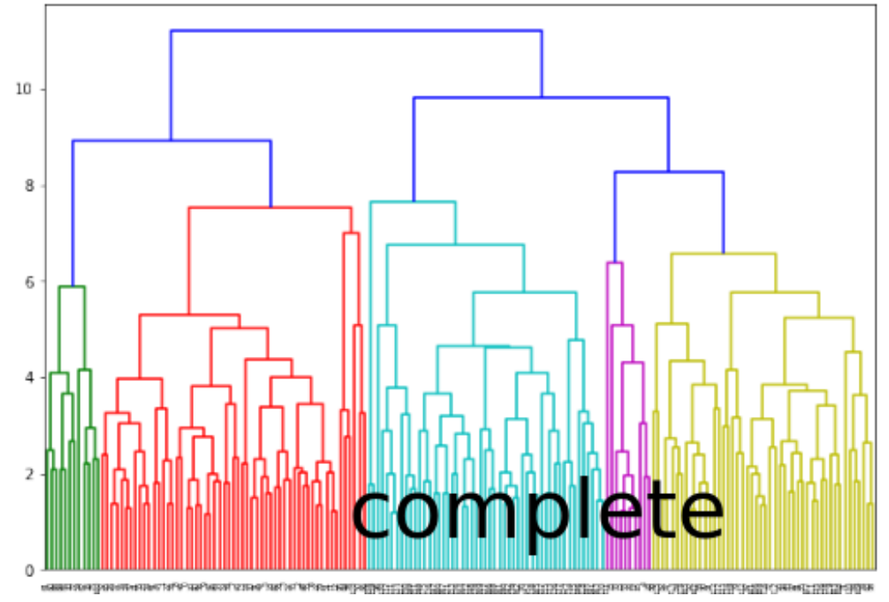
No es necesario realizar la última iteración ya que se entiende que ambos clusters se unen.

Efecto del Linkage Escogido









Clustering Jerárquico: Detalles Técnicos

ⓘ Fortalezas

- No requiere definir el número de Clusters a priori.
- Al tener distintas variantes es posible que los puntos sean agrupados de manera completamente distintas.

⚠ Debilidades

- Muy ineficiente computacionalmente debido a que genera una nueva matriz de distancia en cada iteración lo que entrega una complejidad $O(n^2)$ o $O(n^3)$ dependiendo del linkage.
- Una vez que se decide combinar 2 clusters no es posible revertir esta decisión.
- No tiene capacidad de generalización, ya que no es posible aplicarlo a datos nuevos.

Implementación en Scikit-Learn

```
1 from sklearn.cluster import AgglomerativeClustering
2
3 km = AgglomerativeClustering(n_clusters=2, metric="euclidean",linkage="ward")
4
5 ## Se entrena y se genera la predicción
6 km.fit_predict(X)
```

- **n_clusters**: Define el número de clusters a crear, por defecto 2.
- **metric**: Permite distancias L1, L2 y coseno.
- **linkage**: Permite single, complete, average y ward.
- `.fit_predict()`: Entrenará el modelo en los datos suministrados e inmediatamente genera el cluster asociado a cada elemento.

⚠
- Si bien el método de Aglomeración no requiere el número de clusters a generar, Scikit-Learn lo exige de modo de poder etiquetar cada elemento.

¿Por qué no existen los métodos `.fit()` y `.predict()` por separado?

Otras implementaciones (Dendograma)

```
1  from scipy.cluster.hierarchy import dendrogram, linkage
2
3  # Genera los cálculos necesarios para construir el Histograma.
4  Z = linkage(X, method='single', metric="euclidean")
5
6  # Graficar el Dendograma
7  plt.figure(figsize=(10, 5)) # Define el tamaño del Gráfico
8  plt.title('Dendograma Clustering Jerárquico') # Define un título para el dendograma
9  plt.xlabel('Iris Samples')
10 plt.ylabel('Distance')
11 dendrogram(Z, leaf_rotation=90., leaf_font_size=8.)
12 plt.show()
```

> Principalmente este código permite graficar el Dendograma completo.

L5-L12: Corresponde al código necesario para graficar el Dendograma.

Sugerencias

Pre-procesamientos

Es importante recordar que el clustering aglomerativo también es un Algoritmo basado en `distancias`, por lo tanto se ve afectado por Outliers y por Escala.

Se recomienda preprocesar los datos con:

- `Winsorizer()` para eliminar Outliers.
- `StandardScaler()` o `MinMaxScaler()` para llevar a una escala común.

Otras técnicas como merge y split, no aplican a este tipo de clustering debido a las limitaciones del algoritmo.

Variantes

> En casos en los que no es posible calcular distancias debido a la presencia de datos categóricos, es posible utilizar el **Gower Dissimilarity** como medida de similitud.

	atributos									
	v_1	v_2	v_3	v_4	v_5	v_6	v_7	v_8	v_9	v_{10}
p_1	1	4	3	5	2	3	1	0	4	0
p_2	0	4	3	2	2	3	1	0	4	1

Gower

Se define como la proporción de variables que tienen distinto valor con respecto al total sin considerar donde ambos son ceros.

$$Gower(p1, p2) = \frac{3}{9}$$

C’est fini