PRESENTATION

Hierarchical Clustering

Anggota Kelompok 6 :

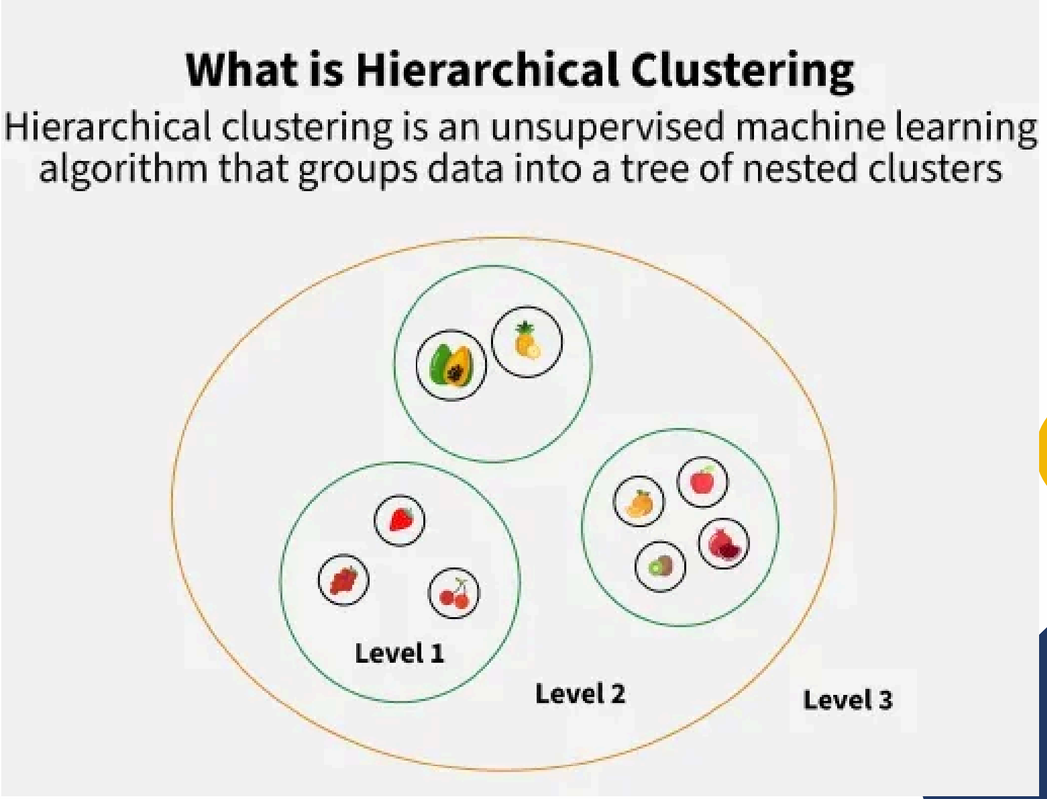
Alif Rahmathul Jadid	(234311030)
Alridzki Innama Nur Razzaaq	(234311031)
Lyan Fairus Atallah	(234311044)
Rahmad Riskiawan H. Saleh	(234311048)

Hierarchical Clustering

Hierarchical clustering adalah metode analisis klaster (clustering) pada machine learning yang bertujuan untuk membangun hierarki klaster, atau pohon klaster, yang disebut dendrogram. Ini adalah metode unsupervised learning, yang berarti metode ini digunakan untuk menemukan pola dalam data tanpa label yang telah ditentukan sebelumnya

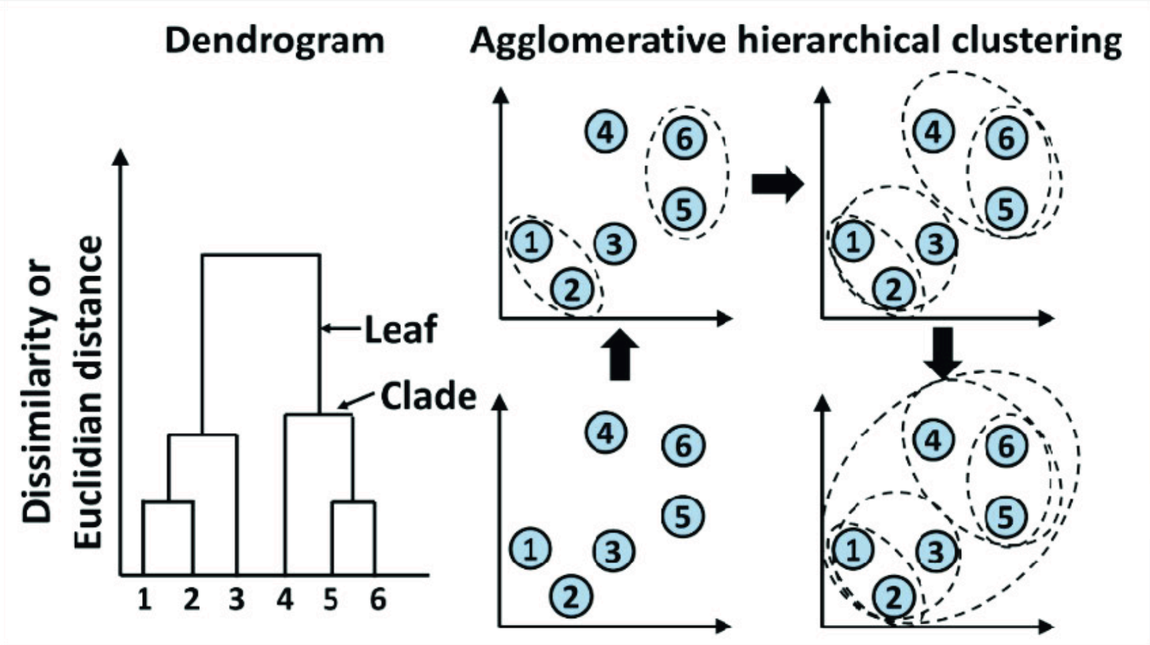
Tujuan:

- Menemukan pola alami dalam data.
- Mengetahui hubungan antar data.
- Melihat proses penggabungan cluster lewat dendrogram.



Agglomerative
Bottom-Up

Agglomerative (Bottom-Up) adalah salah satu metode dalam Hierarchical Clustering yang bekerja dengan cara menggabungkan data dari bawah ke atas.
Setiap data mulanya dianggap sebagai satu cluster terpisah, kemudian cluster yang paling mirip akan digabung secara bertahap hingga akhirnya terbentuk satu cluster besar.



Jenis Linkage

AgglomerativeClustering mendukung strategi linkage Ward, Single, Average, dan Complete :

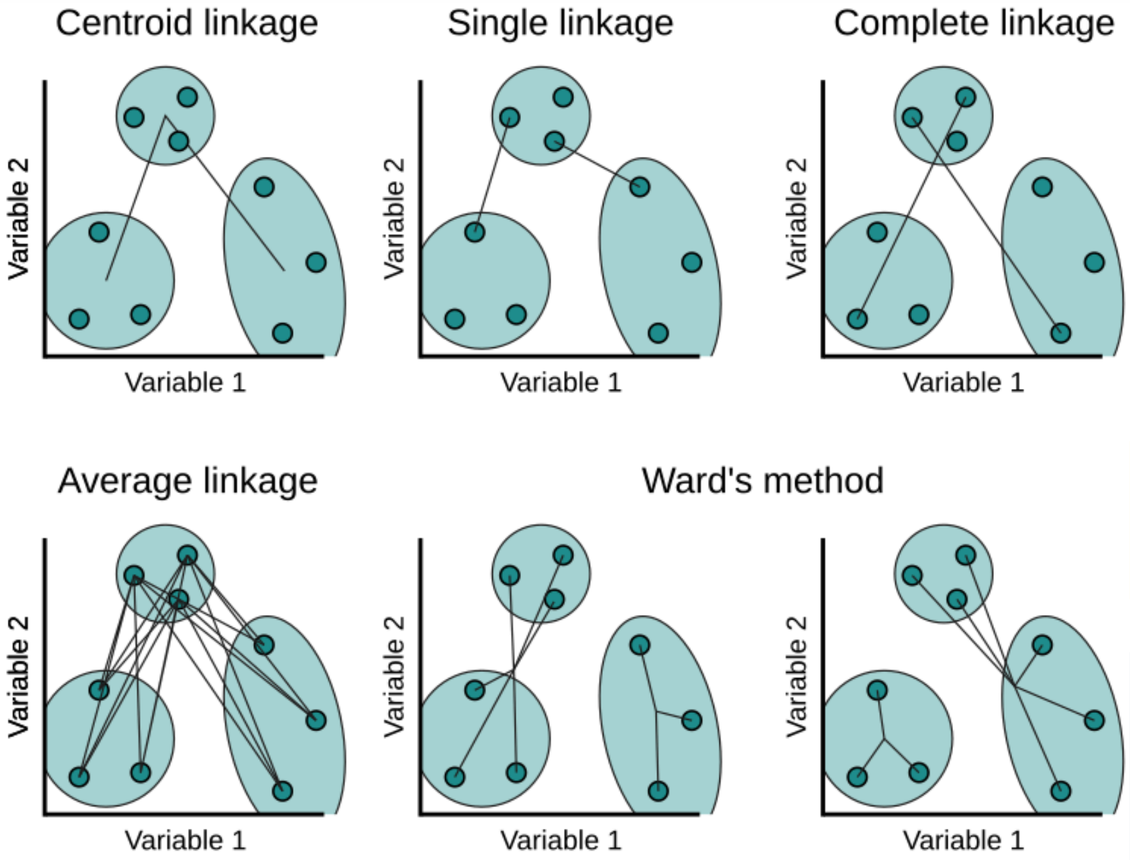
Ward
- Menggabungkan dua klaster dengan kenaikan variansi terkecil.
- Mirip tujuan k-means, tapi menggunakan metode hierarki.
- Menghasilkan ukuran klaster paling seimbang.
- Hanya untuk jarak Euclidean.

Complete Linkage
- Menggunakan jarak maksimum antar titik dalam dua klaster.

Average Linkage
- Menggunakan rata-rata jarak antar titik pada dua klaster.

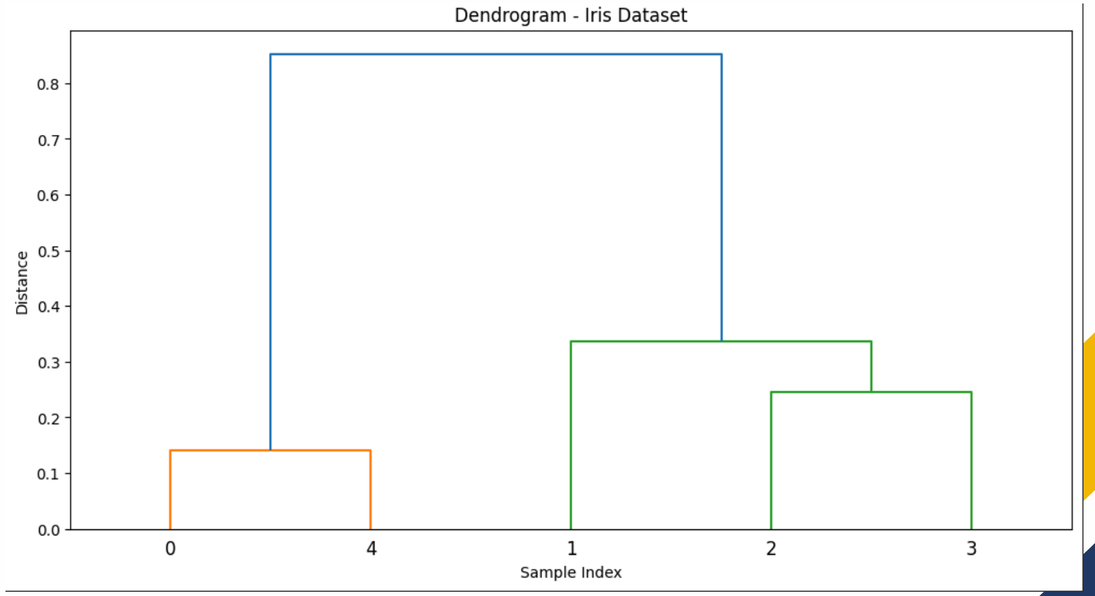
Single Linkage
- Menggunakan jarak minimum antar titik klaster.

Paling sensitif terhadap noise, klaster bisa memanjang (efek chaining).
Tetapi paling cepat dan cocok untuk bentuk cluster tidak bulat.



Visualisasi Hierarki
Dendrogram

Dendrogram adalah representasi visual yang menunjukkan bagaimana cluster terbentuk dan hubungan antar cluster. Ketinggian cabang dalam dendrogram menunjukkan jarak atau perbedaan saat cluster digabungkan. Cabang yang lebih pendek berarti data tersebut lebih mirip.



Hipotesa Function

Hipotesa dalam Hierarchical Clustering adalah bahwa dua cluster yang paling mirip, sesuai dengan cost function atau kriteria jarak yang digunakan, akan digabungkan menjadi satu cluster. Jika menggunakan single linkage, hipotesisnya adalah bahwa cluster yang memiliki pasangan titik paling dekat merupakan cluster yang paling mirip. Pada complete linkage, cluster dianggap mirip jika jarak titik terjauhnya paling kecil. Pada average linkage, kemiripan cluster ditentukan dari rata-rata jarak seluruh titik antar cluster. Sedangkan pada Ward linkage, hipotesisnya adalah bahwa cluster terbaik untuk digabung adalah yang menghasilkan kenaikan SSE (variansi intra-cluster) paling kecil. Dengan demikian, hipotesa function secara konseptual menyatakan bahwa kesamaan antar cluster ditentukan sepenuhnya oleh cost function yang digunakan, dan penggabungan dilakukan berdasarkan nilai cost terkecil.

Cost Function pada Ward Linkage

Cost function dalam Hierarchical Clustering bekerja dengan menentukan dua cluster yang paling mirip berdasarkan kriteria tertentu, lalu menggabungkannya secara bertahap hingga jumlah cluster yang diinginkan tercapai. Pada Ward linkage, kemiripan dihitung dengan mencari pasangan cluster yang menghasilkan kenaikan SSE (Sum of Squared Error) paling kecil, sehingga cluster tetap kompak. Sementara itu, metode lain seperti single, complete, dan average linkage menggunakan jarak antar cluster sebagai dasar penggabungan: single memilih jarak titik terdekat, complete memilih jarak titik terjauh, dan average menghitung rata-rata jarak antar titik kedua cluster. Dengan kata lain, setiap langkah penggabungan ditentukan oleh nilai cost paling kecil sesuai aturan linkage yang digunakan.

$$\Delta E = SSE_{\text{baru}} - SSE_{\text{awal}}$$

SSE (Sum of Squared Error) didefinisikan sebagai:

$$SSE = \sum_{i=1}^{n} \|x_i - \mu\|^2$$

Ward akan memilih dua cluster A dan B yang bila digabung menambah error total paling kecil, yaitu

$$\Delta E(A,B) = \frac{|A||B|}{|A|+|B|} \|\mu_A - \mu_B\|^2$$

Dengan:

- x_i =data ke-i
- μ = centroid/muatan rata-rata cluster
- n = jumlah data dalam cluster

Interpretasi Rumus:

$\|\mu_A - \mu_B\|^2$

jarak kuadrat antara centroid cluster A dan B

$\frac{|A||B|}{|A|+|B|}$

faktor ukuran cluster (semakin besar cluster makin besar dampak penggabungan)

Cluster yang digabung adalah yang menghasilkan nilai ΔE paling kecil.

Cost Function
Linkage

Single link: $D(c_1,c_2) = \min_{x_1 \in c_1, x_2 \in c_2} D(x_1,x_2)$

- distance between closest elements in clusters
- produces long chains a→b→c→…→z

Complete link: $D(c_1,c_2) = \max_{x_1 \in c_1, x_2 \in c_2} D(x_1,x_2)$

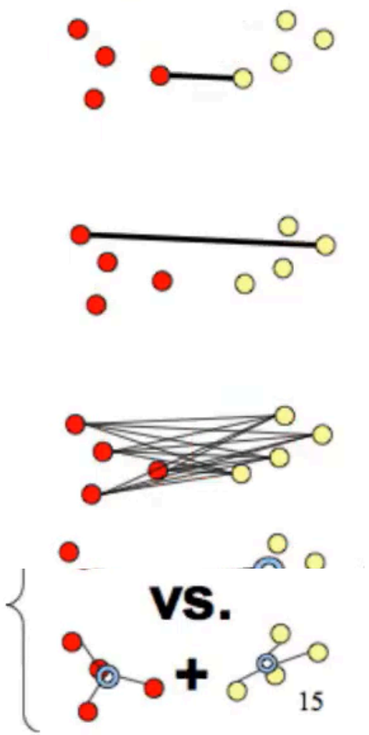
- distance between farthest elements in clusters
- forces "spherical" clusters with consistent "diameter"

Average link: $D(c_1,c_2) = \sum_{x_1 \in c_1} \sum_{x_2 \in c_2} D(x_1,x_2)$

- average of all pairwise distances
- less affected by outliers

Ward's method: $TD_{c_1 \cup c_2} = \sum_{x \in c_1 \cup c_2} D(x, \mu_{c_1 \cup c_2})^2$

- consider joining two clusters, how does it change the total distance (TD) from centroids?



Variasi
Metrik Jarak

distance metrics

Linkage single, average, dan complete dapat digunakan dengan berbagai jenis jarak, seperti:

- Euclidean (L2) : Default, umum digunakan
- Manhattan (L1) : Cocok untuk data sparse
- Cosine distance : Tidak dipengaruhi skala data
- Precomputed affinity matrix : Bisa pakai jarak custom

Pedoman pemilihan metrik:

- pilih metrik yang memperbesar jarak antar kelas
- dan memperkecil jarak dalam kelas

Connectivity Constraints

AgglomerativeClustering dapat diberi batasan konektivitas, yaitu hanya klaster yang saling berdekatan (berdasarkan graf/kedekatan lokal) yang boleh digabung.
Contoh penggunaan:

- Data berbentuk swiss roll: mencegah klaster menyatu pada lipatan yang tidak berdekatan.
- Klaster halaman web hanya jika saling terhubung link.
- Klaster pixel gambar menggunakan grid adjacency.

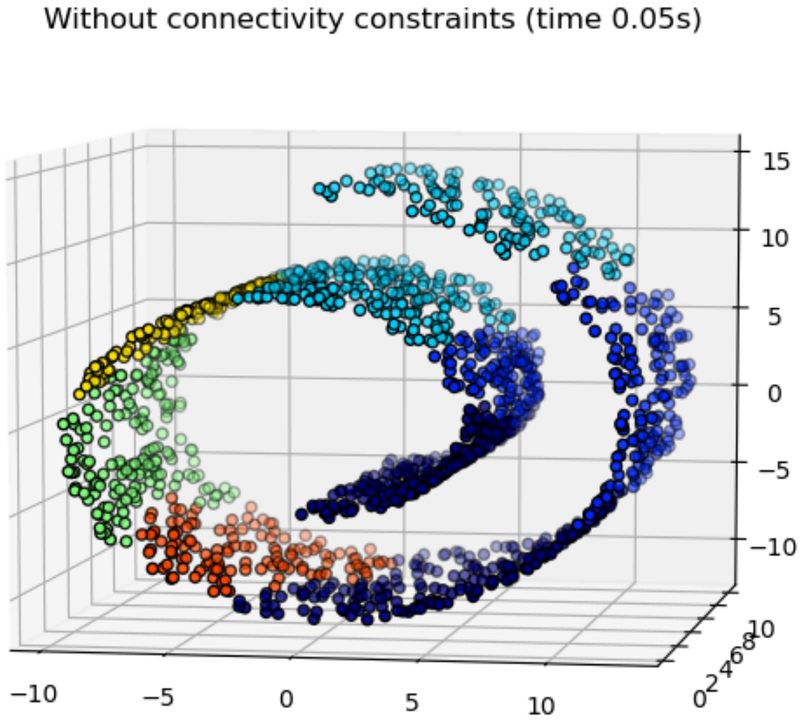
Kelebihan:

- Menjaga struktur lokal
- Membuat algoritma lebih cepat untuk dataset besar

Peringatan:
Jika digabung dengan single/average/complete linkage, bisa memunculkan klaster besar vs klaster kosong (efek "rich get richer").

Konektivitas dibuat dengan:

- kneighbors_graph
- grid_to_graph



StudiKasus

Pada studi kasus ini, Hierarchical Clustering digunakan untuk mengelompokkan data bunga Iris berdasarkan empat fitur fisik seperti panjang dan lebar sepal serta petal. Algoritma bekerja secara agglomerative, yaitu menggabungkan sampel yang paling mirip secara bertahap hingga terbentuk struktur hierarki. Dengan menggunakan Ward linkage, cluster yang terbentuk menjadi lebih rapi dan terpisah jelas. Hasilnya menunjukkan bahwa spesies Iris Setosa terkelompok sangat baik karena memiliki ciri fisik yang berbeda, sementara Versicolor dan Virginica berada dalam cluster yang lebih dekat karena kemiripannya. Dendrogram membantu melihat proses penggabungan cluster dan menunjukkan bahwa tiga cluster adalah jumlah yang optimal. Studi kasus ini menunjukkan bahwa Hierarchical Clustering dapat menemukan pola alami tanpa membutuhkan label data.

Kode

```python
import matplotlib.pyplot as plt
from sklearn.datasets import load_iris
from sklearn.cluster import AgglomerativeClustering
from scipy.cluster.hierarchy import dendrogram, linkage
import numpy as np

# 1. Load dataset asli Iris
iris = load_iris()
X = iris.data[:5]
y = iris.target [:5]  # untuk evaluasi saja (unsupervised)

# 2. Hierarchical Clustering (Agglomerative)
model = AgglomerativeClustering(
    n_clusters=3,
    linkage='ward'
)

labels = model.fit_predict(X)
print("Cluster hasil:", labels)
```

```python

# 3. Visualisasi
plt.scatter(X[:, 2], X[:, 3], c=labels, s=80)
plt.xlabel("Petal Length")
plt.ylabel("Petal Width")
plt.title("Hierarchical Clustering on Iris Dataset")
plt.show()

# 4. Dendrogram
linked = linkage(X, method='ward')

plt.figure(figsize=(12, 6))
dendrogram(linked, truncate_mode='level', p=5)
plt.title("Dendrogram - Iris Dataset")
plt.xlabel("Sample Index")
plt.ylabel("Distance")
plt.show()

```

Penjelasan
Kode

```
# 1. Load dataset asli Iris
iris = load_iris()
X = iris.data[:5]
y = iris.target[:5]   # untuk evaluasi saja (unsupervised)
```

Load dataset Iris

Penjelasan :

- iris.data berisi 5 sampel × 5 fitur
- (sepal length, sepal width, petal length, petal width)

Penjelasan
Kode

```
12 # 2. Hierarchical Clustering (Agglomerative)
13 model = AgglomerativeClustering(
14     n_clusters=3,
15     linkage='ward'
16 )
```

Membuat model Agglomerative Clustering
Penjelasan:

- n_clusters=3 → kita ingin membagi menjadi 3 cluster
- linkage="ward" → menggabungkan cluster berdasarkan minimasi SSE,
- cocok untuk data berbasis jarak Euclidean.

Ward linkage menghasilkan cluster yang bentuknya paling bulat/rapi.

Penjelasan Kode

```
18 labels = model.fit_predict(X)
19 print("Cluster hasil:", labels)
```

Melatih model & mendapatkan hasil clusterPenjelasan:

- fit_predict(X) melakukan 2 hal:
  a. menghitung jarak antar sampel
  b. mengelompokkan sampel menjadi cluster

Penjelasan Kode

```
# 3. Visualisasi
plt.scatter(X[:, 2], X[:, 3], c=labels, s=80)
plt.xlabel("Petal Length")
plt.ylabel("Petal Width")
plt.title("Hierarchical Clustering on Iris Dataset")
plt.show()
```

Visualisasi hasil clustering

Penjelasan:

- visualisasi, hanya dipakai 2 fitur:
  - petal length (kolom 2)
  - petal width (kolom 3)
- c=labels → memberi warna berdasarkan cluster
- Grafik ini menunjukkan apakah cluster terbentuk dengan baik.

Penjelasan Kode

Membuat dendrogram
Penjelasan:

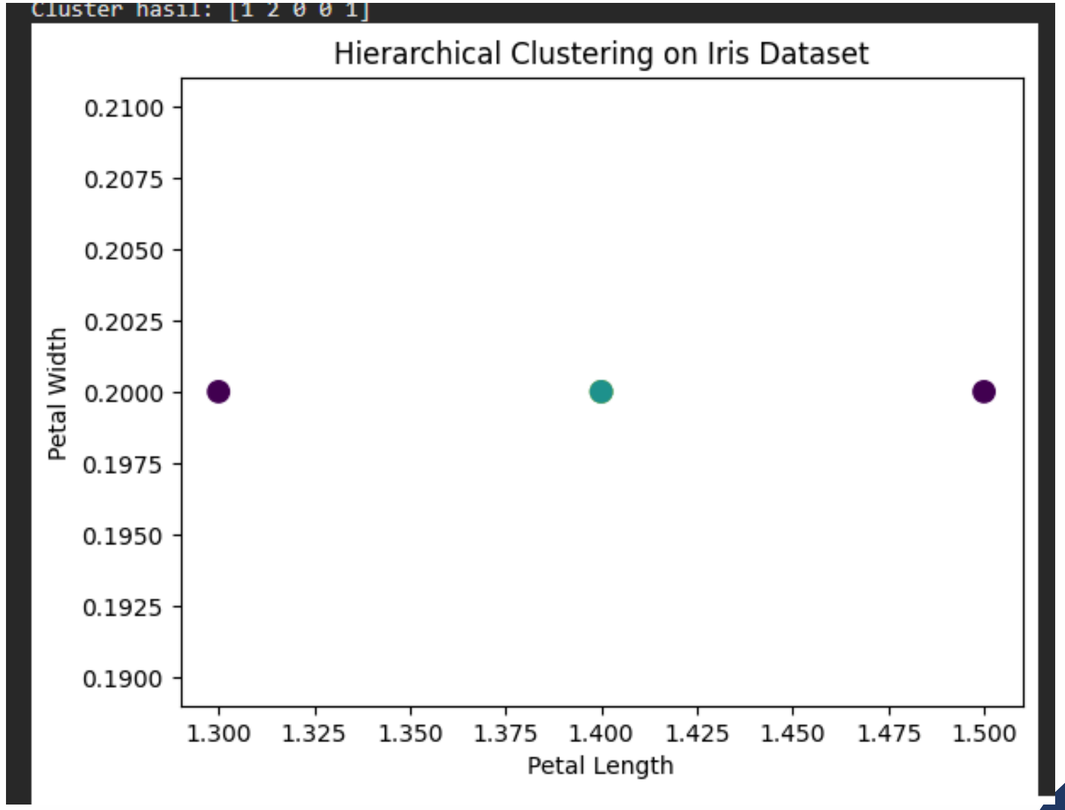
- linkage(X, method='ward') → menghitung jarak antara cluster pada setiap proses penggabungan
- dendrogram() → memvisualisasikan struktur pohon hierarki
- truncate_mode='level', p=5 → memotong dendrogram agar tidak terlalu panjang (lebih rapi)
- Sumbu-Y merepresentasikan “jarak” atau “biaya” saat cluster digabung.

```
28 # 4. Dendrogram
29 linked = linkage(X, method='ward')
30
31 plt.figure(figsize=(12, 6))
32 dendrogram(linked, truncate_mode='level', p=5)
33 plt.title("Dendrogram - Iris Dataset")
34 plt.xlabel("Sample Index")
35 plt.ylabel("Distance")
36 plt.show()
```

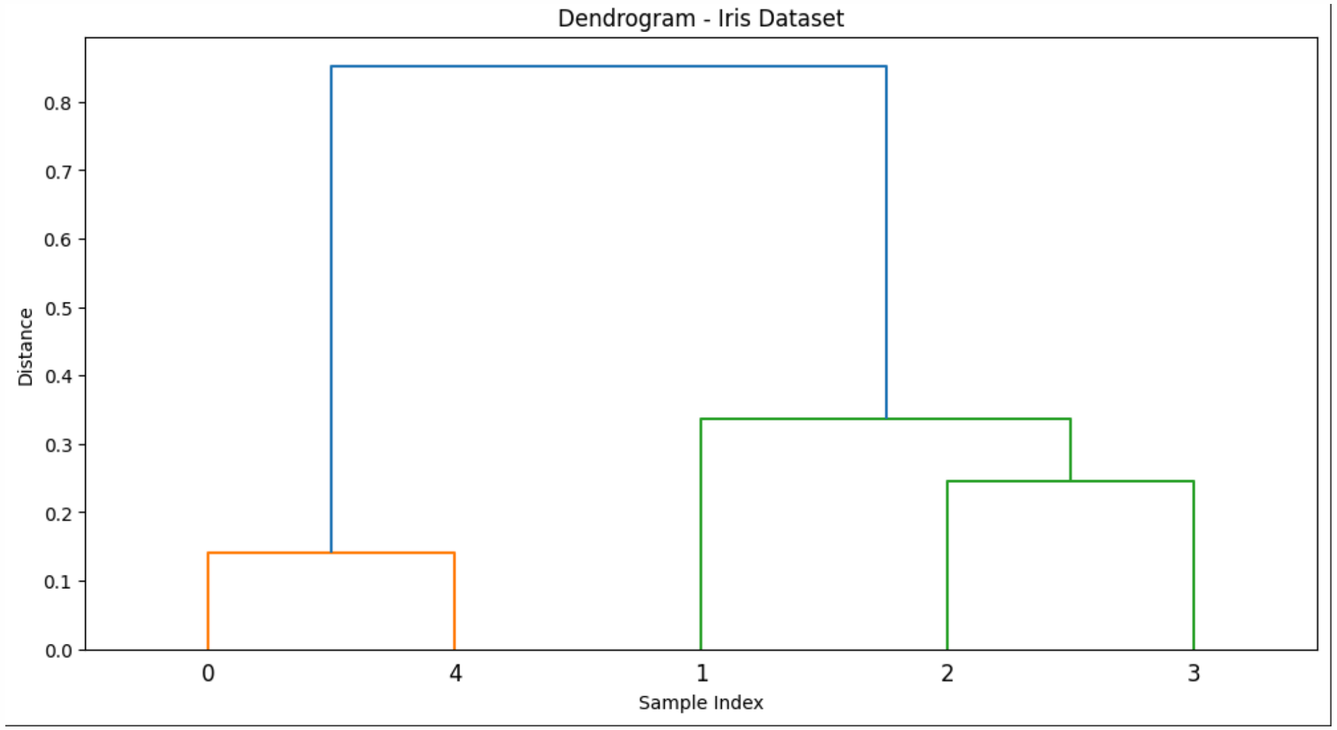
Hasil Kode

Penjelasan :
Cluster Hijau adalah iri Setosa petal pendek dan sempit
Cluster Ungu adalah iris Versicolor petal sedang
Cluster Kuning adalah iris Virginica panjang dan lebar

Hierarchical Clustering berhasil memetakan spesies Iris menjadi 3 cluster yang jelas terpisah berdasarkan kemiripan fitur fisiknya.



Penjelasan Hasil Kode



membaca dendrogram dari bawah ke atas. Garis rendah = objek mirip. Garis tinggi = kurang mirip. Garis potong horizontal menentukan jumlah cluster.

Dendrogram menunjukkan:
Struktur hierarki data.
Seberapa mirip/mirip cluster satu dengan lainnya.
Bahwa Iris dataset secara alami membentuk 3 kelompok besar.

Kesimpulan

Pada studi kasus dataset Iris, hierarchical clustering mampu mengelompokkan 150 data bunga berdasarkan 4 fitur fisik menjadi tiga cluster utama. Hasil visualisasi menunjukkan bahwa:

Cluster Setosa terpisah paling jelas karena memiliki karakteristik petal yang berbeda.
Versicolor dan Virginica berada dalam cluster yang berdekatan karena memiliki kemiripan fitur.
Dendrogram memperlihatkan proses penggabungan dari penggabungan kecil di bagian bawah hingga penggabungan terbesar di bagian atas, dan menunjukkan bahwa tiga cluster merupakan jumlah yang paling optimal.
Secara keseluruhan, metode Hierarchical Clustering cocok digunakan ketika jumlah data tidak terlalu besar, ketika ingin memahami struktur alami data, serta ketika jumlah cluster belum diketahui sejak awal. Metode ini unggul dalam visualisasi, interpretasi pola, dan kemampuannya menemukan struktur hierarki dalam dataset.

Thank You