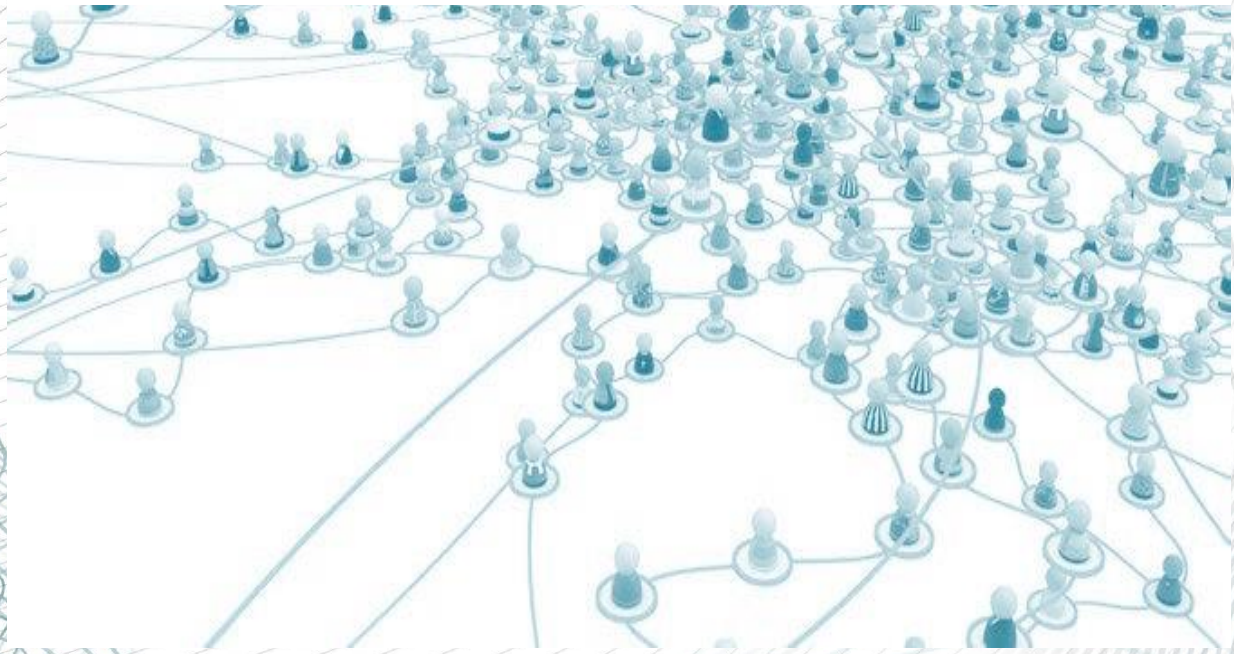
MODUL PRAKTIKUM

SD3107-Pembelajaran Mesin



Program Studi Sains Data
Fakultas Sains
Institut Teknologi Sumatera

2024

MODUL 5

Hierarchical Agglomerative Clustering

A. Konsep Dasar

Hierarchical Agglomerative Clustering (HAC) adalah metode pengelompokan data secara hierarki dengan pendekatan bottom-up. Terdapat dua jenis strategi pengelompokan, yaitu Agglomerative (dari bawah ke atas) dan Divisive (dari atas ke bawah). Agglomerative dimulai dari setiap data adalah klaster itu sendiri, kemudian untuk setiap klaster yang sekiranya memiliki kesamaan, akan dimerge. Divisive dimulai dari seluruh data yang ditempatkan dalam satu klaster, kemudian dibagi sampai setiap data memiliki klasternya sendiri.

Pada Hierarki Aglomeratif, terdapat beberapa metode pengelompokkan yaitu Single Linkage (Jarak terdekat) yang dapat dilakukan dengan menggunakan nilai minimum dari setiap pasangan vektor fitur. Metode kedua yaitu Complete Linkage (Jarak terjauh) dipakai menggunakan nilai maksimum dari setiap pasangan vektor fitur. Ketiga yaitu metode Average Linkage (Jarak Rerata) bekerja dengan menggunakan rerata dari setiap titik yang berada dalah satu klaster. Berikut adalah algoritma Hierarchical Agglomerative Clustering:

a) Hitung matriks jarak antar data
b) Gabungkan dua kelompok terdekat berdasarkan parameter kedekatan yang ditentukan.
c) Perbarui matriks jarak antar data untuk merepresentasikan kedekatan diantara kelompok baru dan kelompok yang masih tersisa.
d) Ulangi langkah 2 dan 3 sampai terdapat satu kelompok yang tersisa.
e) Selesai

Dalam menyelesaikan persoalaan pengelompokan, misalkan x dan y adalah vektor fitur dengan dimensi d. Ada beberapa formula yang dapat digunakan untuk menghitung jarak antara dua vektor fitur yaitu :

- Manhattan Distance:

$$D(x,y) = \sum_{i=1}^{d} |x_i - y_i| \quad \text{..... (1)}$$

- Euclidean Distance:

$$D(x,y) = \sqrt{\sum_{i=1}^{d} (x_i - y_i)^2} \text{ .....(2)}$$

Keunggulan HAC adalah tidak memerlukan jumlah klaster awal dan hasilnya bisa divisualisasikan dengan dendrogram, memberikan gambaran hubungan antar data. Namun, HAC membutuhkan waktu komputasi yang tinggi untuk dataset besar dan sulit diubah setelah proses selesai, sehingga kurang fleksibel untuk data yang sangat beragam.

B. Tujuan Praktikum

I. Tujuan Instruksional Umum

Praktikum bertujuan untuk menerapkan teori Hierarchical Agglomerative Clustering untuk pengelompokan data dalam pembelajaran mesin.

II. Tujuan Instruksional Khusus

1. Mahasiswa mampu menguasai konsep dasar Hierarchical Agglomerative Clustering.
2. Mahasiswa mampu menyelesaikan studi kasus pengelompokan data menggunakan metode Hierarchical Agglomerative Clustering.
3. Mahasiswa mampu menganalisis hasil dari studi kasus pengelompokan data menggunakan metode Hierarchical Agglomerative Clustering.

C. Dataset dan bahasa pemrograman Python

C.1 Dataset dalam praktikum

Dalam praktikum modul 5 ini kita hanya akan menggunakan data masukan berupa array x dan y.

C.2 Bahasa Pemrograman Python

Pada praktikum kali ini, kita akan menggunakan IDE Python Service dari Google Colab, maka dari itu Anda perlu menyiapkan akun google pribadi untuk mengakses servis ini di akun google masing-masing.

Library yang akan digunakan diantaranya numpy, pandas, dan sklearn.

1) NumPy: digunakan untuk operasi matematika berbasis array dan matriks. NumPy sangat penting dalam komputasi numerik dengan Python pada praktikum ini.
2) matplotlib.pyplot : untuk visualisasi data.
3) Scientific Python (Scipy) : menyediakan alat untuk komputasi ilmiah dan

teknis. Scipy memiliki berbagai modul untuk optimasi, aljabar linear, integrasi, dan statistik, serta mendukung analisis data, termasuk fungsi untuk clustering, pemrosesan sinyal, dan pemodelan matematis.

Dalam praktikum :

from scipy.cluster.hierarchy import dendrogram, linkage

4) Scikit-learn (sklearn): library untuk implementasi berbagai algoritma machine learning seperti klasifikasi, regresi, clustering, dan reduksi dimensi.

Dalam praktikum :

from sklearn.cluster import AgglomerativeClustering

Praktikum pembelajaran mesin pada modul ini dapat maksimal anda praktikan dengan pemahaman bidang atau mata kuliah terkait seperti algoritma pemrograman, struktur data, dan data mining (data preprocessing).

~~*Dataset dan Source code akan diberikan saat praktikum berlangsung dan dapat anda unduh pada folder berikut* Modul 5 PM 2024 .~~

D. Implementasi Hierarchical Agglomerative Clustering

D.1 Library Scientific Python (Scipy)

1) Import Library

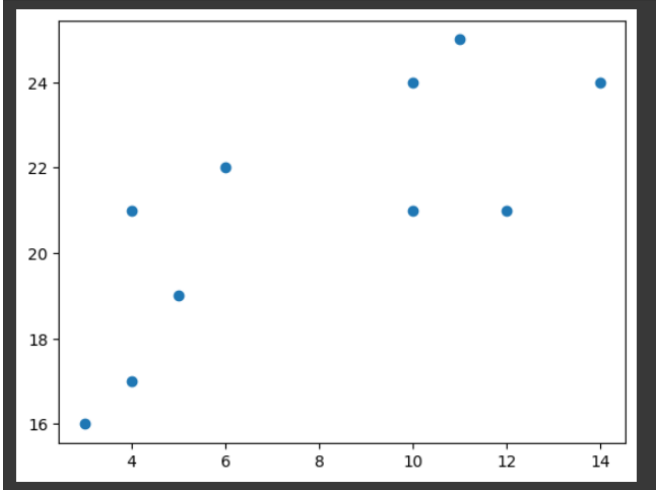
```
import numpy as np
import matplotlib.pyplot as plt
from scipy.cluster.hierarchy import dendrogram, linkage
```

`from scipy.cluster.hierarchy import dendrogram, linkage`
Baris ini mengimpor fungsi **`dendrogram`** dan **`linkage`** dari modul **`hierarchy`** dalam pustaka **`scipy.cluster.`** Fungsi **`linkage`** digunakan untuk menghitung jarak atau kemiripan antar data dan menghasilkan informasi yang dibutuhkan dalam clustering hirarkis. Hasil dari **`linkage`** ini kemudian bisa divisualisasikan menggunakan **`dendrogram`**, yang menampilkan struktur hierarki dalam bentuk diagram pohon.

2) Import Dataset

```
x = [4, 5, 10, 4, 3, 11, 14 , 6, 10, 12]
y = [21, 19, 24, 17, 16, 25, 24, 22, 21, 21]

plt.scatter(x, y)
plt.show()
```



Inputkan data pada x dan y. Lakukan visualisasi data awal untuk dapat melihat titik pada vektor (x,y).

3) Pengelompokan aglomeratif

```
data = list(zip(x, y))

linkage_data = linkage(data, method='complete', metric='euclidean')
dendrogram(linkage_data)
```

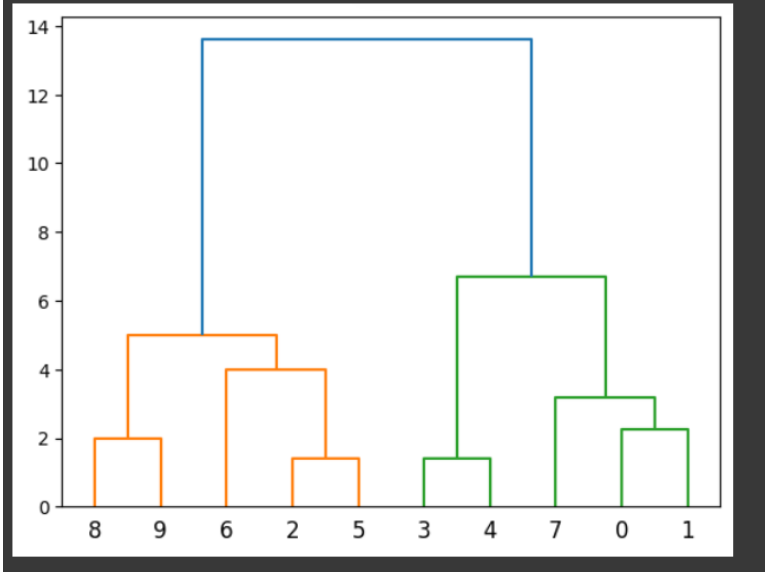
Sesuaikan metode yang ingin digunakan dengan merubah nama metode pada fungsi linkage dan formula metric yang akan digunakan untuk menghitung jarak.

Berikut adalah tabel yang menjelaskan fungsi linkage:

`linkage` (y[, method, metric, optimal_ordering])	Perform hierarchical/agglomerative clustering.
`single` (y)	Perform single/min/nearest linkage on the condensed distance matrix `y`.
`complete` (y)	Perform complete/max/farthest point linkage on a condensed distance matrix.
`average` (y)	Perform average/UPGMA linkage on a condensed distance matrix.
`weighted` (y)	Perform weighted/WPGMA linkage on the condensed distance matrix.
`centroid` (y)	Perform centroid/UPGMC linkage.
`median` (y)	Perform median/WPGMC linkage.
`ward` (y)	Perform Ward's linkage on a condensed distance matrix.

4) Visialisasikan dendrogram

```
plt.show()
```



Dendrogram ini menunjukkan struktur pengelompokan data menggunakan HAC. Sumbu X menampilkan data individu, sedangkan sumbu Y menunjukkan jarak antar klaster pada saat penggabungan terjadi. Pada level rendah, data yang paling mirip digabung terlebih dahulu (misalnya, data 8 dan 9 serta data 6 dan 2). Seiring naiknya level, klaster dengan jarak yang lebih jauh digabung, hingga semua data menyatu pada level tertinggi. Dengan memotong dendrogram di level tertentu (misalnya sekitar Y=6), kita bisa membagi data menjadi dua klaster besar.

D.2 Library Scikit-learn (sklearn)

1) Import Library

```
import numpy as np
import matplotlib.pyplot as plt
from sklearn.cluster import AgglomerativeClustering
```

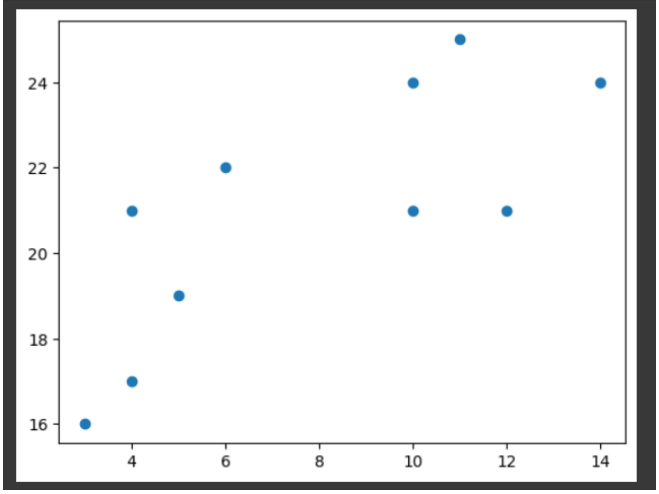
`from sklearn.cluster import AgglomerativeClustering`

Kode ini mengimpor kelas **`AgglomerativeClustering`** dari pustaka **`sklearn.cluster`** yang digunakan untuk menerapkan algoritma clustering aglomeratif. Dengan **`AgglomerativeClustering`**, kita dapat melakukan pengelompokan data secara langsung berdasarkan parameter yang diinginkan, seperti jumlah klaster, metode linkage, dan lain-lain.

2) Import Dataset

```
x = [4, 5, 10, 4, 3, 11, 14 , 6, 10, 12]
y = [21, 19, 24, 17, 16, 25, 24, 22, 21, 21]

plt.scatter(x, y)
plt.show()
```



Inputkan data pada x dan y. Lakukan visualisasi data awal untuk dapat melihat titik pada vektor (x,y).

3) Pengelompokan aglomeratif

```
data = list(zip(x, y))

hierarchical_cluster = AgglomerativeClustering(n_clusters=2, linkage='complete')
labels = hierarchical_cluster.fit_predict(data)
```

hierarchical_cluster membuat sebuah objek AgglomerativeClustering dengan parameter **n_clusters=2** dan **linkage='complete'**. Artinya, model akan melakukan clustering aglomeratif (hierarki) dan membagi data menjadi 2 klaster. Parameter linkage='complete' menunjukkan bahwa model akan menggunakan complete linkage, yaitu metode penggabungan klaster berdasarkan jarak maksimum antara titik-titik di dua klaster. Dengan kata lain, klaster baru akan terbentuk berdasarkan jarak terjauh antar anggota klaster.

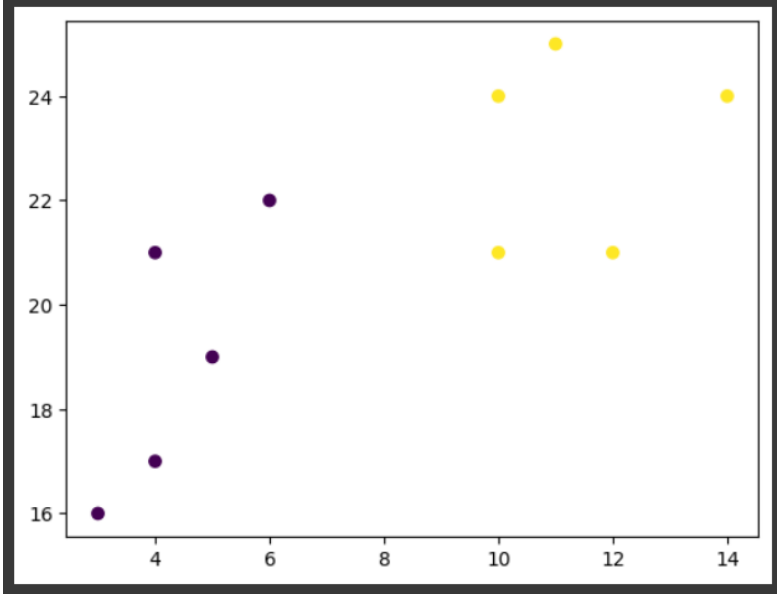
Bagian **labels** didefinisikan melakukan fit dan predict secara bersamaan pada data. Metode **fit_predict** mengelompokkan data sesuai dengan model yang sudah ditentukan dan menghasilkan label klaster untuk setiap titik dalam data. Variabel labels akan menyimpan hasil berupa **label (0 atau 1)** yang menunjukkan klaster mana setiap data termasuk, sesuai dengan dua klaster yang sudah didefinisikan (n_clusters=2).

4) Tampilkan hasil pengelompokan dari label

```
print(labels)
```

```
[0 0 1 0 0 1 1 0 1 1]
```

```
plt.scatter(x, y, c=labels)
plt.show()
```



Baris print(labels) menunjukkan label klaster yang dihasilkan model untuk setiap titik data, yaitu [0 0 1 0 0 1 1 0 1 1]. Angka 0 dan 1 mewakili dua klaster yang berbeda. Titik-titik dengan label 0 berada dalam satu klaster (diwarnai ungu), sementara titik-titik dengan label 1 berada dalam klaster lain (diwarnai kuning). Model *Agglomerative Clustering* mengelompokkan data dengan baik berdasarkan posisi, menghasilkan dua klaster yang jelas terpisah.

TUGAS INDIVIDU

- Tugas individu dikerjakan saat praktikum berlangsung dengan waktu yang ditentukan oleh asisten praktikum.
- Soal tugas individu terbagi menjadi tiga level soal yaitu *Basic*, *Medium*, *dan Hard*. Praktikan dapat menyelesaikan ketiganya untuk mendapatkan poin maksimal.
- Namun praktikan cukup menyelesaikan minimal satu soal jika estimasi waktu praktikum sudah akan berakhir agar tidak melewatkan penilaian test lisan.