

**IMPLEMENTASI DATA MINING UNTUK ANALISIS TREND LAGU DI  
SPOTIFY PADA TAHUN 2023 BERDASARKAN KARAKTERISTIK  
AUDIO MENGGUNAKAN METODE K-MEANS CLUSTERING**

Disusun untuk Memenuhi Tugas Besar Mata Kuliah SPK



Disusun Oleh:

Kelompok 3 Data Mining-A

Trici Ayunda	2311522017
Vania Zhafira Zahra	2311523007
Suci Nurhaliza	2311521009
Muhammad Diaz Ananda S	2311521015
Ikhwan Hamidi	2311521003

Dosen Pengampu:

Aina Hubby Aziira, M.Eng.

Dwi Welly Sukma Nirad, S.Kom, M.T.

**DEPARTEMEN SISTEM INFORMASI**

**FAKULTAS TEKNOLOGI INFORMASI**

**UNIVERSITAS ANDALAS**

**2025**

## KATA PENGANTAR

Segala puji bagi Allah Swt yang telah memberikan rahmat dan karunia-Nya sehingga kami dapat menyelesaikan laporan ini dengan tepat waktu. Tanpa pertolongan-Nya, tentunya kami tidak dapat menyelesaikan laporan ini dengan baik. Shalawat beriringkan salam semoga terlimpah kepada baginda tercinta kita, yakni Nabi Muhammad Saw yang kita nanti-nantikan syafaatnya di akhirat nanti.

Kami juga mengucapkan syukur kepada Allah Swt atas limpahan nikmat sehat-Nya, baik itu berupa fisik maupun akal pikiran sehingga kami mampu untuk menyelesaikan laporan ini sebagai Tugas dari Mata Kuliah Data Mining dengan judul **“IMPLEMENTASI DATA MINING UNTUK ANALISIS TREND LAGU DI SPOTIFY PADA TAHUN 2023 BERDASARKAN KARAKTERISTIK AUDIO MENGGUNAKAN METODE K-MEANS CLUSTERING”**.

Kami mengucapkan terima kasih kepada Ibu Aina Hubby Aziira, M.Eng. dan Ibu Dwi Welly Sukma Nirad, S.Kom, M.T selaku dosen Mata Kuliah Data Mining yang telah membimbing, mengajarkan, serta membantu kami dalam menyelesaikan tugas ini. Kami tentu menyadari bahwa laporan ini masih jauh dari kata sempurna dan masih banyak terdapat kesalahan serta kekurangan di dalamnya. Untuk itu, kami mengharapkan kritik dan saran dari pembaca agar laporan ini nantinya dapat menjadi laporan yang lebih baik lagi. Kemudian apabila terdapat banyak kesalahan pada laporan ini kami mohon maaf yang sebesar-besarnya.

Demikian, semoga laporan ini dapat bermanfaat bagi pembaca maupun penulis sendiri. Terima kasih.

Padang, 2025

Penulis

## DAFTAR ISI

<b>KATA PENGANTAR.....</b>	<b>2</b>
<b>DAFTAR ISI.....</b>	<b>3</b>
<b>DAFTAR GAMBAR.....</b>	<b>4</b>
<b>DAFTAR TABEL.....</b>	<b>5</b>
<b>BAB I PENDAHULUAN.....</b>	<b>1</b>
1.1 Latar belakang.....	1
1.2 Rumusan Masalah.....	1
1.3 Batasan Masalah.....	1
1.4 Tujuan.....	1
1.5 Manfaat.....	1
<b>BAB II METODOLOGI PENELITIAN.....</b>	<b>1</b>
2.1 Spotify.....	2
2.2 Data Mining.....	2
2.2.1 Pengertian Data Mining.....	2
2.2.2 Tujuan Data Mining.....	2
2.2.3 Manfaat Data Mining.....	2
2.3 Clustering.....	2
2.3.1 K-means.....	2
2.3.2 Agglomerative Hierarchical.....	2
2.3.3 DCSCAN.....	2
2.4 Objek Penelitian.....	2
2.5 Metode Penelitian.....	2
2.6 Pengumpulan Data.....	2
2.6.1 Sumber Data.....	2
2.6.2 Atribut.....	2
2.7 Flowchart Penelitian.....	2
<b>BAB III PEMBAHASAN.....</b>	<b>2</b>
3.1 Perhitungan Manual.....	2
3.1.1 Perhitungan K-means.....	2
3.1.2 Perhitungan Agglomerative Hierarchical.....	2
3.1.3 Perhitungan DBSCAN.....	2
3.2 Implementasi.....	2
<b>BAB IV PENUTUP.....</b>	<b>3</b>
4.1 Kesimpulan.....	3
4.2 Saran.....	3
<b>DAFTAR PUSTAKA.....</b>	<b>4</b>

## **DAFTAR GAMBAR**

## **DAFTAR TABEL**

## **BAB I PENDAHULUAN**

### **1.1 Latar belakang**

Musik telah menjadi bagian penting dalam kehidupan sehari-hari, di mana masyarakat cenderung memilih lagu berdasarkan genre dan preferensi pribadi. Dengan perkembangan teknologi, layanan streaming musik seperti Spotify telah memberikan kemudahan bagi pengguna dalam mengakses berbagai macam lagu secara digital. Sejak diluncurkan pada tahun 2008, Spotify telah menjadi salah satu platform musik terbesar yang menyediakan fitur gratis maupun berbayar bagi penggunanya.

Seiring bertambahnya jumlah pengguna, pemahaman tentang tren lagu yang populer di suatu periode menjadi penting, baik bagi industri musik, kreator konten, maupun pengiklan. Salah satu cara untuk menganalisis tren lagu adalah dengan menggunakan teknik Data Mining, khususnya metode K-Means Clustering, yang mampu mengelompokkan lagu berdasarkan karakteristik audio tertentu.

Dalam beberapa penelitian sebelumnya, metode K-Means telah digunakan dalam berbagai kasus klasifikasi, seperti segmentasi pelanggan berdasarkan perilaku belanja, pengelompokan mahasiswa berdasarkan nilai akademik, serta identifikasi pola penyakit berdasarkan data medis. Namun, studi yang mengaplikasikan metode K-Means untuk mengelompokkan lagu berdasarkan karakteristik audio pada Spotify masih terbatas. Oleh karena itu, penelitian ini bertujuan untuk menganalisis tren lagu populer di Spotify pada tahun 2023 berdasarkan karakteristik audio menggunakan metode K-Means Clustering.

### **1.2 Rumusan Masalah**

Berdasarkan latar belakang di atas, rumusan masalah dalam laporan ini adalah sebagai berikut:

1. Bagaimana karakteristik audio dari lagu-lagu populer di Spotify pada tahun 2023?
2. Bagaimana metode K-Means Clustering dapat digunakan untuk mengelompokkan lagu berdasarkan karakteristik audio?
3. Apa saja pola yang dapat diidentifikasi dari hasil analisis clustering terhadap lagu-lagu populer di Spotify?

### **1.3 Batasan Masalah**

Agar penelitian lebih terarah, beberapa batasan yang diterapkan dalam penelitian ini adalah:

#### **1. Sumber Data**

Data yang digunakan dalam penelitian ini berasal dari dataset "spotify 2023" yang diperoleh dari *Kaggle*.

#### **2. Variabel yang dianalisis**

Analisis hanya dilakukan berdasarkan karakteristik audio yang tersedia dalam dataset dengan memilih beberapa fitur yang relevan.

#### **3. Lingkup Geografis**

Analisis tidak membedakan tren musik berdasarkan negara atau wilayah tertentu, melainkan melihat pola secara umum di seluruh pengguna *Spotify*.

### **1.4 Tujuan**

Adapun tujuan dari penelitian adalah sebagai berikut:

1. Menganalisis karakteristik audio dari lagu-lagu populer di Spotify pada tahun 2023.

2. Mengimplementasikan metode K-Means Clustering untuk mengelompokkan lagu berdasarkan karakteristik audio.
3. Mengidentifikasi pola atau kecenderungan dalam tren lagu populer berdasarkan hasil clustering.

## **1.5 Manfaat**

Adapun manfaat yang diharapkan dari penelitian ini adalah:

1. **Bagi Industri Musik**

Membantu industri musik dalam memahami tren lagu berdasarkan karakteristik audio untuk strategi pemasaran yang lebih efektif.

2. **Bagi Pengguna Spotify**

Memberikan wawasan tentang pola lagu populer, sehingga pengguna dapat menemukan lagu yang sesuai dengan preferensi mereka.

3. **Bagi Peneliti di Bidang Data Mining**

Menambah referensi terkait penerapan metode K-Means Clustering dalam analisis tren musik.

4. **Bagi Pengembang Aplikasi Musik**

Memberikan wawasan tentang bagaimana karakteristik audio memengaruhi popularitas lagu, yang dapat digunakan dalam pengembangan fitur rekomendasi musik yang lebih baik.

## **BAB II METODOLOGI PENELITIAN**

### **2.1 Spotify**

### **2.2 Data Mining**

#### **2.2.1 Pengertian Data Mining**

#### **2.2.2 Tujuan Data Mining**

#### **2.2.3 Manfaat Data Mining**

### **2.3 Clustering**

#### **2.3.1 K-means**

### **2.4 Objek Penelitian**

Objek Penelitian ini berfokus pada analisis trend lagu di Spotify pada tahun 2023 berdasarkan karakteristik audio menggunakan metode K-Means Clustering dengan menggunakan data “Spotify 2023”. Variabel yang akan diteliti adalah karakteristik audio dari lagu-lagu yang tersedia di platform Spotify pada tahun 2023. Data yang digunakan mencakup berbagai atribut audio yang menentukan



karakteristik sebuah lagu, seperti *danceability*, *energy*, *valence*. Variabel ini akan digunakan untuk mengelompokkan lagu-lagu berdasarkan kesamaan karakteristik audio menggunakan metode K-Means Clustering.

Pemilihan variabel ini didasarkan pada asumsi bahwa karakteristik audio suatu lagu memiliki pengaruh besar terhadap popularitas dan tren musik di Spotify. Dengan memahami pola dari lagu-lagu yang memiliki kesamaan fitur audio, dapat diidentifikasi genre atau gaya musik yang dominan dalam tren tahun 2023.

## **2.5 Metode Penelitian**

### **2.6 Pengumpulan Data**

Tahapan pengumpulan data dalam proses penelitian merupakan tahapan penting yang berperan dalam pengambilan informasi yang diperlukan untuk menjawab pertanyaan penelitian dan mencapai tujuan penelitian serta memberikan manfaat sesuai dengan kebutuhan.

#### **2.6.1 Sumber Data**

Dataset yang digunakan dalam penelitian ini adalah *spotify\_2023*, yang diperoleh dari platform Kaggle. Dataset ini berisi informasi mengenai lagu-lagu yang dirilis atau populer di Spotify selama tahun 2023, termasuk berbagai karakteristik audio seperti *tempo*, *danceability*, *energy*, *valence*, *loudness*, *acousticness*, *instrumentalness*, dan *speechiness*.

Dataset ini dipilih karena menyediakan data yang cukup lengkap untuk melakukan analisis tren musik berdasarkan karakteristik audio. Dengan menggunakan dataset ini, penelitian dapat mengelompokkan lagu-lagu ke dalam beberapa klaster berdasarkan kemiripan fitur audio menggunakan metode K-Means Clustering.

#### **2.6.2 Atribut**

### **2.7 Flowchart Penelitian**

## BAB III PEMBAHASAN

### 3.1 Perhitungan Manual

Dalam melakukan perhitungan manual berdasarkan dataset yang kami peroleh. Kami hanya mengambil 3 atribut yang relevan untuk melakukan clustering berdasarkan karakteristik audio. sebagai data sampel perhitungan kami mengambil 10 baris data teratas seperti pada tabel dibawah ini.

Tabel 3.1 Data Perhitungan Algoritma K-Means

Data	Danceability_% (A)	Valence_% (B)	Energy_% (C)
1	80	89	83
2	71	61	74
3	51	32	53
4	55	58	72
5	65	23	80
6	92	66	58
7	67	83	76
8	67	26	71
9	85	22	62
10	81	56	48

#### 3.1.1 Perhitungan K-means

Algoritma K Means merupakan salah satu metode partial clustering berbasis titik pusat (centroid) dimana proses clustering dilakukan dengan meminimalkan jarak jumlah kuadrat antara data dengan masing-masing pusat cluster. Algoritma K-Means dalam penerapannya memerlukan tiga parameter yang seluruhnya ditentukan pengguna yaitu jumlah cluster k, inisialisasi cluster, dan jarak.

Berikut langkah-langkah melakukan clustering dengan menggunakan algoritma k-means:

1. Menentukan nilai K sebanyak jumlah cluster atau kelompok yang diinginkan. Jumlah cluster yang diambil pada data ini adalah sebanyak 4 kelompok yang terdiri dari lagu upbeat-ceria, lagu mellow-emosional, lagu energik-tari, dan lagu akustik-tenang
  - a. Lagu upbeat-ceria : Lagu-lagu dalam kelompok ini memiliki tempo cepat, nuansa positif, dan memberi semangat atau kebahagiaan. Biasanya cocok diputar saat suasana santai, bahagia, atau saat ingin meningkatkan mood.
  - b. Lagu mellow-emosional : Kelompok ini mencakup lagu-lagu dengan tempo lambat atau sedang dan menyampaikan emosi yang dalam, seperti sedih, rindu, atau patah hati. Biasanya digunakan untuk suasana reflektif atau melankolis.
  - c. Lagu energik-tari : Lagu-lagu ini memiliki beat yang kuat dan ritme cepat, cocok untuk suasana pesta atau aktivitas fisik seperti menari. Biasanya termasuk dalam genre EDM, pop dansa, atau hip-hop dengan tempo tinggi.
  - d. Lagu akustik-tenang : Kelompok ini mencakup lagu-lagu dengan aransemen musik minimalis, biasanya akustik, dengan nuansa damai dan menenangkan. Cocok untuk relaksasi, belajar, atau saat ingin suasana tenang.

Pada perhitungan ini ditandai dengan angka yang akan menentukan individu tersebut akan masuk ke cluster yang mana dengan rincian sebagai berikut:

- a. Angka 4 : Lagu upbeat-ceria
- b. Angka 3 : Lagu mellow-emosional
- c. Angka 2 : Lagu energik-tari
- d. Angka 1 : Lagu akustik-tenang

2. Setelah menentukan nilai K sebanyak jumlah cluster yang diinginkan, lalu pilih sebanyak K data dari set data sebagai pusat cluster (centroid) secara random. Disini data random yang diambil yaitu pada baris ke 2,6 dan 10 seperti Tabel 3.2 dibawah

Data	Centroid	A	B	C
1	1	80	89	83
2	2	71	61	74
6	3	92	66	58
10	4	81	56	48

3. Menghitung jarak antara objek dengan masing-masing centroid menggunakan rumus euclidean distance

$$d(x_i, x_j) = \sqrt{(|x_{i1} - x_{j1}|^2 + |x_{i2} - x_{j2}|^2 + \dots + |x_{ip} - x_{jp}|^2)}$$

Data	Danceability_ % (A)	Valence_ % (B)	Energy_ % (C)	C1	C2	C3	C4
1	80	89	83	0.000	30.757	36.028	49.830
2	71	61	74	30.757	0.000	26.870	26.777
3	51	32	53	70.640	41.012	53.498	29.275
4	55	58	72	41.316	16.401	40.361	26.907
5	65	23	80	67.750	38.936	55.335	46.011
6	92	66	58	36.028	26.870	0.000	28.723
7	67	83	76	15.937	22.450	35.185	38.897
8	67	26	71	65.437	35.355	48.929	37.802
9	85	22	62	70.392	43.139	44.733	40.939
10	81	56	48	48.114	28.302	17.916	0.000

4. Mengelompokkan objek berdasarkan jarak terdekat dengan centroid. Setelah melakukan perhitungan untuk mencari jarak antar centroid, lalu dikelompokkan data tersebut berdasarkan jarak terdekat dengan centroid nya dengan menggunakan nilai minimum antar tiga centroid tersebut seperti

C1	C2	C3	C4	Min	Cluster
0.000	30.757	36.028	49.830	0.000	
30.757	0.000	26.870	26.777	0.000	
70.640	41.012	53.498	29.275	29.275	

41.316	16.401	40.361	26.907	16.401	
67.750	38.936	55.335	46.011	38.936	
36.028	26.870	0.000	28.723	0.000	
15.937	22.450	35.185	38.897	15.937	
65.437	35.355	48.929	37.802	35.355	
70.392	43.139	44.733	40.939	40.939	
48.114	28.302	17.916	0.000	0.000	

Pada Tabel 4.4 terlihat bahwa jika suatu data dekat dengan centroid pertama atau nilai minimum nya pada centroid pertama maka akan dikelompokkan menjadi cluster 1 begitupun untuk data yang dekat pada centroid 2 dan 3 maka masing-masing nya akan masuk ke dalam cluster 2 dan cluster 3. Berikut hasil dari cluster pada Tabel 3.5 dibawah

<b>DATA</b>	<b>C1</b>	<b>C2</b>	<b>C3</b>	<b>C4</b>
1	1			
2		1		
3				1
4		1		
5		1		
6			1	
7	1			
8		1		
9				1
10				1

5. Setelah mendapatkan kelompok pada iterasi pertama selanjutnya kita akan mencari pada iterasi kedua ataupun selanjutnya hingga tidak ada lagi objek yang berpindah cluster.
6. Pada iterasi kedua ini, tentukan terlebih dahulu centroid baru dengan menggunakan rumus sebagai berikut:

$$C_{m(q)} = \frac{1}{n_m} \sum_{i=1}^{n_m} x_{i(q)}$$

Cara melakukan perhitungan untuk menentukan centroid nya dengan rumus tersebut sebagai berikut:

- 1) Pada tabel 3.5 terlihat bahwa data yang masuk pada centroid 1 sebanyak 2 yaitu data ke-1 dan 7. Jadi untuk mencari centroid baru nya dengan membagi data ke 1 dan 7 yang pada tabel 4.1 dibagi dengan dua.
- 2) Pada tabel 3.5 untuk data yang masuk ke dalam centroid 2 sebanyak 4 terdapat pada data ke- 2,4,5, dan 8. Jadi untuk mencari centroid baru nya dengan membagi data ke 2,4,5, dan 8 yang pada tabel 4.1 dibagi dengan empat.
- 3) Pada tabel 3.5 terlihat bahwa data yang masuk pada centroid 3 hanya 1 pada baris 6. Jadi data baris pertama pada tabel 4.1 akan dibagi dengan satu.
- 4) Pada tabel 3.5 untuk data yang masuk ke dalam centroid 4 sebanyak 3 buah terdapat pada data ke-3,9, dan 10. Jadi untuk mencari centroid baru nya dengan membagi data ke-3,9, dan 10 yang pada tabel 4.1 dibagi dengan tiga.

Setelah mencari perhitungan tersebut maka akan dapat hasil perhitungan yang ada pada Tabel 3.6 berikut

Data	Centroid	A	B	C
1	1	73.5	86,0	79.5
2	2	64.5	42,0	74.25
6	3	92,0	66,0	58,0
10	4	72.33	36.67	54.33

7. Setelah mendapatkan centroid baru, lakukan perhitungan kembali dengan rumus euclidean distance dan didapatkan hasil seperti Tabel 3.7 dibawah:

Data	Danceability_ % (A)	Valence_ % (B)	Energy_ % (C)	C1	C2	C3	C4
1	80	89	83	7.970	50.253	36.028	59.934
2	71	61	74	26.351	20.490	26.870	28.210
3	51	32	53	47.649	27.537	53.498	22.250
4	55	58	72	28.720	16.690	40.361	25.380
5	65	23	80	63.159	19.400	55.335	37.528
6	92	66	58	36.973	34.803	0.000	30.683
7	67	83	76	8.939	41.786	35.185	41.504
8	67	26	71	60.172	19.546	48.929	35.417
9	85	22	62	66.728	30.616	44.733	28.422
10	81	56	48	44.141	34.019	17.916	9.570

8. Mengelompokkan objek berdasarkan jarak terdekat dengan centroid seperti Tabel:

C1	C2	C3	C4	Min	Cluster	Ket
7.970	50.253	36.028	59.934	0.000	1	Tetap
26.351	20.490	26.870	28.210	0.000	2	Tetap
47.649	27.537	53.498	22.250	29.275	4	Tetap
28.720	16.690	40.361	25.380	16.401	2	Tetap
63.159	19.400	55.335	37.528	38.936	2	Tetap

36.973	34.803	0.000	30.683	0.000	3	Tetap
8.939	41.786	35.185	41.504	15.937	1	Tetap
60.172	19.546	48.929	35.417	37.802	2	Tetap
66.728	30.616	44.733	28.422	40.939	4	Tetap
34.036	31.742	17.196	9.570	14.000	4	Tetap

Pada Tabel 3.8 terlihat bahwa hasil cluster yang didapatkan sama dengan hasil cluster pada iterasi pertama. Jadi hasil tersebut berarti sudah benar untuk pengelompokan nya yang terdapat pada Tabel 3.9 dibawah

DATA	C1	C2	C3	C4
1	1			
2		1		
3				1
4		1		
5		1		
6			1	
7	1			
8		1		
9				1
10				1

Kesimpulan :

Berdasarkan hasil perhitungan K-Means yang telah mencapai konvergensi pada iterasi kedua, setiap lagu dari sampel data dapat dikelompokkan ke dalam empat klaster utama yang merepresentasikan suasana dan karakteristik audio. Klaster pertama (Cluster 1) berisi lagu-lagu yang memiliki nilai danceability, valence, dan energy yang cukup tinggi, yaitu lagu pada data ke-1 dan



ke-7. Lagu-lagu ini termasuk dalam kategori lagu akustik-tenang, yang meskipun enerjik, tetap menghadirkan suasana yang positif dan nyaman untuk didengarkan.

Klaster kedua (Cluster 2) terdiri dari lagu-lagu pada data ke-2, ke-4, ke-5, dan ke-8. Lagu-lagu dalam kelompok ini memiliki kombinasi nilai energy yang tinggi dan valence yang sedang hingga rendah, sehingga termasuk ke dalam kategori lagu energik-tari. Lagu-lagu ini cocok untuk aktivitas seperti menari, berolahraga, atau suasana pesta.

Klaster ketiga (Cluster 3) hanya berisi satu lagu menunjukkan bahwa karakteristik ini unik dalam data yang diuji. yaitu pada data ke-6, yang memiliki nilai danceability dan energy tinggi dengan valence yang sedang. Lagu ini dikategorikan sebagai lagu mellow-emosional, karena meskipun cukup aktif secara ritmis, lagu ini menyimpan nuansa emosi yang lebih dalam dan tidak seceria klaster lainnya.

Sementara itu, klaster keempat (Cluster 4) mencakup lagu-lagu pada data ke-3, ke-9, dan ke-10. Lagu-lagu ini memiliki nilai valence rendah hingga sedang serta energy yang menengah ke bawah. Berdasarkan karakteristik tersebut, lagu-lagu ini dikelompokkan sebagai lagu upbeat-ceria, yaitu lagu yang memiliki elemen semangat, namun dengan sentuhan suasana yang lebih ringan atau bahkan sedikit gelap secara emosional.

Dengan pengelompokan ini, hasil clustering tidak hanya berhasil mengelompokkan lagu berdasarkan kemiripan numerik pada fitur audio, tetapi juga sesuai dengan pemaknaan suasana musik yang relevan dengan konteks pengguna.

### **3.1.2 Perhitungan Agglomerative Hierarchical**

Agglomerative Clustering adalah salah satu teknik clustering hirarki yang populer dalam analisis data. Algoritma ini bekerja dengan cara menggabungkan pasangan-pasangan data terdekat ke dalam cluster, kemudian menggabungkan cluster-cluster tersebut hingga semua data tergabung dalam satu cluster besar atau hingga kriteria tertentu terpenuhi. Berikut langkah-langkah melakukan clustering dengan menggunakan algoritma Agglomerative Hierarchical :

1. Menghitung Matriks jarak dengan rumus euclidean.

$$d_{ij} = \sqrt{\sum_{k=1}^p (X_{ik} - X_{jk})^2}$$

keterangan.

$d_{ij}$  : jarak antara objek  $i$  dengan  $j$

$x_{ij}$  : nilai objek  $i$  pada variabel ke- $k$

$x_{jk}$  : nilai objek  $j$  pada variabel ke- $k$

$p$  : banyaknya variabel yang diamati

Setiap data akan dihitung jaraknya dari data pertama ke data kedua hingga dari data kesembilan ke data kesepuluh. Setiap pasangan data akan membentuk satu cluster, sehingga pada tahap awal akan terbentuk 10 cluster. Sebagai ilustrasi, cluster pertama merupakan jarak antara data 1 dan data 2 dengan nilai , yaitu :

$$d_{ij} = \sqrt{(80 - 71)^2 + (89 - 61)^2 + (83 - 74)^2} = 30.757$$

Sehingga didapatkan matriks jaraknya seperti berikut :

Data	1	2	3	4	5	6	7	8	9	10
1	0									
2	30.757	0								
3	70.627	37.716	0							
4	41.311	16.401	26.4	0						
5	67.754	41.658	27.202	38.456	0					
6	36.014	23.537	52.153	44.384	48.547	0				
7	15.905	24.335	61.684	42.154	61.960	33.496	0			
8	65.428	37.336	21.189	35.468	21.401	44.160	57.663	0		
9	70.392	49.457	35.693	48.895	22.226	43.472	64.305	18.330	0	

Data	1	2	3	4	5	6	7	8	9	10
10	48.113	22.135	33.406	30.232	38.704	41.620	41.303	33.376	47.074	0

2. Menggabungkan dua cluster terdekat yaitu cluster 1 dengan 7 karena jaraknya yang terkecil yaitu 15.905 dibandingkan dengan cluster yang lain
3. Memperbarui matriks jarak menggunakan teknik pengelompokan single linkage

$$d_{uv} = \min\{d_{uv}\}, d_{uv} \in D$$

### Pembaruan 1

Dengan menggunakan rumus complete linkage jarak antara cluster 1,7 dengan data 1 sampai 10 akan dihitung dan matriks jarak akan diperbarui. Sebagai contoh untuk jarak cluster 1,7 dengan data 2

$$d(1,7),2 = \min\{d_{1,2}; d_{7,2}\} = \max\{30.757; 24.335\} = 30.757$$

Maka didapatlah matriks pembaruan seperti berikut :

Data	1,7	2	3	4	5	6	8	9	10
1,7	0								
2	24.335	0							
3	61.684	37.716	0						
4	41.311	16.401	26.400	0					
5	61.960	41.658	27.202	38.456	0				
6	33.496	23.537	52.153	44.384	48.547	0			
8	57.663	37.336	21.189	35.468	21.401	44.160	0		

9	64. 305	49.4 57	35.6 93	48.8 95	22.2 26	43.4 72	18.3 30	0	
10	41. 303	22.1 35	33.4 06	30.2 32	38.7 04	41.6 20	33.3 76	47.0 74	0

### Pembaruan tahap 2

Pembaruan tahap 2 dilakukan untuk cluster terdekat yaitu 6,10 sehingga didapatkanlah perhitungan matriks jarak

Data	1,7	2,4	3	5	6	8	9	10
1,7	0							
2,4		0						
3			0					
5				0				
6					0			
8						0		
9							0	
10								0

### 3.1.3 Perhitungan DBSCAN

Berikut adalah langkah-langkah dalam proses perhitungan manual menggunakan algoritma DBSCAN.

#### 1. Memilih atau menentukan epsilon dan MinPts:

- $\epsilon$  merupakan radius maksimum untuk mencari titik tetangga.
- MinPts adalah jumlah minimum tetangga atau titik untuk membentuk sebuah cluster.

Pada kasus ini kami memilih  $\epsilon = 30$  karena rentang data yang cukup besar dan MinPts = 2.

#### 2. Menghitung jarak euclidean setiap pasangan titik.

$$d_{ij} = \sqrt{[(x_i - x_j)^2 + (y_i - y_j)^2]}$$

$x_i$  = koordinat x untuk fasilitas i

$y_i$  = koordinat y untuk fasilitas i

$d_{ij}$  = jarak antar fasilitas I dan j

Contoh perhitungan jarak setiap pasangan titik:

**Rumus Jarak Euclidean:**

$$\begin{aligned} \text{Jarak} &= \sqrt{(x_j - x_i)^2 + (y_j - y_i)^2 + (z_j - z_i)^2} \\ &= \sqrt{(51 - 80)^2 + (32 - 89)^2 + (53 - 83)^2} \\ &= \sqrt{(-29)^2 + (-57)^2 + (-30)^2} \\ &= \sqrt{841 + 3249 + 900} \\ &= \sqrt{4990} \\ &\approx 70.64 \end{aligned}$$

Tabel jarak setiap pasangan titik

	1	2	3	4	5	6	7	8	9	10
1	0,00	30,7 6	70,6 4	41,3 2	67,7 5	36,0 3	15,9 4	65,4 4	70,39	48,11
2	30,7 6	0,00	41,0 1	16,4 0	38,9 4	26,8 7	22,4 5	35,3 6	43,14	28,30
3	70,6 4	41,0 1	0,00	32,4 5	31,7 2	53,5 0	58,1 9	24,8 2	36,57	38,74
4	41,3 2	16,4 0	32,4 5	0,00	37,2 7	40,3 6	28,0 2	34,1 9	47,92	35,44
5	67,7 5	38,9 4	31,7 2	37,2 7	0,00	55,3 4	60,1 7	9,70	26,93	48,67
6	36,0 3	26,8 7	53,5 0	40,3 6	55,3 4	0,00	35,1 9	48,9 3	44,73	17,92
7	15,9 4	22,4 5	58,1 9	28,0 2	60,1 7	35,1 9	0,00	57,2 2	65,12	41,34

8	65,4 4	35,3 6	24,8 2	34,1 9	9,70	48,9 3	57,2 2	0,00	20,52	40,31
9	70,3 9	43,1 4	36,5 7	47,9 2	26,9 3	44,7 3	65,1 2	20,5 2	0,00	36,99
10	48,1 1	28,3 0	38,7 4	35,4 4	48,6 7	17,9 2	41,3 4	40,3 1	36,99	0,00

### 3. Mengidentifikasi Core Point, Border Point, dan Noise ( $\epsilon = 30$ , MinPts = 2)

Berdasarkan matriks jarak Euclidean dan parameter DBSCAN yang digunakan, berikut hasil klasifikasi tiap titik:

a. Titik 2:

Tetangga dengan jarak  $\leq 30$ : Tidak ada

Jumlah tetangga: 0

Klasifikasi: Noise (bukan core dan tidak berada dalam radius core point)

b. Titik 2

Tetangga dengan jarak  $\leq 30$ : Titik 4 (16,40), Titik 7 (22,45), Titik 10 (28,30)

Jumlah tetangga: 3

Klasifikasi: Core Point

c. Titik 3

Tetangga dengan jarak  $\leq 30$ : Titik 8 (24,82)

Jumlah tetangga: 1

Klasifikasi: Core Point (dengan dirinya sendiri, total MinPts = 2)

d. Titik 4

Tetangga dengan jarak  $\leq 30$ : Titik 2 (16,40), Titik 7 (28,02)

Jumlah tetangga: 2

Klasifikasi: Core Point

e. Titik 5

Tetangga dengan jarak  $\leq 30$ : Titik 8 (9,70), Titik 9 (26,93)

Jumlah tetangga: 2

Klasifikasi: Core Point

f. Titik 6

Tetangga dengan jarak  $\leq 30$ : Titik 10 (17,92)  
 Jumlah tetangga: 1  
 Klasifikasi: Core Point

g. Titik 7

Tetangga dengan jarak  $\leq 30$ : Titik 1 (15,94), Titik 2 (22,45)  
 Jumlah tetangga: 2  
 Klasifikasi: Core Point

h. Titik 8

Tetangga dengan jarak  $\leq 30$ : Titik 3 (24,82), Titik 5 (9,70),  
 Titik 9 (20,52)  
 Jumlah tetangga: 3  
 Klasifikasi: Core Point

i. Titik 9

Tetangga dengan jarak  $\leq 30$ : Titik 5 (26,93), Titik 8 (20,52)  
 Jumlah tetangga: 2  
 Klasifikasi: Core Point

j. Titik 10

Tetangga dengan jarak  $\leq 30$ : Titik 2 (28,30), Titik 6 (17,92)  
 Jumlah tetangga: 2  
 Klasifikasi: Core Point

Titik	Tetangga $\leq 30$	Core	Border	Noise?
1	0	TIDAK	TIDAK	YA
2	3	YA	TIDAK	TIDAK
3	1	YA	TIDAK	TIDAK
4	2	YA	TIDAK	TIDAK
5	2	YA	TIDAK	TIDAK

6	1	YA	TIDAK	TIDAK
7	2	YA	TIDAK	TIDAK
8	3	YA	TIDAK	TIDAK
9	2	YA	TIDAK	TIDAK
10	2	YA	TIDAK	TIDAK

#### 4. Hasil Clustering DBSCAN

Berdasarkan perhitungan tersebut, dapat diperoleh pembentukan cluster sebagai berikut:

- Core Points: Titik 2, 3, 4, 5, 6, 7, 8, 9, 10
- Border Points: Tidak ada
- Noise: Titik 1

### 3.2 Implementasi

#### 3.2.1 Implementasi K-Means

#### 3.2.2 Implementasi Agglomerative

#### 3.2.3 Implementasi DBSCAN



## **BAB IV PENUTUP**

### **4.1 Kesimpulan**

### **4.2 Saran**

## **DAFTAR PUSTAKA**

progres report 2