

# Perbandingan Algoritma K-Means dan Algoritma K-Medoids Pada Kasus Covid-19 di Indonesia

Novianti Puspitasari\*, Gidion Lempas, Hamdani, Haviluddin, Anindita Septiarini

Fakultas Teknik, Program Studi Informatika, Universitas Mulawarman, Samarinda, Indonesia

Email: <sup>1,\*</sup>novia.ftik.unmul@gmail.com, <sup>2</sup>gidionlempas@gmail.com, <sup>3</sup>hamdani@unmul.ac.id, <sup>4</sup>haviluddin@unmul.ac.id  
<sup>5</sup>anindita@unmul.ac.id

Email Penulis Korespondensi: novia.ftik.unmul@gmail.com

Submitted: 21/01/2023; Accepted: 31/03/2023; Published: 31/03/2023

**Abstrak**—Analisis pengelompokan data Covid-19 telah banyak dilakukan, namun penelitian tentang pengelompokan setiap kasus dari data Covid-19 di seluruh provinsi di Indonesia belum tersedia. Penelitian ini menggunakan dua algoritma clustering yaitu K-Means dan K-Medoids untuk mengelompokkan kasus positif, sembuh dan meninggal pada data Covid-19 ke dalam tiga cluster yaitu rendah, sedang dan tinggi. Data penelitian merupakan data kasus Covid-19 di seluruh provinsi di Indonesia dari mulai tahun 2020 hingga tahun 2021. Di dalam perhitungan clustering, ketiga metode jarak yang digunakan dalam penelitian ini terdiri dari Chebyshev Distance, Manhattan Distance, dan Euclidean Distance. Berdasarkan hasil pengujian Silhouette Coefficient terhadap ketiga metode perhitungan jarak, ditemukan bahwa Manhattan Distance merupakan metode perhitungan jarak terbaik untuk K-Means dan K-Medoids. Lebih lanjut, hasil pengujian metode Sum Squared Error (SSE), Silhouette Coefficient (SC) dan Davies Index Bouldin (DBI) terhadap cluster yang dihasilkan, menunjukkan bahwa nilai yang dihasilkan oleh algoritma K-Means lebih tinggi pada metode SC dan DBI. Hal ini dibuktikan dengan nilai SC sebesar 0.838; 0.838; dan 0.925 pada kasus positif, sembuh dan meninggal. Sementara nilai DBI sebesar 0.305 untuk kasus positif, 0.295 kasus sembuh dan 1.569 pada kasus meninggal. Berdasarkan nilai tersebut, membuktikan bahwa K-Means lebih unggul dalam mengelompokkan dan menempatkan cluster dibandingkan dengan K-Medoids.

**Kata Kunci:** Covid-19; Clustering; K-Means; K-Medoids; Silhouette Coefficient; Sum Squared Error; Davies Boudlin Index

**Abstract**—Analyzing Covid-19 data has been conducted in many types of research, but research on classifying each case from Covid-19 data in all provinces in Indonesia has yet to be available. This study uses two clustering algorithms, namely K-Means and K-Medoids, to classify positive cases recovered and died in the Covid-19 data into three clusters, namely low, medium and high. The research data is Covid-19 case data in all provinces in Indonesia from 2020 to 2021. In the clustering calculations, the three distance methods used in this study are the Chebyshev Distance, Manhattan Distance, and Euclidean Distance. Based on the Silhouette Coefficient test results for the three distance calculation methods, it was found that Manhattan Distance is the best distance calculation method for K-Means and K-Medoids. Furthermore, the results of testing the Sum Squared Error (SSE), Silhouette Coefficient (SC) and Davies Index Bouldin (DBI) methods for the resulting clusters show that the value generated by the K-Means algorithm is higher in the SC and DBI methods. This result is evidenced by the SC value of 0.838; 0.838; and 0.925 in positive cases, recovered and died. While the DBI value is 0.305 for positive cases, 0.295 for recovered cases and 1.569 for dead cases. Based on these values, it proves that K-Means is superior in grouping and placing clusters compared to K-Medoids.

**Keywords:** Covid-19; Clustering; K-Means; K-Medoids; Silhouette Coefficient; Sum Squared Error; Davies Boudlin Index

## 1. PENDAHULUAN

*Clustering* merupakan salah satu metode *data mining* yang melakukan proses identifikasi terhadap sekumpulan obyek ke dalam kelas-kelas obyek yang sama (mirip) dan merupakan usaha untuk menemukan pola penyebaran dan pola hubungan dari sekumpulan data yang besar [1]. Penelitian ini menggunakan dua algoritma jenis *partitional clustering* yaitu *K-Means* dan *K-Medoids* untuk mengidentifikasi data penyebaran Covid-19 di Indonesia. Covid-19 merupakan penyakit yang muncul pada akhir tahun 2019, dengan penyebaran yang cukup cepat dan merata di Indonesia. Penyakit ini berdampak negatif bagi seluruh masyarakat Indonesia dan dunia [2]. Kementerian Kesehatan Republik Indonesia sebelumnya sudah merincikan jumlah data Covid-19 di Indonesia menjadi tiga kriteria berdasarkan kasus positif, kasus sembuh, dan kasus meninggal. Namun, informasi tentang pengelompokan setiap Provinsi di Indonesia untuk setiap kasus masuk ke dalam *cluster* (kelompok) kasus rendah, kasus sedang, atau kasus tinggi belum tersedia. Berdasarkan hal tersebut, penelitian ini menggunakan algoritma *K-Means* dan *K-Medoid* untuk mengetahui titik-titik pusat kasus Covid-19 dan mengelompokkan wilayah ke dalam tiga *cluster* yaitu rendah, sedang, dan tinggi dari ketiga kasus positif, sembuh dan meninggal.

Penelitian tentang penerapan algoritma *K-Means* dan *K-Medoids* untuk melakukan *clustering* penyakit Covid-19 telah beberapa kali dilakukan. Penelitian *K-Means* untuk pengelompokan kasus *Covid-19* pada tingkat provinsi di Indonesia menggunakan metode pengujian *Shilhouetten Index* yang menampilkan akurasi sebesar 85,7% [1]. Selanjutnya, penelitian lain yang menggunakan *K-Means* untuk melihat penyebaran kasus Covid-19 di suatu provinsi [3]–[7]. Algoritma *K-Medoids* sendiri juga telah diterapkan dalam pengelompokan penyebaran *Covid-19* di Kota Padang [8] dan di wilayah lainnya [9], [10]. Kedua algoritma tersebut dalam penelitian lainnya digunakan untuk mengelompokkan data *Covid-19* di Kota Dumai. Penelitian ini membandingkan algoritma *K-Means* dan *K-Medoids* dengan menggunakan metode pengujian *Davies Bouldin Index* (DBI). Hasil penelitiannya menunjukkan bahwa untuk mengelompokkan data Covid-19 di kota Dumai, *K-Means* lebih unggul dengan indeks DBI 0.139 [11]. Penelitian lainnya yang membandingkan algoritma *K-Means* dan *K-Medoids* juga menghasilkan kesimpulan yang sama bahwa *K-Means* lebih optimal dalam menghasilkan *cluster* atau membuat pengelompokan daripada *K-Medoids* [12]–[16].

Namun, pada penelitian pengelompokan wilayah sebaran cacat pada anak dengan kedua algoritma tersebut, *K-Medoids* lebih unggul daripada *K-Means* berdasarkan indeks *Silhouette Coefficient* [17]. Hasil penelitian [18], [19] juga menyatakan *K-Medoids* lebih unggul daripada *K-Means*. Berdasarkan penelitian terdahulu, metode *K-Means* dan *K-Medoids* untuk mengidentifikasi kasus *Covid-19* di seluruh provinsi Indonesia berdasarkan kasus sembuh, positif dan meninggal belum tersedia. Beberapa penelitian hanya mengidentifikasi kasus *Covid-19* disatu wilayah. Oleh karena itu penelitian ingin melihat keakuratan dari kedua algoritma yaitu *K-Means* dengan *K-Medoids* dalam mengidentifikasi kasus *Covid-19* di seluruh provinsi Indonesia. Metode pengujian akurasi dalam penelitian ini menggunakan tiga metode pengujian yaitu *Sum Square Error*, *Silhouette Coefficient* dan *Davies Bouldin Index* sehingga hasil perbandingan metode *clustering* dapat terukur dan menghasilkan *clustering* yang optimal. Kontribusi yang diharapkan dari penelitian ini adalah mengetahui sebaran data *Covid-19* yang masuk ke dalam *cluster* rendah, sedang, dan tinggi, dari setiap kasus positif, sembuh dan meninggal.

## 2. METODOLOGI PENELITIAN

Metode penelitian merupakan langkah-langkah yang dilakukan untuk mencapai tujuan penelitian. Langkah-langkah dalam penelitian ini antara lain (1) Pengumpulan Data, (2) Transformasi Data, (3) *Clustering* algoritma *K-Means* dan *K-Medoids*, (4) Pengujian Hasil *Clustering*.

### 2.1 Pengumpulan Data

Penelitian ini menggunakan data kasus *Covid-19* di Indonesia yang didapatkan melalui website Kementerian Kesehatan Republik Indonesia <https://infeksiemerging.kemkes.go.id/>. Data *Covid-19* yang digunakan diambil dari bulan Maret 2020 sampai November 2021. Data kasus *Covid-19* dapat dilihat dalam Tabel 1.

**Tabel 1.** Data *Covid-19* di Indonesia

No.	Provinsi	Jumlah Kasus		
		Kasus Positif	Kasus Sembuh	Kasus Meninggal
1	Aceh	38405	36316	2066
2	Sumatera Utara	105987	102989	2889
3	Sumatera Barat	89831	87567	2152
4	Riau	128575	124102	4116
5	Jambi	29764	28956	778
6	Sumatera Selatan	59918	56815	3070
7	Bengkulu	23102	22604	473
8	Lampung	49659	45233	3842
9	Bangka Belitung	52210	50656	1453
10	Kepulauan Riau	53875	52107	1747
11	DKI Jakarta	863482	849417	13594
12	Jawa Barat	707345	691546	14729
13	Jawa tengah	486068	454538	30198
14	DI Yogyakarta	156477	150799	5265
15	Jawa Timur	399147	369268	29683
16	Banten	132624	129731	2685
17	Bali	114108	110019	4040
18	Nusa Tenggara Barat	27768	26922	814
19	Nusa Tenggara Timur	63937	62400	1338
20	Kalimantan Barat	41448	40305	1059
21	Kalimantan Tengah	45601	44154	1404
22	Kalimantan Selatan	69889	67477	2390
23	Kalimantan Timur	158183	152661	5453
24	Kalimantan Utara	35868	34995	810
25	Sulawesi Utara	34684	33527	1051
26	Sulawesi Tengah	47120	45469	1599
27	Sulawesi Selatan	109877	107520	2236
28	Sulawesi Tenggara	20152	19579	527
29	Gorontalo	11833	11369	460
30	Sulawesi Barat	12350	11993	346
31	Maluku	14585	14300	261
32	Maluku Utara	12097	11750	303
33	Papua	34307	32052	556
34	Papua Barat	23322	22753	357

(Sumber: <https://infeksiemerging.kemkes.go.id/>)

Dari Tabel 1 terlihat bahwa data yang diperoleh berjumlah 34 *record* data dari ke-34 provinsi di Indonesia. Data penelitian ini terdiri dari tiga variabel yaitu kasus positif, kasus sembuh, dan kasus meninggal. Kasus positif merupakan jumlah kasus orang yang terdeteksi positif penyakit Covid-19. Kasus sembuh merupakan jumlah kasus dari masyarakat yang telah sembuh dan kasus meninggal adalah jumlah kasus dari masyarakat yang meninggal akibat penyakit Covid-19.

## 2.2 Transformasi Data

Transformasi data merupakan proses perubahan data dan penggabungan data ke dalam format tertentu [20]. Transformasi data dilakukan sebelum masuk ke dalam proses *clustering* (pengelompokan). Transformasi yang dilakukan adalah normalisasi data untuk menyamakan rentang nilai tiap data. Normalisasi yang digunakan dalam penelitian ini adalah normalisasi *Min-Max*. Normalisasi *Min-Max* merupakan metode normalisasi yang melakukan transformasi linier dengan menggunakan nilai minimum dan maksimum. Metode *Min-Max* menghasilkan keseimbangan antara satu dengan yang lain pada rentang yang sama, menggunakan persamaan 1 [21].

$$\text{normalized}(x) = \frac{\text{minRange} + (x - \text{minValue})(\text{maxRange} - \text{minRange})}{\text{maxValue} - \text{minValue}} \quad (1)$$

## 2.3 Algoritma K-Means

Algoritma *K-Means* adalah metode *clustering* berbasis jarak yang membagi data ke dalam sejumlah *cluster*, dan algoritma ini hanya bekerja pada atribut numerik. Algoritma *K-Means* termasuk *partitioning clustering* yang memisahkan data ke  $k$  daerah bagian yang terpisah [22], [23]. Pada awal iterasi pusat setiap *cluster* ditetapkan secara bebas, kemudian untuk iterasi selanjutnya dihitung jarak antara setiap data dengan setiap pusat *cluster* [21], [24]. Penelitian ini menggunakan tiga metode jarak yaitu *Chebyshev Distance*, *Manhattan Distance* dan *Euclidean Distance*. Berikut langkah-langkah dalam algoritma *K-Means*:

- Tentukan  $k$  sebagai jumlah *cluster* yang dibentuk.
- Menentukan titik *centroid* awal secara acak.
- Menghitung jarak data ke titik *centroid*, dalam penelitian ini menggunakan metode *Chebyshev Distance* dengan persamaan 2, *Manhattan Distance* dengan persamaan 3, dan *Euclidean Distance* dengan persamaan 4.

$$\text{dist}(x, y) = \max_k |x_i - y_i| \quad (2)$$

$$d(x, y) = \sum_{i=1}^n |x_i - y_i| \quad (3)$$

$$d(x, y) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2} \quad (4)$$

- Alokasikan masing-masing objek data ke dalam *centroid* yang paling dekat.
- Menentukan titik *centroid* yang baru dengan menghitung nilai rata-rata dari data yang berada dalam satu *cluster* dengan menggunakan persamaan 5.

$$C(i) = \frac{x_1 + x_2 + x_3 + \dots + x_n}{\sum x} \quad (5)$$

- Ulangi langkah 3 jika posisi *centroid* baru tidak sama.

## 2.4 Algoritma K-Medoids

Algoritma *K-Medoids* merupakan algoritma yang mirip dengan algoritma *K-Means* karena kedua algoritma ini merupakan algoritma *partitioning* yang memecah data menjadi kelompok-kelompok. *K-Medoids* adalah salah satu metode *clustering* yang meminimalkan jarak antara titik berada dalam *cluster* dan titik yang ditunjuk sebagai pusat *cluster* itu [25]. Langkah-langkah algoritma *K-Medoids*:

- Menentukan jumlah *cluster* yang akan dibentuk.
- Menentukan *medoid* awal secara acak.
- Menghitung jarak data ke *medoid* menggunakan persamaan 2, persamaan 3, dan persamaan 4.
- Alokasikan setiap data ke *cluster* terdekat.
- Hitung total simpangan ( $S$ ), dengan menghitung total jarak terdekat baru – total jarak terdekat lama, jika  $S < 0$  maka kembali ke langkah 2.
- Ulangi langkah 3 sampai 5 hingga nilai  $S > 0$ , hingga diperoleh *cluster* beserta anggotanya.

## 2.5 Pengujian Hasil Clustering

Metode pengujian hasil *cluster* dievaluasi dengan dua teknik (pengukuran) yaitu internal dan eksternal [26]. Pengujian hasil *clustering* dalam penelitian ini digunakan untuk melihat algoritma yang lebih akurat dalam melakukan *clustering* pada data Covid-19, dengan menggunakan metode pengujian *Sum Squared Error* (SSE), metode *Silhouette Coefficient* (SC), dan metode *Davies Index Bouldin* (DBI).

### 2.5.1 Sum Squared Error (SSE)

Metode *Sum Squared Error* (SSE) digunakan untuk mengukur selisih total dari nilai sebenarnya terhadap nilai yang tercapai. SSE merupakan hasil penjumlahan dari seluruh jarak masing-masing data dengan titik pusat *cluster*-nya [27]. Nilai SSE yang semakin kecil atau mendekati 0 menandakan hasil perhitungan mempunyai nilai kesalahan acak terkecil [21]. Persamaan metode SSE dapat dilihat pada persamaan 6.

$$SSE = \sum_{i=1}^n (X_i - Y_i)^2 \tag{6}$$

### 2.6.2 Silhouette Coefficient (SC)

Metode *Silhouette Coefficient* digunakan untuk melihat kualitas dan kekuatan *cluster*, seberapa baik suatu objek ditempatkan dalam suatu *cluster* [28]. Hasil *clustering* dikatakan memiliki *cluster* yang kuat dan berada pada *cluster* yang tepat, jika nilai *silhouette* lebih dari 0 atau positif. Namun, jika nilai *silhouette* kurang dari 0 atau minus maka struktur *cluster* lemah dan tidak berada pada *cluster* yang tepat [29]. Tahapan dalam pengujian menggunakan metode *silhouette coefficient* adalah:

- a. Pertama hitung rata-rata jarak dari suatu dokumen dengan semua dokumen lain yang berada dalam satu *cluster* yang sama, menggunakan persamaan 7.

$$a(i) = \frac{1}{|A-1|} \sum_{j \in A, j \neq i} C d(i, j) \tag{7}$$

- b. Kemudian hitung rata-rata jarak dari suatu dokumen dengan semua dokumen di *cluster* lain dengan menggunakan persamaan 8, dan kemudian diambil nilai terkecilnya menggunakan persamaan 9.

$$d(i, C) = \frac{1}{|A|} \sum_{j \in C} d(i, j) \tag{8}$$

$$b(i) = \min_{C \neq A} d(i, C) \tag{9}$$

- c. Langkah terakhir, hitung nilai *silhouette* menggunakan persamaan 10.

$$s(i) = \frac{b(i) - a(i)}{\max(a(i), b(i))} \tag{10}$$

### 2.6.3 Davies Bouldin Index (DBI)

*Davies Bouldin Index* (DBI) merupakan salah satu metode yang berbentuk matrik untuk mengevaluasi hasil algoritma *clustering* [29]. DBI digunakan untuk mengevaluasi *cluster* secara umum berdasarkan kuantitas dan kedekatan antar anggota *cluster*. Semakin kecil nilai DBI yang dihasilkan, maka semakin baik *cluster* yang dihasilkan [18]. Terdapat beberapa tahapan dalam metode DBI diantaranya:

- a. Mencari nilai SSW menggunakan persamaan 11.

$$SSW = \frac{1}{m_i} \sum_{j=i}^{m_i} d(x_j, x_i) \tag{11}$$

- b. Mencari nilai SSB menggunakan persamaan 12.

$$SSB = d(c_i, c_j) \tag{12}$$

- c. Mencari nilai rasio menggunakan persamaan 13.

$$R(i, j) = \frac{SSW_i + SSW_j}{SSB_{i,j}} \tag{13}$$

- d. Langkah terakhir yaitu menghitung nilai DBI menggunakan persamaan 14.

$$DBI = \frac{1}{k} \sum_{i=1}^k \max_{i \neq j} (R_{i,j}) \tag{14}$$

## 3. HASIL DAN PEMBAHASAN

Pada bagian ini berisi hasil dan pembahasan dari penerapan algoritma *K-Means* dan *K-Medoids* serta perbandingan akurasi antara kedua algoritma tersebut menggunakan berbagai metode pengujian akurasi.

### 3.1 Normalisasi Data

Tahap pertama sebelum melakukan *clustering* yaitu menormalisasikan data kasus Covid-19 dengan normalisasi *Min-Max* menggunakan persamaan 1. Hasil perhitungan normalisasi *Min-Max* ditampilkan oleh Tabel 2.

**Tabel 2.** Hasil Normalisasi *Min-Max*

Provinsi Ke-	Kasus Positif	Kasus Sembuh	Kasus Meninggal
1	0.031	0.030	0.060
2	0.111	0.109	0.088
3	0.092	0.091	0.063
4	0.137	0.135	0.129
5	0.021	0.021	0.017
6	0.056	0.054	0.094



7	0.013	0.013	0.007
8	0.044	0.040	0.120
9	0.047	0.047	0.040
10	0.049	0.049	0.050
11	1.000	1.000	0.445
12	0.817	0.812	0.483
13	0.557	0.529	1.000
14	0.170	0.166	0.167
15	0.455	0.427	0.983
16	0.142	0.141	0.081
17	0.120	0.118	0.126
18	0.019	0.019	0.018
19	0.061	0.061	0.036
20	0.035	0.035	0.027
21	0.040	0.039	0.038
22	0.068	0.067	0.071
23	0.172	0.169	0.173
24	0.028	0.028	0.018
25	0.027	0.026	0.026
26	0.041	0.041	0.045
27	0.115	0.115	0.066
28	0.010	0.010	0.009
29	0.000	0.000	0.007
30	0.001	0.001	0.003
31	0.003	0.003	0.000
32	0.000	0.000	0.001
33	0.026	0.025	0.010
34	0.013	0.014	0.003

Tabel 2 menampilkan hasil normalisasi data kasus Covid-19 yang telah memiliki rentang yang sama.

### 3.2 Algoritma K-Means

Perhitungan *clustering* menggunakan algoritma *K-Means* dilakukan setelah proses normalisasi. Data Covid-19 yang telah dinormalisasi akan dimasukkan ke dalam tiga *cluster* yaitu rendah, sedang dan tinggi. Proses peng*clustering* dilakukan terhadap data kasus positif, kasus sembuh, dan kasus meninggal. Tahap pertama dari proses *clustering* adalah menentukan pusat *cluster* atau *centroid*. Dalam penelitian ini, penentuan pusat *cluster* sebanyak  $K = 3$  dilakukan dengan mengambil data ke-9 sebagai pusat *cluster* C1, data ke-11 sebagai C2 dan data ke-13 sebagai pusat *cluster* C3. Tabel 3. menampilkan titik *centroid* awal.

Tabel 3. Titik *centroid* awal *K-Means*

Cluster	Data ke-	Kasus Positif	Kasus Sembuh	Kasus Meninggal
C1	9	0.047	0.047	0.040
C2	11	1.000	1.000	0.445
C3	13	0.557	0.529	1.000

Dari penentuan titik *centroid*, proses berikutnya adalah menghitung jarak menggunakan *Chebyshev Distance* (CD), *Manhattan Distance* (MD) dan *Euclidean Distance* (ED). Penggunaan ketiga metode jarak ini berfungsi untuk mengetahui metode yang terbaik dan hasilnya menjadi model untuk dasar pengelompokan data kasus Covid-19. Pada proses perhitungan metode jarak, *sample* data yang digunakan adalah kasus positif. Perhitungan metode jarak pada kasus positif menggunakan persamaan 2 untuk metode CD, persamaan 3 untuk MD dan persamaan 4 untuk ED. Hasil perhitungan metode jarak untuk CD, MD dan ED berhenti di iterasi ke 2 yang ditampilkan oleh Tabel 4.

Tabel 4. Perhitungan ketiga metode Jarak Kasus Positif *K-Means*

Provinsi Ke-	Jarak ke <i>Centroid</i>								
	CD			MD			ED		
	C <sub>1</sub>	C <sub>2</sub>	C <sub>3</sub>	C <sub>1</sub>	C <sub>2</sub>	C <sub>3</sub>	C <sub>1</sub>	C <sub>2</sub>	C <sub>3</sub>
1	0.025	0.877	0.475	0.025	0.877	0.475	0.025	0.877	0.475
2	0.054	0.798	0.395	0.054	0.798	0.395	0.054	0.798	0.395
3	0.035	0.817	0.414	0.035	0.817	0.414	0.035	0.817	0.414
4	0.081	0.771	0.369	0.081	0.771	0.369	0.081	0.771	0.369
5	0.035	0.887	0.485	0.035	0.887	0.485	0.035	0.887	0.485
6	0.000	0.852	0.449	0.000	0.852	0.449	0.000	0.852	0.449



7	0.043	0.895	0.493	0.043	0.895	0.493	0.043	0.895	0.493
8	0.012	0.864	0.461	0.012	0.864	0.461	0.012	0.864	0.461
:	:	:	:	:	:	:	:	:	:
27	0.059	0.793	0.391	0.059	0.793	0.391	0.059	0.793	0.391
28	0.047	0.899	0.496	0.047	0.899	0.496	0.047	0.899	0.496
29	0.056	0.908	0.506	0.056	0.908	0.506	0.056	0.908	0.506
30	0.056	0.908	0.505	0.056	0.908	0.505	0.056	0.908	0.505
31	0.053	0.905	0.503	0.053	0.905	0.503	0.053	0.905	0.503
32	0.056	0.908	0.506	0.056	0.908	0.506	0.056	0.908	0.506
33	0.030	0.882	0.479	0.030	0.882	0.479	0.030	0.882	0.479
34	0.043	0.895	0.492	0.043	0.895	0.492	0.043	0.895	0.492

Langkah selanjutnya adalah membandingkan ketiga metode jarak menggunakan *Silhouette Coefficient (SC)* untuk menemukan metode jarak terbaik yang digunakan dalam proses perhitungan selanjutnya. Perhitungan menggunakan persamaan 7 sampai 10. Tabel 5 menampilkan nilai akurasi dari ketiga metode jarak.

**Tabel 5.** Nilai Akurasi Metode Jarak Kasus Positif *K-Means*

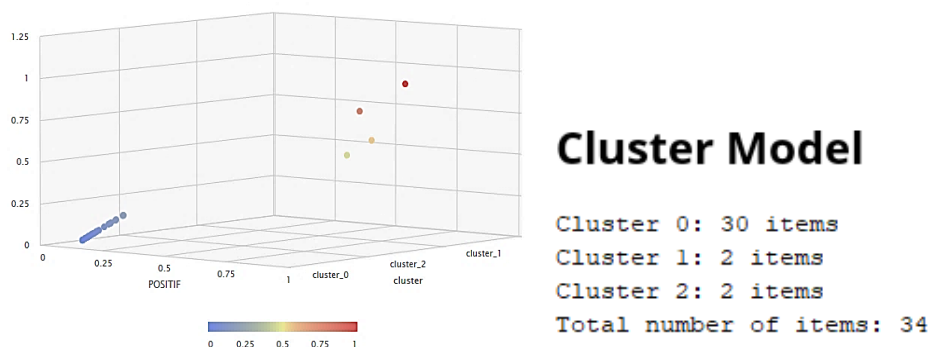
Akurasi	<i>Chebyshev</i>	<i>Manhattan</i>	<i>Euclidian</i>
SC	0.767	0.838	0.825

Berdasarkan Tabel 5, terlihat bahwa metode jarak *Manhattan* merupakan metode jarak terbaik di *K-Means*, karena memiliki nilai akurasi paling tinggi. Hasil ini menjadi dasar untuk mengelompokkan data Covid-19 yang menghasilkan pusat *cluster* pada iterasi terakhir dan pemetaan *cluster* yang ditampilkan oleh Tabel 6.

**Tabel 6.** Hasil *Clustering K-Means* Kasus Positif

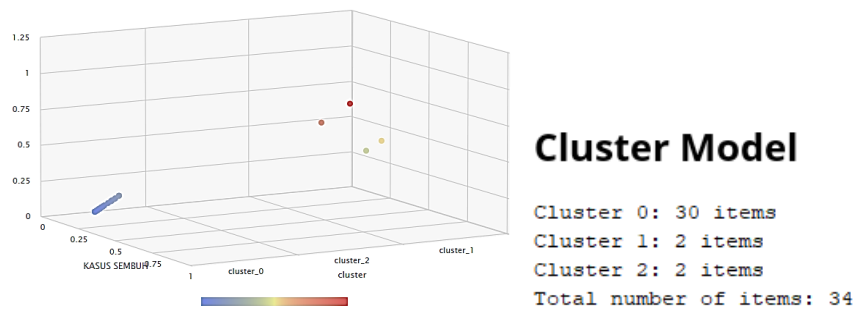
Provinsi	Jarak Ke <i>Centroid</i>			Hasil <i>Cluster</i>		
	Ke- C <sub>1</sub>	C <sub>2</sub>	C <sub>3</sub>	C <sub>1</sub>	C <sub>2</sub>	C <sub>3</sub>
1	0.025	0.877	0.475	*		
2	0.054	0.798	0.395	*		
3	0.035	0.817	0.414	*		
4	0.081	0.771	0.369	*		
5	0.035	0.887	0.485	*		
6	0.000	0.852	0.449	*		
7	0.043	0.895	0.493	*		
8	0.012	0.864	0.461	*		
:	:	:	:	:	:	:
27	0.059	0.793	0.391	*		
28	0.047	0.899	0.496	*		
29	0.056	0.908	0.506	*		
30	0.056	0.908	0.505	*		
31	0.053	0.905	0.503	*		
32	0.056	0.908	0.506	*		
33	0.030	0.882	0.479	*		
34	0.043	0.895	0.492	*		

Hasil *clustering* berdasarkan Tabel 6 menunjukkan bahwa terdapat dua provinsi untuk kasus positif yang masuk ke dalam *cluster* C2 dan C3, sementara ketiga puluh provinsi lainnya masuk ke dalam *cluster* C1. Penentuan *cluster* mana yang termasuk ke dalam *cluster* rendah, sedang dan tinggi dapat dilihat dari hasil pengelompokkan menggunakan grafik dengan alat bantu *Rapidminer* yang ditunjukkan oleh Gambar 1.



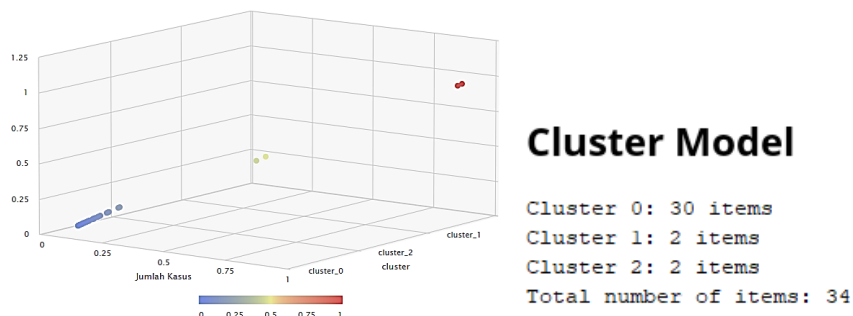
**Gambar 1.** Grafik Plot *K-Means* Hasil *Clustering* Kasus Positif

Gambar 1 menampilkan *cluster* C0 berwarna biru yang nilainya lebih rendah daripada dua *cluster* lainnya dengan jumlah data sebanyak 30 data. *Cluster* C1 berwarna merah dengan nilai diantara *cluster* C2 dan *cluster* C0 dengan item sebanyak 2 data. *Cluster* C2 berwarna jingga yang memiliki nilai indeks diantara C1 dan C2 dengan jumlah data yang sama yaitu 2 data. Dari hasil grafik tersebut maka ditemukan korelasi antara *cluster* C0 pada *Rapidminer* merupakan *cluster* C1 pada proses perhitungan yang termasuk ke dalam *cluster* rendah. Lebih lanjut, *cluster* C1 pada *Rapidminer* adalah *cluster* C2 pada proses perhitungan dengan *cluster* tinggi dan *cluster* C2 pada *Rapidminer* adalah C3 pada proses perhitungan yang merupakan *cluster* sedang. Proses penentuan *cluster* untuk kasus sembuh dan meninggal menggunakan tahapan yang sama seperti yang telah dilakukan pada kasus positif. Pada kasus sembuh hasil *clustering* menampilkan hasil yang berbeda dengan kasus positif seperti pada Gambar 2.



Gambar 2. Grafik Plot K-Means Hasil Clustering Kasus Sembuh

Gambar 2 menampilkan korelasi antara *cluster* C0 pada *Rapidminer* merupakan *cluster* C1 pada proses perhitungan yang termasuk ke dalam *cluster* rendah. Lebih lanjut, *cluster* C1 pada *Rapidminer* adalah *cluster* C2 yang termasuk *cluster* sedang dan *cluster* C2 pada *Rapidminer* adalah C3 pada proses perhitungan yang merupakan *cluster* tinggi. Lebih lanjut, Pada kasus meninggal hasil *clustering* menampilkan hasil yang sama dengan kasus positif seperti pada Gambar 3.



Gambar 3. Grafik Plot K-Means Hasil Clustering Kasus Meninggal

Berdasarkan Gambar 3, terlihat bahwa *cluster* C0 pada *Rapidminer* merupakan *cluster* C1 pada proses perhitungan yang termasuk ke dalam *cluster* rendah dengan jumlah data sebanyak 30 data. Lebih lanjut, *cluster* C1 pada *Rapidminer* adalah *cluster* C2 yang termasuk *cluster* tinggi dengan 2 data dan *cluster* C2 pada *Rapidminer* adalah C3 pada proses perhitungan dengan jumlah data sebanyak 2 data yang termasuk *cluster* sedang. Hasil *clustering* dari penerapan Algoritma K-Means untuk keseluruhan kasus dapat dilihat pada Tabel 7.

Tabel 7. Hasil Clustering Algoritma K-Means

No	Provinsi	Cluster		
		Kasus Positif	Kasus Sembuh	Kasus Meninggal
1	Aceh	C1	C1	C1
2	Sumatera Utara	C1	C1	C1
3	Sumatera Barat	C1	C1	C1
4	Riau	C1	C1	C1
5	Jambi	C1	C1	C1
6	Sumatera Selatan	C1	C1	C1
7	Bengkulu	C1	C1	C1
8	Lampung	C1	C1	C1
9	Bangka Belitung	C1	C1	C1
10	Kepulauan Riau	C1	C1	C1
11	DI Yogyakarta	C2	C2	C2
12	Banten	C2	C2	C2
13	Bali	C3	C3	C3

14	Nusa Tenggara Barat	C1	C1	C1
15	Nusa Tenggara Timur	C3	C3	C3
16	Kalimantan Barat	C1	C1	C1
17	Kalimantan Tengah	C1	C1	C1
18	Kalimantan Selatan	C1	C1	C1
19	Kalimantan Timur	C1	C1	C1
20	Kalimantan Utara	C1	C1	C1
21	Sulawesi Utara	C1	C1	C1
22	Sulawesi Tengah	C1	C1	C1
23	Sulawesi Selatan	C1	C1	C1
24	Sulawesi Tenggara	C1	C1	C1
25	Gorontalo	C1	C1	C1
26	Sulawesi Barat	C1	C1	C1
27	Maluku	C1	C1	C1
28	Maluku Utara	C1	C1	C1
29	Papua	C1	C1	C1
30	Papua Barat	C1	C1	C1
31	DKI Jakarta	C1	C1	C1
32	Jawa Barat	C1	C1	C1
33	Jawa tengah	C1	C1	C1
34	Jawa Timur	C1	C1	C1

Hasil *clustering* yang disajikan oleh Tabel 7 menunjukkan bahwa untuk kasus positif, ketiga puluh provinsi termasuk ke dalam jumlah kasus positif rendah. Dua provinsi yaitu D.I Yogyakarta dan Banten termasuk ke dalam kasus positif dengan jumlah tertinggi dan dua provinsi lainnya yaitu Bali dan Nusa Tenggara Timur termasuk ke dalam provinsi dengan jumlah penderita Covid-19 kategori sedang. Selanjutnya, untuk kasus sembuh, ketiga puluh provinsi memiliki tingkat kesembuhan rendah. Provinsi D.I Yogyakarta dan Banten termasuk ke dalam tingkat kesembuhan sedang dan dua provinsi lainnya memiliki tingkat kesembuhan dengan presentase tertinggi. Pada kasus meninggal, polanya sama dengan kasus positif dimana dua provinsi termasuk ke dalam provinsi dengan jumlah kasus kematian tertinggi. Dua provinsi berikutnya yaitu Bali dan Nusa Tenggara Timur termasuk ke dalam provinsi dengan jumlah kasus kematian sedang dan ketiga puluh provinsi lainnya masuk ke dalam kategori jumlah kasus kematian sedikit.

### 3.3 Algoritma K-Medoids

Penentuan *cluster* menggunakan algoritma *K-Medoids* untuk data Covid-19 kasus positif, kasus sembuh, dan kasus meninggal, menggunakan tahapan yang sama seperti K-Means yaitu menentukan titik pusat *cluster* awal atau *medoid*. Tabel 8 menampilkan titik *medoid* awal.

**Tabel 8.** Titik *Medoid* Awal

Cluster	Data ke-	Kasus Positif	Kasus Sembuh	Kasus Meninggal
C1	7	0.013	0.013	0.007
C2	27	0.115	0.115	0.066
C3	25	0.027	0.026	0.026

Dari Tabel 8, penentuan titik *medoid* awal sebanyak  $K = 3$  untuk kasus positif dilakukan dengan mengambil data ke-7 sebagai pusat *medoid* C1, data ke-27 sebagai C2 dan data ke-25 sebagai C3. Setelah menentukan titik *medoid*, kemudian melakukan perhitungan jarak data ke *medoid* dengan ketiga metode perhitungan jarak, *Chebyshev Distance* (CD), *Manhattan Distance* (MD) dan *Euclidean Distance* (ED). Pada proses perhitungan metode jarak, *sample* data yang digunakan adalah kasus positif. Hasil perhitungan metode jarak kasus positif untuk CD, MD dan ED berhenti di iterasi ke 2 yang ditampilkan oleh Tabel 9.

**Tabel 9.** Perhitungan ketiga metode Jarak *Medoid* Kasus Positif

Provinsi Ke-	Jarak ke <i>Medoid</i>								
	CD			MD			ED		
	C <sub>1</sub>	C <sub>2</sub>	C <sub>3</sub>	C <sub>1</sub>	C <sub>2</sub>	C <sub>3</sub>	C <sub>1</sub>	C <sub>2</sub>	C <sub>3</sub>
1	0.018	0.004	0.031	0.084	0.526	0.969	0.106	0.025	0.089
2	0.061	0.084	0.110	0.005	0.446	0.889	0.027	0.054	0.010
3	0.042	0.065	0.091	0.024	0.465	0.908	0.045	0.035	0.029
4	0.088	0.110	0.136	0.022	0.420	0.863	0.000	0.081	0.017
5	0.028	0.006	0.020	0.094	0.536	0.979	0.116	0.035	0.099
6	0.007	0.030	0.056	0.059	0.500	0.944	0.081	0.000	0.064
7	0.036	0.014	0.013	0.102	0.544	0.987	0.124	0.043	0.107
8	0.005	0.018	0.044	0.071	0.512	0.956	0.093	0.012	0.076

⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
27	0.066	0.088	0.115	0.885	0.442	0.000	0.022	0.059	0.005
28	0.040	0.017	0.009	0.990	0.547	0.105	0.127	0.047	0.110
29	0.049	0.027	0.001	1.000	0.557	0.115	0.137	0.056	0.120
30	0.049	0.026	0.000	0.999	0.556	0.115	0.136	0.056	0.119
31	0.046	0.024	0.003	0.997	0.554	0.112	0.134	0.053	0.117
32	0.049	0.027	0.000	1.000	0.557	0.115	0.137	0.056	0.120
33	0.023	0.000	0.026	0.974	0.530	0.089	0.111	0.030	0.094
34	0.036	0.013	0.013	0.987	0.543	0.102	0.124	0.043	0.107

Tahapan selanjutnya yaitu membandingkan nilai *medoid* dari ketiga metode jarak menggunakan SC seperti yang ditampilkan oleh Tabel 10.

**Tabel 10.** Nilai Akurasi Metode Jarak Kasus Positif *K-Medoids*

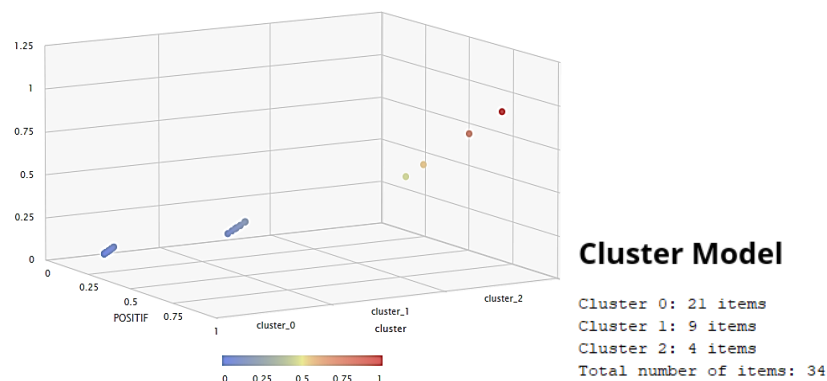
Akurasi	<i>Chebyshev</i>	<i>Manhattan</i>	<i>Euclidian</i>
SC	0.341	0.651	0.537

Berdasarkan Tabel 10, terlihat bahwa metode jarak *Manhattan* merupakan metode jarak terbaik untuk *K-Medoids* karena memiliki nilai akurasi paling tinggi. Dari hasil tersebut, menjadi dasar untuk mengelompokkan data Covid-19 yang menghasilkan titik *medoid* terkecil pada iterasi terakhir dan pemetaan *cluster* yang ditampilkan oleh Tabel 11.

**Tabel 11.** Hasil *Clustering* Kasus Positif *K-Medoids*

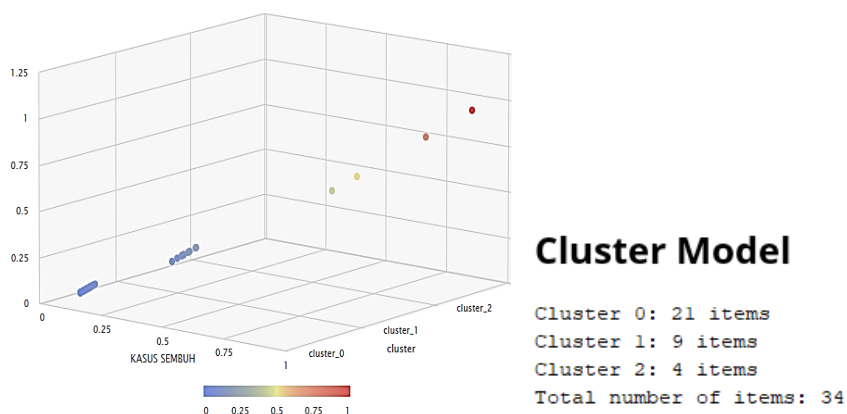
Provinsi	Jarak Ke <i>Medoid</i>			Hasil <i>Cluster</i>		
	Ke- C <sub>1</sub>	C <sub>2</sub>	C <sub>3</sub>	C <sub>1</sub>	C <sub>2</sub>	C <sub>3</sub>
1	0.084	0.526	0.969	*		
2	0.005	0.446	0.889	*		
3	0.024	0.465	0.908	*		
4	0.022	0.420	0.863	*		
5	0.094	0.536	0.979	*		
6	0.059	0.500	0.944	*		
7	0.102	0.544	0.987	*		
8	0.071	0.512	0.956	*		
⋮	⋮	⋮	⋮	⋮	⋮	⋮
27	0.885	0.000	0.442		*	
28	0.990	0.105	0.547		*	
29	1.000	0.115	0.557		*	
30	0.999	0.115	0.556		*	
31	0.997	0.554	0.112			*
32	1.000	0.557	0.115			*
33	0.974	0.530	0.089			*
34	0.987	0.543	0.102			*

Dari informasi yang berada di Tabel 11 menunjukkan bahwa terdapat dua puluh satu provinsi untuk kasus positif yang masuk ke dalam *cluster* C1. Selanjutnya, empat provinsi yang masuk ke dalam *cluster* C2 dan sembilan provinsi berada di *cluster* C3. Penentuan *cluster* mana yang termasuk ke dalam *cluster* rendah, sedang dan tinggi dapat dilihat dari hasil pengelompokkan menggunakan grafik dengan alat bantu *Rapidminer* yang ditunjukkan oleh Gambar 4.



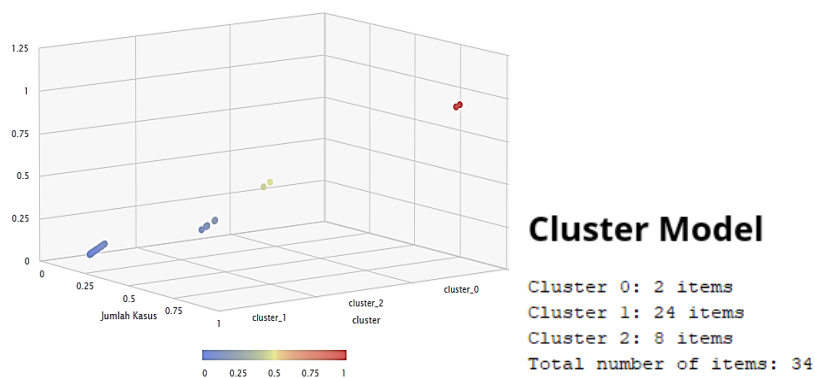
Gambar 4. Grafik Plot *K-Medoids* Hasil *Clustering* Kasus Positif

Plot grafik pada Gambar 4 menampilkan *cluster* C0 berwarna biru yang nilainya lebih rendah daripada dua *cluster* lainnya dengan jumlah data sebanyak 21. *Cluster* C1 berwarna jingga yang letaknya berada diantara *cluster* C0 dan *cluster* C1 dengan item sebanyak 9 data. *Cluster* C2 berwarna merah yang memiliki nilai indeks diatas C0 dan C1 dengan jumlah data yaitu 4 data. Dari hasil grafik tersebut maka ditemukan korelasi antara *cluster* C0 pada *Rapidminer* merupakan *cluster* C1 pada proses perhitungan yang termasuk ke dalam *cluster* rendah. Lebih lanjut, *cluster* C1 pada *Rapidminer* adalah *cluster* C2 pada proses perhitungan dengan *cluster* sedang dan *cluster* C2 pada *Rapidminer* adalah C3 pada proses perhitungan yang merupakan *cluster* tinggi. Setelah mendapatkan hasil *clustering* kasus positif, proses selanjutnya adalah menentukan hasil *clustering* kasus sembuh dan meninggal dengan menggunakan tahapan yang sama. Pada kasus sembuh hasil *clustering* menampilkan hasil yang sama dengan kasus positif seperti pada Gambar 5.



Gambar 5. Grafik Plot K-Medoids Hasil Clustering Kasus Sembuh

Hasil plot grafik *clustering* pada Gambar 5 untuk kasus sembuh tidak memiliki perbedaan dengan kasus positif. Pada kasus meninggal hasil *clustering* menampilkan hasil yang berbeda dengan dua kasus lainnya seperti pada Gambar 6.



Gambar 6. Grafik Plot K-Medoids Hasil Clustering Kasus Meninggal

Pada Gambar 6, hasil grafik menunjukkan korelasi antara *cluster* C0 pada *Rapidminer* merupakan *cluster* C1 pada proses perhitungan yang termasuk ke dalam *cluster* tinggi sebanyak 2 data. Lebih lanjut, *cluster* C1 pada *Rapidminer* adalah *cluster* C2 pada proses perhitungan yang termasuk *cluster* rendah berjumlah 24 data dan *cluster* C2 pada *Rapidminer* adalah C3 pada proses perhitungan berjumlah 8 data yang merupakan *cluster* sedang. Hasil *clustering* dari penerapan algoritma K-Medoids untuk keseluruhan kasus dapat dilihat pada Tabel 12.

Tabel 12. Hasil Clustering Algoritma K-Medoids

No	Provinsi	Cluster		
		Kasus Positif	Kasus Sembuh	Kasus Meninggal
1	Aceh	C1	C1	C1
2	Jambi	C1	C1	C1
3	Sumatera Selatan	C1	C1	C2
4	Bengkulu	C1	C1	C2
5	Lampung	C1	C1	C2
6	Bangka Belitung	C1	C1	C2
7	Kepulauan Riau	C1	C1	C2
8	Nusa Tenggara Barat	C1	C1	C2
9	Nusa Tenggara Timur	C1	C1	C2

10	Kalimantan Barat	C1	C1	C2
11	Kalimantan Tengah	C1	C1	C2
12	Kalimantan Utara	C1	C1	C2
13	Sulawesi Utara	C1	C1	C2
14	Sulawesi Tengah	C1	C1	C2
15	Sulawesi Tenggara	C1	C1	C2
16	Gorontalo	C1	C1	C2
17	Sulawesi Barat	C1	C1	C2
18	Maluku	C1	C1	C2
19	Maluku Utara	C1	C1	C2
20	Papua	C1	C1	C2
21	Papua Barat	C1	C1	C2
22	Sumatera Utara	C2	C2	C2
23	Sumatera Barat	C2	C2	C2
24	Riau	C2	C2	C2
25	DI Yogyakarta	C2	C2	C2
26	Banten	C2	C2	C2
27	Bali	C2	C2	C3
28	Kalimantan Selatan	C2	C2	C3
29	Kalimantan Timur	C2	C2	C3
30	Sulawesi Selatan	C2	C2	C3
31	DKI Jakarta	C3	C3	C3
32	Jawa Barat	C3	C3	C3
33	Jawa tengah	C3	C3	C3
34	Jawa Timur	C3	C3	C3

Tabel 12 menampilkan hasil pada kasus positif terdapat dua puluh satu provinsi yang termasuk ke dalam kasus positif berjumlah rendah, sembilan provinsi termasuk ke dalam jumlah kasus positif sedang dan empat provinsi lainnya memiliki jumlah penderita covid-19 dengan jumlah kasus positif tertinggi. Selanjutnya, pada kasus sembuh, menampilkan pola yang sama seperti pada kasus positif. Pada kasus meninggal, terdapat dua provinsi yang termasuk ke dalam provinsi dengan jumlah kasus kematian tertinggi. Selanjutnya, dua puluh empat provinsi lainnya masuk ke dalam kategori provinsi dengan jumlah kasus kematian rendah dan delapan provinsi lainnya masuk ke dalam kategori jumlah kasus kematian sedang.

### 3.4 Pengujian Akurasi Algoritma

Pengujian untuk menentukan algoritma terbaik dalam mengelompokkan data kasus Covid-19 dilakukan menggunakan tiga metode pengujian. Metode pengujian yang pertama yaitu *Sum Squared Error* (SSE), dimana semakin kecil nilai yang dihasilkan atau mendekati nol maka hasil *clustering* tersebut memiliki nilai kesalahan acak terkecil. Metode pengujian yang kedua adalah *Silhouette Coefficient* untuk melihat kualitas dan kekuatan *cluster* yang dihasilkan. Selanjutnya, metode *Davies Bouldin Index* (DBI) digunakan untuk melihat kuantitas dan kedekatan antar *cluster* yang dihasilkan. Hasil pengujian SSE, SC dan DBI untuk kedua algoritma ditampilkan oleh Tabel 13.

**Tabel 13.** Hasil Pengujian SSE, SC, DBI

Kasus	SSE		SC		DBI	
	<i>K-Means</i>	<i>K-Medoids</i>	<i>K-Means</i>	<i>K-Medoids</i>	<i>K-Means</i>	<i>K-Medoids</i>
Positif	3.381	3.248	0.838	0.651	0.305	0.752
Sembuh	3.251	3.154	0.838	0.647	0.295	0.849
Meninggal	4.049	4.869	0.925	0.609	1.569	0.560

Berdasarkan Tabel 13, hasil pengujian SSE untuk kedua algoritma diperoleh bahwa *K-Medoids* lebih unggul, karena memiliki nilai SSE terkecil pada kasus positif dan sembuh. Algoritma *K-Means* hanya memiliki nilai SSE terkecil di di kasus meninggal. Pada pengujian SC, diketahui bahwa algoritma *K-Means* unggul dalam pengujian ini. Nilai yang dihasilkan pada kasus positif, kasus sembuh dan kasus meninggal lebih tinggi daripada *K-Medoids*. Hal ini mengindikasikan bahwa kualitas *cluster* yang dihasilkan oleh *K-Means* untuk mengelompokkan ketiga kasus tersebut tergolong sesuai. Lebih lanjut, dari pengujian DBI, *K-Means* memiliki nilai terkecil pada kasus positif dan kasus sembuh dibandingkan algoritma *K-Medoids*. Nilai DBI yang terkecil pada *K-Means* membuktikan bahwa algoritma *K-Means* lebih unggul daripada *K-Medoids*. Hal ini dikarenakan pada pengujian DBI, semakin kecil nilai yang didapat, maka semakin baik *cluster* yang dihasilkan. Faktor yang mempengaruhi nilai DBI pada *K-Means* lebih kecil adalah penentuan titik pusat dari algoritma *K-Means* yang menghasilkan sebuah *cluster* berdasarkan kuantitas dan kedekatan antar anggota *cluster*. Lebih lanjut, dari hasil pengujian terlihat bahwa algoritma *K-Medoids* tidak unggul karena memiliki komputasi lebih sulit daripada *K-Means*, dimana *K-Medoids* menghitung *medoid* (titik pusat) menggunakan frekuensi kejadian.

#### 4. KESIMPULAN

Pengelompokan (*clustering*) terhadap data Covid-19 telah dilakukan menggunakan algoritma *K-Means* dan *K-Medoids*. Dalam proses *clustering*, ketiga metode jarak juga telah diterapkan yaitu *Chebyshev Distance*, *Manhattan Distance* dan *Euclidean Distance*. Dari ketiga jarak tersebut, metode *Manhattan Distance* merupakan metode jarak terbaik untuk kedua algoritma berdasarkan perhitungan akurasi menggunakan *Silhouette Coefficient* dengan nilai 0.838. Lebih lanjut, berdasarkan pengujian hasil *clustering* menggunakan metode *Silhouette Coefficient*, *Davies Bouldin Index* diperoleh kesimpulan bahwa *K-Means* unggul karena mampu menghasilkan kualitas dan kekuatan *cluster* secara maksimal berdasarkan jarak antar *cluster*. Sementara, *K-Medoids* memiliki keunggulan dalam menghasilkan *cluster* dengan nilai kesalahan acak terkecil berdasarkan pengujian metode *Sum Squared Error (SSE)*.

#### REFERENCES

- [1] A. N. Sihananto, A. P. Sari, H. Khariono, R. A. Fernanda, and D. C. M. Wijaya, "Implementasi Metode K-Means untuk Pengelompokan Kasus Covid-19 Tingkat Provinsi di Indonesia," *J. Inform. dan Sist. Inf.*, vol. 3, no. 1, pp. 76–85, 2022, doi: 10.33005/jifosi.v3i1.472.
- [2] G. D. Rembulan, T. Wijaya, D. Palullungan, K. N. Alfina, and M. Qurthuby, "Kebijakan Pemerintah Mengenai Coronavirus Disease (COVID-19) di Setiap Provinsi di Indonesia Berdasarkan Analisis Klaster," *Jiems (Journal Ind. Eng. Manag. Syst.)*, vol. 13, no. 2, 2020, doi: 10.30813/jiems.v13i2.2280.
- [3] M. W. Goni, D. Gustian, and F. Sembiring, "Implementasi K-Means Dalam Pengelompokan Penyebaran COVID-19 di Jawa Barat," *Progresif J. Ilm. Komput.*, vol. 17, no. 2, pp. 107–118, 2021.
- [4] Y. F. S. Y. Damanik, S. Sumarno, I. Gunawan, D. Hartama, and I. O. Kirana, "Penerapan Data Mining Untuk Pengelompokan Penyebaran Covid-19 Di Sumatera Utara Menggunakan Algoritma K-Means," *J. Ilmu Komput. dan Inform.*, vol. 1, no. 2, 2021.
- [5] A. Solichin and K. Khairunnisa, "Klasterisasi persebaran virus Corona (Covid-19) di DKI Jakarta menggunakan metode K-Means," *Fountain Informatics J.*, vol. 5, no. 2, pp. 52–59, 2020.
- [6] Z. Nabila, A. R. Isnain, P. Permata, and Z. Abidin, "Analisis Data Mining Untuk Clustering Kasus Covid-19 Di Provinsi Lampung Dengan Algoritma K-Means," *J. Teknol. Dan Sist. Inf.*, vol. 2, no. 2, pp. 100–108, 2021.
- [7] D. P. Sari, "Implementasi Algoritma K-Means Dalam Menentukan Tingkat Penyebaran Pandemi Covid-19 Di Sumatera Barat," *Comput. Based Inf. Syst. J.*, vol. 9, no. 1, 2021.
- [8] D. Sepri and Y. Fimazid, "Pengelompokan Penyebaran Covid-19 di Kota Padang Menggunakan Algoritma K-Medoids," *Inf. Syst. Res. J.*, vol. 1, no. 2, pp. 39–45, 2021.
- [9] S. Samudi, S. Widodo, and H. Brawijaya, "Algoritma K-Medoids untuk Menentukan Clustering Data Covid-19 di DKI Jakarta," *JURSIMA (Jurnal Sist. Inf. dan Manajemen)*, vol. 10, no. 1, pp. 122–127, 2022.
- [10] S. Sindi, W. R. O. Ningse, I. A. Sihombing, F. I. R. H. Zer, and D. Hartama, "Analisis Algoritma K-Medoids Clustering dalam Pengelompokan Penyebaran Covid-19 di Indonesia," *J. Teknol. Inf.*, vol. 4, no. 1, pp. 166–173, 2020.
- [11] U. R. Gurning and Mustakim, "Penerapan Algoritma K-Means dan K-Medoid untuk Pengelompokan Data Pasien Covid-19," *Build. Informatics, Technol. Sci.*, vol. 3, no. 1, p. 48–55, 2021, doi: 10.47065/bits.v3i1.1003.
- [12] A. Supriyadi, A. Triayudi, and I. D. Sholihati, "Perbandingan Algoritma K-Means dengan K-Medoids pada Pengelompokan Armada Kendaraan Truk Berdasarkan Produktivitas," *J. Ilm. Penelit. dan Pembelajaran Inform.*, vol. 6, no. 2, pp. 229–240, 2021, doi: 10.29100/jipi.v6i2.2008.
- [13] N. T. Luchia and Mustakim, "Perbandingan Algoritma K-Means dan K-Medoids pada Pengelompokan Humidity , Temperature , dan Voltage di Data Center Perawang," *J. Inf. Syst. Res.*, vol. 4, no. 1, pp. 184–190, 2022, doi: 10.47065/josh.v4i1.2385.
- [14] F. Farahdinna, I. Nurdiansyah, A. Suryani, and A. Wibowo, "Perbandingan Algoritma K-Means dan K-Medoids dalam Klasterisasi Produk Asuransi Perusahaan Nasional," *J. Ilm. FIFO*, vol. 11, no. 2, pp. 208–214, 2019, doi: 10.22441/fifo.2019.v11i2.010.
- [15] Athifaturrofifah, R. Goejantoro, and D. Yuniarti, "Perbandingan Pengelompokan K-Means dan K-Medoids Pada Data Potensi Kebakaran Hutan/Lahan Berdasarkan Persebaran Titik Panas (Studi Kasus : Data Titik Panas Di Indonesia Pada 28 April 2018)," *J. EKSPONENSIAL*, vol. 10, no. 2, pp. 143–152, 2019.
- [16] M. Herviany, S. P. Delima, T. Nurhidayah, and Kasini, "Perbandingan Algoritma K-Means dan K-Medoids untuk Pengelompokan Daerah Rawan Tanah Longsor di Provinsi Jawa Barat," *MALCOM (ndonesian J. Mach. Learn. Comput. Sci.)*, vol. 1, no. 1, pp. 34–40, 2021.
- [17] D. Marlina, N. F. Putri, A. Fernando, and A. Ramadhan, "Implementasi Algoritma K-Medoids dan K-Means untuk Pengelompokan Wilayah Sebaran Cacat pada Anak," *J. CoreIT J. Has. Penelit. Ilmu Komput. dan Teknol. Inf.*, vol. 4, no. 2, p. 64, 2018, doi: 10.24014/coreit.v4i2.4498.
- [18] M. Wahyudi, S. Solikhun, and L. Pujiastuti, "Komparasi K-Means Clustering dan K-Medoids Clustering dalam Mengelompokkan Produksi Susu Segar di Indonesia Berdasarkan Nilai DBI," *J. Bumigora Inf. Technol.*, vol. 4, no. 2, pp. 243–254, 2022.
- [19] E. Luthfi and A. W. Wijayanto, "Analisis Perbandingan Metode Hierarchy, K-Means, dan K-Medoids Clustering dalam Pengelompokan Indeks Pembangunan Manusia Indonesia," *INOVASI*, vol. 17, no. 4, pp. 761–773, 2021.
- [20] K. Anam, B. Nurhakim, and C. Juliane, "Komparasi Algoritma Klasifikasi Data Mining Menggunakan Optimize Selection untuk Peminatan Program Studi," *Build. Informatics, Technol. Science*, vol. 4, no. 2, pp. 606–613, 2022, doi: 10.47065/bits.v4i2.2160.
- [21] Haviluddin *et al.*, "A Performance Comparison of Euclidean, Manhattan and Minkowski Distances in K-Means Clustering," in *2020 6th International Conference on Science in Information Technology: Embracing Industry 4.0: Towards Innovation in Disaster Management, ICSITech 2020*, 2020, pp. 184–188, doi: 10.1109/ICSITech49800.2020.9392053.
- [22] I. S. Damanik, S. R. Andani, and D. Sehendo, "Teknik Data Mining dalam Clustering Produksi Susu Segar di Indonesia



- dengan Algoritma K-Means,” *BRAHMANA (Jurnal Penerapan Kecerdasan Buatan)*, vol. 1, no. 1, pp. 31–39, 2019, doi: 10.30645/brahmana.v1i1.5.
- [23] A. Septiarini, I. A. Thaher, and N. Puspitasari, “Pengelompokan Kualitas Kinerja Pegawai Menggunakan Metode K-Means Clustering,” *Komputika J. Sist. Komput.*, vol. 11, no. 2, pp. 131–141, 2022, doi: 10.34010/komputika.v11i2.5518.
- [24] E. Irfiani and S. S. Rani, “Algoritma K-Means Clustering untuk Menentukan Nilai Gizi Balita,” *JUSTIN (Jurnal Sist. dan Teknol. Informasi)*, vol. 6, no. 4, pp. 165–172, 2018.
- [25] N. L. Anggreini, “Teknik Clustering Dengan Algoritma K-Medoids Untuk Menangani Strategi Promosi Di Politeknik Tede Bandung,” *J. Teknol. Inf. dan Pendidik.*, vol. 12, no. 2, pp. 1–7, 2019, doi: 10.24036/tip.v12i2.215.
- [26] R. R. Muhima *et al.*, *Kupas Tuntas Algoritma Clustering: Konsep, Perhitungan Manual, dan Program*. Yogyakarta: Andi, 2022.
- [27] L. P. Refialy, H. Maitimu, and M. S. Pesulima, “Perbaikan Kinerja Clustering K-Means pada Data Ekonomi Nelayan dengan Perhitungan Sum of Square Error (SSE) dan Optimasi nilai K cluster,” *Techno.Com*, vol. 20, no. 2, pp. 321–329, 2021, doi: 10.33633/tc.v20i2.4572.
- [28] B. Wira, A. E. Budianto, and A. S. Wiguna, “Implementasi Metode K-Medoids Clustering Untuk Mengetahui Pola Pemilihan Program Studi Mahasiswa Baru Tahun 2018 Di Universitas Kanjuruhan Malang,” *RAINSTEK J. Terap. Sains Teknol.*, vol. 1, no. 3, pp. 53–68, 2019, doi: 10.21067/jtst.v1i3.3046.
- [29] I. Kamila, U. Khairunnisa, and Mustakim, “Perbandingan Algoritma K-Means dan K-Medoids untuk Pengelompokan Data Transaksi Bongkar Muat di Provinsi Riau,” *J. Ilm. Rekayasa dan Manaj. Sist. Inf.*, vol. 5, no. 1, pp. 119–125, 2019.