

# Pengelompokkan Kasus Covid-19 Di Indonesia Menggunakan Metode *Density-Based Spatial Clustering Of Applicatios With Noise (Dbscan)*

Oleh: Irnawati J Sinyor  
Univeristas Muhammadiyah Semarang

Article history	Abstract
Submission :	In early March, Indonesia entered Corona Virus Disease 2019 (Covid-19), every day cases of the spread of Covid-19 in Indonesia continued to increase. So that the government needs to establish more effective policies to handle the Covid-19 case in Indonesia. So a solution is needed to find out this problem is to create a system that can provide information about the data grouping of Covid-19 sufferers into clusters. One of the grouping methods is the Density-Based Spatial Clustering Application With Noise (DBSCAN) method. DBSCAN is a grouping algorithm based on data density. So that it can be seen which provinces have high, moderate and low number of sufferers. The use of DBSCAN does not need to determine the cluster itself because it is able to determine the number of clusters generated by itself. DBSCAN requires two parameters as a benchmark for grouping, namely Epsilon ( $\epsilon$ ) which is the radius (maximum distance) of group member data from the core data, and Minimal Points (MinPts) which is the minimum number of group data within the Epsilon radius ( $\epsilon$ ). The results of this study, using the optimum Epsilon (optimum) which is 20,000 and MinPts as much as 6, obtained the number of clusters as much as 2 and noise as much as 1.
Revised :	
Accepted :	
<b>Keyword:</b> <i>Data Mining, Clustering, Covid-19, DBSCAN</i>	

## PENDAHULUAN

Indonesia sedang dilanda pandemi Covid-19, virus ini pertama kali muncul di Wilayah Wuhan-tiongkok ditemukan pada akhir Desember 2019. *Corona Virus Disease* 2019 yang disingkat dengan Covid-19 merupakan penyakit yang baru ditemukan dan disebabkan oleh jenis Corona virus baru yaitu Sars-CoV-2. Virus ini menyebar dengan cepat ke berbagai Negara hampir ke seluruh dunia sehingga *World Health Organization* (WHO) menyatakan Covid-19 sebagai *pandemic* global yang secara resmi diumumkan pada tanggal 11 Maret 2020 (WHO, 2020) dan menjadi salah satu kejadian luar biasa yang tidak pernah diperkirakan sebelumnya. Covid-19 dapat berpindah secara tidak langsung dengan menyebarkan virusnya melalui benda-benda mati yang bertahan selama

dua jam sampai enam hari akibat terpapar virus dari sentuhan tangan seseorang yang terpapar.

Covid-19 menyebar dengan cepat ke seluruh dunia salah satunya Indonesia. Indonesia mengumumkan kasus pertama Covid-19 pada tanggal 2 Maret 2020 dengan penemuan 2 kasus di Depok. Jumlah ini terus meningkat dengan cepat hingga tanggal 19 Desember terdata total kasus Covid-19 terkonfirmasi sebanyak 657.948 jiwa, sembuh sebanyak 536.260 jiwa dan total kasus mening-gal sebanyak 19.659 jiwa. (covid19.go.id).

*Data mining* adalah sebuah proses pencarian secara otomatis informasi yang berguna dalam tempat penyimpanan data berukuran besar. *Data mining* adalah analisa terhadap data untuk menemukan hubungan yang

jelas serta menyimpulkannya yang belum diketahui sebelumnya dengan cara terkini dipahami dan berguna bagi pemilik data tersebut. Teknik *data mining* yang termasuk *descriptive mining* adalah *clustering*, *asosiation*, dan *sequential mining*.

*Clustering* adalah mengelompokkan data (objek) yang didasarkan hanya pada informasi yang ditemukan dalam data yang menggambarkan objek tersebut dan hubungan antar data (Tan et al., 2006). Tujuan analisis kelompok adalah agar objek-objek yang bergabung dalam sebuah kelompok merupakan objek-objek yang mirip (atau berhubungan) satu sama lain dan berbeda (atau tidak berhubungan) dengan objek dalam kelompok yang lain (Prasetyo, 2012).

Terdapat banyak metode yang bisa digunakan untuk mengelompokkan data, di antaranya ada K-Means, pengelompokkan hierarki, Fuzzy C-Means, self-organizing map, dan DBSCAN. *Density-Based Spatial Clustering Application With Noise* (DBSCAN) adalah algoritma pengelompokan yang didasarkan pada kepadatan (*density*) data. Konsep kepadatan dalam DBSCAN menghasilkan tiga macam status dari setiap data, yaitu inti (*core*), batas (*border*), dan noise. Jumlah kelompok yang terbentuk bergantung pada dua nilai yaitu Epsilon (*Eps*) yang merupakan radius (*jarak maksimal*) data anggota kelompok dari data inti, dan Minimal Points (*MinPts*) yang merupakan banyaknya minimal data kelompok dalam radius *Eps*. DBSCAN masuk kedalam pengelompokan sekatan (*partitioning*) dan eksklusif, yang artinya DBSCAN merupakan metode pengelompokan yang membagi set data ke dalam sejumlah kelompok yang tidak tumpang-tindih antara satu kelompok dengan kelompok yang lain. DBSCAN dapat mendeteksi data yang mempunyai jarak yang jauh dari suatu kelompok (*noise*) yaitu data yang tidak ikut dalam kelompok mayoritas (Tan et al., 2006).

Salah satu penelitian terdahulu terkait Metode *Density-Based Spatial Clustering Application With Noise* (DBSCAN) pernah dilakukan oleh ahamsyah (2020) untuk mengelompokkan penyebaran peting di Wilayah Pasuruan pada tahun 2018. Hasil yang didapat yaitu terdapat 2 cluster dan 15 titik *noise*, dengan anggota *cluster* pertama adalah lokasi yang paling banyak terjadi petir dan sangat rapat,

berjumlah 25535 titik sambaran yang menyebar hampir di seluruh Wilayah Pasuruan kecuali Prigen dan sekitarnya, anggota kedua adalah lokasi petir yang cukup rapat, namun kejadiannya tidak sebanyak petir pada *cluster* pertama, berjumlah 16 titik sambaran yang berada di sekitar Prigen. Sedangkan *noise* berada di sekitaran *cluster* kedua dan beberapa titik di sepanjang garis lintang -7,7 derajat.

Peneliti mengenai terkait *Density-Based Spatial Clustering Application With Noise* (DBSCAN) dan K-Means pernah dilakukan oleh Isnarwaty dkk (2019) untuk mengelompokkan pendapat menjadi beberapa kategori. Hasil yang didapat yaitu nilai *silhouette coefficient* tertinggi yang diperoleh, didapatkan hasil bahwa metode DBSCAN merupakan metode terbaik dibandingkan dengan metode K-Means untuk mengelompokkan tweet yang ditujukan kepada akun media sosial Twitter layanan ekspedisi JNE, J&T, dan Pos Indonesia. *Clustering* dengan metode terbaik menghasilkan 18 cluster untuk layanan ekspedisi JNE, 22 cluster untuk layanan ekspedisi J&T, dan 11 cluster untuk layanan ekspedisi Pos Indonesia. Kemudian ada penelitian tentang Covid-19 yang dilakukan oleh Sindi dkk (2020) menggunakan metode K-Medoids, pengklasteran terbaik dilakukan dengan 3 *cluster*. Dari 34 record diperoleh 1 record pada *cluster* pertama, 2 record pada *cluster* kedua, 31 record pada *cluster* ketiga.

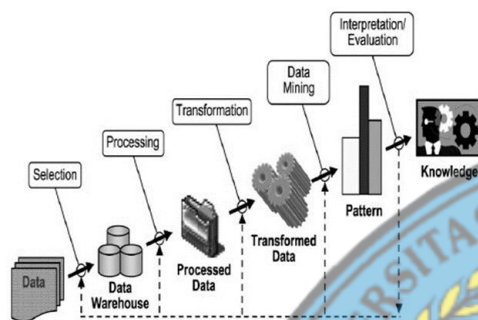
Dari uraian di atas, peneliti ingin menerapkan metode *Density-Based Spatial Clustering Application With Noise* (DBSCAN) pada perkembangan kasus Covid-19 di Indonesia untuk mengetahui pola pemilihan penentuan pengelompokan penyebaran Covid-19 di masing-masing Wilayah di Indonesia.

## LANDASAN TEORI

### Data Mining

*Data mining* merupakan suatu proses untuk menemukan informasi dari jumlah data yang besar (Zaki & Meira, 2014). *Data mining* mempunyai lima peran utama yaitu estimasi, prediksi, klasifikasi, klaster dan asosiasi (P.-N. Tan, Steinbach, & Kumar, 2006). Peran *data mining* yang sering digunakan adalah klasifikasi dan klaster karena dapat digunakan untuk atribut yang banyak (Fan, Wallace, Rich, & Zhang, 2006). Pengelompokan *Data Mining* dibagi menjadi beberapa kelompok yaitu: estimasi, prediksi, klasifikasi, pengklasteran dan asosiasi

*Data mining* adalah proses mencari pola atau informasi menarik dalam data terpilih dengan menggunakan teknik atau metode tertentu. Teknik-teknik, metode-metode, atau algoritma dalam *data mining* sangat bervariasi. Pemilihan metode atau algoritma yang tepat sangat bergantung pada tujuan dan proses *Knowledge Discovery in Database (KDD)* secara keseluruhan. *Knowledge Discovery in Database (KDD)* mengacu pada keseluruhan proses menemukan pengetahuan yang bermanfaat dari data. Ini melibatkan evaluasi dan kemungkinan interpretasi pola untuk membuat keputusan tentang apa yang memenuhi syarat sebagai pengetahuan.



Gambar 1. Tahapan dalam KDD

salah satu tahapan dalam keseluruhan proses KDD adalah data mining;

#### 1. Data Selection

Pemilihan (seleksi) data dari sekumpulan data operasional perlu dilakukan sebelum tahap penggalan informasi dalam *Knowledge Discovery in Database (KDD)* dimulai. Data hasil seleksi yang akan digunakan untuk proses *data mining*, disimpan dalam suatu berkas terpisah dari basis data operasional.

#### 2. Pre-processing / Cleaning

Sebelum proses data mining dapat dilaksanakan, perlu dilakukan proses cleaning pada data yang menjadi fokus *Knowledge Discovery in Database (KDD)*. Proses cleaning mencakup antara lain membuang duplikasi data, memeriksa data yang inkonsisten, dan memperbaiki kesalahan pada data, seperti kesalahan cetak. Juga dilakukan proses enrichment, yaitu proses “memperkaya” data yang sudah ada dengan data atau informasi lain yang relevan dan diperlukan untuk *Knowledge Discovery in Database (KDD)*, seperti data atau informasi eksternal lainnya yang diperlukan.

#### 3. Transformation

Transformasi data merupakan proses perubahan atau penggabungan data ke dalam format yang sesuai untuk diproses dalam data mining. Beberapa metode data mining membutuhkan format data yang khusus sebelum bisa diaplikasikan. Sebagai contoh beberapa metode standar seperti analisis asosiasi dan clustering hanya bisa menerima input data kategorikal. Karenanya data berupa angka numerik yang berlanjut perlu dibagi-bagi menjadi beberapa interval. (Eska, 2016).

#### 4. Data Mining

Data mining adalah proses mencari pola atau informasi menarik dalam data terpilih dengan menggunakan teknik atau metode tertentu. Teknik-teknik, metode-metode, atau algoritma dalam data mining sangat bervariasi. Pemilihan metode atau algoritma yang tepat sangat bergantung pada tujuan dan proses *Knowledge Discovery in Database (KDD)* secara keseluruhan.

#### 5. Interpretation / Evaluation

Pola informasi yang dihasilkan dari proses data mining perlu ditampilkan dalam bentuk yang mudah dimengerti oleh pihak yang berkepentingan. Tahap ini merupakan bagian dari proses *Knowledge Discovery in Database (KDD)* yang disebut interpretation. Tahap ini mencakup pemeriksaan apakah pola atau informasi yang ditemukan bertentangan dengan fakta atau hipotesis yang ada sebelumnya.

Evaluasi pola (pattern evaluation) digunakan untuk mengidentifikasi pola-pola menarik kedalam knowledge based yang ditemukan. Dalam tahap ini hasil dari teknik data mining berupa pola-pola yang khas maupun model prediksi dievaluasi untuk menilai apakah hipotesa yang ada memang tercapai (Eska, 2016).

#### Density-Based Spatial Clustering of Application with Noise (DBSCAN)

*Density Based Clustering Density-Based Spatial Clustering of Application with Noise (DBSCAN)* adalah metode pengelompokan berdasarkan tingkat kepadatan data (*density-based*). DBSCAN algoritma menumbuhkan area-area dengan kepadatan yang cukup tinggi ke dalam *cluster-cluster* dan menemukan *cluster-cluster* dalam bentuk yang sembarang dalam suatu *database spatial* yang memuat *noise* (Sander *et al.*, 1998). DBSCAN mendefinisikan

*cluster* sebagai himpunan maksimum dari titik-titik kepadatan yang terkoneksi (*density-connected*). Semua objek yang tidak masuk ke dalam *cluster* manapun dianggap sebagai *noise*.

Tahapan DBSCAN yaitu menghitung jarak titik pusat ke titik yang lain menggunakan jarak Euclidean lalu dinyatakan seperti Persamaan:

$$\text{Jarak} = \sqrt{(x-x_p)^2 + (y-y_p)^2} \quad (1)$$

Dengan:

- x : Koordinat sumbu x titik tujuan
- y : Koordinat sumbu y titik tujuan
- x<sub>p</sub> : Koordinat pusat sumbu x
- y<sub>p</sub> : Koordinat pusat sumbu y

Setelah terbentuk kelompok dilanjutkan dengan menghitung silhouette yang hasilnya bervariasi antara -1 hingga 1. Pengertian nilai dalam silhouette jika 1 maka berada dalam kelompok yang tepat. Jika 0 maka berada diantara dua kelompok sehingga tidak jelas masuk kelompok A atau B. Jika -1 maka struktur kelompok overlapping dan lebih tepat dimasukkan kekelompok lain. Jika lebih besar dari 0 dan mendekati 1 maka kelompok yang dihasilkan sudah optimal.

## Clustering

Clustering merupakan salah satu metode yang ada dalam data mining yang bertujuan untuk mengelompokkan data set ke dalam kelompok tertentu (*cluster*) dengan menggunakan parameter tertentu sehingga objek yang terdapat dalam sebuah cluster memiliki tingkat similaritas yang tinggi satu sama lain. Saat ini, para peneliti terus melakukan perbandingan model clustering guna mendapatkan cluster yang cocok digunakan pada sebuah penelitian (A. Ramadhan, Efendi, dan Mustakim, 2017).

Clustering merupakan suatu proses pengelompokan record suatu, observasi, atau mengelompokkan kelas yang memiliki kesamaan objek. Perbedaan clustering dengan klasifikasi yaitu tidak adanya variabel target dalam melakukan suatu pengelompokan pada proses *clustering*. Clustering sering dilakukan sebagai untuk langkah awal dalam proses data mining saat melakukan suatu metode analisis. Terdapat banyak algoritma Clustering yang telah digunakan oleh peneliti sebelumnya seperti K-

Means, Improved K-Means, Fuzzy C-Means, DBSCAN, K-Medoids (PAM), CLARANS dan Fuzzy Subtractive. Setiap algoritma memiliki kelebihan dan kekurangan masing-masing, namun prinsip algoritma sama, yaitu mengelompokkan data sesuai dengan karakteristik dan mengukur jarak kemiripan antar data dalam satu kelompok. Kualitas cluster diukur menggunakan *Silhouette Coefficient*

## Konsep Kepadatan (*Density Concept*)

DBSCAN menentukan sendiri jumlah *cluster* yang akan dihasilkan sehingga kita tidak perlu lagi untuk menentukan jumlah *cluster* yang diinginkan, tapi memerlukan 2 input lain, yaitu:

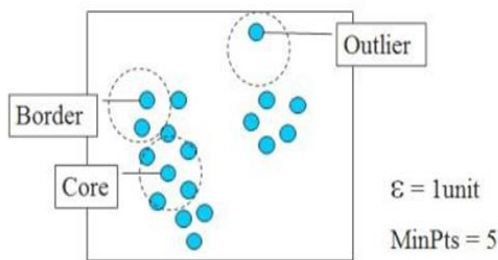
- a. *MinPts*: minimum banyak items dalam cluster
- b. *Eps*: nilai untuk jarak antar-items yang menjadi dasar pembentukan *neighborhood* dari suatu titik item.

*Neighborhood* yang terletak di dalam radius ( $\epsilon$ ) disebut  $\epsilon$ -neighborhood dari objek data.

Jika  $\epsilon$ -neighborhood dari suatu objek berisi paling sedikit suatu angka yang minimum, *MinPts* dari suatu objek, objek tersebut disebut *core object*.

Kepadatan data merupakan jumlah data yang berada dalam radius *MinPts* berupa jumlah data minimum dalam radius  $\epsilon$ . Konsep kepadatan memiliki tiga status (Prasetyo, 2012), yaitu:

1. Poin inti (*core*) : jumlah tetangga dan dirinya sendiri berada dalam radius  $\epsilon \geq \text{MinPts}$ .
2. Poin batas (border) : jumlah tetangga dan dirinya sendiri dalam radius  $\epsilon$  kurang dari *MinPts*, tetapi tetangga menjadi inti karena kehadirannya.
3. Poin pencilan (outlier) : jumlah tetangga dan dirinya sendiri dalam radius  $\epsilon$  kurang dari *MinPts* dan tidak ada tetangga yang menjadi inti karna kehadirannya.



Gambar 2. Gambar 1 Core dan Border

Menurut definisi, ada 2 jenis titik (*points*) dalam suatu *cluster*: di dalam *cluster* (*core points*) dan di tepian *cluster* (*border points*) di mana *neighborhood* dari border points berisi jauh lebih sedikit items daripada *neighborhood* dari core points (Ester et al., 1996).

DBSCAN menelusuri *cluster-cluster* dengan memeriksa  $\epsilon$  - *neighborhood* (*Eps-neighborhood*) dari tiap-tiap point dalam *database*. Jika  $\epsilon$  - *neighborhood* dari point *p* mengandung lebih dari *MinPts*, *cluster* baru dengan *p* sebagai *core object* diciptakan.

Kemudian DBSCAN secara iteratif mengumpulkan secara langsung objek-objek *density-reachable* dari core object tersebut, dimana mungkin melibatkan penggabungan dari beberapa *cluster-cluster* yang *density-reachable*.

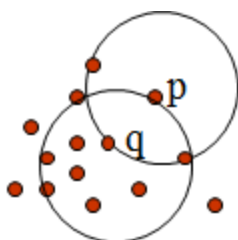
a. *Directly density-reachable*

Sebuah titik item dikatakan *directly density-reachable* dari titik lainnya jika jarak di antara mereka tidak lebih dari nilai *Eps*. *Directly density-reachable* = titik *q* dikatakan *directly density-reachable* dari titik *p* jika titik *q* adalah  $N_{eps}(p)$  dan *p* adalah core point.

$$N_{eps}(p) = \{q \text{ belongs to } D \mid \text{dist}(p,q) \leq Eps\}$$

Jarak dari titik ke titik lainnya tidak lebih dari nilai *Eps*.

$$\text{Core object } p: |N_{eps}(p)| \geq \text{MinPts}$$

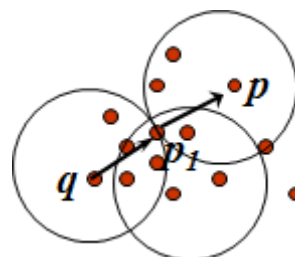


Gambar 3. *Directly Density-Reachable*

b. *Density-reachable*

Sebuah titik *item* dikatakan *density-reachable* dari titik *item* yang lain jika ada suatu rantai yang menghubungkan keduanya yang berisi hanya titik-titik yang *directly density-reachable* dari titik-titik sebelumnya.

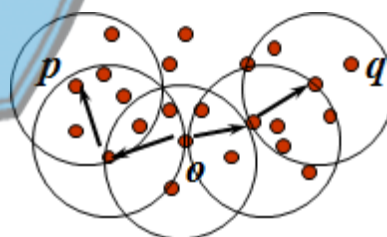
Suatu objek *p* adalah *density reachable* dari objek *q* dengan respek ke  $\epsilon$  dan *MinPts* dalam suatu set objek *D* jika terdapat suatu rantai objek  $p_1, p_2, \dots, p_n$ , dimana  $p_1 = q$  dan  $p_n = p$ , di mana  $p_{i+1}$  *density reachable* secara langsung dari  $p_i$  dengan respek ke  $\epsilon$  dan *MinPts*.



Gambar 4. *Density-Reachable*

c. *Density-connected*

Sebuah obyek *p* adalah *density-connected* terhadap obyek *q* dengan memperhatikan *Eps* dan *MinPts* dalam set obyek *D*, jika ada sebuah obyek *o* elemen *D* sehingga *p* dan *q* keduanya *density-reachable* dari *o* dengan memperhatikan *Eps* dan *MinPts*.



Gambar 5. *Density-Connected*

**Algoritma K-Nearest Neighbor (KNN)**

Algoritma k-nearest neighbor (Pencarian tetangga terdekat) merupakan teknik klasifikasi yang sangat populer yang diperkenalkan oleh Fix dan Hodges (1951), yang telah terbukti menjadi algoritma sederhana yang baik. KNN merupakan salah satu metode yang digunakan dalam pengklasifikasian dengan menggunakan algoritma supervised (Chan et al. 2010).

Nilai k yang terbaik untuk algoritma ini tergantung pada data. Umumnya, nilai k yang tinggi akan mengurangi efek noise pada klasifikasi, tetapi membuat batasan antara setiap klasifikasi menjadi lebih kabur. Nilai k yang bagus dapat dipilih dengan optimasi parameter, misalnya dengan menggunakan cross-validation. Kasus khusus di mana klasifikasi diprediksikan berdasarkan data pembelajaran yang paling dekat (dengan kata lain, k = 1) disebut algoritma nearest neighbor.

### Silhouette Coefficient

*Silhouette coefficient* merupakan metode evaluasi untuk menguji optimal atau ketepatan sebuah cluster yang telah terbentuk dari proses clustering (Tanzil Furqon and Muflikhah 2016). *Silhouette coefficient* memberikan hasil kualitas visual objek dalam tiap cluster, memberikan informasi sesuai dengan jumlah cluster pada data set. Untuk setiap objek dinotasikan oleh cluster dimana dia berasal (Swindiarito 2018). Metode ini merupakan gabungan dari metode separation dan cohesion. Tahapan perhitungan *Silhouette coefficient* adalah sebagai berikut (Handoyo, Rumani, and Nasution, 2014):

1. Hitung rata-rata jarak dari suatu data, menggunakan Persamaan 2 maka didapatkan rata-rata dengan cara memisalkan  $i$  terhadap semua data lain yang berada dalam satu *cluster* sebagai berikut.

$$\alpha(i) = \frac{1}{|A|-1} \sum_{j \in A, j \neq i} d(i, j) \quad (2)$$

Dimana :

$\alpha(i)$  : Perbedaan rata-rata objek (i) ke semua objek lain pada A  
 $d(i, j)$  : jarak antara data i dengan j  
 A : Cluster

2. Hitung rata-rata jarak data  $i$  tersebut dengan semua data di *cluster* lain, dan diambil nilai terkecilnya menggunakan Persamaan 2.

$$d(i, C) = \frac{1}{|C|} \sum_{j \in C} d(i, j) \quad (3)$$

Dimana :

$d(i, C)$  : Perbedaan rata-rata objek (i) ke semua objek lain pada C  
 C : Cluster lain selain cluster A atau cluster C tidak sama dengan cluster A.

3. Setelah menghitung  $d(i, C)$  untuk semua C, maka diambil nilai terkecil dengan menggunakan Persamaan 4.

$$b(i) = \min_{c \neq A} d(i, C) \quad (4)$$

Cluster B yang mencapai minimum (yaitu,  $d(i, B) = b(i)$ ) disebut tetangga dari objek (i). Ini adalah cluster terbaik kedua untuk objek (i).

4. Nilai *Silhouette Coefficient* didefinisikan seperti pada Persamaan 4 berikut:

$$S(i) = \frac{b(i) - \alpha(i)}{\max(\alpha(i), b(i))} \quad (5)$$

Dimana :

S(i) = Nilai Silhouette Coefficient.

b(i) = Nilai minimum objek i dengan objek pada cluster lain C.

$\alpha(i)$  = Rata-rata jarak objek ke i dengan semua objek yang berada di dalam suatu cluster.

Tabel 1. Tabel Nilai *Silhouette Kaufman* dan *Rousseeuw*

Nilai Silhouette Coefficient	Struktur
$0.7 < SC \leq 1$	Struktur Kuat
$0.5 < SC \leq 0.7$	Struktur Sedang
$0.25 < SC \leq 0.5$	Struktur Lemah
$SC \leq 0.25$	Tidak terstruktur

Setelah terbentuk kelompok dilanjutkan dengan menghitung silhouette yang hasilnya bervariasi antara -1 hingga 1. Pengertian nilai dalam silhouette jika 1 maka berada dalam kelompok yang tepat. Jika 0 maka berada diantara dua kelompok sehingga tidak jelas masuk kelompok A atau B. Jika -1 maka struktur kelompok overlapping dan lebih tepat dimasukkan kekelompok lain. Jika lebih besar dari 0 dan mendekati 1 maka kelompok yang dihasilkan sudah optimal.

## METODE PENELITIAN

### Sumber Data

Data yang digunakan adalah data bulanan Covid-19 pada tiap Provinsi di Indonesia dari tanggal 2 Maret 2020 sampai dengan tanggal 31 Desember 2020, dengan variable pasien positif, pasien sembuh, dan

pasien meninggal. Adapun populasi yang digunakan yaitu seluruh masyarakat di Indonesia. Data yang digunakan dalam penelitian ini adalah data sekunder yang diperoleh pada laman Kaggle pada halaman website <https://www.kaggle.com/hendratno/covid19-indonesia>.

### Variabel dan Struktur Data

Tabel 1. Variabel Data

Variabel	Keterangan
X1	Kasus Positif
X2	Kasus Sembuh
X3	Kasus Meninggal

Tabel 2. Struktur Data

Provinsi	X1	X2	X3
1	X <sub>1.1</sub>	X <sub>1.2</sub>	X <sub>1.3</sub>
...	...	...	...
...	...	...	...
...	...	...	...
...	...	...	...
...	...	...	...
34	X <sub>34.1</sub>	X <sub>34.2</sub>	X <sub>34.3</sub>

### Langkah Penelitian

Langkah-langkah dalam penelitian ini yaitu :

1. Menyiapkan data yang akan dianalisis
2. Melakukan analisis statistika deskriptif pada data untuk mengetahui gambaran awal persebaran kasus Covid-19 di Indonesia.
3. Penentuan Parameter *Epsilon*.
4. Inialisasi parameter *minpts*.
5. Inialisasi *Core* poin (*p*) menggunakan data koordinat yang pertama berdasarkan nilai minimum dari hasil statistika deskriptif.
6. Hitung radius  $\alpha$  atau semua jarak densitas terjangkau terhadap *r* menggunakan jarak Euclidean

$$\text{Jarak} = \sqrt{(x - xp)^2 + (y - yp)^2}$$

7. Menentukan titik-titik yang *density-reachable* terhadap *p* yang berada dalam radius  $\epsilon$ . Jika banyaknya titik yang *density-reachable* terhadap *p* lebih dari atau sama dengan *MinPts* maka titik *p* adalah

core point sehingga cluster terbentuk dan dilanjutkan ke titik lain disekitarnya yang merupakan titik yang *density-reachable* terhadap *p*. Jika *p* adalah border point dan tidak ada titik yang *density-reachable* terhadap *p*, maka proses dilanjutkan ke titik yang lain.

8. Ulangi langkah 5-7 hingga semua titik diproses.
9. Membuat peta tematik penyebaran Covid-19 di seluruh Provinsi di Indonesia berdasarkan *DBSCAN*.
10. Hasil pengelompokkan dianalisis lebih lanjut dengan melakukan interpretasi kelompoknya berdasarkan karakteristik pada setiap kelompok yang terbentuk.

## HASIL PENELITIAN dan PEMBAHASAN

### Analisis Statistika Deskriptif

Melihat gambaran umum dari kasus Covid-19 di Indonesia. Didapatkan hasil sebagai berikut:

Tabel 3. Hasil Statistika Deskriptif

	Positiv	Sembuh	Meninggal
Min	1941	1200	27,0
Median	3806	6850	260,0
Mean	21859	17619	639,0
Max	183735	164776	5827,0

Kasus Positiv Covid-19 tertinggi di Indonesia sebesar 183735 jiwa di Provinsi DKI Jakarta dan terendah sebesar 1941 jiwa di Provinsi Sulawesi Barat. Kasus Sembuh Covid-19 tertinggi sebesar 164776 jiwa di Provinsi DKI Jakarta dan terendah sebesar 1200 jiwa di Provinsi Nusa Tenggara Timur. Serta kasus meninggal tertinggi sebesar 5827 jiwa di Provinsi Jawa Timur dan Terendah sebesar 27 jiwa di Provinsi Kalimantan Barat.

### Deskripsi Data

Sebelum melakukan proses mining, dilakukan deskripsi data terlebih dahulu untuk melihat gambaran umum faktor yang mempengaruhi perkembangan kasus Covid-19 di Indonesia. Hasil dari perhitungan Normalisasi data diperoleh dengan program Rstudio. Normalisasi digunakan ketika data/atribut pada skala yang berbeda jauh. Agar memiliki skala/ukuran/dimensi yang sama maka dilakukan normalisasi data. Didapatkan hasil sebagai berikut:

Tabel 4. Hasil Normalisasi Data

No	Provinsi	Positiv	Sembuh	meninggal
1	Aceh	-0,362	-0,332	-0,236
2	Bali	-0,118	-0,163	-0,145
3	Banten	-0,102	-0,234	-0,197
4	Bengkulu	-0,504	-0,477	-0,443
5	DKI Jakarta	4,464	4,653	2,204
.	.	.	.	.
.	.	.	.	.
.	.	.	.	.
34	Sumatera Utara	-0,103	-0,071	0,034

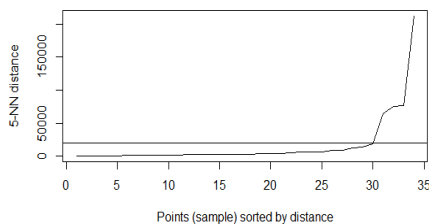
Tahap evaluasi cluster dilakukan dengan menggunakan *Average Silhouette width*. Skor nilai *Average Silhouette width* antara 0 sampai dengan 1, lebih besar dan mendekati 1 menunjukkan kualitas cluster yang lebih baik, yaitu memiliki jarak inter-cluster (jarak antar satu cluster dengan cluster lainnya) tinggi dan jarak intracluster (jarak antar obyek dengan obyek lainnya di dalam satu cluster yang sama) rendah. Hal tersebut menggambarkan bahwa setiap cluster terpisah cukup jauh dengan cluster yang lain sedang setiap anggota cluster memiliki kepadatan yang tinggi dengan obyek-obyek lainnya yang berada dalam satu cluster yang sama. Dapat dilihat pada Tabel 5.

Tabel 5. *Average Silhouette Width*

No	Eps	Minpts	jumlah Cluster	Jumlah Noise	Average Silhouette width
1	20.00	1	3	0	0
2	20.00	2	2	1	<b>0,83</b>
3	20.00	3	2	1	0,81
4	20.00	4	1	4	0,78
5	20.00	5	1	4	0,75
6	20.00	6	1	4	0,58

### Penentuan Parameter Epsilon

Pada tahap ini dilakukan beberapa skenario eksperimen clustering. Sebelum menentukan berapa eps yang akan digunakan untuk eksperimen DBSCAN, terlebih dahulu menghitung k-Nearest Neighbor Distance (kNNdist) yaitu penghitungan cepat jarak tetangga k-terdekat dalam matriks poin. Plot kNNdist dapat digunakan untuk membantu menemukan nilai eps yang tepat untuk DBSCAN. Pada plot kNNdist akan dicari knee karena menunjukkan nilai eps optimal. Berikut ini adalah gambar plot k-NN distance dengan k=5 untuk dataset yang digunakan pada penelitian ini.

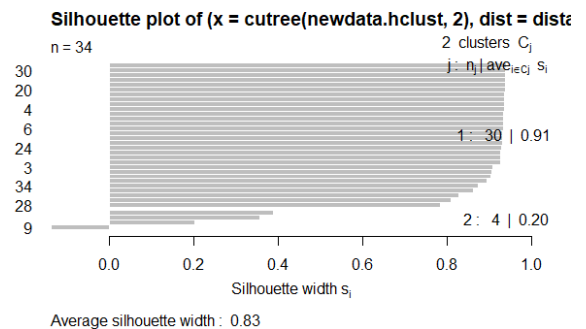


Gambar 6. Plot k-NN Distance

Dari Gambar 7 Plot k-NN Distance diperkirakan *knee* terletak di angka 20.000, hal ini menunjukkan potensi titik optimal di antara titik itu dan uji coba minPts 1 sampai dengan 6, sehingga eksperimen ditentukan beberapa eps dan minPts.

### Inisialisasi Parameter Minpts

Salah satu hasil cluster yang optimal pada epsilon 20.000 dengan minPts 2, terdiri dari dua cluster. Cluster 1 terdiri dari 30 item, cluster 2 hanya 3 item dan noise 1, nilai *average silhouette width* dapat dilihat pada gambar 8 :

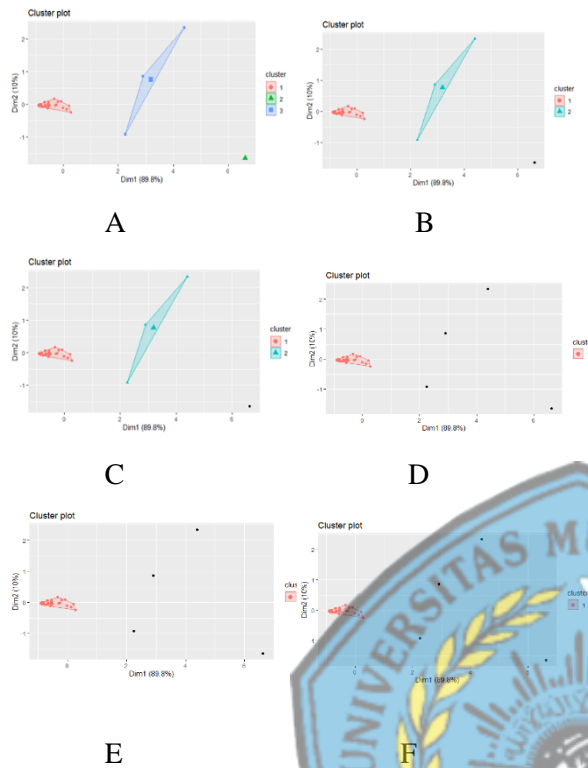


Gambar 7. *Silhouette Plot*

### Hasil Visualisasi Data



Setelah semua data sudah membentuk *cluster* dan menemukan *noise*, maka langkah terakhir adalah melakukan visualisasi data hasil *DBSCAN Clustering* seperti pada Gambar di bawah berikut:



Berikut hasil visualisasi data pada gambar di atas:

1. Gambar A menunjukkan hasil pengelompokan dari *DBSCAN Clustering* dengan  $Eps=20.000$  dan  $MinPts=1$ . Dari eksperimen tersebut dihasilkan 3 cluster, 0 noise, dan silhouette score sebesar 0.
2. Gambar B menunjukkan hasil pengelompokan dari *DBSCAN Clustering* dengan  $Eps=20.000$  dan  $MinPts=2$ . Dari eksperimen tersebut dihasilkan 2 cluster, 1 noise, dan silhouette score sebesar 0,83.
3. Gambar C menunjukkan hasil pengelompokan dari *DBSCAN Clustering* dengan  $Eps=20.000$  dan  $MinPts=3$ . Dari eksperimen tersebut dihasilkan 2 cluster, 1 noise, dan silhouette score sebesar 0,81.
4. Gambar D menunjukkan hasil pengelompokan dari *DBSCAN Clustering* dengan  $Eps=20.000$  dan  $MinPts=4$ . Dari eksperimen tersebut dihasilkan 1 cluster, 4 noise, dan silhouette score sebesar 0,78.

5. Gambar E menunjukkan hasil pengelompokan dari *DBSCAN Clustering* dengan  $Eps=20.000$  dan  $MinPts=5$ . Dari eksperimen tersebut dihasilkan 1 cluster, 4 noise, dan silhouette score sebesar 0,75.

6. Gambar F menunjukkan hasil pengelompokan dari *DBSCAN Clustering* dengan  $Eps=20.000$  dan  $MinPts=6$ . Dari eksperimen tersebut dihasilkan 1 cluster, 4 noise, dan silhouette score sebesar 0,58.

Setelah didapatkan hasil yang paling optimum dari *DBSCAN Clustering*. Selanjutnya adalah pembuatan peta tematik seperti pada gambar 8.

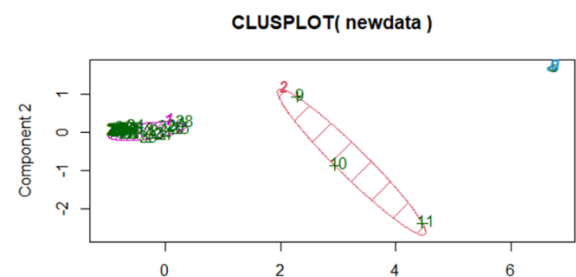


Gambar 8. Visualisasi Hasil *DBSCAN Clustering*

Memperjelas keberadaan hasil *DBSCAN Clustering* yang telah dilakukan dengan Rstudio. Hal ini dilakukan agar visualisasinya tampak jelas dengan adanya tambahan fitur Provinsi di Indonesia berdasarkan peta asli.

### Analisis Pola Penyebaran

Analisis Hasil *DBSCAN Clustering* Setelah model hasil *clustering* terbentuk, langkah terakhir adalah melakukan analisis berdasarkan hasil dari *DBSCAN Clustering*. Analisis dari hasil tersebut ditampilkan pada tabel berikut:



Gambar 9. Hasil *DBSCAN Clustering* dengan  $Epsilon (\epsilon) = 20.000$  dan  $MinPts=2$

Tabel 6. Hasil Pengelompokan dengan metode DBSCAN

Clas ter	Kode Provinsi	Anggota
0 Noise	5	DKI Jakarta
1	9,10,11	Jawa Barat, Jawa Tengah dan Jawa Timur
2	1,2,3,4,6,7,8,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30,31,32,33 dan 34	Aceh, Bali, Banten, Bengkulu, Daerah Istimewa Yogyakarta, Gorontalo, Jambi, Kalimantan Barat, Kalimantan Selatan, Kalimantan Tengah, Kalimantan Timur, Kalimantan Utara, Kepulauan Bangka Belitung, Kepulauan Riau, Lampung, Maluku, Maluku Utara, Nusa Tenggara Barat, Nusa Tenggara Timur, Papua, Papua Barat, Riau, Sulawesi Barat, Sulawesi Selatan, Sulawesi Tengah, Sulawesi Tenggara, Sulawesi Utara, Sumatera Barat, Sumatera Selatan dan Sumatera Utara

1 noise. dengan anggota cluster pertama berjumlah 3 titik sebaran, anggota cluster kedua berjumlah 30 titik sebaran dan berjumlah 1 titik sebaran.

2. Analisis dari hasil DBSCAN Clustering tersebut adalah sebagai berikut:

- a) Penyebaran Covid-19 yang tergolong noise adalah tingkat penyebaran Covid-19 yang terlalu tinggi, yang tergolong noise berjumlah 1 titik sebaran. Penyebaran Covid-19 ini terjadi di DKI Jakarta.
- b) penyebaran Covid-19 yang tergolong pada cluster pertama adalah tingkat penyebaran Covid-19 yang rendah, yaitu berjumlah 3 titik sebaran. Penyebaran Covid-19 ini terjadi di Jawa Barat, Jawa Tengah dan Jawa Timur.
- c) Penyebaran Covid-19 yang tergolong pada cluster kedua adalah tingkat penyebaran Covid-19 yang sedang, yaitu berjumlah 30 titik sebaran. Penyebaran Covid-19 ini terjadi di Aceh, Bali, Banten, Bengkulu, Daerah Istimewa Yogyakarta, Gorontalo, Jambi, Kalimantan Barat, Kalimantan Selatan, Kalimantan Tengah, Kalimantan Timur, Kalimantan Utara, Kepulauan Bangka Belitung, Kepulauan Riau, Lampung, Maluku, Maluku Utara, Nusa Tenggara Barat, Nusa Tenggara Timur, Papua, Papua Barat, Riau, Sulawesi Barat, Sulawesi Selatan, Sulawesi Tengah, Sulawesi Tenggara, Sulawesi Utara, Sumatera Barat, Sumatera Selatan dan Sumatera Utara

#### Saran

Berdasarkan Tabel 7. Hasil Pengelompokan dengan metode DBSCAN didapatkan dari hasil program Rstudio dapat disimpulkan bahwa cluster 1 yaitu cluster dengan tingkat penyebaran Covid-19 rendah, cluster 2 memiliki tingkat kebutuhan sedang dan Noise itu merupakan tingkat penyebaran Covid-19 tinggi.

#### SIMPULAN dan SARAN

##### Simpulan

Berdasarkan rumusan masalah beserta pembahasan di atas, dapat disimpulkan bahwa:

1. Hasil analisis cluster menggunakan metode DBSCAN didapatkan bahwa kombinasi parameter yang optimal adalah  $\epsilon = 20.000$  dan  $MinPts = 2$  dengan nilai Silhouette Score sebesar 0,83 yang menghasilkan 2 cluster dan

Pada penelitian ini, penulis terfokus pada pembuatan cluster dengan menggunakan algoritma DBSCAN pada perkembangan kasus Covid-19 di Indonesia. Adapun saran yang dapat diberikan adalah :

1. Pemerintah lebih memperhatikan penyebaran kasus Covid-19 pada provinsi di DKI Jakarta, upaya dan kebijakan pemerintah perlu di perketat, agar rantai penularan kasusu Covid-19 di Indonesia berkurang.
2. Menambahkan/Menggabungkan algoritma lain untuk melakukan prediksi pada penelitian selanjutnya.

#### Daftar Pustaka

Misbah, Fahamsyah. (2020). Metode DBSCAN Untuk Analisis Pola Penyebaran Petir di Pasuruan. *Disertasi*, tidak dipublikasikan.

- Universitas Islam Negeri Maulana Malik Ibrahim
- Devi Putri Isnarwaty dan Irhamah. (2019). Text Clustering pada Akun TWITTER Layanan Ekspedisi JNE, J&T, dan Pos Indonesia Menggunakan Metode Density-Based Spatial Clustering of Applications with Noise (DBSCAN) dan K-Means. *Jurnal. Institut Teknologi Sepuluh Nopember*
- Fajrin, A.A. & Maulana, Algifanri. 2018. Penerapan Data Mining Untuk Analisis Pola Pembelian Konsumen Dengan Algoritma FP-Growth Pada Data Transaksi Penjualan Spare Part Motor. *KLIK*. Vol 05, No. 01, Februari 2018, hlmn. 27-36. Batam: Universitas Putera Batam.
- Prasetyo, E. (2012). *Data Mining: Konsep dan Aplikasi menggunakan Matlab*. Yogyakarta: Andi Offset.
- Purwanto, Barus, U. Y., Adrianto, B., & Agung, H. (2012). Spatial Hotspots Clustering of Forest and Land Fires using DBSCAN and ST-DBSCAN. Bogor
- Ningsih, L., Atastina, I., Herdiani, A., 2018. Analisis dan Implementasi Community Detection Menggunakan Algoritma DBSCAN pada Twitter, Vol.5, No.1
- Atmaja, E. H. S. 'Implementation of k-Medoids Clustering Algorithm to Cluster Crime Patterns in Yogyakarta', *International Journal of Applied Sciences and Smart Technologies*, 1(1), pp. 33-44. doi: 10.24071/ijasst.v1i1.1859. 2019.
- Fatihatul, F., Setiawan, A., dan Rosadi, R. (2011). Asosiasi data mining menggunakan algoritma fp-growth untuk market basket analysis. *Jatinangor Univ. Padjadjaran*, 1-8.
- Furqon, M. T., dan Muflikhah, L. (2016). Clustering the potential risk of tsunami using density-based spatial clustering of application with noise (dbscan). *Journal of Environmental Engineering and Sustainable Technology*, 3(1), 1- 8.
- Han, J., Kamber, M., dan Pei, J. (2012). *Data mining: concepts and techniques*, waltham, ma. Morgan Kaufman Publishers, 10, 978-1.
- Pramudiono, I. (2006). Apa itu data mining. Dalam <http://datamining.japati.net/bin/indodm.cgi>, 28.
- Safitri, Diah dkk. 2017. Metode DBSCAN Untuk Pengelompokan Kabupaten/Kota di Provinsi Jawa Tengah Berdasarkan Produksi Padi Sawah dan Padi Ladang. *Statistika*. Vol 5, No. 1, Mei 2017, hlmn. 8-13. Semarang: Departemen Statistika, Fakultas Sains dan Matematika, Universitas Diponegoro.
- Sukma sindi, dkk. 2020. Analisis algoritma K-medoids clustering dalam pengelompokan penyebaran covid-19 di Indonesia. *Jurnal di Program Studi Teknik Informatika, STIKOM Tunas Bangsa Pematangsiantar*.
- Sulastrri, Heni & Gufroni, Acep Irham. 2017. Penerapan Data Mining Dalam Pengelompokan Penderita Thalassaemia. *Jurnal Nasional Teknologi dan Sistem Informasi*. Vol. 03, No. 02. Tasikmalaya: Universitas Siliwangi.
- Catchmeup. 2020. <https://catchmeup.id/covid-19/> diakses pada tanggal 19 Desember 2020.
- Kaggle. 2020. <https://www.kaggle.com/hendratno/covid19-indonesia> diakses pada tanggal 03 Desember 2020.

