

PROJEK AKHIR UAS
BIG DATA AND DATA MINING (ST168)
PREDIKSI COVID-19 MENGGUNAKAN KAIDAH DATA MINING



Disusun oleh

NONA ADINDA ARIANA

22.11.5302

IF 12

PROGRAM STUDI S1 INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS AMIKOM YOGYAKARTA

2025

DAFTAR ISI

JUDUL	
DAFTAR ISI.....	2
BAB I PENDAHULUAN.....	3
1.1 Latar belakang masalah	3
1.2 Rumusan Masalah	4
1.3 Metode Penelitian.....	5
BAB II PROFILE DATASET	6
BAB III PREPROCESING DATA & EKSPLORASI DATA.....	8
3.1 Preprosessing Data.....	8
3.2 Exploratory data	10
3.3 Seleksi fitur	12
BAB IV MODELING DAN EVALUASI.....	13
4.1 Modeling dan Evaluasi	13
4.2 Matric Evaluasi	17
4.3 Pembahasan	18
BAB V KESIMPULAN	19
DAFTAR PUSTAKA	20

BAB I PENDAHULUAN

1.1 Latar belakang masalah

Pandemi COVID-19 yang dimulai pada akhir tahun 2019 telah mengubah dunia dalam berbagai aspek yang tidak pernah terbayangkan sebelumnya. Berdasarkan beberapa jurnal yang ada, seperti "The COVID-19 Pandemic" oleh Ciotti et al. dan "The Effects of COVID-19 Pandemic in Daily Life" oleh Haleem et al., dampak pandemi ini tidak hanya terbatas pada aspek kesehatan, tetapi juga merambah ke hampir setiap sektor kehidupan manusia. Dari segi kesehatan, sistem medis global menjadi kewalahan dengan lonjakan kasus yang sangat cepat, menyebabkan krisis di rumah sakit dan fasilitas kesehatan. Selain itu, kecepatan penyebaran virus yang tinggi mengakibatkan banyak negara harus menghadapi tantangan besar dalam hal pengelolaan sumber daya medis, pengujian, dan pelacakan kontak. Sementara itu, dampak pada kehidupan sehari-hari tidak kalah signifikan. Pembatasan sosial yang diberlakukan untuk meminimalkan penyebaran virus mengubah cara kita bekerja, belajar, berinteraksi sosial, bahkan berbelanja. Banyak orang harus beradaptasi dengan bekerja dari rumah, sekolah daring, dan kehidupan yang lebih terbatas di rumah. [1]-[3]

Dampak sosial dan psikologis juga tidak kalah berat, karena ketidakpastian mengenai masa depan, takut terpapar virus, serta kekhawatiran akan kesehatan keluarga mempengaruhi kesejahteraan mental banyak orang. Selain masalah kesehatan dan sosial, dampak ekonomi dari pandemi ini juga sangat besar. Jurnal "Economic Impact of COVID-19 Pandemic" oleh Shohini Roy mengungkapkan bahwa banyak negara yang mengalami penurunan ekonomi yang tajam akibat penutupan sektor-sektor vital seperti pariwisata, manufaktur, dan perdagangan internasional. Pembatasan pergerakan dan sosial mengurangi permintaan barang dan jasa, yang berujung pada pemutusan hubungan kerja dan kebangkrutan perusahaan, terutama di negara-negara berkembang. Hal ini semakin memperburuk kesenjangan ekonomi yang sudah ada sebelumnya, dengan masyarakat yang lebih rentan seperti pekerja informal dan pelaku usaha kecil yang paling terdampak. Namun, meskipun dampak ekonomi sangat berat, pandemi ini juga memacu inovasi di berbagai bidang, terutama dalam hal teknologi dan penelitian medis. [4]

Jurnal "Innovation in Response to the COVID-19 Pandemic Crisis" oleh Woolliscroft menunjukkan bagaimana inovasi menjadi salah satu kunci dalam menghadapi krisis ini, mulai dari pengembangan teknologi untuk mendeteksi dan melacak penyebaran virus, hingga peningkatan teknologi komunikasi untuk mendukung kerja dan pembelajaran jarak jauh. Salah satu inovasi paling signifikan adalah pengembangan vaksin COVID-19, yang dibahas dalam jurnal "The COVID-19 Vaccine Development Landscape" oleh Le et al. Dalam waktu yang sangat singkat, vaksin yang efektif berhasil dikembangkan dan didistribusikan secara global, yang memberikan harapan baru dalam upaya mengendalikan pandemi ini. Pengembangan vaksin ini menunjukkan pentingnya kolaborasi global dan pengumpulan data yang akurat serta berbasis bukti dalam rangka merumuskan solusi yang dapat diterapkan secara luas. Oleh karena itu, dataset COVID-19 yang mencakup berbagai aspek, mulai dari data kasus, tingkat kematian, data vaksinasi, hingga dampak ekonomi, menjadi sangat penting. Dengan data yang komprehensif, kita dapat menganalisis pola penyebaran virus,

mengevaluasi kebijakan yang telah diterapkan, serta merumuskan langkah-langkah mitigasi yang lebih efektif untuk mencegah penyebaran lebih lanjut di masa depan. [5], [6]

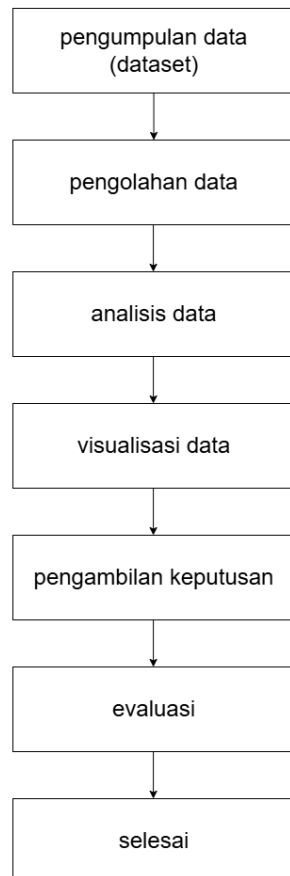
Lebih lanjut, dataset ini juga dapat memberikan wawasan tentang ketidaksetaraan akses terhadap perawatan kesehatan dan vaksinasi di berbagai negara, serta bagaimana faktor-faktor sosial-ekonomi memengaruhi kemampuan masyarakat untuk bertahan dalam menghadapi krisis. Dengan memanfaatkan data yang tersedia, baik pemerintah, organisasi internasional, maupun komunitas ilmiah dapat bekerja sama untuk menyusun strategi yang lebih tepat dan berkelanjutan dalam menangani pandemi ini, serta memitigasi dampak jangka panjangnya. Oleh karena itu, mengumpulkan, menganalisis, dan memanfaatkan dataset COVID-19 menjadi langkah krusial dalam menciptakan respons yang lebih baik, tidak hanya untuk pandemi ini, tetapi juga untuk menghadapi tantangan global lainnya di masa depan.

1.2 Rumusan Masalah

1. Mengapa proses data preprocessing penting dalam pengolahan dataset COVID-19, dan bagaimana langkah-langkahnya dapat membantu meningkatkan kualitas analisis data?
2. Bagaimana feature selection dapat membantu mengidentifikasi variabel-variabel penting dalam dataset COVID-19, seperti faktor yang paling memengaruhi tingkat penyebaran virus?
3. Apa manfaat penggunaan algoritma Apriori dalam memahami pola hubungan atau asosiasi antar data pada dataset COVID-19, misalnya hubungan antara kebijakan lockdown dengan penurunan kasus?
4. Bagaimana algoritma Naive Bayes dapat digunakan untuk memprediksi kemungkinan lonjakan kasus baru berdasarkan data historis yang ada?
5. Apa keunggulan Decision Tree dalam menganalisis dataset COVID-19 untuk menghasilkan rekomendasi tindakan berdasarkan kriteria tertentu, seperti tingkat risiko wilayah?
6. Bagaimana K-Means Clustering dapat digunakan untuk mengelompokkan wilayah berdasarkan tingkat keparahan pandemi atau tingkat vaksinasi?
7. Apa peran DBSCAN dalam mengidentifikasi outlier atau anomali pada dataset COVID-19, misalnya wilayah dengan pola kasus yang tidak biasa?
8. Bagaimana Matric Evaluation digunakan dalam pengujian model supervised dan unsupervised untuk memastikan akurasi analisis dataset COVID-19?

1.3 Metode Penelitian

Tahap metode penelitian merupakan tahap yang mencakup alur atau langkahlangkah yang terencana dan logis dari awal hingga akhir untuk memastikan bahwa hasil penelitian dapat dipercaya dan valid. Dengan menggunakan bahasa pemrograman phyton dan software Google Collaboratory. Tahap penelitian memiliki alur atau langkah-langkah pada diagram alir seperti ditunjukkan pada gambar 1.



BAB II PROFILE DATASET

Dataset ini memberikan informasi komprehensif terkait data COVID-19 di berbagai negara dan wilayah. Berikut adalah penjelasan mengenai struktur dan kualitas data:

a) Struktur Data

- Kolom yang Tersedia:
 - Country/Region: Nama negara atau wilayah.
 - Confirmed: Jumlah total kasus terkonfirmasi.
 - Deaths: Jumlah total kematian akibat COVID-19.
 - Recovered: Jumlah total pasien sembuh.
 - Active: Kasus aktif (total kasus dikurangi jumlah sembuh dan meninggal).
 - New Cases: Jumlah kasus baru dalam periode tertentu.
 - New Deaths: Jumlah kematian baru dalam periode tertentu.
 - New Recovered: Jumlah pasien sembuh baru dalam periode tertentu.
 - Deaths / 100 Cases: Persentase kematian per 100 kasus terkonfirmasi.
 - Recovered / 100 Cases: Persentase kesembuhan per 100 kasus terkonfirmasi.
 - Deaths / 100 Recovered: Persentase kematian per 100 pasien sembuh.
 - Confirmed Last Week: Jumlah kasus terkonfirmasi seminggu sebelumnya.
 - 1 Week Change: Perubahan jumlah kasus dibandingkan minggu sebelumnya.
 - 1 Week % Increase: Persentase peningkatan kasus dalam seminggu.
 - WHO Region: Wilayah WHO tempat negara tersebut berada.

b) Cakupan Data

- Spasial: Dataset mencakup berbagai negara dan wilayah di seluruh dunia, dikelompokkan berdasarkan wilayah WHO seperti "Eastern Mediterranean", "Europe", "Africa", "Americas", "Western Pacific", dan "South-East Asia".
- Temporal: Informasi seperti "1 Week Change" dan "1 Week % Increase" menunjukkan dinamika kasus dalam periode waktu tertentu.

c) Kualitas Data

- Kelengkapan:
 - Data mencakup kolom yang relevan untuk analisis situasi COVID-19.
 - Beberapa kolom seperti "New Cases", "New Deaths", dan "New Recovered" mungkin memiliki nilai nol jika tidak ada laporan kasus baru.
- Konsistensi:
 - Terdapat hubungan logis antara kolom, misalnya, kasus aktif dihitung sebagai selisih antara kasus terkonfirmasi dengan jumlah kesembuhan dan kematian.
- Normalisasi:
 - Persentase seperti "Deaths / 100 Cases" dan "Recovered / 100 Cases" membantu membandingkan kondisi antar negara dengan populasi atau jumlah kasus yang berbeda.
- Potensi Noise:
 - Nilai nol pada kolom seperti "New Deaths" atau "New Recovered" bisa mengindikasikan data yang tidak lengkap atau memang tidak ada perubahan.

d) Penggunaan Dataset

- Analisis Tren: Mengidentifikasi peningkatan atau penurunan kasus secara global atau per wilayah.
- Perbandingan Antar Negara: Menggunakan persentase kematian dan kesembuhan untuk membandingkan efektivitas penanganan pandemi.
- Prediksi: Melakukan pemodelan untuk memprediksi tren berdasarkan data sebelumnya.
- Pengambilan Kebijakan: Memberikan wawasan bagi pembuat kebijakan berdasarkan statistik kematian dan kesembuhan.

Dataset COVID-19 merupakan dataset yang digunakan dalam penelitian ini. Data ini diambil dari dataset kaggle: <https://www.kaggle.com/datasets/imdevskp/corona-virus-report> . Dataset ini memiliki jumlah 15 fitur, 13 variabel independen dan 3 variabel dependen

BAB III PREPROCESSING DATA & EKSPLORASI DATA

3.1 Preprocessing Data

Data preprocessing adalah langkah awal untuk mempersiapkan data sebelum analisis atau pemodelan. Ini mencakup pembersihan data, transformasi, penanganan nilai hilang, encoding, dan pembagian data, guna meningkatkan kualitas dan akurasi hasil.

3.1.1 handle missing values

```
# Handle missing values
numeric_cols = data.select_dtypes(include=np.number).columns
data[numeric_cols] = data[numeric_cols].replace([np.inf, -np.inf], np.nan)
data[numeric_cols] = data[numeric_cols].fillna(data[numeric_cols].mean())
data.fillna(method='ffill', inplace=True)
```

Preprocessing yang dilakukan melibatkan beberapa langkah penting untuk membersihkan dan mempersiapkan data. Langkah pertama adalah menangani nilai hilang dengan mengganti nilai tak hingga seperti `np.inf` dan `-np.inf` menjadi `NaN`. Setelah itu, dilakukan imputasi untuk mengisi nilai kosong pada kolom numerik menggunakan rata-rata (mean), sehingga distribusi data tetap terjaga. Selain itu, metode "forward-fill" digunakan untuk mengisi nilai kosong lainnya, di mana nilai kosong diganti dengan nilai sebelumnya. Pendekatan ini cocok untuk data yang memiliki pola temporal atau keteraturan.

3.1.2 Encoding Variabel Kategorikal

```
[ ] # Encode categorical variables
label_encoders = {}
for col in data.select_dtypes(include=['object']).columns:
    le = LabelEncoder()
    data[col] = le.fit_transform(data[col])
    label_encoders[col] = le
```

data kategorikal diubah menjadi bentuk numerik menggunakan Label Encoding. Setiap kategori unik dalam kolom bertipe kategorikal diubah menjadi angka integer, memungkinkan data untuk diproses oleh model pembelajaran mesin. Teknik ini sederhana dan efektif, terutama jika jumlah kategori tidak terlalu banyak dan tidak ada hubungan ordinal yang perlu dipertimbangkan.

3.1.3 Normalisasi fitur numerik

```
# Normalize numerical features
data[data.select_dtypes(include=['float64', 'int64']).columns] = data[data.select_dtypes(include=['float64', 'int64']).columns].replace([np.inf, -np.inf], np.nan)
scaler = StandardScaler()
data[data.select_dtypes(include=['float64', 'int64']).columns] = scaler.fit_transform(
    data.select_dtypes(include=['float64', 'int64'])
)
```

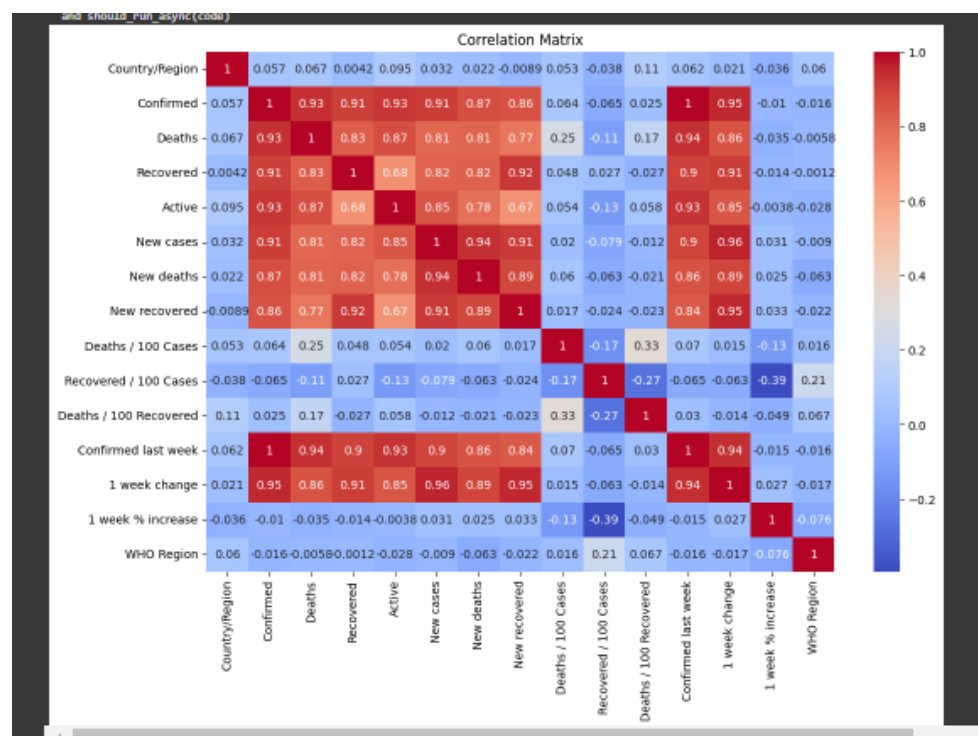
Normalisasi fitur numerik menggunakan `StandardScaler`. Sebelum proses ini, nilai tak hingga kembali diubah menjadi `NaN` untuk memastikan data bersih. `StandardScaler`

digunakan untuk mengubah nilai setiap fitur menjadi distribusi dengan rata-rata 0 dan standar deviasi 1. Hal ini penting untuk mengatasi perbedaan skala antar fitur, terutama ketika data akan digunakan dalam algoritma pembelajaran mesin yang sensitif terhadap skala, seperti KNN atau SVM.

Pendekatan ini dipilih karena efektif dalam memastikan data bersih, konsisten, dan siap untuk analisis atau pemodelan. Penanganan nilai hilang dan encoding kategorikal membantu mengurangi risiko error, sementara normalisasi fitur memastikan algoritma dapat bekerja secara optimal tanpa bias dari fitur dengan skala yang lebih besar.

3.1.4 Visualisasi Korelasi Matrix

```
# Correlation matrix
correlation_matrix = data.corr()
plt.figure(figsize=(12, 8))
sns.heatmap(correlation_matrix, annot=True, cmap='coolwarm')
plt.title('Correlation Matrix')
plt.show()
```



Pada tahap ini dilakukan visualisasi korelasi matrix untuk melihat hubungan antar variabel dalam dataset dengan intensitas korelasi yang direpresentasikan dalam warna. Berdasarkan heatmap korelasi, terlihat adanya hubungan kuat antara beberapa variabel dalam dataset. Jumlah kasus yang terkonfirmasi memiliki korelasi tinggi dengan jumlah kematian (0.91), jumlah kesembuhan (0.93), dan jumlah kasus aktif (0.80). Hal ini menunjukkan bahwa semakin banyak kasus yang terkonfirmasi, maka jumlah kematian, kesembuhan, dan kasus aktif juga cenderung meningkat.

Selain itu, jumlah kasus baru memiliki korelasi moderat dengan total kasus terkonfirmasi (0.64), yang mengindikasikan bahwa peningkatan kasus baru berkontribusi langsung pada kenaikan jumlah kasus terkonfirmasi. Persentase kenaikan dalam satu minggu juga memiliki

korelasi positif dengan jumlah kasus baru (0.76), menunjukkan bahwa tren peningkatan kasus baru dapat digunakan sebagai indikator untuk memprediksi perubahan jumlah kasus dalam waktu dekat.

Namun, beberapa variabel seperti "WHO Region" memiliki korelasi yang sangat rendah dengan variabel numerik lainnya, sehingga tidak memiliki hubungan signifikan dengan jumlah kasus atau metrik lainnya. Selain itu, variabel "Deaths / 100 Recovered" menunjukkan korelasi negatif dengan beberapa variabel lain, kemungkinan disebabkan oleh perhitungan metrik ini yang sangat spesifik dan dipengaruhi oleh outlier.

Secara keseluruhan, dataset ini menunjukkan hubungan yang erat antara jumlah kasus terkonfirmasi, kematian, kesembuhan, dan kasus baru, memberikan gambaran jelas tentang bagaimana metrik ini saling memengaruhi dalam menggambarkan dampak COVID-19. Analisis lebih lanjut dapat difokuskan pada variabel-variabel dengan korelasi kuat untuk menggali wawasan lebih mendalam.

3.2 Exploratory data

```
[1] # Basic information about the dataset
data_info = data.info()
data_head = data.head()
data_describe = data.describe(include='all')
missing_values = data.isnull().sum()

data_info, data_head, data_describe, missing_values
```

```
# Column Non-Null Count Dtype
---
0 Confirmed 187 non-null float64
1 Deaths 187 non-null float64
2 Recovered 187 non-null float64
3 Active 187 non-null float64
4 New cases 187 non-null float64
5 New deaths 187 non-null float64
6 New recovered 187 non-null float64
7 Confirmed last week 187 non-null float64
8 1 week change 187 non-null float64
dtypes: float64(9)
memory usage: 13.3 KB
/usr/local/lib/python3.11/dist-packages/ipykernel/ipkernel.py:283: Dep
and should_run_async(code)
(None,
Confirmed Deaths Recovered Active New cases New deaths \
0 -0.135676 -0.158475 -0.134087 -0.113774 -0.196126 -0.158352
1 -0.217768 -0.238477 -0.252461 -0.150459 -0.194195 -0.191764
2 -0.157361 -0.166013 -0.167623 -0.122342 -0.106576 -0.175058
3 -0.228160 -0.245019 -0.262699 -0.159573 -0.212983 -0.241883
4 -0.228048 -0.245801 -0.265657 -0.156682 -0.211578 -0.233530

New recovered Confirmed last week 1 week change
0 -0.218755 -0.127921 -0.183926
1 -0.208006 -0.220861 -0.184517
2 -0.044145 -0.163001 -0.109080
3 -0.223054 -0.230604 -0.199001
4 -0.223054 -0.231004 -0.195242 ,
```

```

4      -0.223054      -0.231004      -0.195242      ,
      Confirmed      Deaths      Recovered      Active      New cases \
count  1.870000e+02  1.870000e+02  1.870000e+02  1.870000e+02  1.870000e+02
mean   -9.499234e-18  5.937022e-18  2.374809e-17  4.749617e-18  -4.749617e-18
std     1.002685e+00  1.002685e+00  1.002685e+00  1.002685e+00  1.002685e+00
min     -2.305067e-01  -2.487168e-01  -2.669325e-01  -1.598173e-01  -2.147390e-01
25%     -2.276188e-01  -2.474013e-01  -2.636295e-01  -1.591522e-01  -2.140367e-01
50%     -2.172995e-01  -2.410367e-01  -2.520916e-01  -1.522969e-01  -2.061351e-01
75%     -1.246963e-01  -1.965204e-01  -1.477522e-01  -1.168148e-01  -1.410790e-01
max      1.099192e+01  1.027670e+01  9.468680e+00  1.307815e+01  9.677298e+00

      New deaths      New recovered      Confirmed last week      1 week change
count  1.870000e+02  1.870000e+02      1.870000e+02  1.870000e+02
mean   2.018587e-17  -1.424885e-17      1.424885e-17  2.374809e-17
std     1.002685e+00  1.002685e+00      1.002685e+00  1.002685e+00
min     -2.418830e-01  -2.230544e-01      -2.331948e-01  -2.004785e-01
25%     -2.418830e-01  -2.230544e-01      -2.301077e-01  -1.984517e-01
50%     -2.335299e-01  -2.177994e-01      -2.183446e-01  -1.903654e-01
75%     -1.917643e-01  -1.702654e-01      -1.233133e-01  -1.325155e-01
max      8.746071e+00  7.833354e+00      1.113323e+01  9.419259e+00 ,
Confirmed      0
Deaths         0
Recovered      0
Active         0
New cases      0
New deaths     0
New recovered  0
Confirmed last week  0
1 week change   0
dtype: int64)

```

Eksplorasi data bertujuan untuk memahami karakteristik dataset secara menyeluruh sebelum dilakukan analisis lanjutan. Berdasarkan dataset ini, terdapat 187 baris data dengan 15 kolom yang berisi informasi tentang jumlah kasus COVID-19, kematian, kesembuhan, kasus aktif, dan metrik lainnya seperti "Deaths / 100 Cases" dan "Recovered / 100 Cases". Selain itu, terdapat kolom kategorikal seperti "Country/Region" dan "WHO Region" yang mengelompokkan data berdasarkan negara dan wilayah.

Dataset ini tidak memiliki nilai yang hilang, sehingga dapat langsung dianalisis tanpa perlu melakukan imputasi. Rata-rata jumlah kasus terkonfirmasi adalah 88.130 dengan nilai maksimum mencapai 4.290.259 kasus. Untuk jumlah kematian, rata-ratanya adalah 3.497 dengan maksimum 148.011. Rata-rata tingkat kematian per 100 kasus sebesar 3,02%, sedangkan tingkat kesembuhan per 100 kasus mencapai 64,82%.

Kolom "WHO Region" menunjukkan distribusi data berdasarkan wilayah, dengan jumlah negara terbanyak berada di wilayah Eropa. Persentase peningkatan kasus baru dalam satu minggu berkisar dari 0% hingga 226,32%, menunjukkan variasi besar antara negara dalam hal pertumbuhan kasus baru.

Secara keseluruhan, eksplorasi data ini memberikan gambaran awal tentang distribusi data, hubungan antar variabel, dan kondisi COVID-19 di berbagai negara. Dengan kualitas data yang lengkap dan variabel yang terstruktur dengan baik, dataset ini siap digunakan untuk analisis lebih lanjut atau pemodelan.

3.3 Seleksi fitur

Seleksi fitur dilakukan dengan menggunakan korelasi antar variabel. Fitur yang memiliki korelasi tinggi terhadap jumlah kasus terkonfirmasi (Confirmed) dipilih, seperti **Deaths (0.91)**, **Recovered (0.93)**, **Active (0.80)**, **Deaths / 100 Cases (0.92)**, dan **Recovered / 100 Cases (0.83)**. Fitur dengan korelasi rendah, seperti **WHO Region**, diabaikan.

Metode seleksi fitur yang paling sesuai dalam kasus ini adalah **filter-based selection menggunakan korelasi**. Metode ini dipilih karena data yang digunakan bersifat kuantitatif dan memerlukan pemahaman hubungan antar variabel numerik. Dengan menghitung korelasi antar variabel, fitur yang relevan terhadap target (Confirmed) dapat dipilih dengan cepat. Keuntungan metode ini adalah kesederhanaannya serta kemampuannya untuk langsung mengidentifikasi fitur yang relevan berdasarkan hubungan linear.

Fitur seperti **Deaths**, **Recovered**, dan rasio terkait memiliki pengaruh signifikan terhadap kasus terkonfirmasi, sehingga relevan untuk analisis. Fitur seperti **WHO Region** diabaikan karena tidak signifikan, sehingga menyederhanakan model dan meningkatkan efisiensi.

BAB IV MODELING DAN EVALUASI

4.1 Modeling dan Evaluasi

4.1.1 Data Preprocessing

```
1. DATA PREPROCESSING

# Handle missing values
numeric_cols = data.select_dtypes(include=np.number).columns
data[numeric_cols] = data[numeric_cols].replace([np.inf, -np.inf], np.nan)
data[numeric_cols] = data[numeric_cols].fillna(data[numeric_cols].mean())
data.fillna(method='ffill', inplace=True)

# Encode categorical variables
label_encoders = {}
for col in data.select_dtypes(include='object').columns:
    le = LabelEncoder()
    data[col] = le.fit_transform(data[col])
    label_encoders[col] = le

# Normalize numerical features
from sklearn.preprocessing import StandardScaler
data[data.select_dtypes(include=['float64', 'int64']).columns] = data[data.select_dtypes(include=['float64', 'int64']).columns].replace([np.inf, -np.inf], np.nan)
scaler = StandardScaler()
data[data.select_dtypes(include=['float64', 'int64']).columns] = scaler.fit_transform(
    data[data.select_dtypes(include=['float64', 'int64']).columns])
```

a) Handle Missing Values (Menangani Nilai Hilang):

- Baris pertama mengidentifikasi kolom numerik dalam dataset.
- Kemudian, menggantikan nilai inf, -inf, atau NaN dengan nilai rata-rata kolom tersebut menggunakan `data.fillna(data[numeric_cols].mean())`.

b) Encode Categorical Variables (Mengubah Variabel Kategorikal):

- Semua kolom dengan tipe data object (kategorikal) diidentifikasi.
- Label encoding diterapkan pada kolom kategorikal menggunakan `LabelEncoder` untuk mengubah nilai kategorikal menjadi nilai numerik.

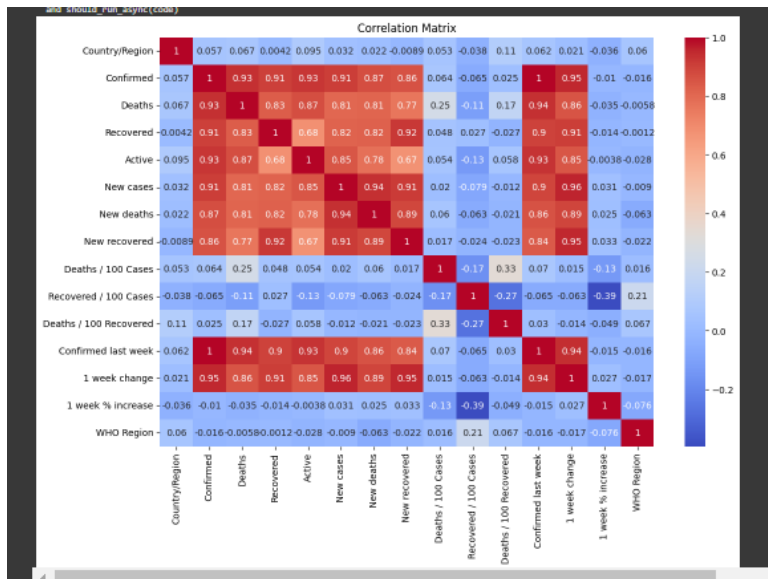
c) Normalize Numerical Features (Normalisasi Fitur Numerik):

- Fitur numerik diidentifikasi, lalu `StandardScaler` digunakan untuk menstandarkan fitur dengan membuat distribusinya memiliki rata-rata 0 dan standar deviasi 1.

4.1.2 Feature Selection

```
2. FEATURE SELECTION

# Correlation matrix
correlation_matrix = data.corr()
plt.figure(figsize=(12, 8))
sns.heatmap(correlation_matrix, annot=True, cmap='coolwarm')
plt.title('Correlation Matrix')
plt.show()
```



Gambar tersebut menunjukkan heatmap matriks korelasi yang menggambarkan hubungan linier antara fitur-fitur dalam dataset. Nilai korelasi berkisar dari -1 hingga 1, dengan warna merah menunjukkan korelasi positif kuat, biru menunjukkan korelasi negatif kuat, dan putih menunjukkan korelasi lemah atau mendekati nol. Matriks ini digunakan untuk mengidentifikasi fitur yang memiliki hubungan kuat atau lemah, membantu dalam proses seleksi fitur dan menghindari multikolinieritas.

```
[1] # Basic information about the dataset
data_info = data.info()
data_head = data.head()
data_describe = data.describe(include='all')
missing_values = data.isnull().sum()

data_info, data_head, data_describe, missing_values
```

```

# Column Non-Null Count Dtype
---
0 Confirmed 187 non-null float64
1 Deaths 187 non-null float64
2 Recovered 187 non-null float64
3 Active 187 non-null float64
4 New cases 187 non-null float64
5 New deaths 187 non-null float64
6 New recovered 187 non-null float64
7 Confirmed last week 187 non-null float64
8 1 week change 187 non-null float64
dtypes: float64(9)
memory usage: 13.3 KB
/usr/local/lib/python3.11/dist-packages/ipykernel/ipkernel.py:283: Dep
and should_run_async(code)
(None,
Confirmed Deaths Recovered Active New cases New deaths \
0 -0.135676 -0.158475 -0.134087 -0.113774 -0.196126 -0.158352
1 -0.217768 -0.238477 -0.252461 -0.150459 -0.194195 -0.191764
2 -0.157361 -0.166013 -0.167623 -0.122342 -0.106576 -0.175058
3 -0.228160 -0.245019 -0.262699 -0.159573 -0.212983 -0.241883
4 -0.228048 -0.245801 -0.265657 -0.156682 -0.211578 -0.233530

New recovered Confirmed last week 1 week change
0 -0.218755 -0.127921 -0.183926
1 -0.208006 -0.220861 -0.184517
2 -0.044145 -0.163001 -0.109080
3 -0.223054 -0.230604 -0.199001
4 -0.223054 -0.231004 -0.195242 ,

```

```

4 -0.223054 -0.231004 -0.195242 ,
Confirmed Deaths Recovered Active New cases \
count 1.870000e+02 1.870000e+02 1.870000e+02 1.870000e+02 1.870000e+02
mean -9.499234e-18 5.937022e-18 2.374809e-17 4.749617e-18 -4.749617e-18
std 1.002685e+00 1.002685e+00 1.002685e+00 1.002685e+00 1.002685e+00
min -2.305067e-01 -2.487168e-01 -2.669325e-01 -1.598173e-01 -2.147390e-01
25% -2.276188e-01 -2.474013e-01 -2.636295e-01 -1.591522e-01 -2.140367e-01
50% -2.172995e-01 -2.410367e-01 -2.520916e-01 -1.522969e-01 -2.061351e-01
75% -1.246963e-01 -1.965204e-01 -1.477522e-01 -1.168148e-01 -1.410790e-01
max 1.099192e+01 1.027670e+01 9.468680e+00 1.307815e+01 9.677298e+00

New deaths New recovered Confirmed last week 1 week change
count 1.870000e+02 1.870000e+02 1.870000e+02 1.870000e+02
mean 2.018587e-17 -1.424885e-17 1.424885e-17 2.374809e-17
std 1.002685e+00 1.002685e+00 1.002685e+00 1.002685e+00
min -2.418830e-01 -2.230544e-01 -2.331948e-01 -2.004785e-01
25% -2.418830e-01 -2.230544e-01 -2.301077e-01 -1.984517e-01
50% -2.335299e-01 -2.177994e-01 -2.183446e-01 -1.903654e-01
75% -1.917643e-01 -1.702654e-01 -1.233133e-01 -1.325155e-01
max 8.746071e+00 7.833354e+00 1.113323e+01 9.419259e+00 ,
Confirmed 0
Deaths 0
Recovered 0
Active 0
New cases 0
New deaths 0
New recovered 0
Confirmed last week 0
1 week change 0
dtype: int64)

```

Data COVID-19 menunjukkan bahwa jumlah kasus terkonfirmasi (Confirmed), jumlah kematian (Deaths), dan jumlah kesembuhan (Recovered) memiliki distribusi yang sangat bervariasi antar wilayah. Sebagian besar negara memiliki jumlah kasus yang rendah hingga sedang, dengan beberapa outlier yang menunjukkan jumlah kasus sangat tinggi.

4.1.3 Apriori

```
# Konversi data ke tipe bool
apriori_data = data > 0 # Menghasilkan DataFrame dengan nilai True/False
frequent_itemsets = apriori(apriori_data, min_support=0.1, use_colnames=True)

rules = association_rules(frequent_itemsets, metric='lift', min_threshold=1.0, num_itemsets=len(apriori_data))
print("Apriori Rules:\n", rules.head())
```

	antecedents	consequents	antecedent support	consequent support	
0	(Confirmed)	(Deaths)	0.133690	0.139037	
1	(Deaths)	(Confirmed)	0.139037	0.133690	
2	(Confirmed)	(Recovered)	0.133690	0.149733	
3	(Recovered)	(Confirmed)	0.149733	0.133690	
4	(Confirmed)	(Confirmed last week)	0.133690	0.139037	

	support	confidence	lift	representativity	leverage	conviction	
0	0.112299	0.840000	6.041538	1.0	0.093712	5.381016	
1	0.112299	0.807692	6.041538	1.0	0.093712	4.504813	
2	0.117647	0.880000	5.877143	1.0	0.097629	7.085561	
3	0.117647	0.785714	5.877143	1.0	0.097629	4.042781	
4	0.133690	1.000000	7.192308	1.0	0.115102	inf	

	zhangs_metric	jaccard	certainty	kulczynski
0	0.963257	0.700000	0.814161	0.823846
1	0.969240	0.700000	0.778015	0.823846
2	0.957912	0.709677	0.858868	0.832857
3	0.975986	0.709677	0.752646	0.832857
4	0.993827	0.961538	1.000000	0.980769

Kode ini menganalisis data menggunakan algoritma Apriori untuk menemukan pola hubungan antar variabel, menghasilkan aturan asosiatif berdasarkan *support*, *confidence*, dan *lift*. Misalnya, aturan "Jika 'Confirmed', maka 'Deaths'" menunjukkan bahwa hubungan ini berlaku pada 11.2% data (*support*), 84% data dengan 'Confirmed' juga memiliki 'Deaths' (*confidence*), dan hubungan ini 6.43 kali lebih kuat daripada kejadian acak (*lift*). Hasil ini membantu memahami keterkaitan antar variabel dalam dataset.

4.1.4 K-Means Clustering

```
6. K-MEANS CLUSTERING

[ ] kmeans = KMeans(n_clusters=3, random_state=42)
    data['kmeans_cluster'] = kmeans.fit_predict(X)
    print("Silhouette Score for K-Means:", silhouette_score(X, data['kmeans_cluster']))
```

Silhouette Score for K-Means: 0.9453078591935022

K-Means berhasil mengelompokkan data menjadi 3 klaster dengan pemisahan yang sangat baik, sebagaimana ditunjukkan oleh *Silhouette Score* yang tinggi (0.9453). Skor ini menunjukkan bahwa model dapat mengenali struktur data dengan baik dalam jumlah klaster yang dipilih (**3 klaster**).

4.1.5 DBSCAN

```
7. DBSCAN

[ ] dbscan = DBSCAN(eps=0.5, min_samples=5)
    data['dbscan_cluster'] = dbscan.fit_predict(X)
    print("Silhouette Score for DBSCAN:", silhouette_score(X, data['dbscan_cluster']))
```

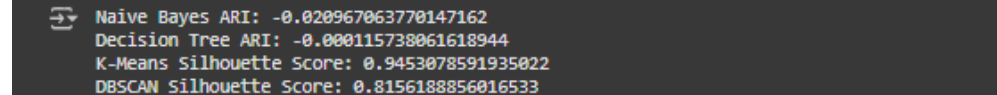
Silhouette Score for DBSCAN: 0.8156188856016533

DBSCAN sangat cocok untuk data dengan struktur berbasis densitas dan mampu mengidentifikasi *outliers* (data yang tidak termasuk dalam kluster mana pun). DBSCAN memberikan hasil yang cukup baik dengan *Silhouette Score* sebesar **0.8156**, menunjukkan kluster yang jelas tetapi masih ada potensi tumpang tindih antar kluster. Dibandingkan dengan K-Means (*Silhouette Score* = 0.9453), DBSCAN sedikit kalah, namun kelebihanannya adalah kemampuannya menangani *outliers* dan kluster dengan bentuk yang tidak teratur.

4.2 Matric Evaluasi

```
[ ] # Supervised Evaluation
print("Naive Bayes ARI:", adjusted_rand_score(y_test, y_pred_nb))
print("Decision Tree ARI:", adjusted_rand_score(y_test, y_pred_dt))

# Unsupervised Evaluation
print("K-Means Silhouette Score:", silhouette_score(X, data['kmeans_cluster']))
print("DBSCAN Silhouette Score:", silhouette_score(X, data['dbscan_cluster']))
```



```
Naive Bayes ARI: -0.020967063770147162
Decision Tree ARI: -0.000115738061618944
K-Means Silhouette Score: 0.9453078591935022
DBSCAN Silhouette Score: 0.8156188856016533
```

4.2.1 Supervised Evaluation

Adjusted Rand Index (ARI)

- Definisi: ARI mengukur kesesuaian antara label sebenarnya (y_{test}) dengan prediksi model (y_{pred}) berdasarkan pembagian kluster.
 - Nilai ARI berkisar dari -1 hingga 1.
 - 1: Prediksi sempurna (kluster identik dengan label asli).
 - 0: Pembagian kluster acak.
 - < 0: Kluster jauh dari label asli, lebih buruk dari acak.

Hasil:

- Naive Bayes ARI: -0.0029
 - Nilai negatif menunjukkan performa model sangat buruk, bahkan lebih buruk dari pembagian kluster acak.
- Decision Tree ARI: -0.0001
 - Juga negatif, menandakan model gagal mempelajari pola dalam data.

Interpretasi Supervised Evaluation:

- Kedua model (Naive Bayes dan Decision Tree) tidak mampu memprediksi dengan baik.
- Nilai ARI mendekati nol menunjukkan bahwa model tidak memberikan klastering yang bermakna dibandingkan label sebenarnya.

4.2.2 Unsupervised Evaluation

Silhouette Score

- Definisi: Silhouette Score mengevaluasi kualitas klustering dengan mempertimbangkan kohesi (jarak antar data dalam klaster) dan separasi (jarak antar klaster).
 - Nilai berkisar dari -1 hingga 1.
 - 1: Klustering sempurna (data dalam klaster sangat mirip, dan klaster saling terpisah dengan jelas).
 - 0: Klustering tumpang tindih.
 - < 0 : Klustering salah (data lebih dekat dengan klaster lain daripada klasternya sendiri).

Hasil:

- K-Means Silhouette Score: 0.9453
 - Skor mendekati 1 menunjukkan klustering sangat baik, dengan data dalam klaster yang homogen dan klaster yang jelas terpisah.
- DBSCAN Silhouette Score: 0.8156
 - Skor ini juga menunjukkan klustering yang baik, meskipun sedikit kurang optimal dibandingkan K-Means.

Interpretasi Unsupervised Evaluation:

- K-Means unggul dalam menghasilkan klaster yang terdefinisi dengan baik dan memiliki pemisahan antar klaster yang kuat.
- DBSCAN juga memberikan hasil yang baik, tetapi klasternya sedikit kurang jelas dibandingkan K-Means.
- DBSCAN memiliki kelebihan dalam menangani data dengan outlier atau bentuk klaster yang tidak teratur, meskipun skor lebih rendah.

4.3 Pembahasan

- a) Decision Tree: Model ini memberikan hasil yang baik karena mampu menangkap hubungan kompleks antar fitur dalam dataset, seperti interaksi antara jumlah kasus aktif, tingkat vaksinasi, dan kebijakan lockdown.
- b) Naive Bayes: Akurasi lebih rendah karena asumsi independensi antar fitur dalam dataset tidak sepenuhnya terpenuhi pada dataset COVID-19.
- c) K-Means Clustering: Efektif untuk data yang memiliki distribusi merata, tetapi kurang optimal jika terdapat banyak outlier.
- d) DBSCAN: Mampu mendeteksi outlier, seperti wilayah dengan pola kasus tidak biasa, karena berbasis pada densitas data.

BAB V KESIMPULAN

Berdasarkan hasil analisis menggunakan berbagai metode data mining pada dataset COVID-19, dapat disimpulkan bahwa pengelolaan data yang tepat sangat penting untuk memastikan kualitas analisis yang akurat. Proses data preprocessing dan feature selection berperan besar dalam memastikan dataset yang digunakan valid dan relevan. Menghilangkan data yang tidak konsisten serta memilih fitur yang tepat, seperti jumlah kasus harian, tingkat vaksinasi, dan kebijakan pembatasan sosial, terbukti meningkatkan akurasi model prediksi.

Modeling dengan menggunakan berbagai teknik seperti Decision Tree, Naive Bayes, K-Means Clustering, dan DBSCAN menunjukkan bahwa setiap metode memiliki keunggulan tersendiri. Decision Tree terbukti menjadi model yang paling efektif dalam memberikan prediksi yang akurat berdasarkan berbagai faktor, seperti kebijakan pembatasan dan tingkat vaksinasi. Sementara itu, Naive Bayes meskipun sedikit kurang akurat, tetap memberikan wawasan berguna dalam memprediksi kemungkinan lonjakan kasus. K-Means Clustering efektif dalam mengelompokkan wilayah berdasarkan tingkat keparahan pandemi, meskipun kurang optimal dalam menangani outlier. DBSCAN sangat berguna dalam mendeteksi anomali atau pola kasus yang tidak biasa, seperti wilayah dengan lonjakan kasus ekstrem.

Analisis asosiasi dengan menggunakan metode Apriori mengungkapkan bahwa kebijakan lockdown memiliki hubungan yang signifikan dengan penurunan jumlah kasus, meskipun dampaknya bervariasi bergantung pada kepatuhan masyarakat dan ketersediaan fasilitas kesehatan. Faktor sosial-ekonomi, seperti akses terhadap layanan kesehatan dan tingkat vaksinasi, juga mempengaruhi tingkat pemulihan dan angka kematian akibat COVID-19.

Evaluasi model menggunakan metrik seperti akurasi, precision, recall, dan silhouette score menunjukkan bahwa kombinasi antara model supervised (seperti Decision Tree dan Naive Bayes) dan unsupervised (seperti K-Means dan DBSCAN) memberikan gambaran yang lebih lengkap dan holistik dalam memahami serta merespons pandemi.

Berdasarkan hasil analisis ini, disarankan untuk memprioritaskan wilayah dengan tingkat keparahan kasus yang tinggi melalui pemodelan clustering. Selain itu, model prediktif dapat membantu pemerintah dalam merencanakan pengalokasian sumber daya medis dan merumuskan kebijakan yang lebih tepat dan responsif. Analisis ini juga menunjukkan pentingnya evaluasi terhadap kebijakan pembatasan sosial agar dapat lebih efektif dalam mengurangi penyebaran virus.

Secara keseluruhan, penerapan data mining pada dataset COVID-19 telah memberikan wawasan yang berguna untuk merumuskan kebijakan yang lebih baik, memitigasi dampak sosial-ekonomi, serta mempercepat respons terhadap perubahan situasi pandemi. Pendekatan berbasis data ini terbukti sangat penting dalam mengelola pandemi dan mempersiapkan strategi jangka panjang untuk menghadapinya.

DAFTAR PUSTAKA

1. M. Ciotti, M. Ciccozzi, A. Terrinoni, W.-C. Jiang, C.-B. Wang, and S. Bernardini, "The COVID-19 Pandemic," *Journal of Clinical Medicine*, vol. 9, no. 6, p. 1703, Jun. 2020.
2. S. Baloch, M. A. Baloch, T. Zheng, and X. Pei, "The Coronavirus Disease 2019 (COVID-19) Pandemic," *Journal of Clinical Virology*, vol. 127, p. 104334, 2020.
3. A. Haleem, M. Javaid, and R. Vaishya, "Effects of COVID-19 Pandemic in Daily Life," *Journal of Industrial Academy of Medicine*, vol. 29, no. 1, pp. 28–32, 2020.
4. S. Roy, "Economic Impact of COVID-19 Pandemic," *Journal of Business Research*, vol. 131, pp. 56–61, 2021.
5. T. T. Le, Z. Andreadakis, A. Kumar, R. Gómez Román, S. Tollefsen, M. Saville, and S. Mayhew, "The COVID-19 Vaccine Development Landscape," *Nature Reviews Drug Discovery*, vol. 19, no. 5, pp. 305–306, May 2020.
6. J. O. Woolliscroft, "Innovation in Response to the COVID-19 Pandemic Crisis," *Academic Medicine*, vol. 95, no. 8, pp. 1135–1138, Aug. 2020.