

**LAPORAN TUGAS BESAR PENAMBANGAN DATA  
SEGMENTASI DAN PREDIKSI PENDUDUK JAWA BARAT**



**Disusun oleh:**

Kelompok DREW SI-47-03

**Anggota:**

Daniel Indra Kusuma 102022300337

Naufal Rafi 102022300279

Eldryan Azri 102022300138

Wildan Zaaqi 102022300154

**FAKULTAS REKAYASA INDUSTRI**

**SISTEM INFORMASI**

**TELKOM UNIVERSITY**

**BANDUNG**

**2025**

## **Kata Pengantar**

Puji dan syukur penulis panjatkan ke hadirat Tuhan Yang Maha Esa karena atas rahmat dan karunia-Nya, laporan tugas besar dengan judul “Segmentasi dan Prediksi Penduduk Jawa Barat” ini dapat diselesaikan dengan baik. Laporan ini disusun sebagai salah satu syarat dalam memenuhi tugas akhir mata kuliah Penambangan Data pada Program Studi Sistem Informasi, Fakultas Rekayasa Industri, Telkom University.

Dalam penyusunan laporan ini, kelompok kami mengangkat studi kasus persebaran penduduk di Provinsi Jawa Barat dengan menerapkan metode spasial clustering dan regresi spasial. Kami berharap analisis yang kami lakukan dapat memberikan gambaran yang komprehensif mengenai kondisi demografis wilayah serta memberikan kontribusi bagi pengambilan keputusan yang berbasis data.

Kami menyadari bahwa laporan ini masih jauh dari sempurna. Oleh karena itu, kami terbuka terhadap kritik dan saran yang membangun guna perbaikan di masa mendatang. Terima kasih kami ucapan kepada dosen pengampu mata kuliah, Bapak/Ibu [Nama Dosen], serta semua pihak yang telah memberikan bimbingan, data, dan dukungan selama proses penyusunan laporan ini.

Akhir kata, semoga laporan ini dapat memberikan manfaat bagi mahasiswa, akademisi, masyarakat, maupun instansi pemerintah yang berkepentingan terhadap analisis dan pengelolaan data kependudukan.

## **Daftar Isi**

Kata Pengantar	2
Daftar Isi	3
<b>Daftar Tabel</b>	<b>4</b>
<b>Daftar Gambar</b>	<b>5</b>
<b>Ringkasan</b>	<b>6</b>
<b>Bab I Pendahuluan</b>	<b>7</b>
1.1 Latar Belakang	7
1.2 Tujuan	7
1.3 Manfaat	7
<b>Bab II Landasan Teori/ Tinjauan Pustaka</b>	<b>8</b>
2.1 Data Mining	8
2.2 Clustering	8
2.3 Regresi Spasial	9
2.4 Sistem Informasi Geografis (SIG)	9
2.5 Visualisasi Data dan Dashboard	9
2.6 Studi Terdahulu	10
<b>Bab III Metodologi Penelitian</b>	<b>10</b>
3.1 Jenis penelitian dan dataset yang digunakan	10
3.2 Sistematika penyelesaian permasalahan	10
3.3 Detail data exploration	10
3.4 Detail data preparation	10
3.5 Detail modelling dan evaluation	11
3.6 Detail perancangan dashboard	11
<b>Bab IV Hasil dan Pembahasan</b>	<b>11</b>

## **Daftar Tabel**

Tabel Bobot Pengerjaan	25
Tabel Pembagian Tugas	25

## **Daftar Gambar**

Gambar 4. 1 Data Exploration	13
Gambar 4. 2 Data Preparation 1	13
Gambar 4. 3 Data Preparation 2	14
Gambar 4. 4 Modelling dan evaluasi spasial regresi	14
Gambar 4.5 modelling dan evaluasi spasial regresi 2	15
Gambar 4. 6 Modelling dan evaluasi spasial regresi 3	15
Gambar 4. 7 Modelling dan evaluasi spasial klastering	16
Gambar 4. 8 Hasil evaluasi spasial regresi	16
Gambar 4. 9 Deployment Diagram 1	16
Gambar 4. 10 Deployment Diagram 2	17
Gambar 4. 11 Deployment Diagram 3	17
Gambar 4. 12 Deployment Diagram 4	18
Gambar 4. 13 Deployment Diagram 5	18
Gambar 4. 14 Deployment Diagram 6	19
Gambar 4. 15 Hasil Interface 1	19
Gambar 4. 16 Hasil interface 2	20
Gambar 4. 17 Hasil interface 3	20
Gambar 4. 18 Hasil interface 4	21
Gambar 4. 19 Hasil interface 5	21
Gambar 4. 20 Hasil interface 6	22
Gambar 4. 21 Hasil interface 7	22

## **Ringkasan**

Laporan tugas besar ini membahas penerapan teknik data mining dalam menganalisis pola persebaran penduduk di Provinsi Jawa Barat. Studi ini menggunakan metode **spasial clustering** dan **regresi spasial** untuk mengelompokkan wilayah berdasarkan kepadatan penduduk serta memprediksi jumlah penduduk di masa mendatang berdasarkan data historis dan lokasi geografis.

Data yang digunakan diperoleh dari Badan Pusat Statistik (BPS) untuk periode tahun 2018 hingga 2020, mencakup jumlah penduduk, luas wilayah, dan koordinat kabupaten/kota di Jawa Barat. Proses analisis dimulai dari pengumpulan dan pembersihan data, perhitungan kepadatan penduduk, penerapan algoritma clustering seperti K-Means dan DBSCAN, hingga pembangunan model regresi spasial menggunakan metode Geographically Weighted Regression (GWR).

Hasil analisis divisualisasikan dalam bentuk **peta klaster** dan **dashboard interaktif**, serta dikembangkan juga **API prediksi** yang memungkinkan estimasi jumlah penduduk pada wilayah baru. Penelitian ini diharapkan dapat memberikan insight berbasis data untuk mendukung pengambilan kebijakan oleh pemerintah daerah serta meningkatkan pemahaman mahasiswa terhadap implementasi data mining dalam konteks geografis.

## **Bab I Pendahuluan**

### **1.1 Latar Belakang**

Provinsi Jawa Barat merupakan salah satu wilayah dengan jumlah penduduk terbesar di Indonesia. Tingginya angka populasi yang tersebar di berbagai kabupaten dan kota menjadikan Jawa Barat sebagai daerah strategis dalam perencanaan pembangunan, distribusi layanan publik, dan kebijakan sosial-ekonomi. Namun, persebaran penduduk yang tidak merata menimbulkan tantangan dalam aspek pemerataan pembangunan, ketersediaan infrastruktur, dan pengelolaan sumber daya.

Dengan memanfaatkan metode *data mining*, analisis terhadap pola persebaran penduduk dapat dilakukan secara lebih mendalam dan berbasis data. Penerapan teknik ini dapat membantu pengambil keputusan dalam mengidentifikasi konsentrasi penduduk, tren migrasi, serta potensi wilayah yang perlu mendapatkan perhatian lebih. Oleh karena itu, dibutuhkan sebuah model data mining yang dapat digunakan untuk mengeksplorasi dan mengekstraksi informasi tersembunyi dari data persebaran penduduk di wilayah Jawa Barat.

### **1.2 Tujuan**

Tujuan dari studi ini adalah untuk mengembangkan model data mining yang mampu menganalisis dan memetakan pola persebaran penduduk di Provinsi Jawa Barat berdasarkan data kependudukan yang tersedia. Model ini diharapkan dapat memberikan insight yang dapat digunakan untuk mendukung pengambilan keputusan oleh pemerintah daerah atau pihak-pihak terkait dalam perencanaan kebijakan.

### **1.3 Manfaat**

Adapun manfaat yang diharapkan dari pelaksanaan studi kasus ini antara lain:

- Memberikan gambaran visual dan statistik terkait konsentrasi penduduk di setiap wilayah di Jawa Barat.
- Menyediakan informasi berbasis data untuk mendukung perencanaan pembangunan daerah dan pengalokasian sumber daya secara efektif.
- Menjadi acuan awal untuk riset lanjutan dalam bidang data mining dan sistem informasi geografis (SIG).
- Meningkatkan pemahaman dan keterampilan mahasiswa dalam menerapkan metode data mining pada permasalahan nyata.

## **1.4 Batasan Penelitian**

Penelitian ini memiliki beberapa batasan sebagai berikut:

1. Ruang lingkup wilayah dibatasi hanya pada kabupaten/kota di Provinsi Jawa Barat.
2. Periode data yang digunakan terbatas pada tahun 2018 hingga 2020, sesuai dengan ketersediaan data jumlah penduduk dan luas wilayah dari BPS.
3. Variabel yang digunakan dalam model terbatas pada jumlah penduduk, luas wilayah, dan koordinat geografis. Faktor lain seperti ekonomi, migrasi, atau tingkat urbanisasi tidak dimasukkan ke dalam model.
4. Metodologi yang digunakan hanya mencakup spasial clustering dan regresi spasial, tanpa melibatkan teknik prediktif lain seperti deep learning atau time series forecasting.
5. Implementasi model hanya difokuskan pada penyajian peta klaster dan API sederhana, tidak mencakup pengembangan sistem informasi berbasis web secara utuh.

## Bab II Landasan Teori/ Tinjauan Pustaka

### 2.1 Data Mining

Data mining adalah proses untuk menemukan pola tersembunyi, korelasi, tren, dan informasi berguna lainnya dari kumpulan data besar dengan menggunakan teknik statistik, matematika, dan pembelajaran mesin. Dalam konteks persebaran penduduk, data mining memungkinkan analisis mendalam terhadap karakteristik dan distribusi data kependudukan sehingga mendukung pengambilan keputusan berbasis data.

Menurut Han dan Kamber (2011), proses data mining mencakup tahapan pembersihan data, integrasi data, seleksi data, transformasi data, penambangan pola, evaluasi pola, dan penyajian informasi. Proses ini sangat relevan untuk studi ini karena mampu mengelola volume data besar dari BPS dan mengungkap klaster wilayah berdasarkan kepadatan penduduk.

### 2.2 Clustering

Clustering merupakan salah satu teknik dalam data mining yang digunakan untuk mengelompokkan data ke dalam grup atau klaster berdasarkan kemiripan karakteristik. Dalam studi ini, clustering digunakan untuk segmentasi wilayah berdasarkan kepadatan penduduk.

Algoritma yang umum digunakan dalam clustering meliputi:

- **K-Means:** algoritma populer untuk membagi data menjadi  $k$  kelompok berdasarkan jarak centroid.
- **DBSCAN:** cocok untuk menemukan klaster dengan kepadatan yang berbeda-beda dan mengabaikan outlier.
- **Hierarchical Clustering:** menghasilkan struktur pohon hirarki dari klaster yang dapat divisualisasikan sebagai *dendrogram*.

Teknik ini berguna dalam konteks pemetaan wilayah dengan konsentrasi penduduk tinggi, sedang, dan rendah di Jawa Barat.

### 2.3 Regresi Spasial

Regresi spasial adalah metode analisis statistik yang mempertimbangkan hubungan antara variabel dependen dan variabel independen dengan memperhitungkan aspek spasial atau lokasi geografis. Teknik ini sangat penting dalam analisis persebaran penduduk karena memperhitungkan kedekatan geografis antar wilayah.

Model regresi spasial yang umum digunakan antara lain:

- **Spatial Lag Model (SLM)**: mempertimbangkan nilai variabel dependen dari lokasi sekitar sebagai faktor prediksi.
- **Spatial Error Model (SEM)**: memperhitungkan kesalahan yang terstruktur secara spasial.
- **Geographically Weighted Regression (GWR)**: melakukan regresi lokal di tiap titik geografis, berguna untuk menangkap heterogenitas spasial.

Dengan menggunakan regresi spasial, penelitian ini mampu membangun API prediksi jumlah penduduk berdasarkan lokasi dan tren historis.

## 2.4 Sistem Informasi Geografis (SIG)

SIG adalah sistem berbasis komputer yang digunakan untuk mengelola, menganalisis, dan memvisualisasikan data yang memiliki referensi geografis. Dalam penelitian ini, SIG digunakan untuk memetakan hasil klasterisasi dan prediksi jumlah penduduk di wilayah Jawa Barat.

Penggunaan SIG memungkinkan visualisasi data dalam bentuk peta tematik yang intuitif, sehingga hasil analisis data mining dapat dengan mudah dipahami dan digunakan oleh pemangku kebijakan.

## 2.5 Visualisasi Data dan Dashboard

Visualisasi data merupakan aspek penting dalam menyampaikan informasi hasil analisis secara efektif. Salah satu cara penyajiannya adalah melalui dashboard interaktif. Dashboard memungkinkan pengguna untuk mengeksplorasi data, melihat tren historis, serta memantau persebaran penduduk secara real-time berdasarkan output dari model clustering dan regresi spasial.

Teknologi dashboard modern seperti Tableau, Power BI, atau visualisasi berbasis web (misalnya dengan D3.js atau Leaflet untuk peta) sangat membantu dalam mendukung transparansi dan penyajian insight berbasis data.

## 2.6 Studi Terdahulu

Beberapa penelitian sebelumnya yang relevan sebagai referensi antara lain:

- “Clustering Analysis for Population Density Using K-Means Algorithm in GIS” oleh Zhang et al. (2019) menunjukkan bahwa kombinasi K-Means dan GIS mampu memetakan konsentrasi penduduk dengan akurasi tinggi.
- “Application of Spatial Regression Models to Analyze Urban Population Growth” oleh Lee et al. (2021) membuktikan efektivitas GWR dalam menangkap variasi spasial dalam pertumbuhan penduduk.

Penelitian-penelitian tersebut memperkuat bahwa pendekatan yang digunakan dalam studi ini relevan dan telah terbukti efektif dalam konteks serupa.

## Bab III Metodologi Penelitian

### 3.1 Jenis penelitian dan dataset yang digunakan

Penelitian ini termasuk ke penelitian kuantitatif, karena melibatkan data numerik (jumlah penduduk, luas wilayah, kepadatan) yang diolah dengan teknik statistik dan algoritma komputasional.

Dataset yang digunakan dalam studi kasus ini terdiri dari dua sumber utama:

- **Jumlah Penduduk Jawa Barat (2018–2020)**: diambil dari publikasi Badan Pusat Statistik (BPS) tingkat provinsi dan kabupaten/kota.
- **Luas Wilayah per Kabupaten/Kota**: data tahun 2020 dari BPS Jawa Barat.

### 3.2 Sistematika penyelesaian permasalahan

Berikut tahapan metode yang digunakan:

1. **Pengumpulan dan Normalisasi Data**
  - Data dari BPS disatukan dalam satu database.
  - Normalisasi dilakukan untuk menyamakan skala data.
2. **Perhitungan Kepadatan Penduduk**
  - Ditambahkan kolom kepadatan sebagai fitur utama untuk clustering.
3. **Spasial Clustering**
  - Menggunakan algoritma seperti DBSCAN, K-Means, atau aglomeratif clustering yang mempertimbangkan jarak spasial dan nilai kepadatan.
  - Hasil berupa klaster wilayah padat, sedang, dan kurang padat.
4. **Regresi Spasial**
  - Menggunakan teknik SAR (Spatial Autoregressive) atau GWR (Geographically Weighted Regression) untuk memodelkan hubungan antara kepadatan dengan luas wilayah dan distribusi populasi sekitar.
5. **Visualisasi Peta Klaster**
  - Menggunakan GeoPandas atau Folium untuk menampilkan hasil klaster pada peta Jawa Barat.
6. **Pengembangan API Prediksi**
  - API dibuat untuk menginput data baru dan mendapatkan prediksi kepadatan berdasarkan model regresi

### 3.3 Detail data exploration

Beberapa eksplorasi data yang dilakukan mencakup:

- Distribusi Jumlah Penduduk per Kabupaten/Kota
- Distribusi Luas Wilayah
- Perhitungan Kepadatan Penduduk

- Visualisasi Geografis menggunakan scatter map atau heatmap

### 3.4 Detail data preparation

Tahapan ini meliputi :

- **Pembersihan data:** Menghapus kolom yang tidak dipakai serta menghapus data yang kosong
- **Transformasi data:** Menambahkan atribut baru seperti kepadatan penduduk dengan cara membagi data jumlah penduduk dengan luas wilayahnya
- **Normalisasi:** Agar algoritma clustering tidak berat sebelah terhadap atribut dengan skala besar.
- **Penggabungan Data:** Menggabungkan file penduduk, luas wilayah, dan koordinat ke dalam satu dataset analisis.

### 3.5 Detail modelling dan evaluation

#### Clustering

- Algoritma: K-Means (dengan parameter k optimal) dan DBSCAN.
- Input: Atribut koordinat dan kepadatan penduduk.

Evaluasi:

- K-Means: Silhouette Score, Davies-Bouldin Index
- DBSCAN: Visualisasi klaster spasial

#### Regresi Spasial

- Algoritma: Geographically Weighted Regression (GWR)
  - Input: Lokasi (latitude, longitude), luas wilayah
  - Target: Jumlah penduduk
- Evaluasi: R<sup>2</sup> Score, RMSE

### 3.6 Detail perancangan dashboard

Dashboard dirancang untuk menampilkan:

- **Peta Klaster** (hasil clustering) menggunakan Leaflet atau Python Folium.
- **Grafik Tren Historis** jumlah penduduk.
- **Peta Prediksi** berdasarkan model regresi spasial.
- **Fitur Interaktif:** Filter wilayah, slider tahun, dan tooltip info wilayah

## Bab IV Hasil dan Pembahasan

Berisi tentang:

1. Hasil dari Bab III mulai dari data exploration sampai *dashboard*

### Data Exploration

The screenshot shows a Jupyter Notebook interface with two code cells. The first cell imports pandas and reads a CSV file named 'Jumlah\_penduduk\_menujur\_kabupaten\_kota\_2018\_2020.csv'. The second cell reads a CSV file named 'Luas\_daerah\_menujur\_kabupaten\_kota\_jawabarat\_2020.csv'. Both cells output tables.

```
import pandas as pd
[49]    ✓ 0.0s
[50] DATA JUMLAH PENDUDUK
df = pd.read_csv('Jumlah_penduduk_menujur_kabupaten_kota_2018_2020.csv')
df.head()
[50]    ✓ 0.0s
[51] DATA LUAS WILAYAH
dd_2 = pd.read_csv('Luas_daerah_menujur_kabupaten_kota_jawabarat_2020.csv')
dd_2.head()
```

Gambar 4. 1 Data Exploration

### Data Preparation

The screenshot shows a Jupyter Notebook interface with two code cells. The first cell drops the 'Jumlah Pulau' column from the 'dd\_2' DataFrame and prints its info. The second cell merges 'dd\_2' with 'df' on 'Wilayah Jawa Barat' and 'Kabupaten/Kota', then drops the 'Kabupaten/Kota' column. Both cells output tables.

```
dd_2.drop(['Jumlah Pulau'], axis=1, inplace=True)
dd_2.info()
[52]    ✓ 0.0s
[53] GABUNGIN DATA PENDUDUK
Cleaning Data Luas Daerah
df_final = df.merge(dd_2, left_on="Wilayah Jawa Barat", right_on="Kabupaten/Kota")
df_final.drop(columns=["Kabupaten/Kota"], inplace=True)
df_final.head()
```

Gambar 4. 2 Data Preparation 1

```

df_final = df.merge(dd_2, left_on="Wilayah Jawa Barat", right_on="Kabupaten/Kota")
df_final.drop(columns=["Kabupaten/Kota"], inplace=True)
df_final.head()

```

	Wilayah Jawa Barat	2018	2019	2020	Ibu Kota Wilayah	Luas Wilayah (Km2)	Percentase Terhadap Luas Wilayah
0	Bogor	5840907	5965410	6088233	Cibinong	2710.62	7.66
1	Sukabumi	2460693	2466272	2470219	Pelabuhan Ratu	4145.70	11.72
2	Cianjur	2260620	2263072	2264328	Cianjur	3840.16	10.85
3	Bandung	3717291	3775279	3831505	Soreang	1767.96	5.00
4	Garut	2606399	2622425	2636637	Torogong Kidul	3074.07	8.69

*Gambar 4. 3 Data Preparation 2*

### *Modelling dan evaluasi spasial regresi*

```

kepadatan_tahunan = {}

for tahun in range(2018, 2021):
    kolom_kepadatan = f'Kepadatan_{tahun}'
    df_final[kolom_kepadatan] = df_final[str(tahun)] / df_final["Luas Wilayah (Km2)"]
    kepadatan_tahunan[tahun] = df_final[["Wilayah Jawa Barat", kolom_kepadatan]]

```

	Wilayah Jawa Barat	Luas Wilayah (Km2)	2018	2019	2020	Ibu Kota Wilayah	Percentase Terhadap Luas Wilayah	Kepadatan_2018	Kepadatan_2019	Kepadatan_2020
0	Bogor	2710.62	5840907	5965410	6088233	Cibinong	7.66	2154.823251	2200.754809	2246.066583
1	Sukabumi	4145.70	2460693	2466272	2470219	Pelabuhan Ratu	11.72	593.553079	594.898811	595.850882
2	Cianjur	3840.16	2260620	2263072	2264328	Cianjur	10.85	588.678597	589.317112	589.644181
3	Bandung	1767.96	3717291	3775279	3831505	Soreang	5.00	2102.587728	2133.387113	2167.189869
4	Garut	3074.07	2606399	2622425	2636637	Torogong Kidul	8.69	847.865859	853.079143	857.702329

*Gambar 4. 4 Modelling dan evaluasi spasial regresi*

The screenshot shows a Jupyter Notebook interface within VS Code. The code cell contains Python code for splitting a dataset and evaluating a K-Nearest Neighbors regression model. The output shows the execution time for each step and the final evaluation metrics: MAE, MSE, and RMSE.

```
import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split, cross_val_predict, KFold
from sklearn.metrics import mean_squared_error, mean_absolute_error
from sklearn.neighbors import KNeighborsRegressor
import matplotlib.pyplot as plt

X = df_final.drop(columns=["Wilayah Jawa Barat", "Kepadatan_2020", "Ibu Kota Wilayah", "Persentase Terhadap Luas Wilayah"])
y = df_final["Kepadatan_2020"]

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

knn_model = KNeighborsRegressor(n_neighbors=5)
cv = KFold(n_splits=10, shuffle=True, random_state=42)
y_pred_cv = cross_val_predict(knn_model, X, y, cv=cv)

mse = mean_squared_error(y, y_pred_cv)
mae = mean_absolute_error(y, y_pred_cv)
rmse = np.sqrt(mse)

print(f"MAE: {mae:.2f}")
print(f"MSE: {mse:.2f}")
print(f"RMSE: {rmse:.2f}")

[60] ✓ 0.0s
```

Gambar 4.5 modelling dan evaluasi spasial regresi 2

This screenshot is identical to the one above, showing the same Jupyter Notebook code and its execution results. The output cell shows the calculated values for MAE, MSE, and RMSE.

```
MAE: 4406.17
MSE: 39726319.68
RMSE: 5545.13
```

Gambar 4.6 Modelling dan evaluasi spasial regresi 3

## Modelling dan evaluasi spasial klastering

The screenshot shows a Jupyter Notebook interface with several cells of Python code. The code performs data cleaning, scales the data using StandardScaler, and then applies K-Means clustering with 5 clusters. It also calculates silhouette scores and Davies-Bouldin Index. The output cells show the execution time (0.0s) and the resulting evaluation metrics.

```
import pandas as pd
import numpy as np
from sklearn.cluster import KMeans, DBSCAN
from sklearn.metrics import silhouette_score, davies_bouldin_score
from sklearn.preprocessing import StandardScaler
import matplotlib.pyplot as plt

X = df_final.drop(columns=["Wilayah Jawa Barat", "Ibu Kota Wilayah", "Percentase Terhadap Luas Wilayah", "Kepadatan_2020"])

scaler = StandardScaler()
X_scaled = scaler.fit_transform(X)

k = 5
kmeans = KMeans(n_clusters=k, random_state=42)
kmeans_labels = kmeans.fit_predict(X_scaled)

sil_score = silhouette_score(X_scaled, kmeans_labels)
db_score = davies_bouldin_score(X_scaled, kmeans_labels)

print("== K-Means Evaluation ==")
print(f"Silhouette Score: {sil_score:.3f}")
print(f"Davies-Bouldin Index: {db_score:.3f}")
```

Gambar 4. 7 Modelling dan evaluasi spasial klastering

A terminal window showing the output of the K-Means evaluation code. It displays the Silhouette Score (0.422), Davies-Bouldin Index (0.654), and a message indicating the auth token was saved to the configuration file.

```
== K-Means Evaluation ==
Silhouette Score: 0.422
Davies-Bouldin Index: 0.654
Authtoken saved to configuration file: C:\Users\WILDAN\AppData\Local\ngrrok\ngrok.yaml
```

Gambar 4. 8 Hasil evaluasi spasial regresi

## Deployment Dashboard

The screenshot shows a Jupyter Notebook interface with Python code for deploying a Streamlit application. The code uses joblib to load a model and streamlit to create a dashboard. The Streamlit app is titled "Dashboard Prediksi Kepadatan Penduduk". The notebook also includes code to read a CSV dataset.

```
import joblib
joblib.dump(knn_model, 'modelprediksi.pkl')

import streamlit as st
import pandas as pd
import numpy as np
import joblib
from sklearn.linear_model import LinearRegression
import matplotlib.pyplot as plt

st.set_page_config(
    page_title="Dashboard Prediksi Kepadatan Penduduk",
)

# Judul
st.title("Prediksi Kepadatan Penduduk Jawa Barat")
st.write("Model ini memprediksi kepadatan penduduk untuk tahun berikutnya berdasarkan data sebelumnya dan luas wilayah.")

model = joblib.load('modelprediksi.pkl')

# Load Dataset
df = pd.read_csv("Jumlah_penduduk_menurut_kabupaten_kota_2018_2020.csv")
```

Gambar 4. 9 Deployment Diagram 1

```

File Edit Selection View Go Run Terminal Help < - > TUBES DATMIN
OPEN EDITORS TUBES_DATMIN.ipynb
TUBES DATMIN
devcontainer.json
app.py
Jumlah_penduduk_menurut_kab...
Luas_dasar_menurut_kabupaten...
modelprediksi.pkl
requirements.txt
TUBES_DATMIN.ipynb

# Preprocessing
df_2.drop(["Jumlah Pulau"], axis=1, inplace=True)
df_final = df.merge("Wilayah Jawa Barat", right_on="Kabupaten/Kota")
df_final.drop(columns="Kabupaten/Kota", inplace=True)

df_final["Kepadatan_2018"] = df_final["2018"] / df_final["Luas Wilayah (Km2)"]
df_final["Kepadatan_2019"] = df_final["2019"] / df_final["Luas Wilayah (Km2)"]
df_final["Kepadatan_2020"] = df_final["2020"] / df_final["Luas Wilayah (Km2)"]

# Fitting model untuk demo
X = df_final[["Kepadatan_2018", "Kepadatan_2019"]]
y = df_final["Kepadatan_2020"]
model = LinearRegression()
model.fit(X, y)

st.subheader("Visualisasi Kepadatan Penduduk per Tahun (Bar Chart)")

# Pilih tahun (2018 atau 2019)
tahun_pilihan = st.selectbox("Pilih Tahun", [2018, 2019], key="tahun_bar_chart")

fig, ax = plt.subplots(figsize=(10, 5))
ax.bar(df_final["Wilayah Jawa Barat"], df_final["Kepadatan_[tahun_pilihan]"])
ax.set_title("Kepadatan Penduduk Tahun {tahun_pilihan}")
ax.set_xlabel("Wilayah Jawa Barat")
ax.set_xticks(np.arange(len(df_final["Wilayah Jawa Barat"])))
ax.set_xticklabels(df_final["Wilayah Jawa Barat"], rotation=90)
ax.grid(True)
st.pyplot(fig)

st.pyplot(fig)

st.subheader("Tabel Kategori Kepadatan Penduduk")

data_kategori = {
    "Rantang Kepadatan (jiwa/km²)": [
        "0-1.000",
        "1.000 < 5.000",
        "5.000 < 10.000",
        "10.000 < 15.000",
        "≥ 15.000"
    ],
    "Kategori": [
        "Sangat Rendah",
        "Rendah",
        "Sedang",
        "Tinggi",
        "Sangat Tinggi"
    ]
}

df_kategori = pd.DataFrame(data_kategori)
st.table(df_kategori)

```

Gambar 4. 10 Deployment Diagram 2

```

File Edit Selection View Go Run Terminal Help < - > TUBES DATMIN
OPEN EDITORS TUBES_DATMIN.ipynb
TUBES DATMIN
devcontainer.json
app.py
Jumlah_penduduk_menurut_kab...
Luas_dasar_menurut_kabupaten...
modelprediksi.pkl
requirements.txt
TUBES_DATMIN.ipynb

# Preprocessing
df_2.drop(["Jumlah Pulau"], axis=1, inplace=True)
df_final = df.merge("Wilayah Jawa Barat", right_on="Kabupaten/Kota")
df_final.drop(columns="Kabupaten/Kota", inplace=True)

df_final["ax_grid(True)"]
st.pyplot(fig)

st.pyplot(fig)

st.subheader("Tabel Kategori Kepadatan Penduduk")

data_kategori = {
    "Rantang Kepadatan (jiwa/km²)": [
        "0-1.000",
        "1.000 < 5.000",
        "5.000 < 10.000",
        "10.000 < 15.000",
        "≥ 15.000"
    ],
    "Kategori": [
        "Sangat Rendah",
        "Rendah",
        "Sedang",
        "Tinggi",
        "Sangat Tinggi"
    ]
}

df_kategori = pd.DataFrame(data_kategori)
st.table(df_kategori)

# Input dari user
st.subheader("Input Data Tahun Sebelumnya")
kepadatan_2018 = st.number_input("Kepadatan 2018", min_value=0.0, step=10.0)
kepadatan_2019 = st.number_input("Kepadatan 2019", min_value=0.0, step=10.0)

if st.button("Prediksi Kepadatan 2020"):
    pred = model.predict([[kepadatan_2018, kepadatan_2019]])

```

Gambar 4. 11 Deployment Diagram 3

The screenshot shows a Jupyter Notebook interface with a dark theme. The left sidebar lists files: 'TUBES\_DATMIN.ipynb' (the active file), 'devcontainer.json', 'app.py', 'Jumlah\_penduduk\_menurut\_kab...', 'Luas\_derauh\_menurut\_kabupaten...', 'modelprediksi.pkl', and 'requirements.txt'. The main area contains Python code for data processing and prediction:

```
# Tabel Kepadatan Pulau
df.drop(["Jumlah Pulau"], axis=1, inplace=True)
st.subheader("Input Data Tahun Sebelumnya")
kepadatan_2018 = st.number_input("Kepadatan 2018", min_value=0.0, step=10.0)
kepadatan_2019 = st.number_input("Kepadatan 2019", min_value=0.0, step=10.0)

if st.button("Prediksi Kepadatan 2020"):
    pred = model.predict([[kepadatan_2018, kepadatan_2019]])
    nilai_prediksi = pred[0]

    # Tentukan label kategori
    if nilai_prediksi < 1000:
        kategori = "Sangat Rendah"
    elif nilai_prediksi < 5000:
        kategori = "Rendah"
    elif nilai_prediksi < 10000:
        kategori = "Sedang"
    elif nilai_prediksi < 15000:
        kategori = "Tinggi"
    else:
        kategori = "Sangat Tinggi"

    st.success(f"Prediksi Kepadatan Tahun 2020: {nilai_prediksi:.2f} jiwa/km")
    st.info(f"Kategori Kepadatan: {kategori}")

# Tabel Prediksi vs Aktual
df_final["Prediksi_2020"] = model.predict(df_final[['Kepadatan_2018', 'Kepadatan_2019']])
df_final["Selisih_Error"] = df_final["Prediksi_2020"] - df_final["Kepadatan_2020"]

st.subheader("Tabel Prediksi vs Aktual")
st.dataframe(df_final[['Wilayah Jawa Barat', 'Kepadatan_2020', 'Prediksi_2020', 'Selisih_Error']].round(2))

# Bar Chart Perbandingan
st.subheader("Visualisasi Prediksi vs Aktual per Wilayah")
fig, ax = plt.subplots(figsize=(12, 5))
width = 0.4
X = np.arange(len(df_final["Wilayah Jawa Barat"]))

```

Gambar 4. 12 Deployment Diagram 4

The screenshot shows a Jupyter Notebook interface with a dark theme. The left sidebar lists files: 'TUBES\_DATMIN.ipynb' (the active file), 'devcontainer.json', 'app.py', 'Jumlah\_penduduk\_menurut\_kab...', 'Luas\_derauh\_menurut\_kabupaten...', 'modelprediksi.pkl', and 'requirements.txt'. The main area contains Python code for data processing, clustering, and visualization:

```
st.subheader("Tabel Prediksi vs Aktual")
st.dataframe(df_final[['Wilayah Jawa Barat', 'Kepadatan_2020', 'Prediksi_2020', 'Selisih_Error']].round(2))

# Bar Chart Perbandingan
st.subheader("Visualisasi Prediksi vs Aktual per Wilayah")
fig, ax = plt.subplots(figsize=(12, 5))
width = 0.4
X = np.arange(len(df_final["Wilayah Jawa Barat"]))
ax.bar(X - width/2, df_final["Kepadatan_2020"], width=width, label="Aktual")
ax.bar(X + width/2, df_final["Prediksi_2020"], width=width, label="Prediksi")
ax.set_xticks(X)
ax.set_xticklabels(df_final["Wilayah Jawa Barat"], rotation=90)
ax.set_ylabel("Kepadatan (jiwa/km)")
ax.legend()
st.pyplot(fig)

# Clustering Visualisasi
from sklearn.preprocessing import StandardScaler
from sklearn.cluster import KMeans
import seaborn as sns

st.subheader("Clustering Wilayah Berdasarkan Kepadatan per Tahun")

# Dropdown untuk memilih tahun
tahun_ktaster = st.selectbox("Pilih Tahun untuk Clustering", [2018, 2019, 2020], key="tahun_clustering")

# Lakukan clustering berdasarkan tahun yang dipilih
X_cluster_single = df_final[['Kepadatan_(tahun_ktaster)']]
X_scaled_single = StandardScaler().fit_transform(X_cluster_single)

kmeans_single = KMeans(n_clusters=5, random_state=42)
df_final["Cluster_Tahun"] = kmeans_single.fit_predict(X_scaled_single)

# Tampilkan hasil clustern per wilayah

```

Gambar 4. 13 Deployment Diagram 5

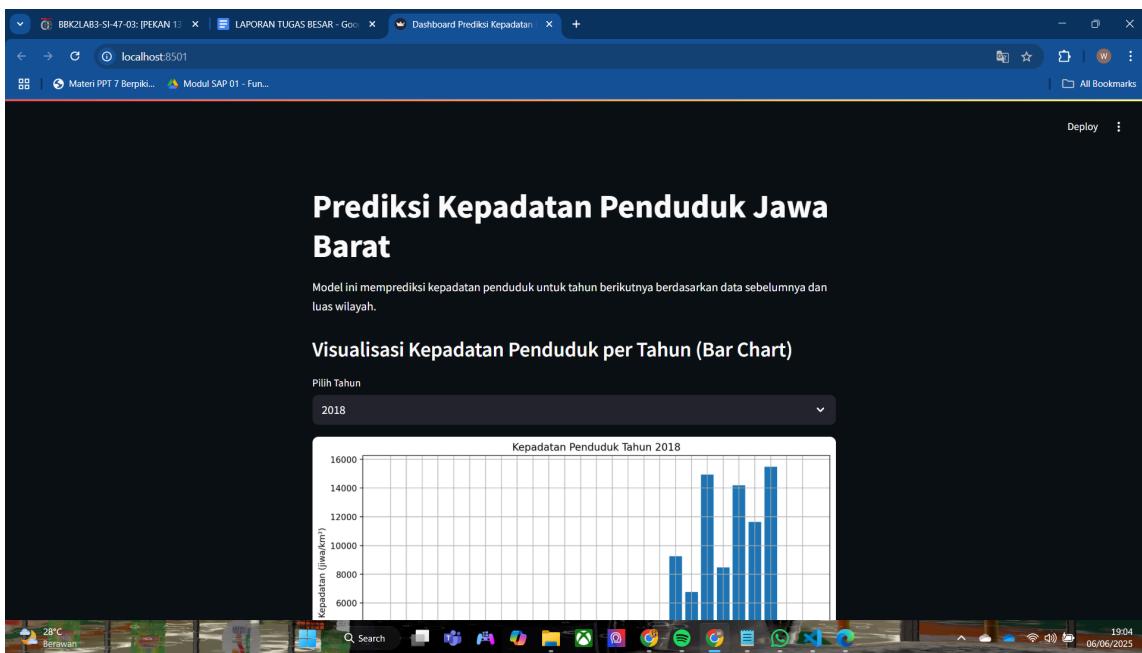
The screenshot shows a Jupyter Notebook interface with the file 'TUBES\_DATMIN.ipynb' open. The code in the cell performs several tasks:

- It drops the 'Jumlah\_Pulau' column from the DataFrame.
- It generates a bar chart titled 'Prediksi\_2020' showing population density ('Kepadatan') for various regions ('Wilayah').
- It performs K-Means clustering on the data, creating a new column 'Cluster\_Tahun'.
- It visualizes the distribution of clusters over time using a bar chart.

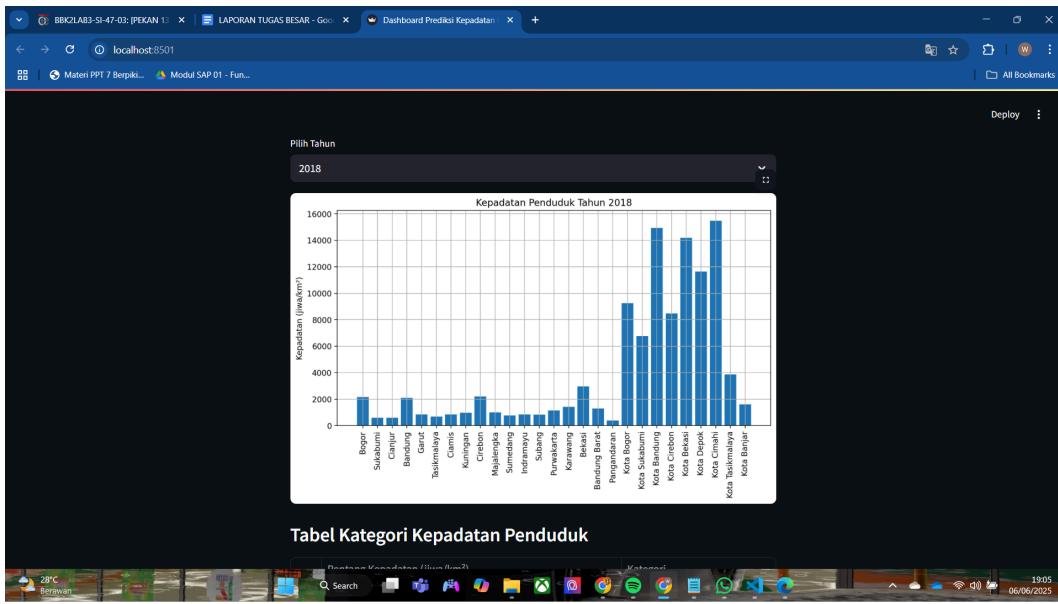
The Jupyter interface includes an Explorer sidebar with files like 'app.py', 'requirements.txt', and 'modelprediksi.pkl'. The bottom status bar shows the Python version (3.12.0) and the date (06/06/2025).

Gambar 4. 14 Deployment Diagram 6

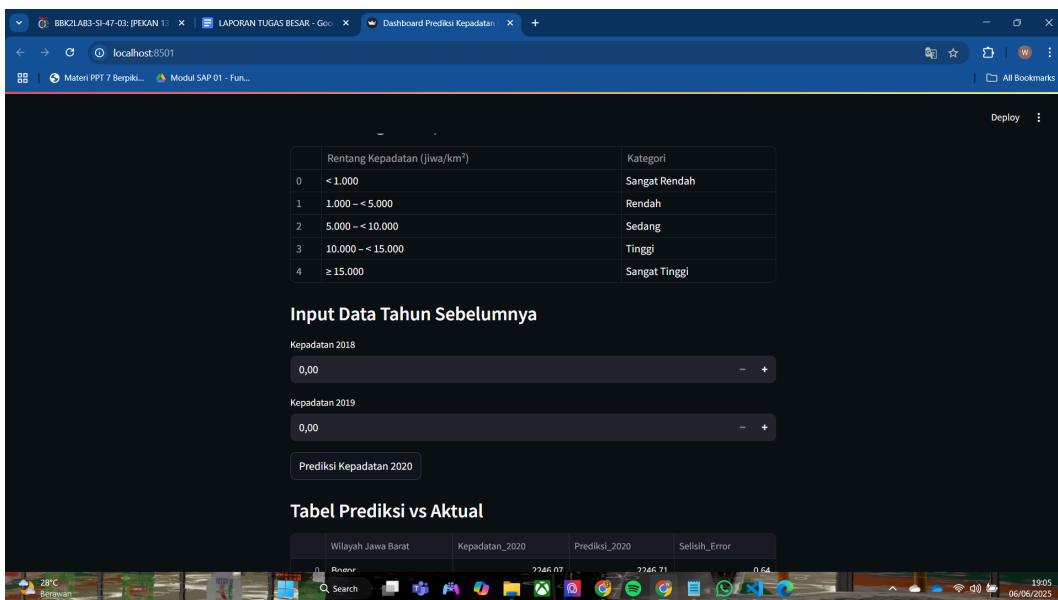
### Hasil Interface



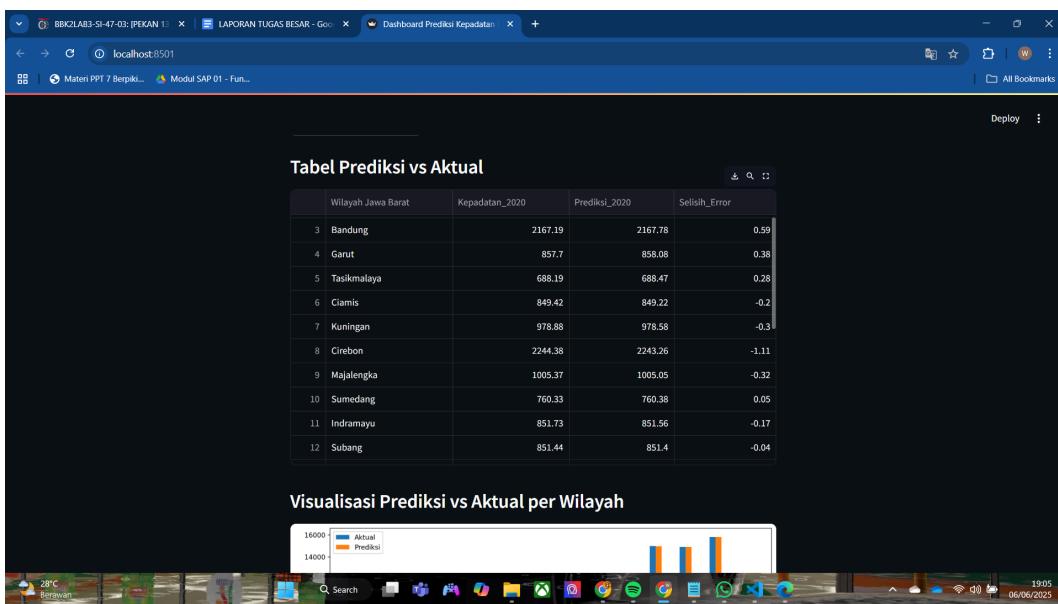
Gambar 4. 15 Hasil Interface 1



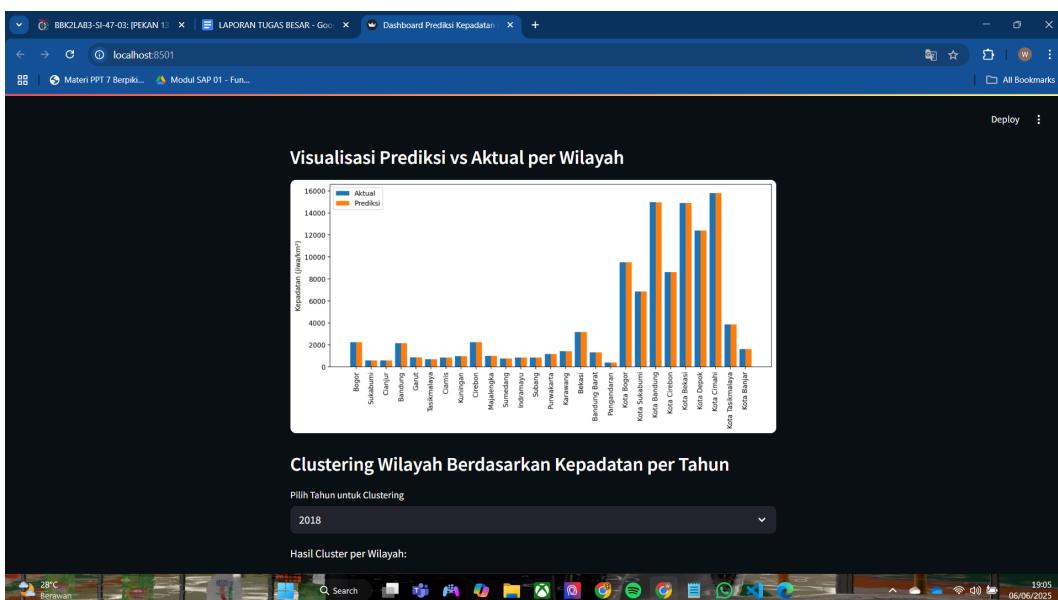
Gambar 4. 16 Hasil interface 2



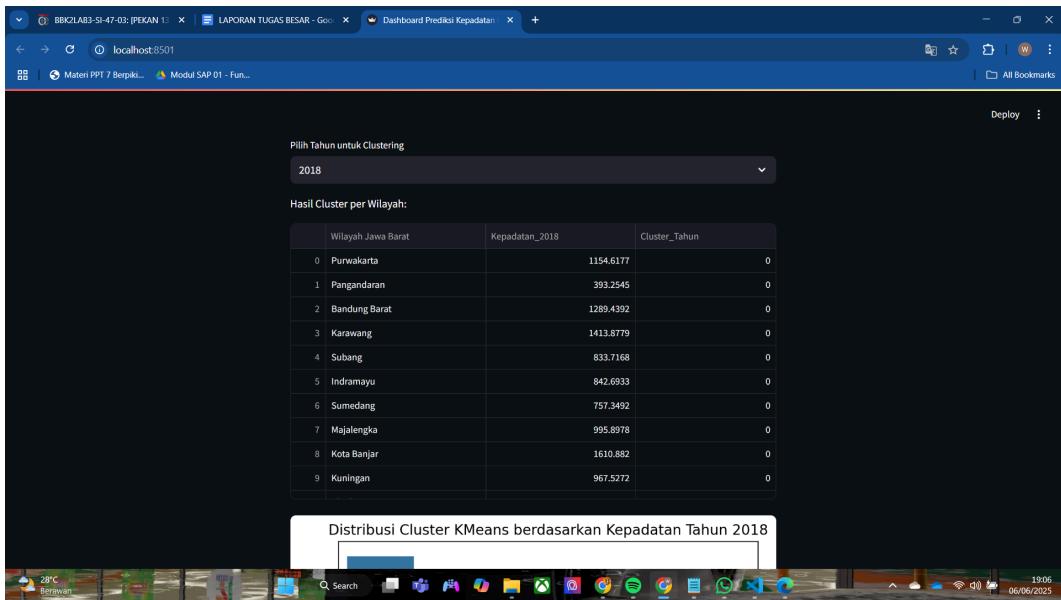
#### Gambar 4. 17 Hasil interface 3



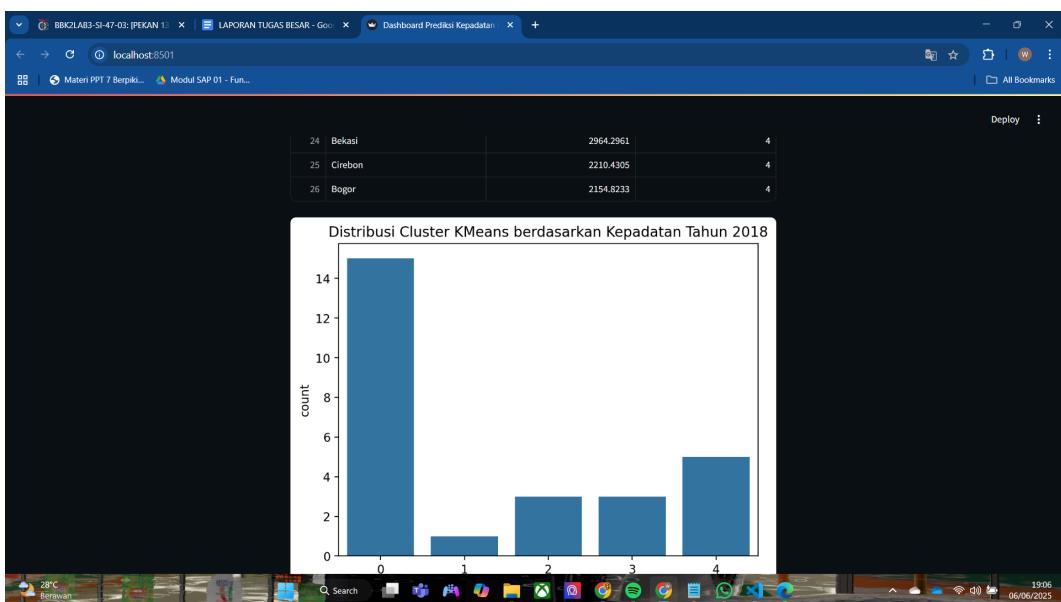
Gambar 4. 18 Hasil interface 4



Gambar 4. 19 Hasil interface 5



Gambar 4. 20 Hasil interface 6



Gambar 4. 21 Hasil interface 7

Berdasarkan hasil eksplorasi, pemodelan, dan evaluasi data yang dilakukan, terdapat beberapa temuan penting yang dapat ditafsirkan dan memberikan makna dalam konteks pengambilan keputusan dan perencanaan wilayah di Provinsi Jawa Barat:

Melalui penerapan metode *clustering* seperti K-Means :

1. Wilayah Jawa Barat berhasil dikelompokkan ke dalam klaster kepadatan tinggi, sedang, dan rendah.
2. Wilayah seperti Kota Bandung, Bekasi, dan Depok termasuk dalam klaster dengan kepadatan tinggi
3. Sebaliknya, kabupaten seperti Pangandaran dan Sukabumi tergolong dalam klaster kepadatan rendah,

Grafik yang dihasilkan dari eksplorasi data memperlihatkan bahwa sebagian besar wilayah mengalami pertumbuhan penduduk yang stabil dan meningkat dari 2018 ke 2020, meskipun laju pertumbuhan berbeda-beda. Model K-Means terbaik diperoleh dengan nilai Silhouette Score tertinggi dan distribusi klaster yang seimbang secara spasial dan Model KNeighborsRegressor memiliki nilai koefisien determinasi ( $R^2$ ) yang cukup tinggi, menunjukkan kemampuan prediktif yang baik.

**Tabel Bobot Pengerjaan**

Daniel Indra Kusuma	102022300337	25%
Naufal Rafi	102022300279	25%
Eldryan Azri	102022300138	25%
Wildan Zaaqi	102022300154	25%

**Tabel Pembagian Tugas**

Daniel Indra Kusuma	102022300337	Data Eksploration, data pre-paration, data modelling & evaluation, deployment
Naufal Rafi	102022300279	Data Eksploration, data pre-paration, data modelling & evaluation, deployment
Eldryan Azri	102022300138	Data Eksploration, data pre-paration, data modelling & evaluation, deployment
Wildan Zaaqi	102022300154	Data Eksploration, data pre-paration, data modelling & evaluation, deployment