

FINAL PROJECT

Praktikum Data Mining

Kelompok 4 B1

OUR TEAM

OUR TEAM



**MUHAMMAD AKBAR
GULUNNA**
2109116046



**FIRZIAN CAESAR
ANANTA**
2109116058

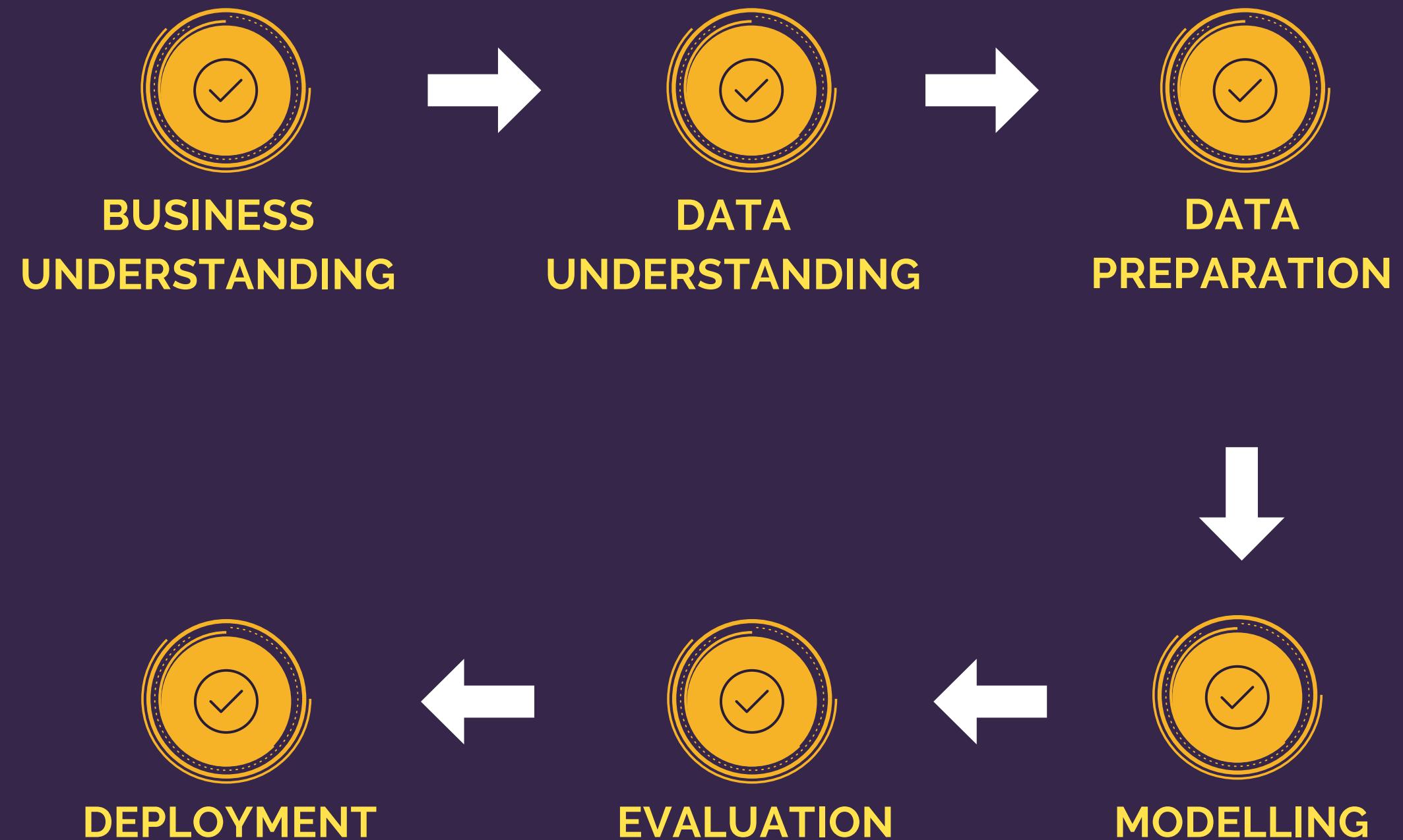


WILDA AZIZAH
2109116062



SELAMAT RIYANDI
2109116064

ALUR CRISP-DM

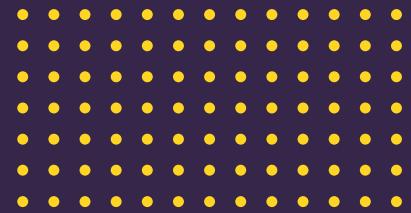


CRISP-DM (SUPERVISED DATASET)

Tanaman Padi Sumatera Dataset

01

BUSINESS UNDERSTANDING



Tanaman Padi Sumatera Dataset

Dataset ini berisi informasi tentang produksi padi di Sumatera dari tahun 2010 hingga 2020. Variabel yang termasuk dalam dataset ini antara lain provinsi, tahun, luas panen, produksi, dan produktivitas.

Goals

1. **Dataset Sumatera Memahami sebaran produksi padi di Sumatera berdasarkan provinsi dan waktu.**
2. **Mengidentifikasi faktor-faktor yang mempengaruhi produksi dan produktivitas padi di Sumatera.**
3. **Membuat prediksi produksi dan produktivitas padi di Sumatera.**



Tujuan penggunaan data mining:

1. Melakukan analisis exploratory data untuk mendapatkan pemahaman awal tentang dataset produksi padi di Sumatera.
2. Melakukan analisis regresi untuk melihat hubungan antara variabel yang ada dalam dataset.
3. Melakukan analisis faktor untuk mengidentifikasi faktor-faktor yang mempengaruhi produksi dan produktivitas padi di Sumatera.
4. Membuat model prediksi produksi dan produktivitas padi di Sumatera berdasarkan variabel-variabel yang ada dalam dataset.
5. Dengan menggunakan data mining pada dataset ini, diharapkan dapat memberikan insight atau wawasan tentang produksi padi di Sumatera, serta membantu dalam pengambilan keputusan dalam perencanaan dan pengembangan usaha pertanian di Sumatera.



02

DATA

UNDERSTANDING



Dataset Tanaman Padi Sumatera, Indonesia

Dataset Hasil Produksi dari tahun 1993-2020

 kaggle.com

123 Curah hujan

123 Kelembapan

123 Luas Panen

123 Produksi

RBC Provinsi

123 Suhu rata-rata

123 Tahun



COLLECT DATA

Dataset ini diperoleh dari laman :
<https://www.kaggle.com/datasets/ardikasatria/datasetsettanamanpadisumatera>

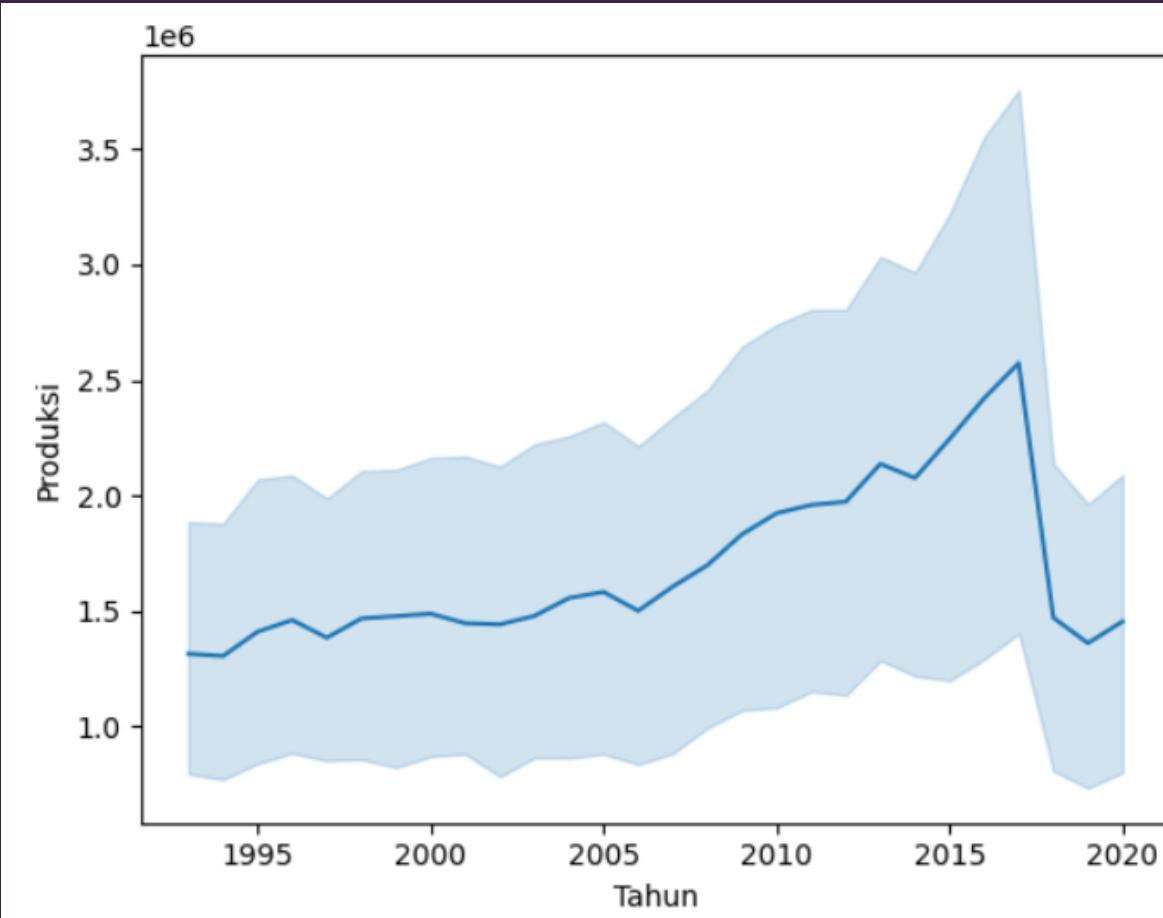
Didalam dataset tanaman padi Sumatera ini terdapat 7 kolom

DESCRIBE DATASET

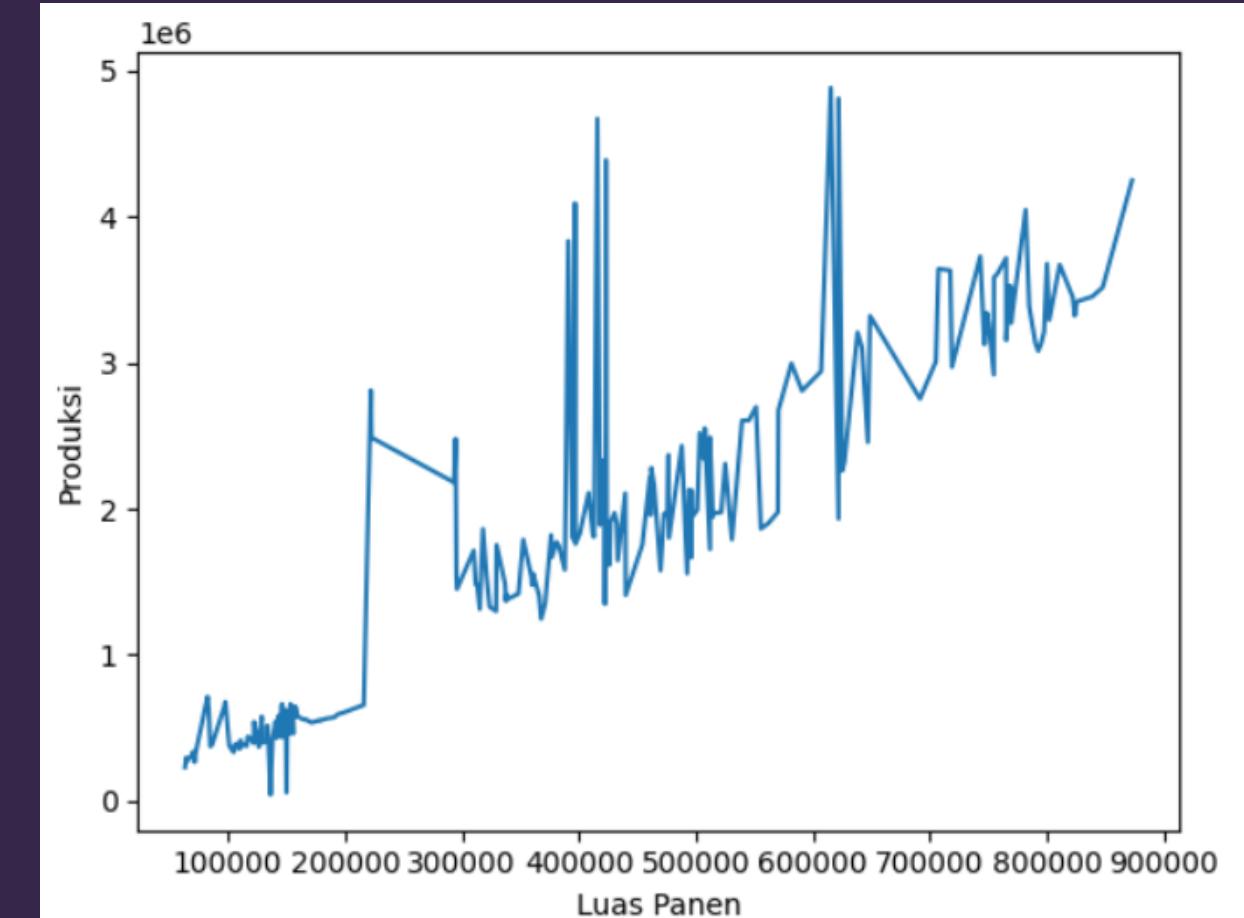
 Provinsi	Kolom ini menunjukkan lokasi pengamatan pada tiap-tiap provinsi.
 Tahun	Kolom ini menunjukkan waktu pengamatan pada setiap tahunnya.
 Produksi	Kolom ini menunjukkan jumlah produksi padi pada setiap provinsi dan tahunnya.
 Luas Panen	Kolom ini menunjukkan luas lahan yang ditanami padi pada setiap provinsi dan tahunnya.
 Curah Hujan	Kolom ini menunjukkan rata-rata curah hujan pada setiap provinsi dan tahunnya.
 Kelembapan	Kolom ini menunjukkan rata-rata kelembapan udara pada setiap provinsi dan tahunnya.
 Suhu rata-rata	Kolom ini menunjukkan rata-rata suhu udara pada setiap provinsi dan tahunnya.

EXPLORE DATA

Menampilkan perubahan produksi padi di wilayah Sumatera dari tahun ke tahun dalam periode tertentu

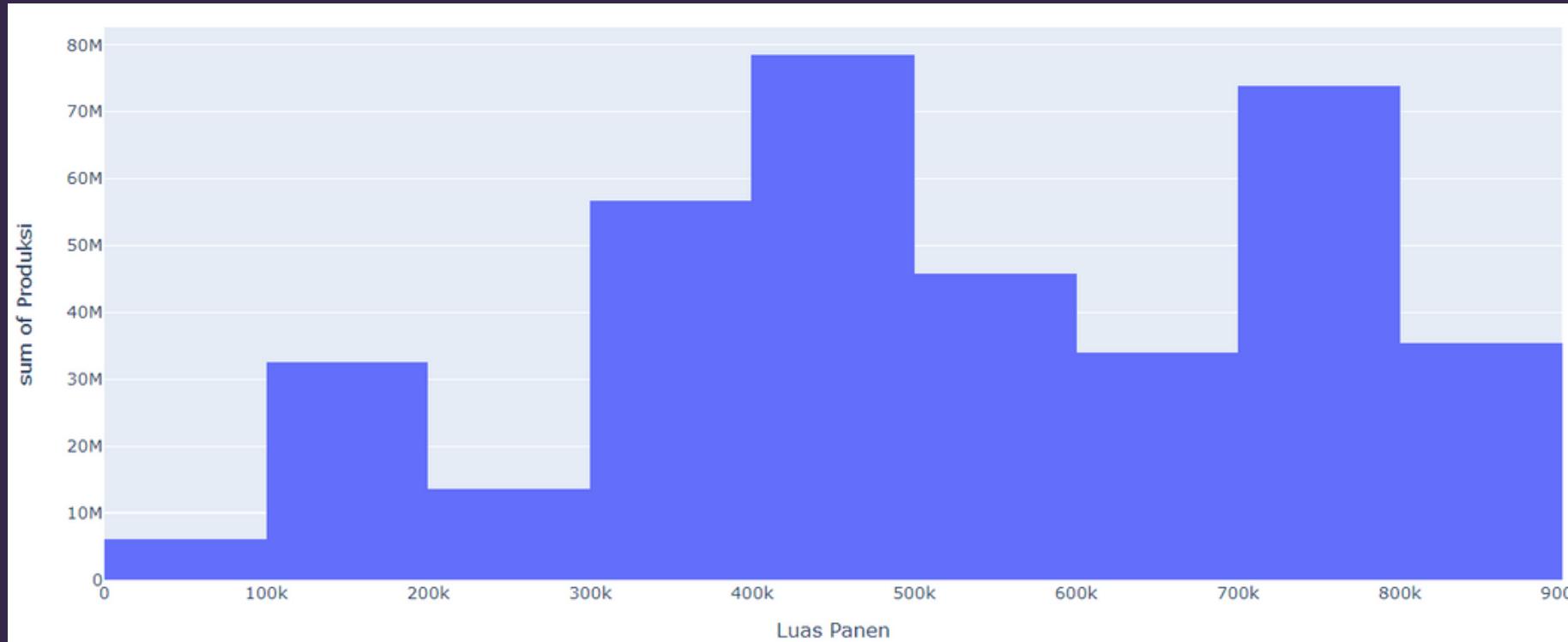


Menampilkan perubahan jumlah Produksi padi berdasarkan Luas Panen.

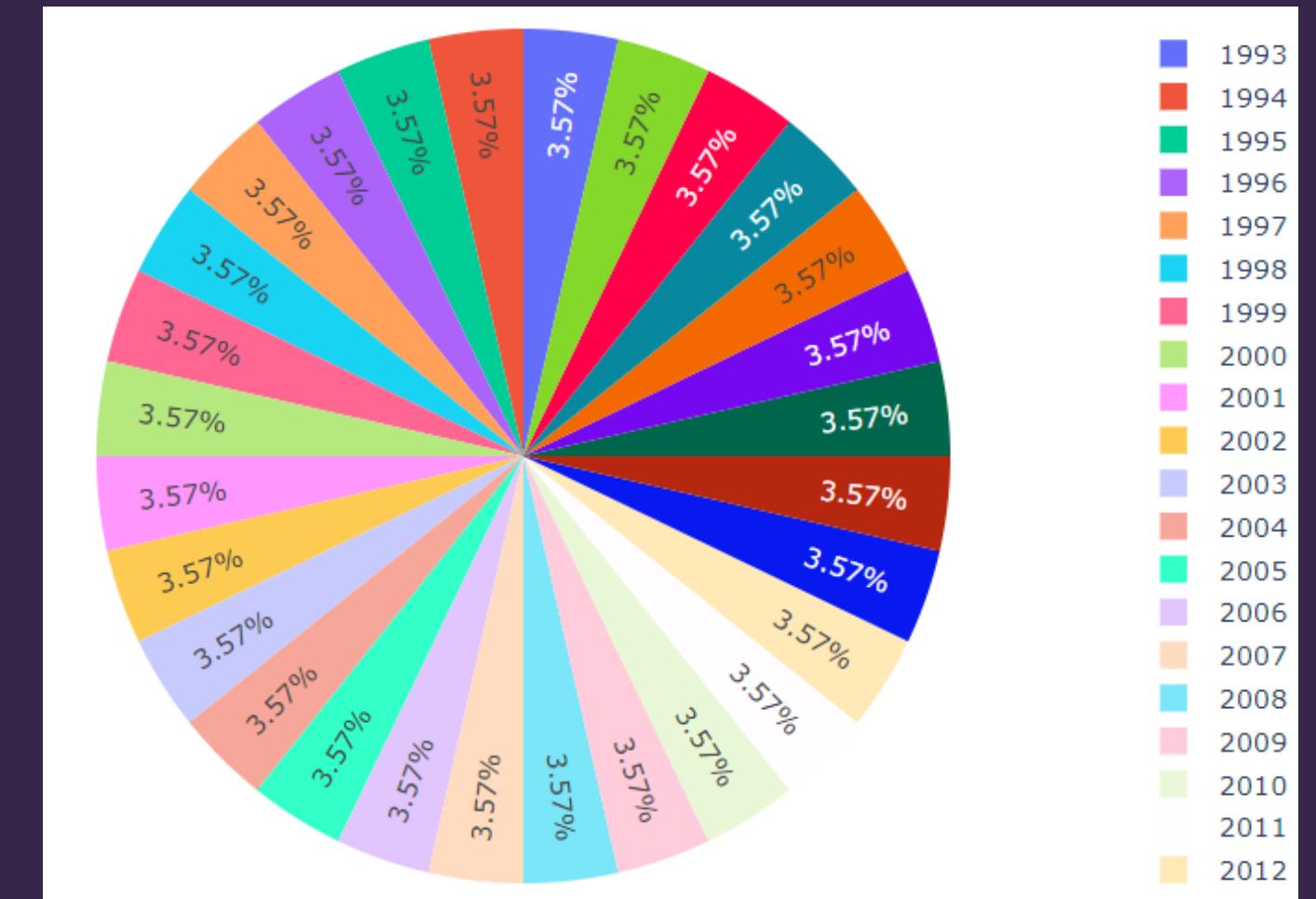


EXPLORE DATA

Menampilkan distribusi frekuensi produksi padi berdasarkan luas panen pada wilayah Sumatera



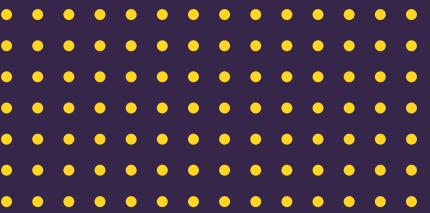
Menampilkan distribusi data untuk setiap tahun dalam dataset tersebut, untuk memperkirakan trend produksi padi di wilayah Sumatera dalam jangka waktu tertentu.



03

DATA

PREPARATION



MISSING VALUE CHECK

Hasil kode tersebut menunjukkan bahwa tidak ada nilai kosong pada setiap kolom dalam dataset

```
df.isna().sum()
```

```
Provinsi      0  
Tahun         0  
Produksi      0  
Luas Panen    0  
Curah hujan   0  
Kelembapan    0  
Suhu rata-rata 0  
dtype: int64
```

	Provinsi	Tahun	Produksi	Luas Panen	Curah hujan	Kelembapan	Suhu rata-rata
0	Aceh	1993	1329536.0	323589.0	1627.0	82.00	26.06
1	Aceh	1994	1299699.0	329041.0	1521.0	82.12	26.92
2	Aceh	1995	1382905.0	339253.0	1476.0	82.72	26.27
3	Aceh	1996	1419128.0	348223.0	1557.0	83.00	26.08
4	Aceh	1997	1368074.0	337561.0	1339.0	82.46	26.31

Hasil kode tersebut menampilkan lima baris pertama dari dataset tersebut yang mencakup beberapa kolom seperti Provinsi, Tahun, Produksi, Luas Panen, Curah hujan, Kelembapan, Suhu rata-rata





DATA TRANSFORMATION

```
selected_data = df[['Provinsi', 'Tahun', 'Produksi', 'Luas Panen', 'Curah hujan', 'Kelembapan', 'Suhu rata-rata']]  
  
print(selected_data.head())  
print(selected_data.tail())
```

Kode tersebut akan menampilkan 5 baris pertama dan terakhir dari DataFrame yang dipilih, sehingga memberikan gambaran singkat tentang isi dari dataset tersebut.

```
df=df.drop('Provinsi',axis=1)  
df.sample(5)
```

	Tahun	Produksi	Luas Panen	Curah hujan	Kelembapan	Suhu rata-rata
206	2003	1966293.0	472635.0	1682.2	68.71	29.67
129	2010	628828.0	153897.0	3207.0	84.60	27.10
141	1994	1347611.0	422109.0	1800.0	79.89	26.53
63	2000	1759059.0	396919.0	3040.6	86.01	25.86
119	2000	536779.0	171395.0	840.3	84.85	27.76

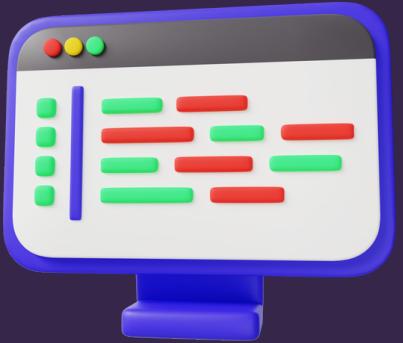
Hasil kode tersebut dilakukan untuk menghapus kolom 'Provinsi' dari DataFrame dan menampilkan 5 baris acak dari DataFrame tanpa kolom 'Provinsi'. 

04 & 05

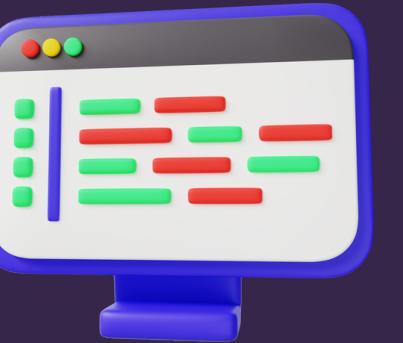
MODELLING & EVALUATION



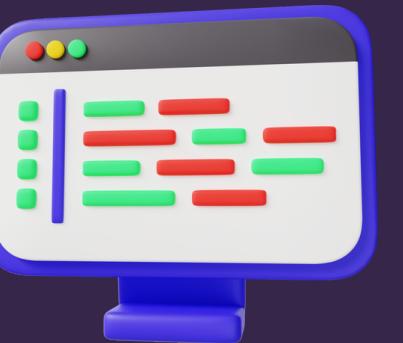
SELECT MODELLING TECHNIQUES



NAIVE BAYES



KNN



C.45



PENJELASAN SINGKAT METRIK

Accuracy : Seberapa banyak prediksi yang benar dari total data

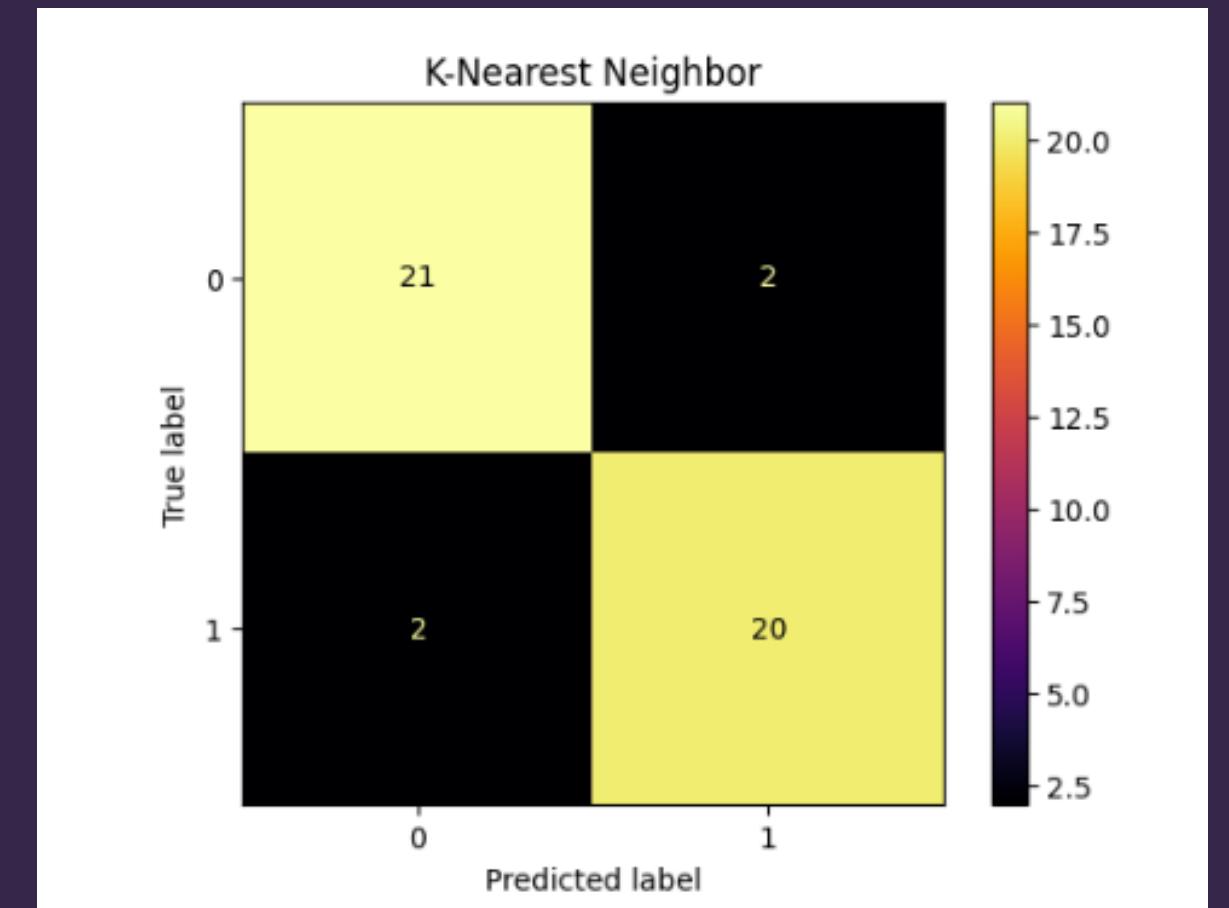
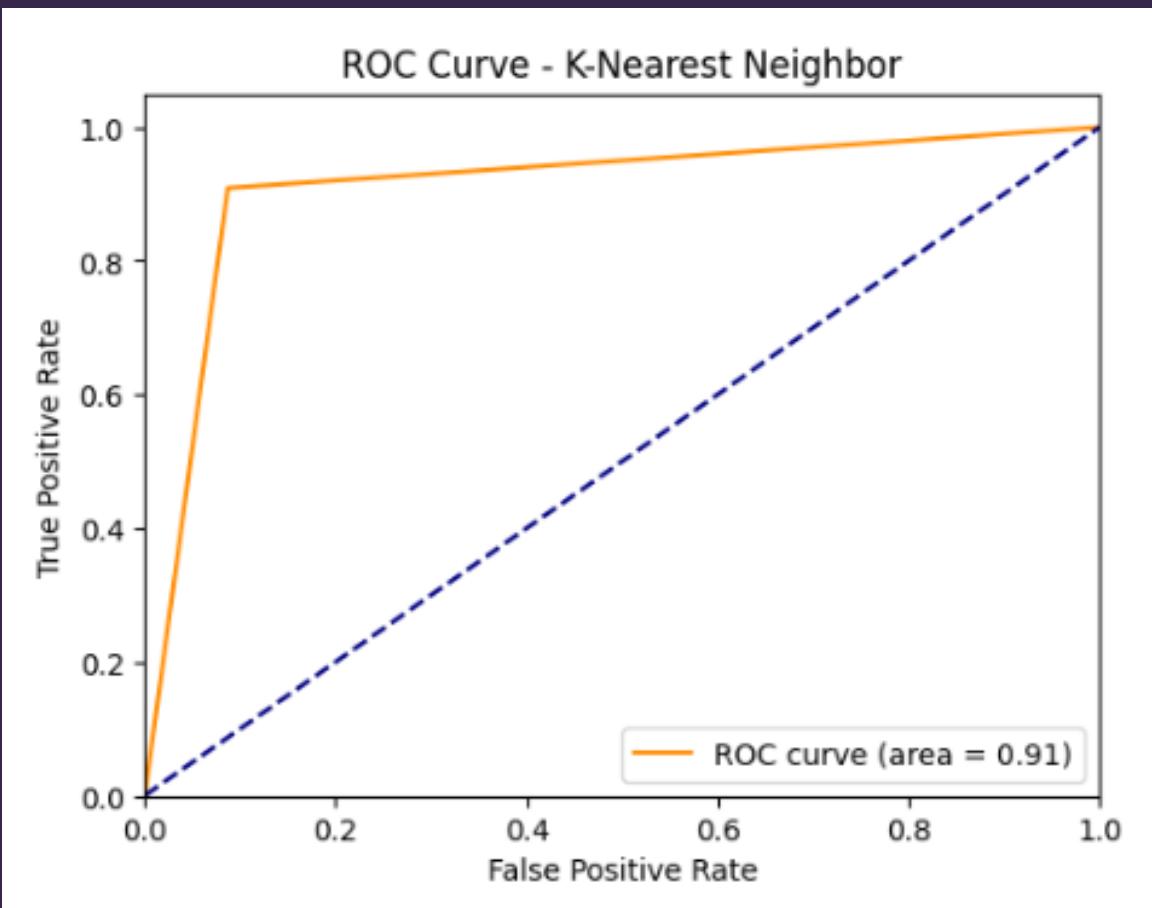
Precision : Seberapa banyak hasil true positive dari gabungan true positive dan false positive

Recall : Seberapa banyak hasil true positive dari gabungan true positive dan true negative

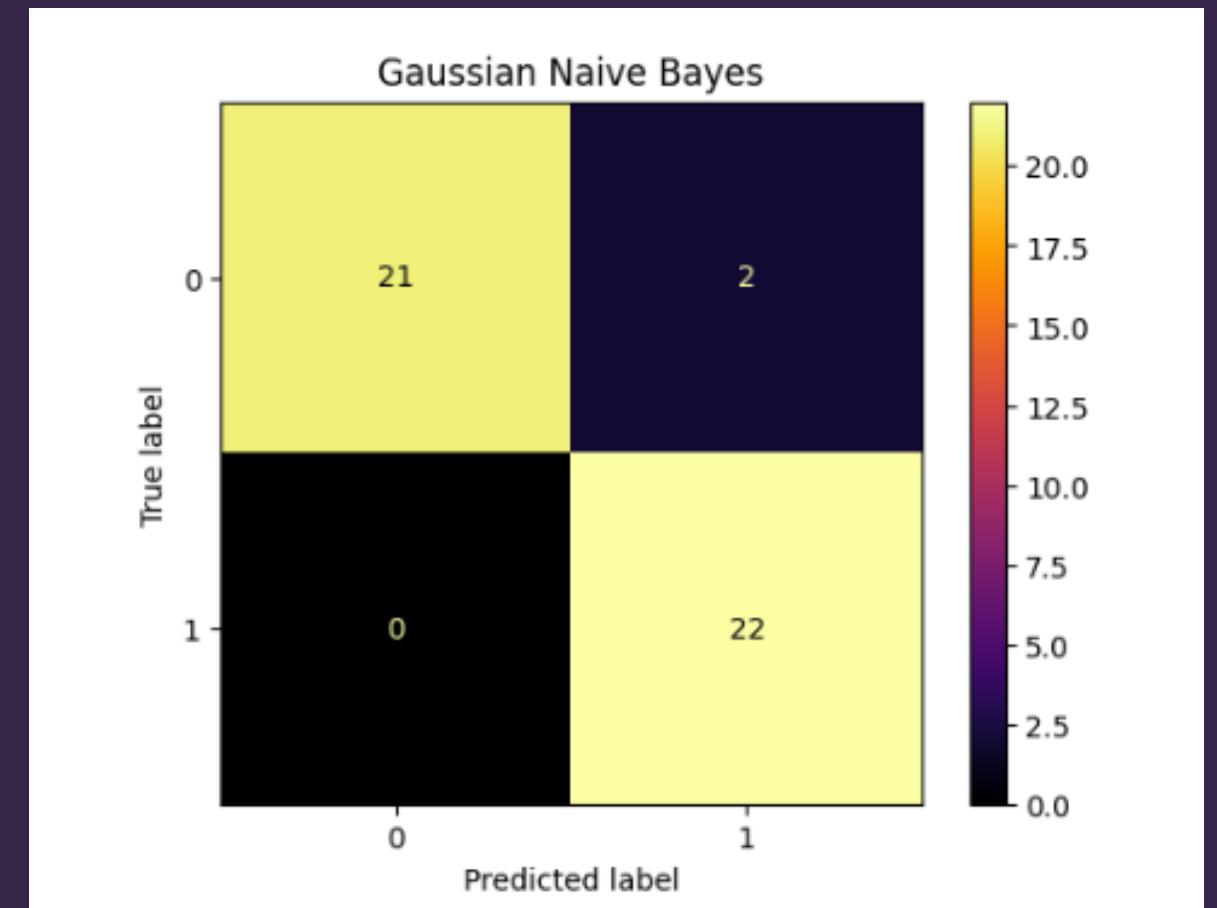
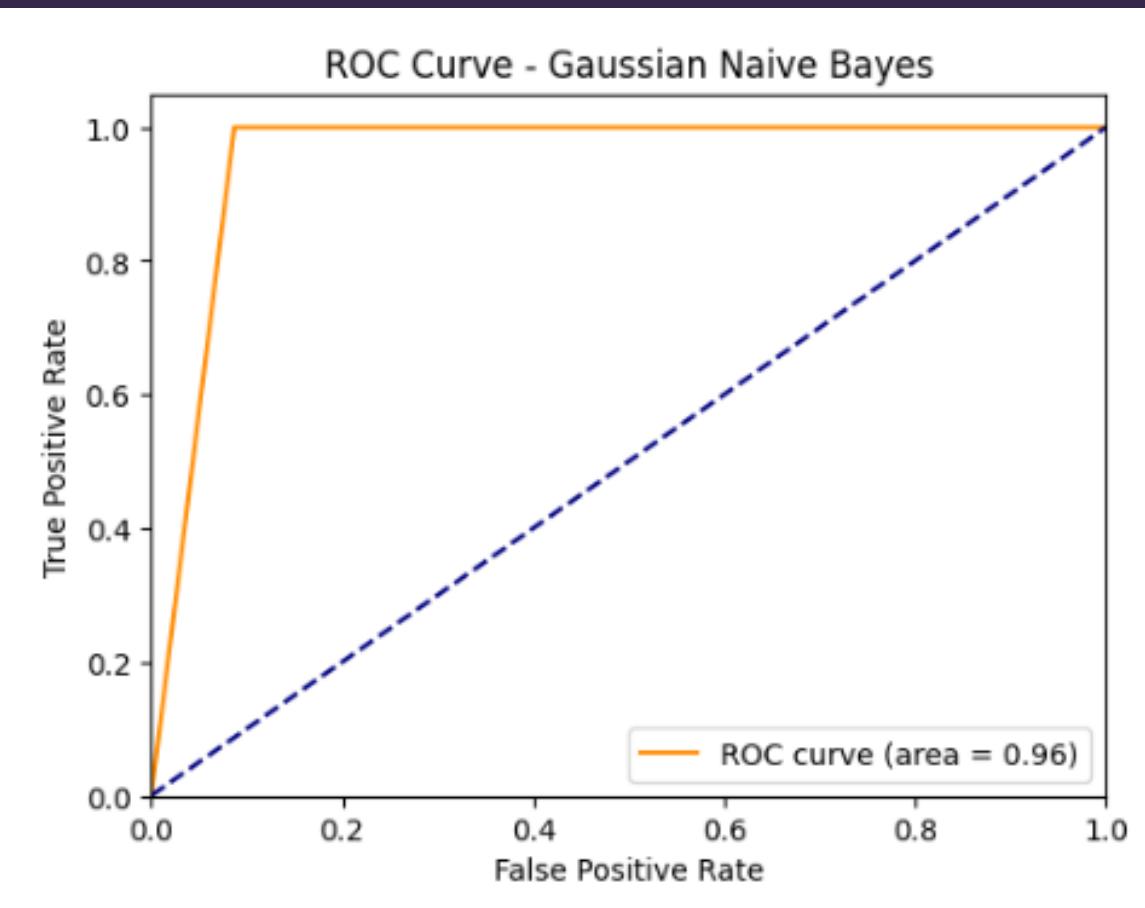
ROC-AUC : Seberapa baik model dapat membedakan antara kelas positif dan negatif



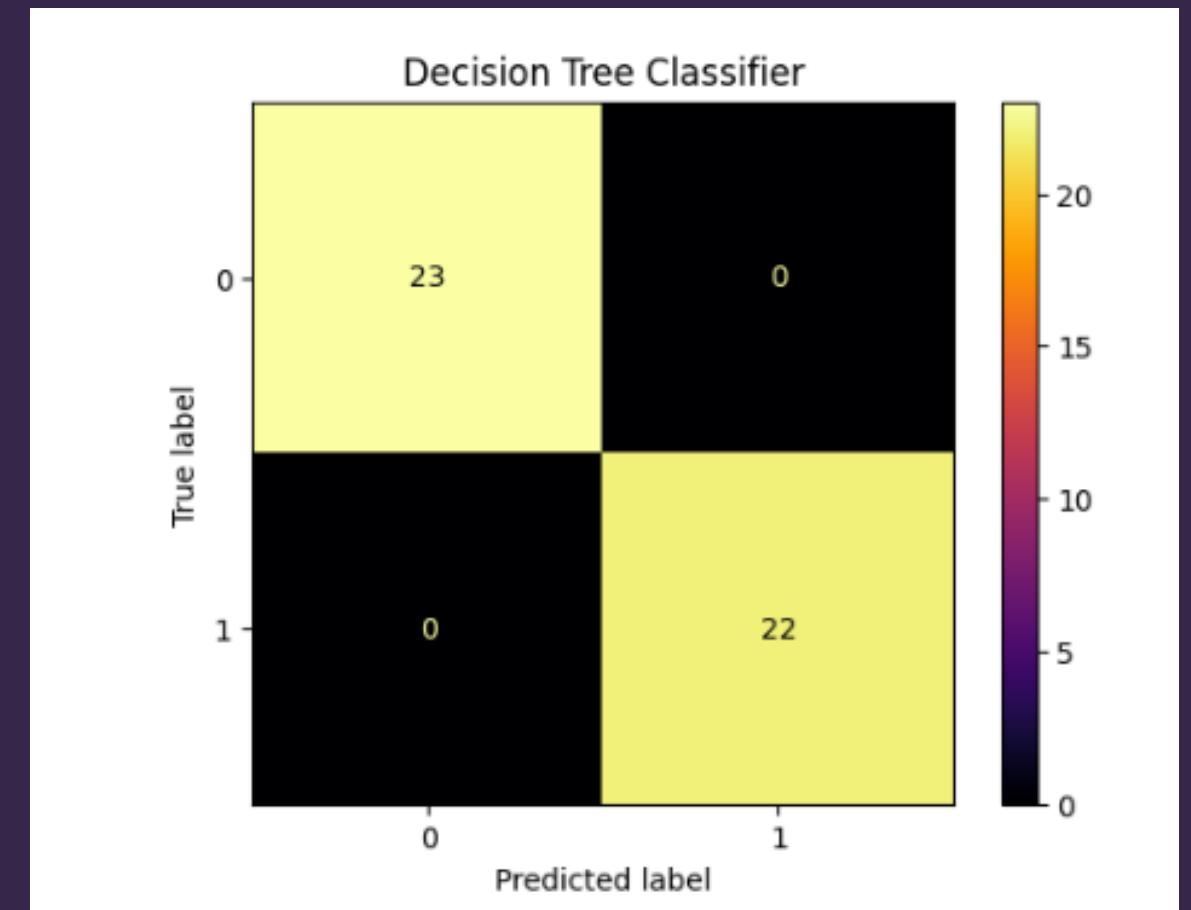
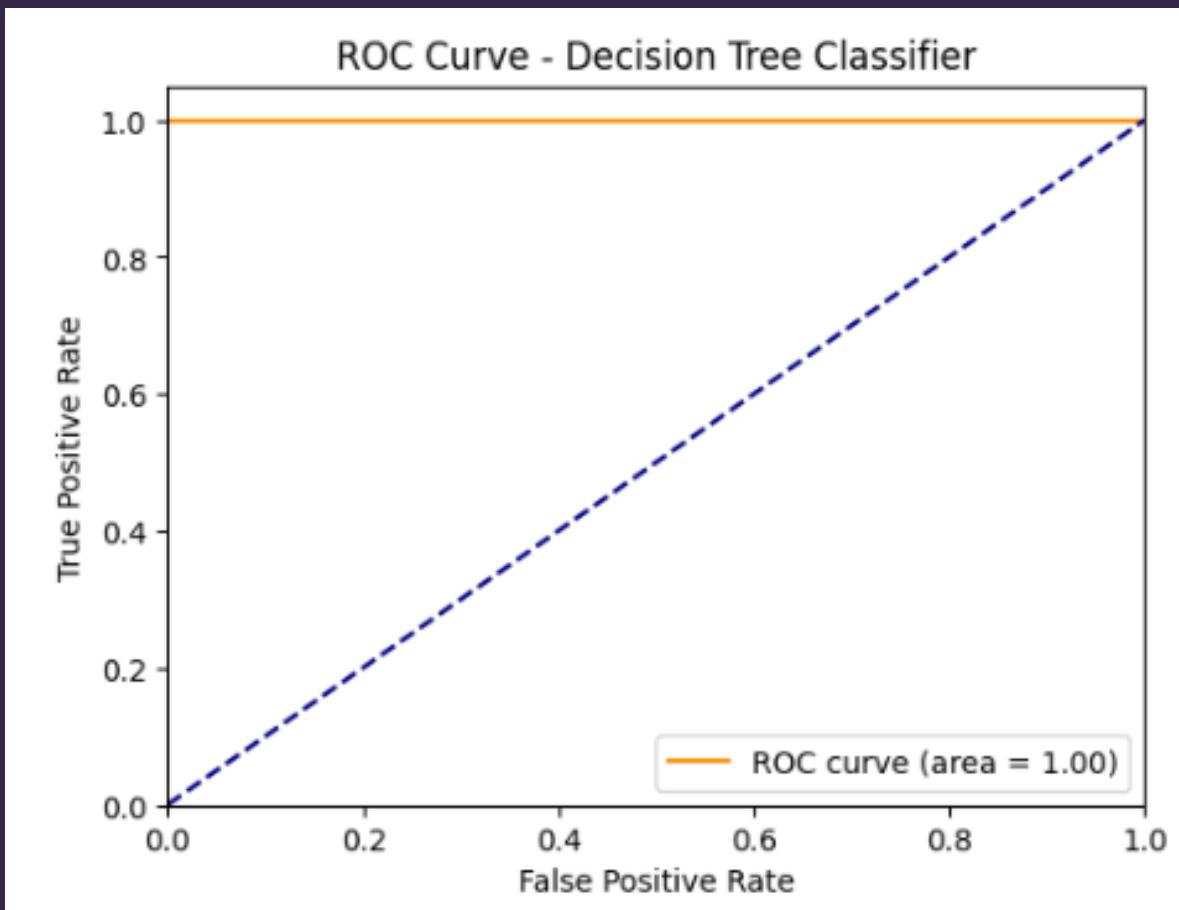
K-Nearest Neighbor (KNN)



Gaussian Naive Bayes



Decision Tree Classifier



KESIMPULAN

	Accuracy (%)	Precision (%)	Recall (%)	ROC-AUC (%)
K-Nearest Neighbor	90.476190	86.956522	95.238095	90.476190
Gaussian Naive Bayes	90.476190	86.956522	95.238095	90.476190
Decision Tree Classifier	97.619048	95.454545	100.000000	97.619048

Dari hasil evaluasi yang dilakukan, terlihat bahwa model Decision Tree Classifier memiliki performa paling baik dibandingkan dengan dua model lainnya dalam memprediksi apakah hasil produksi padi suatu wilayah di Sumatera dapat dikategorikan sebagai baik atau buruk. Hal ini terlihat dari nilai Accuracy, Precision, Recall, dan ROC-AUC yang paling tinggi dibandingkan dengan model K-Nearest Neighbor dan Gaussian Naive Bayes.

06

DEPLOYMENT

DEPLOYMENT

```
df.to_csv('Supervised - Deployment.csv')
```

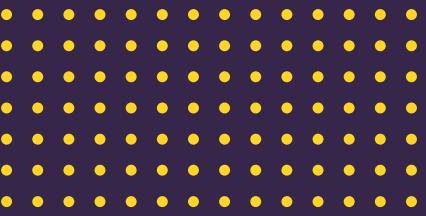
Kode tersebut akan mengonversi DataFrame (df) ke dalam format file CSV dan menyimpannya sebagai file dengan nama "Supervised - Deployment.csv". File ini dapat digunakan untuk menyimpan data hasil preproses atau data yang telah siap untuk dianalisis.

CRISP-DM (UN-SUPERVISED DATASET)

Life Expectancy Dataset

01

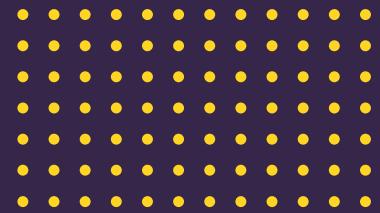
BUSINESS UNDERSTANDING



Life Expectancy Dataset

Dataset ini berisi kumpulan data tentang harapan hidup di negara-negara di seluruh dunia. Dataset ini berisi 6 kolom dan 223 Baris. Setiap baris sesuai dengan negara dalam urutan peringkat harapan hidup mereka. Dataset memiliki tiga kolom numerik, Harapan Hidup Keseluruhan, Harapan Hidup Pria dan Harapan Hidup Wanita.





Goals

Tujuan analisis terhadap data ini yaitu untuk mengidentifikasi faktor-faktor yang mempengaruhi harapan hidup penduduk di berbagai negara di dunia. Hal tersebut penting untuk dilakukan dalam upaya meningkatkan kesehatan dan kesejahteraan masyarakat, oleh karena itu penting untuk memahami faktor-faktor yang mempengaruhi harapan hidup penduduk. Sebab dibeberapa kasus, tingkat harapan hidup penduduk digunakan sebagai salah satu indikator penilaian terhadap suatu negara dalam menjamin kesejahteraan penduduknya.



Tujuan penggunaan data mining:

1. Melakukan analisis exploratory data untuk mendapatkan pemahaman awal mengenai statistik angka harapan hidup penduduk di suatu negara.
2. Melakukan analisis regresi untuk melihat hubungan antara variabel yang mempengaruhi angka harapan hidup penduduk.
3. Melakukan analisis faktor untuk mengidentifikasi faktor-faktor yang mempengaruhi harapan hidup penduduk pada suatu negara.
4. Membuat model prediksi kemungkinan angka harapan hidup penduduk pada suatu negara berdasarkan variabel-variabel yang ada dalam dataset.



02

DATA

UNDERSTANDING



COLLECT DATA

Dataset ini diperoleh dari laman :
<https://www.kaggle.com/datasets/amansaxena/lifeexpectancy>



Didalam Life Expectancy Dataset ini terdapat 6 kolom

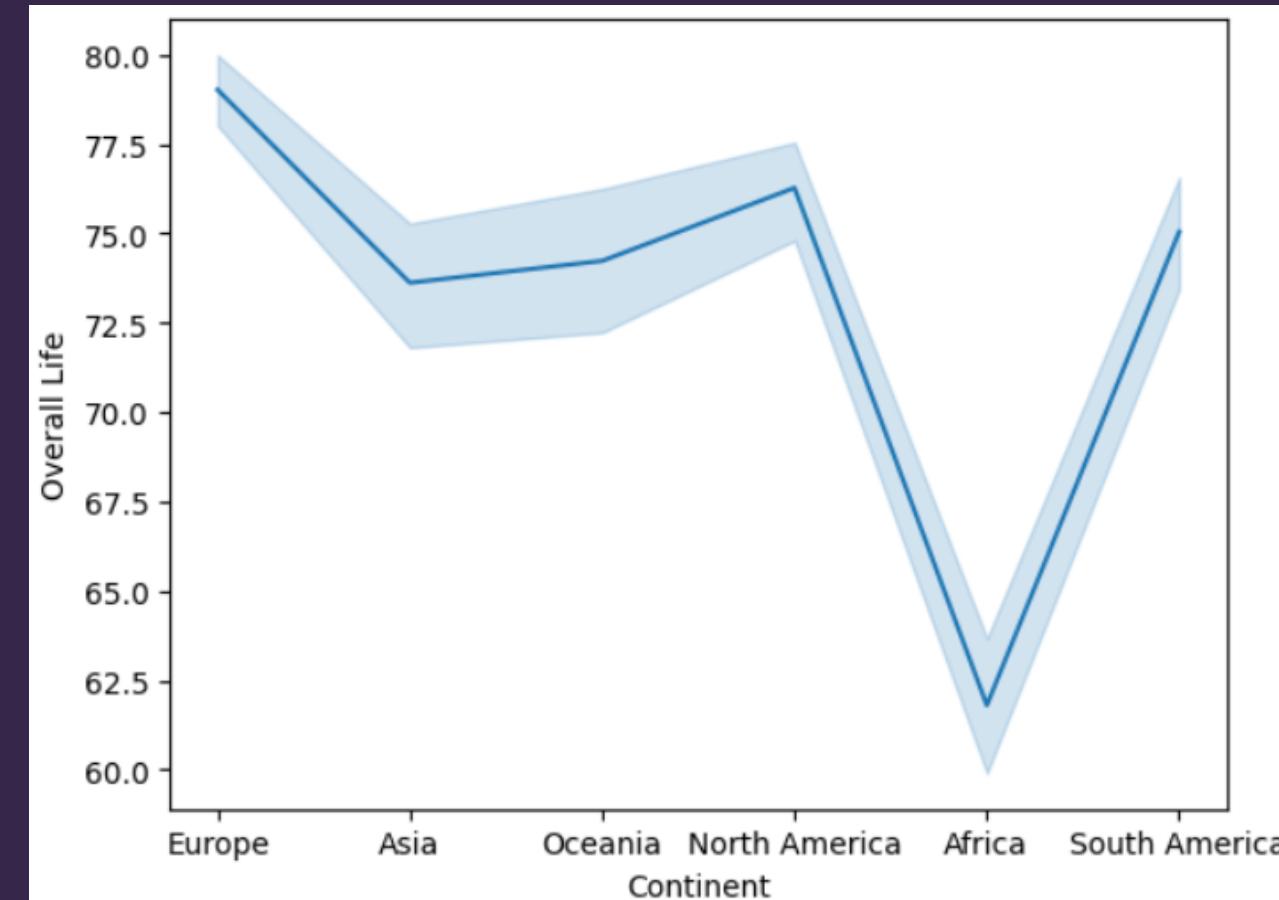


DESCRIBE DATASET

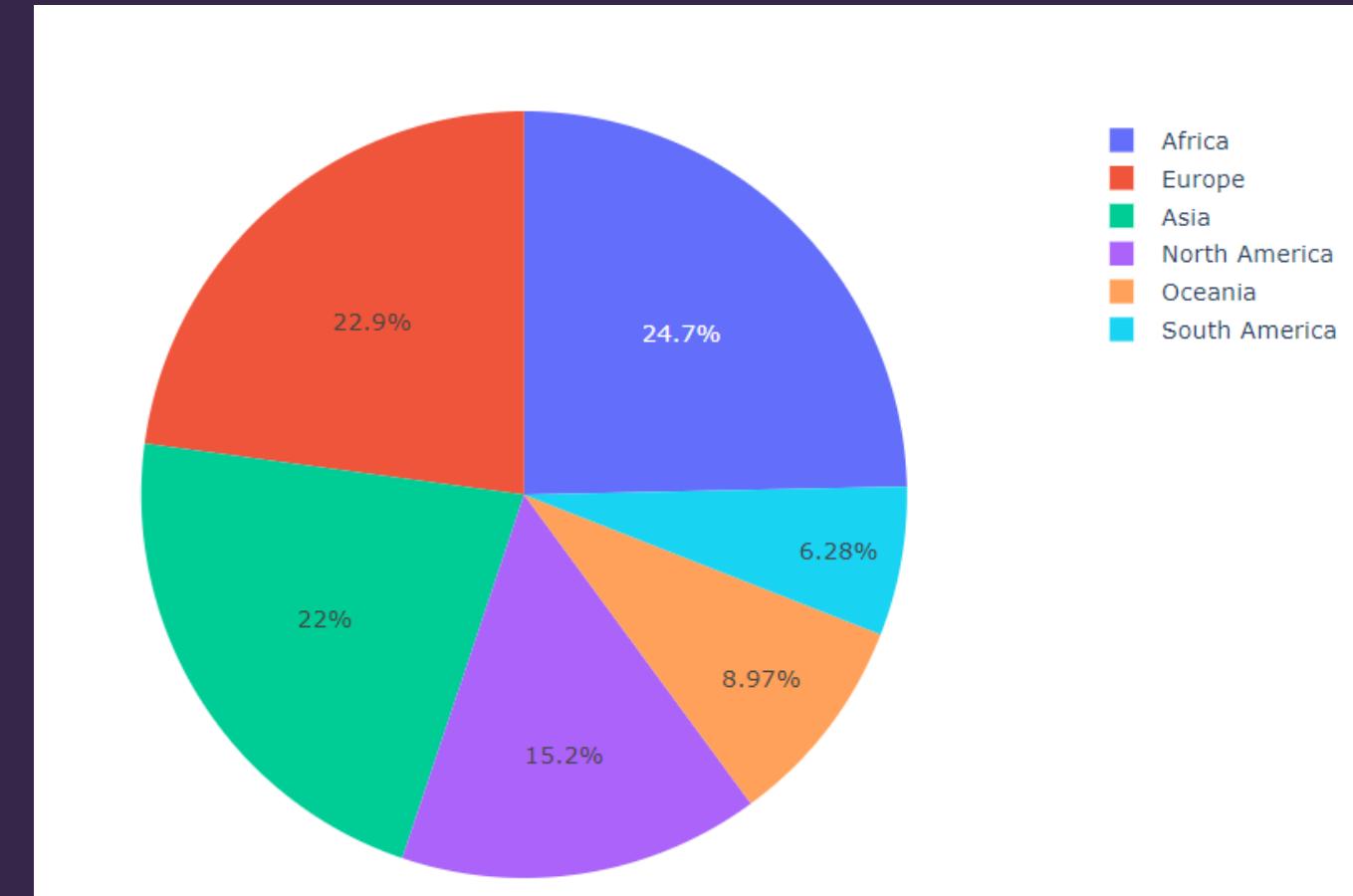
	Rank	Kolom yang menunjukkan peringkat dari suatu negara berdasarkan angka harapan hidup penduduk tertinggi.
	Country	Kolom yang menunjukkan daftar negara-negara yang terdata pada dataset harapan hidup penduduk.
	Overall Life	kolom yang menunjukkan angka harapan hidup penduduk secara rata-rata kombinasi dari penduduk berjenis kelamin laki-laki dan penduduk berjenis kelamin perempuan.
	Male Life	kolom yang menunjukkan angka harapan hidup penduduk berjenis kelamin laki-laki pada suatu negara.
	Female Life	kolom yang menunjukkan angka harapan hidup penduduk berjenis kelamin perempuan pada suatu negara.
	Continent	kolom yang menunjukkan asal wilayah regional atau benua dari suatu negara.

EXPLORE DATA

Menunjukkan bagaimana ekspektasi hidup pada setiap benua, serta perbedaan antara ekspektasi hidup di berbagai benua.



Menampilkan distribusi harapan hidup (Life Expectancy) di seluruh negara dalam dataset Life Expectancy,

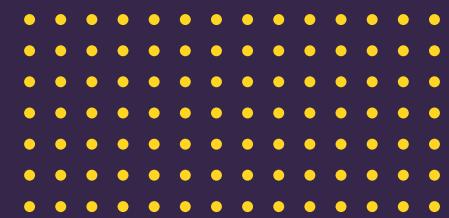




03

DATA

PREPARATION



MISSING VALUE CHECK

Hasil perintah tersebut menunjukkan bahwa tidak ada nilai kosong pada setiap kolom dalam dataset

```
df2.isna().sum()
```

```
Rank      0  
Country   0  
Overall Life 0  
Male Life 0  
Female Life 0  
Continent 0  
dtype: int64
```

```
df2.head()
```

	Rank	Country	Overall Life	Male Life	Female Life	Continent
0	1	Monaco	89.5	85.6	93.5	Europe
1	2	Japan	85.0	81.7	88.5	Asia
2	3	Singapore	85.0	82.3	87.8	Asia
3	4	Macau; China	84.5	81.6	87.6	Asia
4	5	San Marino	83.3	80.7	86.1	Europe

Hasil perintah tersebut menampilkan lima baris pertama dari dataset tersebut yang mencakup beberapa kolom seperti Country, Overall Life, Male Life, Female Life, dan Continent





DATA TRANSFORMATION

```
df2 = df2.drop('Rank',axis = 1)  
  
df2 = df2.drop(3)
```

Pada hasil perintah tersebut menghapus kolom 'Rank' dari DataFrame dan baris ke-3 dari DataFrame yang diambil dari dataset "Life Expectancy".

	Country	Overall Life	Male Life	Female Life	Continent
0	Monaco	89.5	85.6	93.5	Europe
1	Japan	85.0	81.7	88.5	Asia
2	Singapore	85.0	82.3	87.8	Asia
4	San Marino	83.3	80.7	86.1	Europe
5	Iceland	83.0	80.9	85.3	Europe
...
218	Gabon	52.1	51.6	52.5	Africa
219	Swaziland	51.6	52.2	51.0	Africa
220	Afghanistan	51.3	49.9	52.7	Asia
221	Guinea-Bissau	50.6	48.6	52.7	Africa
222	Chad	50.2	49.0	51.3	Africa

222 rows × 5 columns



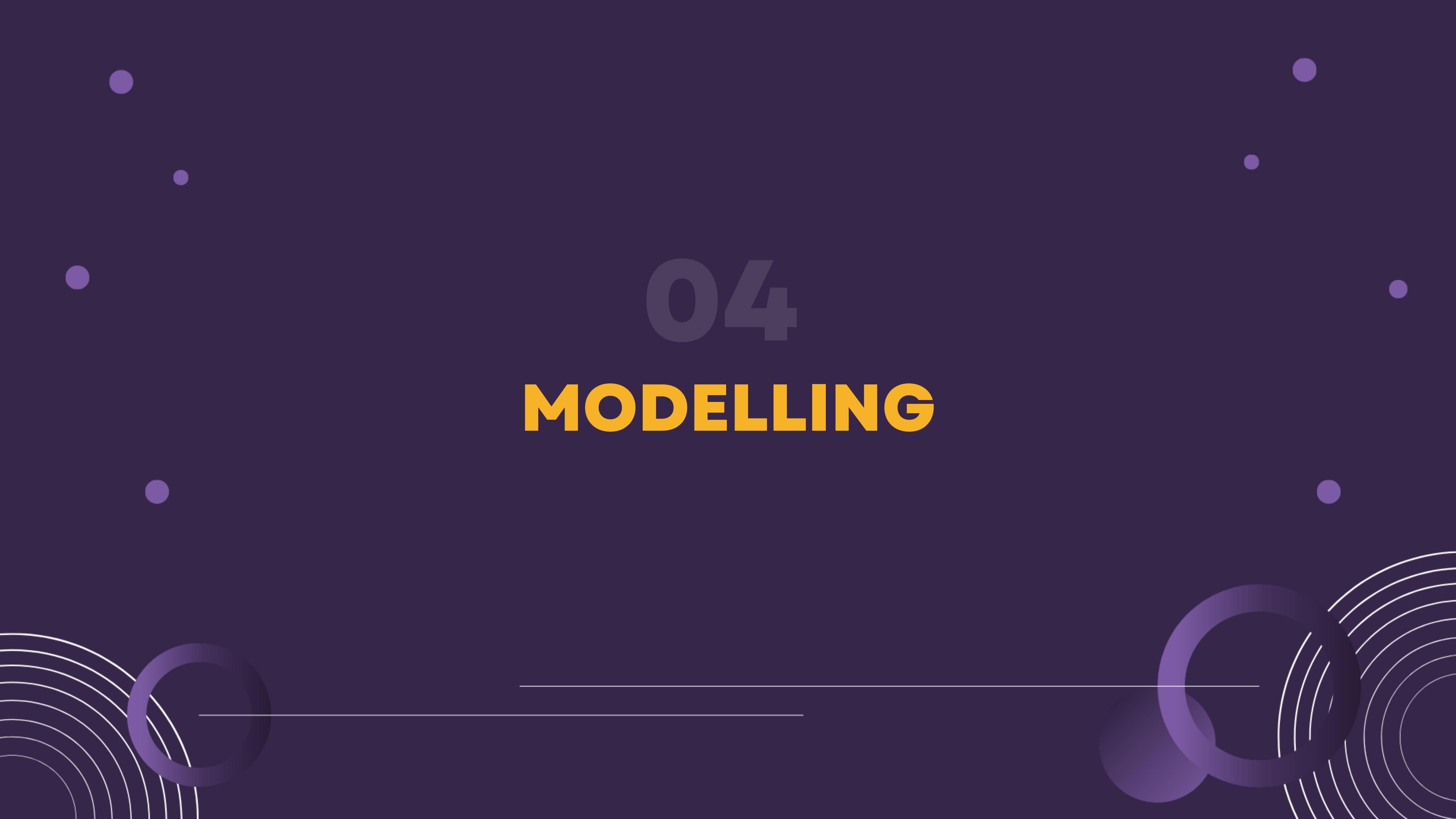


DATA TRANSFORMATION

```
# Pisahkan fitur  
features = ['Overall Life', 'Male Life', 'Female Life']
```

```
# Buat variabel X berisi data numerik yang akan digunakan  
X = df2[features].values  
print(X)  
  
# Normalisasi data menggunakan z-score  
X = (X - np.mean(X, axis=0)) / np.std(X, axis=0)
```

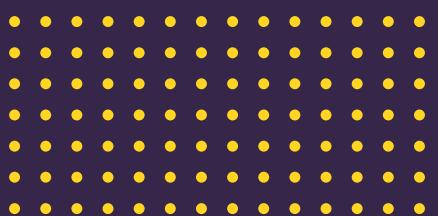
Pada Kode tersebut digunakan untuk mengambil fitur-fitur tertentu dari dataset dan kemudian melakukan normalisasi menggunakan z-score.



04

MODELLING

PREPROCESSING DATA



Mendefinisikan algoritma clustering K-Means

```
def kmeans(X, k, max_iterations=100):
    # Initialize centroids randomly
    centroids = X[np.random.choice(len(X), k, replace=False)]

    for i in range(max_iterations):
        # Assign labels based on closest centroid
        distances = np.sqrt(((X - centroids[:, np.newaxis])**2).sum(axis=2))
        labels = np.argmin(distances, axis=0)

        # Update centroids based on mean of points in each cluster
        for j in range(k):
            centroids[j] = X[labels == j].mean(axis=0)

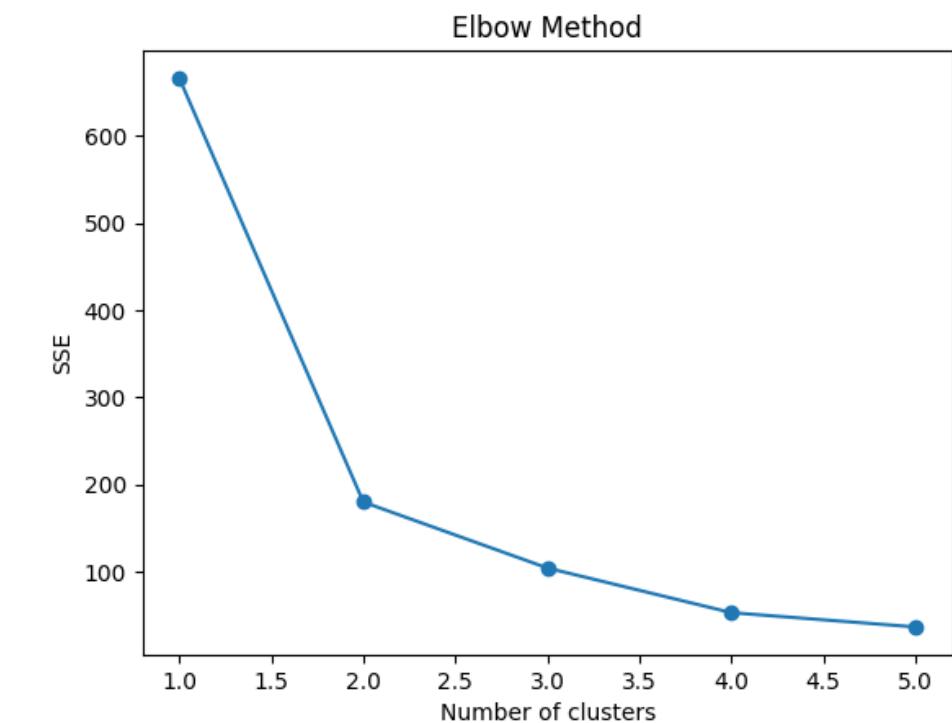
    return centroids, labels

# Define the function to compute SSE
def compute_sse(X, centroids, labels):
    sse = 0
    for i in range(len(X)):
        centroid = centroids[labels[i]]
        sse += np.sum((X[i] - centroid) ** 2)
    return sse
```

Menentukan jumlah cluster

```
# Calculate SSE for different numbers of clusters
sse = []
for k in range(1, 6):
    centroids, labels = kmeans(X, k)
    sse.append(compute_sse(X, centroids, labels))

# Plot SSE vs number of clusters
plt.plot(range(1, 6), sse, marker='o')
plt.xlabel('Number of clusters')
plt.ylabel('SSE')
plt.title('Elbow Method')
plt.show()
```



```
[ ] k = 3
```

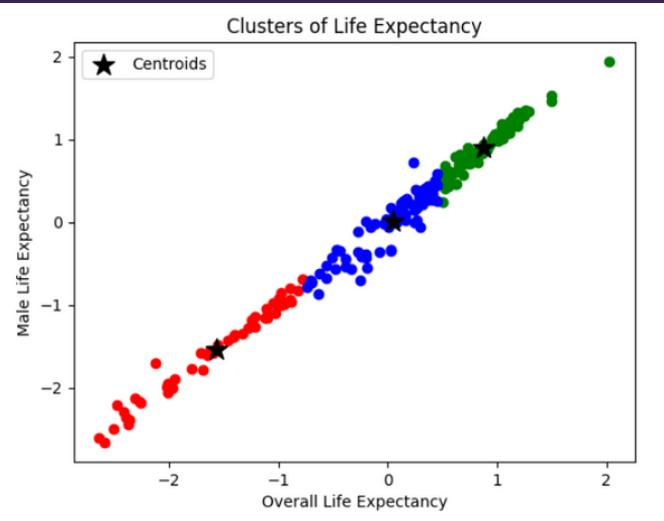


PREPROCESSING DATA

```
#jalankan K-means  
centroids, labels = kmeans(X, k)  
  
Kmeans = kmeans(X, k)  
df2['Cluster'] = labels
```

Eksekusi dataset ke dalam algoritma K-Means

```
# Plot the data points and centroids  
colors = ['r', 'g', 'b','y']  
for i in range(k):  
    plt.scatter(X[labels == i, 0], X[labels == i, 1], color=colors[i])  
plt.scatter(centroids[:, 0], centroids[:, 1], marker='*', s=200, color='black', label="Centroids")  
plt.title('Clusters of Life Expectancy')  
plt.xlabel('Overall Life Expectancy')  
plt.ylabel('Male Life Expectancy')  
plt.legend()  
plt.show()
```



Menampilkan output proses clustering dengan algoritma K-Means pada grafik 2d

```
cluster1 = df2[df2['Cluster'] == 0]  
cluster2 = df2[df2['Cluster'] == 1]  
cluster3 = df2[df2['Cluster'] == 2]
```

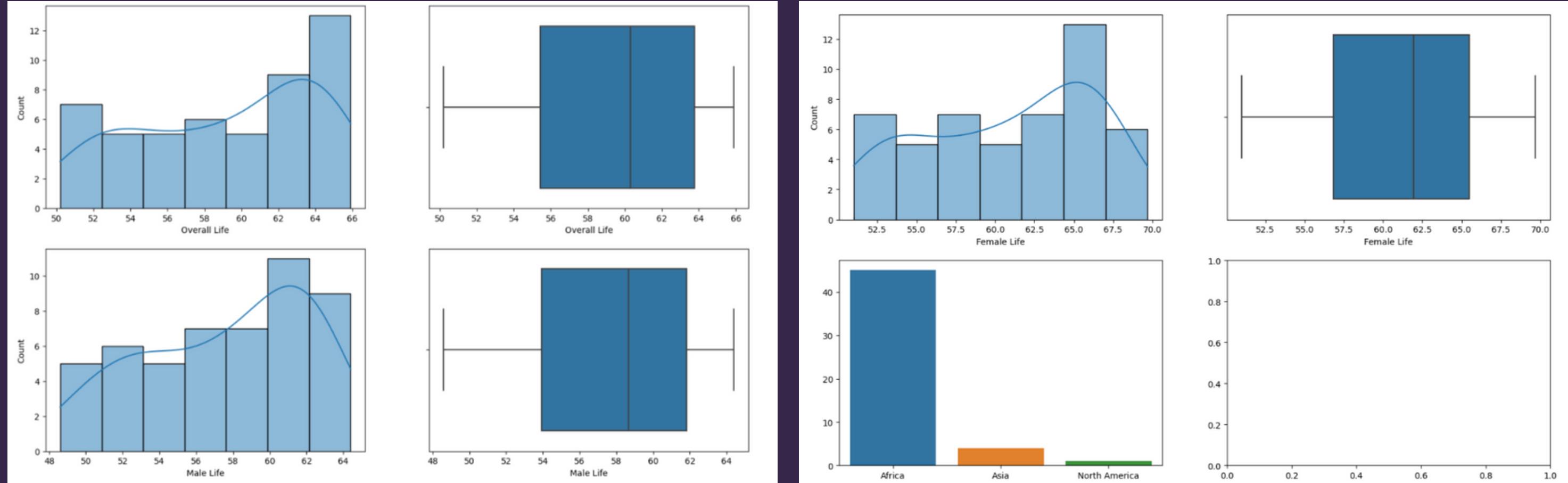
Baris tersebut merupakan baris untuk menampung setiap baris data ke dalam kluster yang sesuai, dan baris hasil kluster tersebut akan disimpan dalam sebuah variabel.

PREPROCESSING DATA



Cluster 1

```
plot_cluster(cluster1)
```



```
countclus1 = cluster1['Cluster'].count()  
countclus1
```

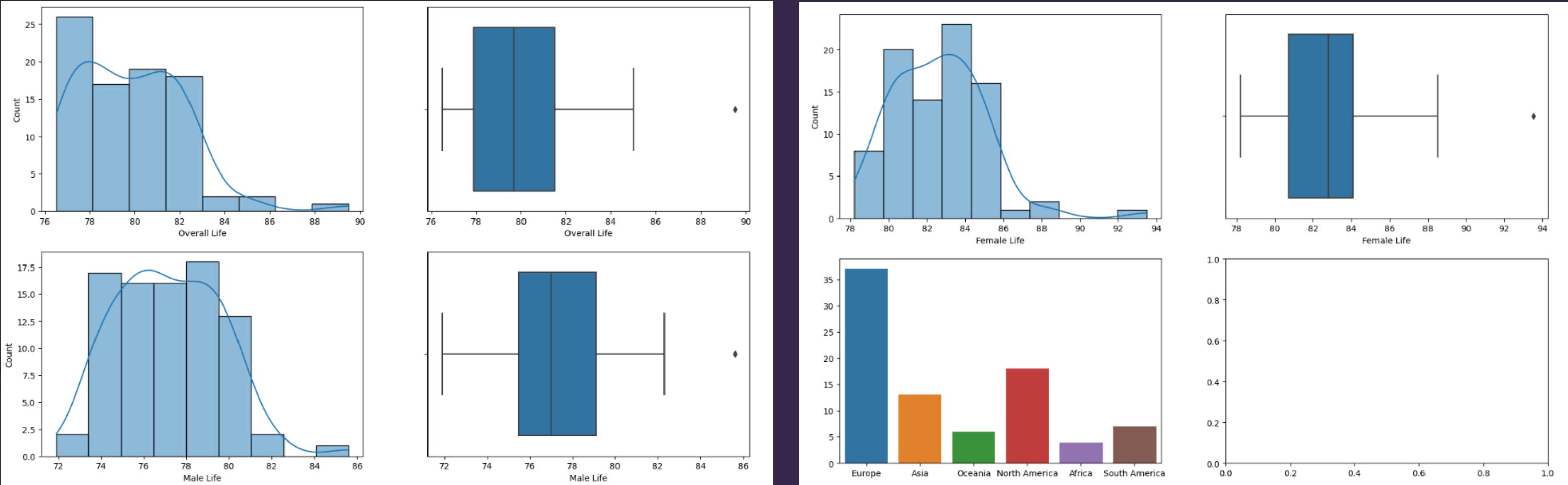
```
50
```

Menghitung jumlah data pada kolom 'Cluster' dari sebuah DataFrame dengan nama 'cluster1'

PREPROCESSING DATA

Cluster 2

```
plot_cluster(cluster2)
```



```
countclus2 = cluster2['Cluster'].sum()  
countclus2
```

85

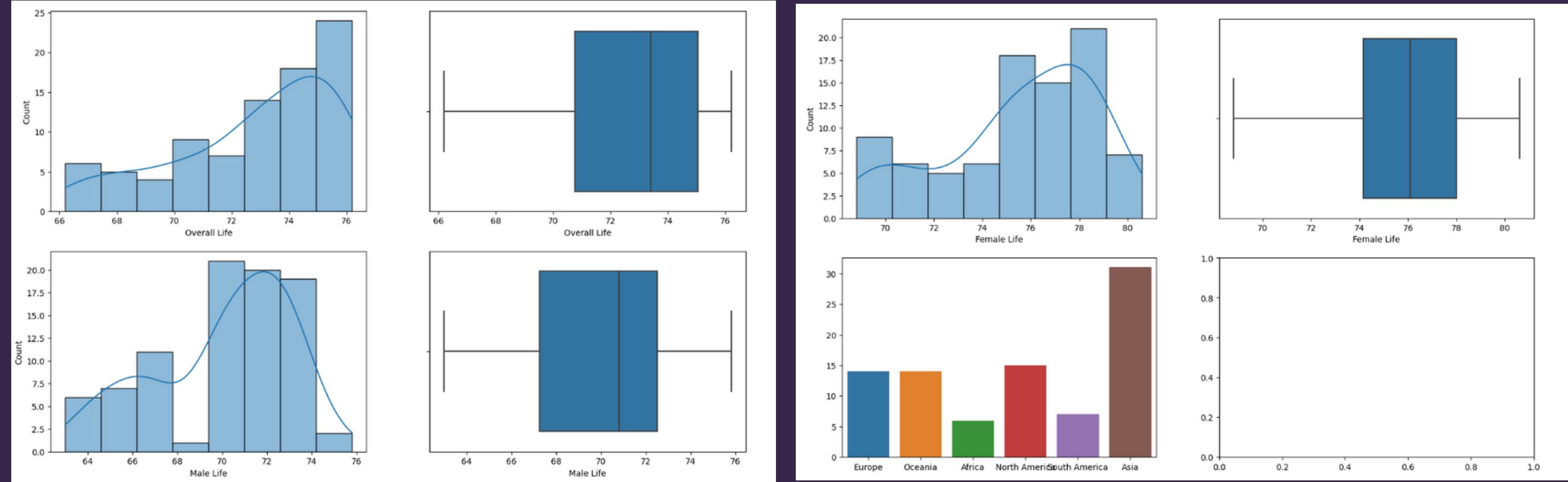
Menghitung jumlah data pada kolom 'Cluster' dari sebuah DataFrame dengan nama 'cluster1'

PREPROCESSING DATA



Cluster 3

```
plot_cluster(cluster3)
```



```
countclus3 = cluster3['Cluster'].count()  
countclus3
```

87

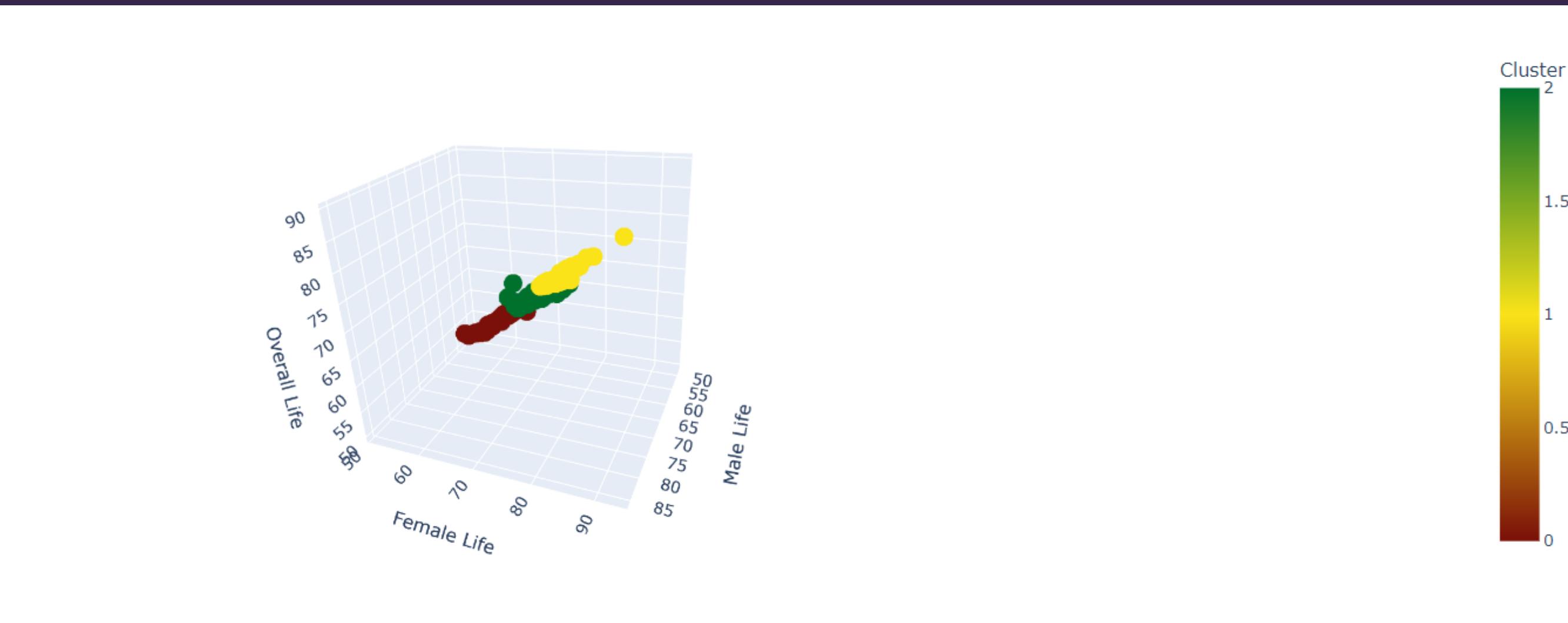
Menghitung jumlah data pada kolom 'Cluster' dari sebuah DataFrame dengan nama 'cluster1'

PREPROCESSING DATA

Menampilkan output proses clustering dalam diagram 3d

```
px.scatter_3d(df2, x='Male Life', y='Female Life', z='Overall Life', color='Cluster',color_continuous_scale=['#7a1009', '#fae319', '#00702d'])
```

Output :



05

EVALUATION

EVALUATING

Menghitung tiga metrik evaluasi clustering

```
from sklearn.metrics import silhouette_score,calinski_harabasz_score,davies_bouldin_score
# Menghitung nilai Silhouette Score
silhouette_avg = silhouette_score(train.drop('Cluster', axis=1), labels)

# Menghitung nilai Calinski-Harabasz Index
ch_score= calinski_harabasz_score(train.drop('Cluster', axis=1), labels)

# Menghitung nilai Davies-Bouldin Index
db_score= davies_bouldin_score(train.drop('Cluster', axis=1), labels)

# Create dataframe for metrics and scores
metrics = ["Silhouette Score", "Calinski-Harabasz Score", "Davies-Bouldin Score"]
scores = [silhouette_avg, ch_score, db_score]
df_scores = pd.DataFrame({'Metric': metrics, 'Score': scores})

# reset index dataframe
df_scores = df_scores.reset_index(drop=True)

# ubah nama axis pada index dan kolom
df_scores = df_scores.rename_axis('', axis=0).rename_axis('Metrics', axis=1)

# Display dataframe as table without index column
display(df_scores.style.hide_index())
```

Output :

```
<ipython-input-742-0f7be67053f9>:23: FutureWarning:  
this method is deprecated in favour of `Styler.hide(axis="index")`
```

Metric	Score
Silhouette Score	0.507805
Calinski-Harabasz Score	591.491460
Davies-Bouldin Score	0.605073

EVALUATING



Menampilkan data statistik dari setiap cluster

```
# Add cluster labels to the data
df2['Cluster'] = labels

# Analyze each cluster
for cluster in sorted(df2['Cluster'].unique()):
    print(f"Cluster {cluster}:")
    print(df2[df2['Cluster'] == cluster].describe())
```

Output:

Cluster 0:				
	Overall Life	Male Life	Female Life	Cluster
count	50.000000	50.00000	50.000000	50.0
mean	59.218000	57.64800	60.870000	0.0
std	4.888875	4.51868	5.447345	0.0
min	50.200000	48.60000	51.000000	0.0
25%	55.425000	53.92500	56.825000	0.0
50%	60.300000	58.65000	61.950000	0.0
75%	63.750000	61.82500	65.475000	0.0
max	65.900000	64.40000	69.700000	0.0
Cluster 1:				
	Overall Life	Male Life	Female Life	Cluster
count	85.000000	85.00000	85.000000	85.0
mean	79.838824	77.189412	82.622353	1.0
std	2.389814	2.493517	2.467039	0.0
min	76.500000	71.900000	78.200000	1.0
25%	77.900000	75.500000	80.700000	1.0
50%	79.700000	77.000000	82.800000	1.0
75%	81.500000	79.100000	84.100000	1.0
max	89.500000	85.600000	93.500000	1.0
Cluster 2:				
	Overall Life	Male Life	Female Life	Cluster
count	87.000000	87.000000	87.000000	87.0
mean	72.794253	70.048276	75.578161	2.0
std	2.823216	3.042790	3.151686	0.0
min	66.200000	63.000000	68.800000	2.0
25%	70.750000	67.250000	74.150000	2.0
50%	73.400000	70.800000	76.100000	2.0
75%	75.050000	72.500000	78.000000	2.0
max	76.200000	75.800000	80.600000	2.0

06

DEPLOYMENT

DEPLOYMENT

Telah dilakukan analisis pada dataset life expectancy menggunakan algoritma klustering K-Means, dan membagi data yang terdapat pada dataset tersebut ke dalam 3 cluster dengan cirinya masing-masing. Setelah dilakukan evaluasi dari hasil modeling algoritma K-Means pada dataset ini menunjukkan performa yang cukup baik dengan nilai matriks Silhouette sebesar 0,494, matriks Calinski-Harabasz sebesar 585, dan matriks Davies-Bouldin sebesar 0,635. Dari proses clustering yang telah dilakukan, diperoleh ciri-ciri unik pada setiap kluster sebagai berikut:

- Cluster 1 : Cluster 1 merupakan cluster tingkat harapan hidup menengah dengan rerata angka harapan hidup hingga usia 72 tahun. Cluster ini umumnya terdiri dari negara-negara berkembang dari kawasan benua Asia
- Cluster 2 : Cluster 2 merupakan cluster tingkat harapan hidup paling tinggi dengan rerata angka harapan hidup hingga usia 79 tahun. Cluster ini umumnya terdiri dari negara-negara maju dari kawasan benua Eropa
- Cluster 3 : Cluster 1 merupakan cluster tingkat harapan hidup paling rendah dengan rerata angka harapan hidup hingga usia 59 tahun. Cluster ini umumnya terdiri dari negara-negara berkembang dari kawasan benua Afrika

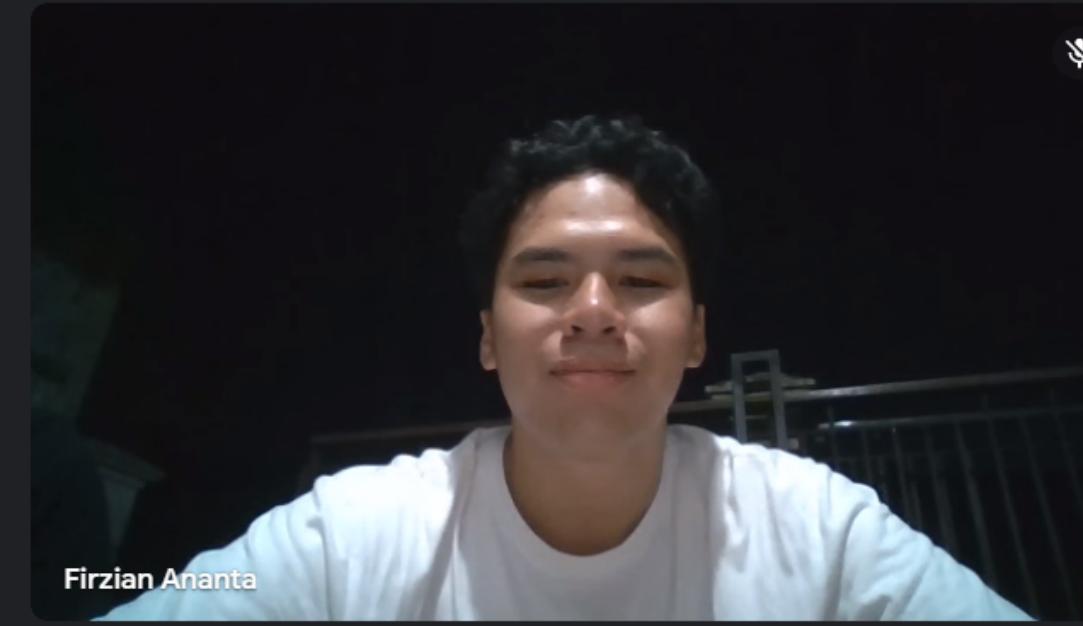
Selain itu dari hasil analisis data juga dapat ditarik kesimpulan lain bahwasanya angka harapan hidup penduduk dengan jenis kelamin wanita pada suatu negara selalu lebih tinggi dibandingkan penduduk dengan jenis kelamin laki-laki.

```
df2.to_csv('Unsupervised - Deployment.csv',index = False)
```

TABEL KONTRIBUSI

Muhammad Akbar Gulunna	Unsupervised (Modelling, Evaluation, Deployment)
Firzian Caesar Ananta	Supervised (Business Understanding, Data Understanding, Explore Data, Modelling, Evaluation, Deployment) dan Visualisasi Google Data Studio
Selamat Riyandi	Unsupervised (Business Understanding, Data Understanding, Explore Data, Data Preparation, Evaluation, Deployment)
Wilda Azizah	Supervised (Data Preparation) dan Membuat Powerpoint

FOTO BERSAMA



THANK YOU
THANK YOU