K Means Clustering

Aim:

To Write a Python program to understand and perform the K-Means clustering algorithm on the given dataset.

Algorithm:

1. Load essential libraries like NumPy, Pandas, Matplotlib, Seaborn, and K Means from sklearn.
2. Read the Mall_Customers.csv file into a Data Frame using pd.readcsv().
3. Use .info() and .head() to understand the structure and preview the dataset.
4. Apply sns.pairplot() to explore pairwise relationships between features.
5. Select relevant features and fit KMeans with a specified number of clusters
6. Add cluster labels to the DataFrame and visualize clusters using scatter plots.
7. Use the elbow method by plotting within-cluster sum of squares for cluster counts from 1 to 10.

Program:

```
[12]: import warnings
      import os

      os.environ["OMP_NUM_THREADS"] = "1"

      warnings.filterwarnings("ignore", category=UserWarning)
```

```
[13]: import numpy as np
      import pandas as pd
      import matplotlib.pyplot as plt
      import seaborn as sns
      %matplotlib inline
```

```
[14]: df=pd.read_csv(r"C:\Users\siddesh\Downloads\Mall_Customers.csv")
```

```
[15]: df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 200 entries, 0 to 199
Data columns (total 5 columns):
 #   Column                  Non-Null Count  Dtype
---  ------                  --------------  -----
 0   CustomerID              200 non-null    int64
 1   Gender                  200 non-null    object
 2   Age                     200 non-null    int64
 3   Annual Income (k$)      200 non-null    int64
 4   Spending Score (1-100)  200 non-null    int64
dtypes: int64(4), object(1)
memory usage: 7.9+ KB
```

```
[16]: df.head()
```

[16]:

	CustomerID	Gender	Age	Annual Income (k$)	Spending Score (1-100)
0	1	Male	19	15	39
1	2	Male	21	15	81
2	3	Female	20	16	6
3	4	Female	23	16	77
4	5	Female	31	17	40

```
[17]: sns.pairplot(df)
      plt.show()
```
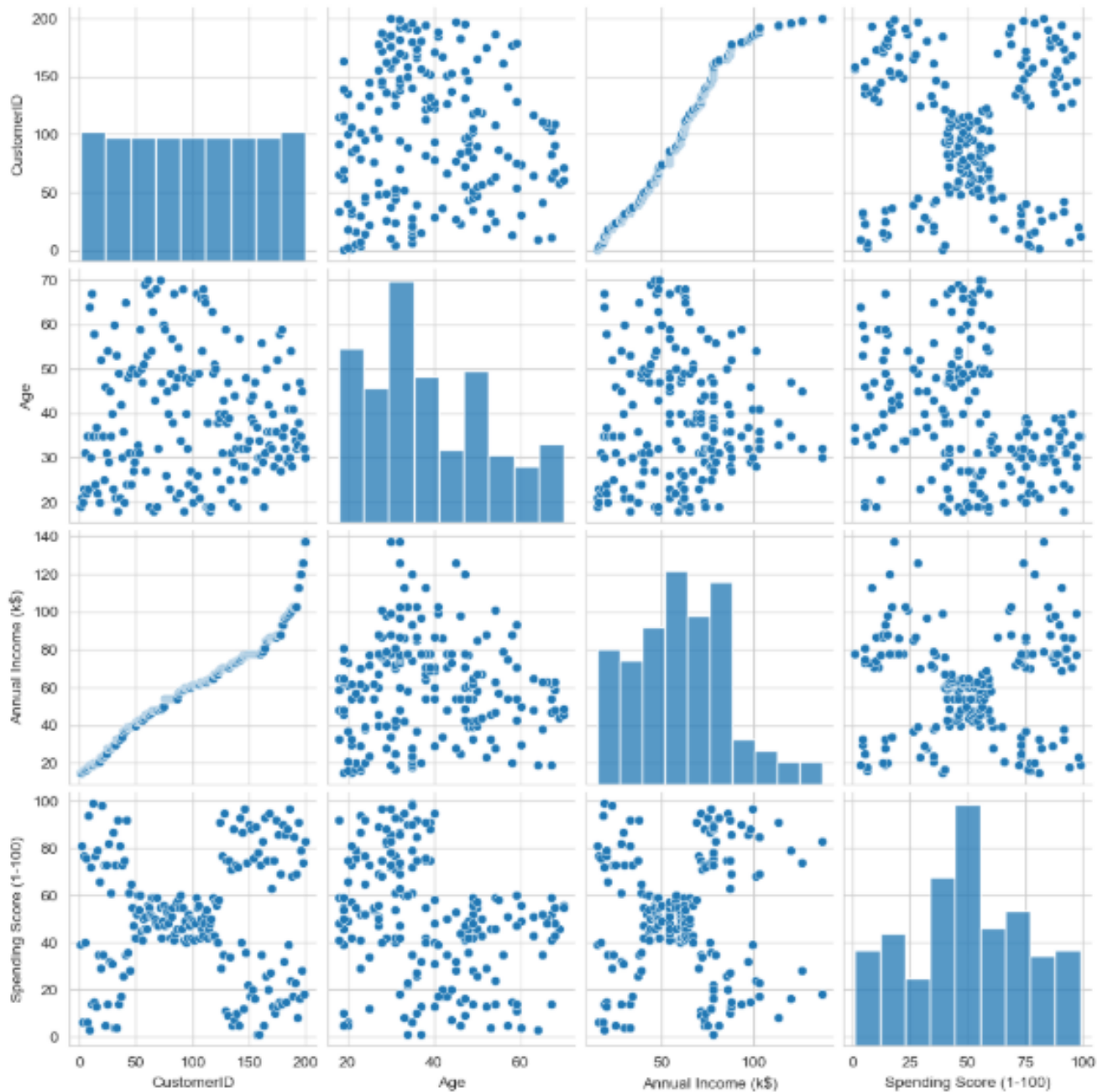


```
[18]: features=df.iloc[:,[3,4]].values
```

```
[19]: from sklearn.cluster import KMeans
      model=KMeans(n_clusters=5)
      model.fit(features)
      KMeans(n_clusters=5)
```
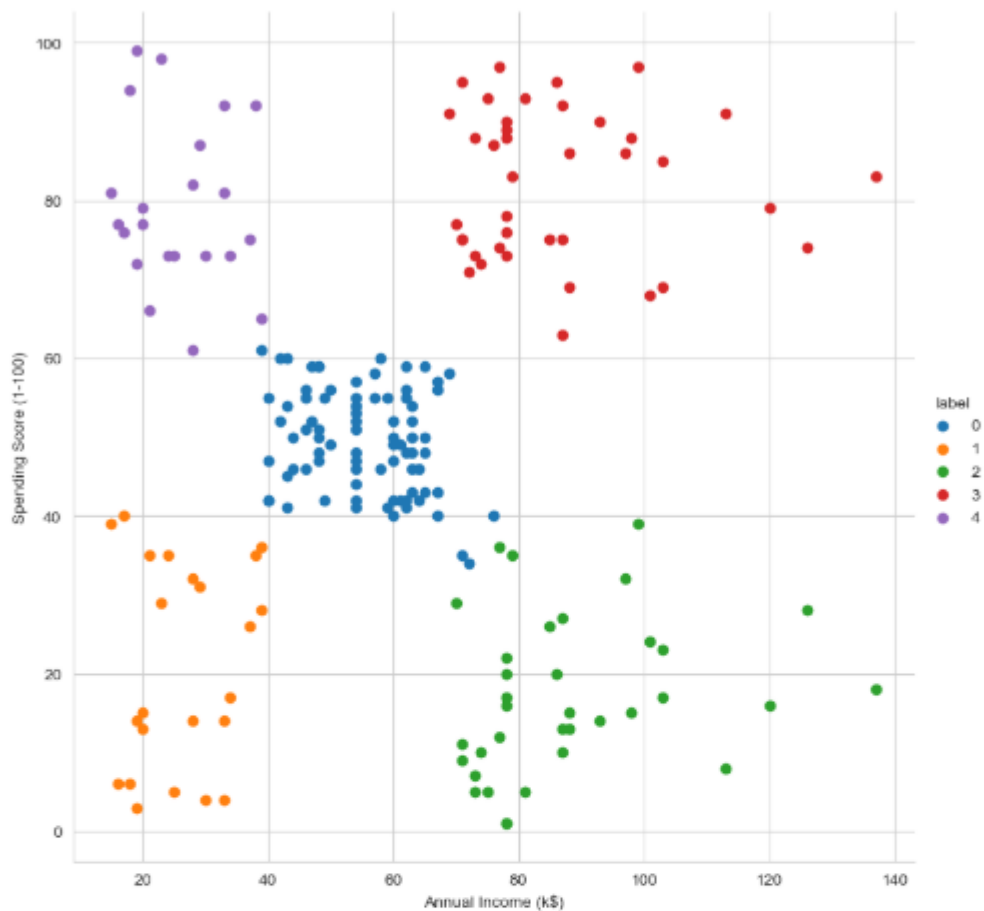
```
[19]: ▸   KMeans
      KMeans(n_clusters=5)
```

```
[20]: Final=df.iloc[:,[3,4]]
      Final['label']=model.predict(features)
      Final.head()
```

[20]:

	Annual Income (k$)	Spending Score (1-100)	label
0	15	39	1
1	15	81	4
2	16	6	1
3	16	77	4
4	17	40	1

```
[21]: sns.set_style("whitegrid")
      sns.FacetGrid(Final,hue="label",height=8) \
      .map(plt.scatter,"Annual Income (k$)", "Spending Score (1-100)") \
      .add_legend();
      plt.show()
```



```
[22]: features_el=df.iloc[:,[2,3,4]].values
      from sklearn.cluster import KMeans
      wcss=[]
      for i in range(1,10):
          model=KMeans(n_clusters=i)
          model.fit(features_el)
          wcss.append(model.inertia_)
      plt.plot(range(1,10),wcss)
      plt.show()
```
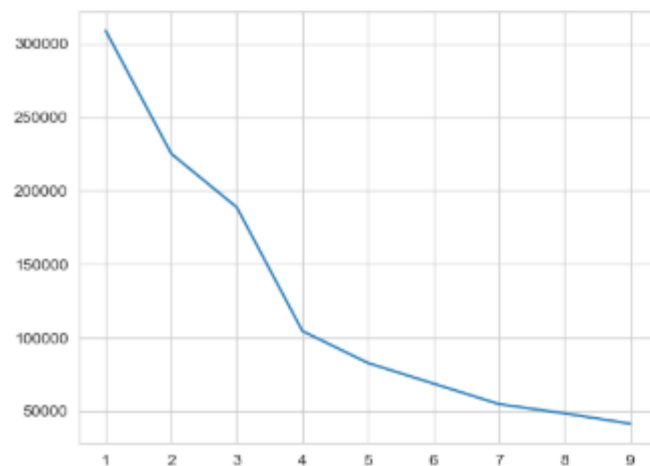

Result:

Thus, the Python program is executed successfully for detecting customer clusters from the given dataset using K-Means clustering.