EXPERIMENT NO: 10

Customer Segmentation using K-Means Clustering

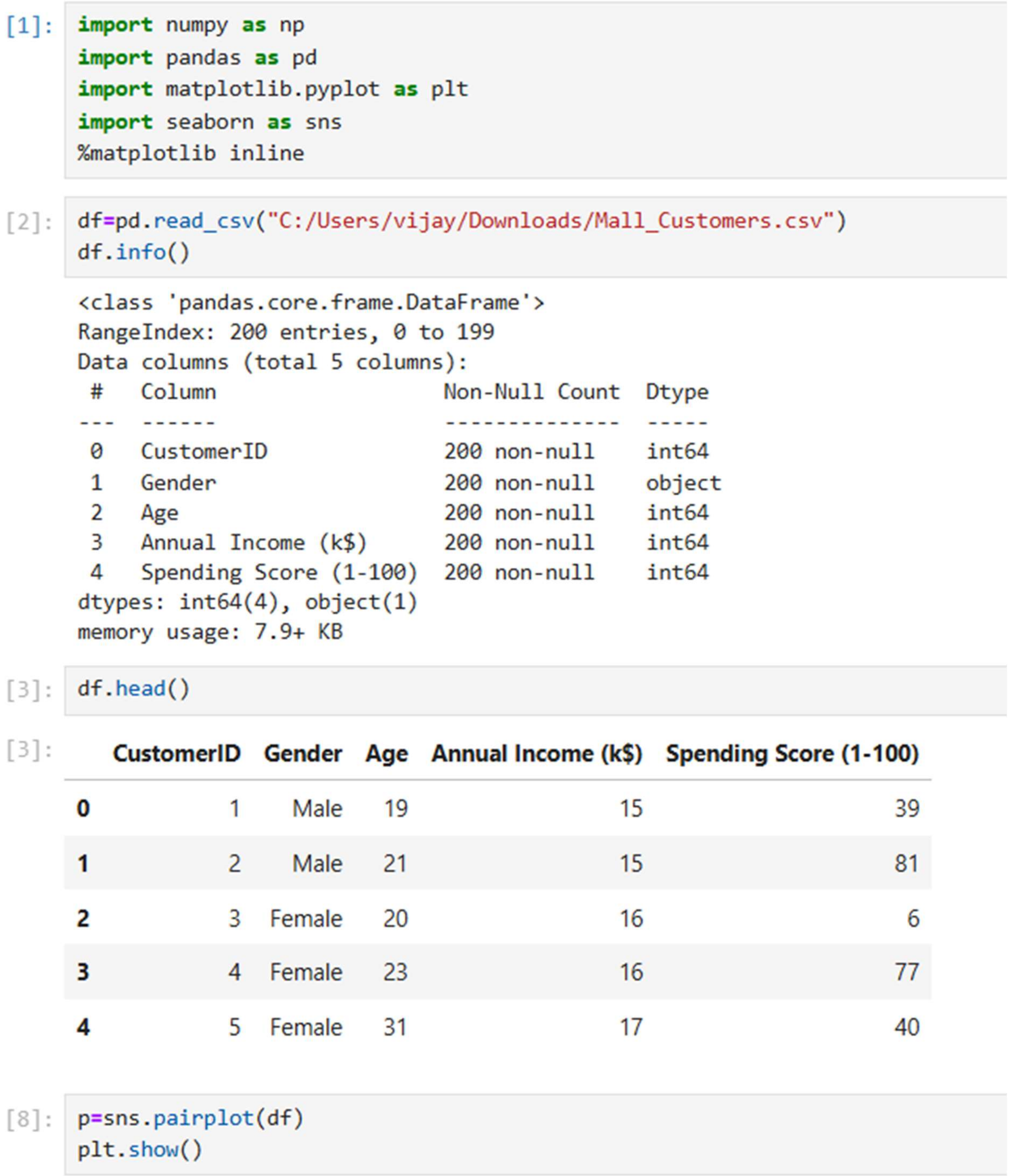
Aim:

To segment mall customers into distinct groups based on their annual income and spending score using the K-Means clustering algorithm.

Algorithm:

1. **Load Data: Import the Mall_Customers.csv dataset using pandas.**
2. **Visualize Data: Use pairplot() to observe relationships among features.**
3. **Select Features: Extract Annual Income (k$) and Spending Score (1-100) for clustering.**
4. **Apply K-Means:**
5. **Initialize K-Means with 5 clusters.**
6. **Fit the model and assign cluster labels to each customer.**
7. **Visualize Clusters:**
8. **Plot scatter points showing different clusters based on income and spending.**
9. **Find Optimal Clusters:**
10. **Calculate WCSS (Within-Cluster Sum of Squares) for cluster counts 1–9.**
11. **Plot the Elbow Curve to determine the best number of clusters.**

Program:



[1]:

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline
```

[2]:

```
df=pd.read_csv("C:/Users/vijay/Downloads/Mall_Customers.csv")
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
```

[3]:

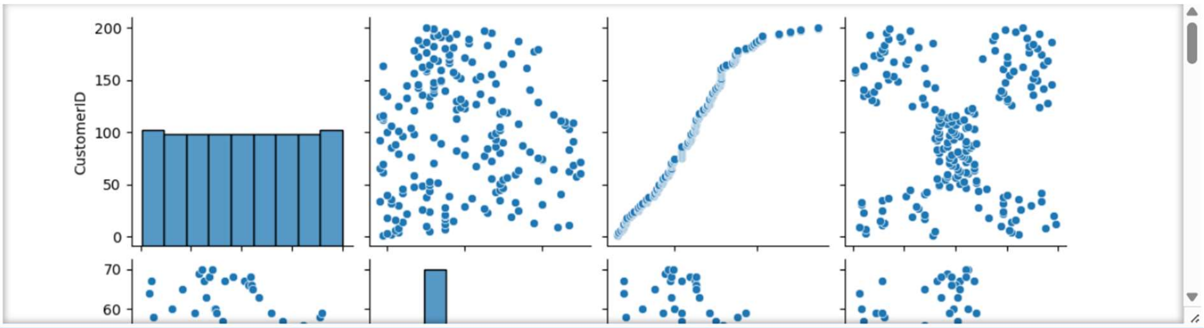
```
df.head()
```

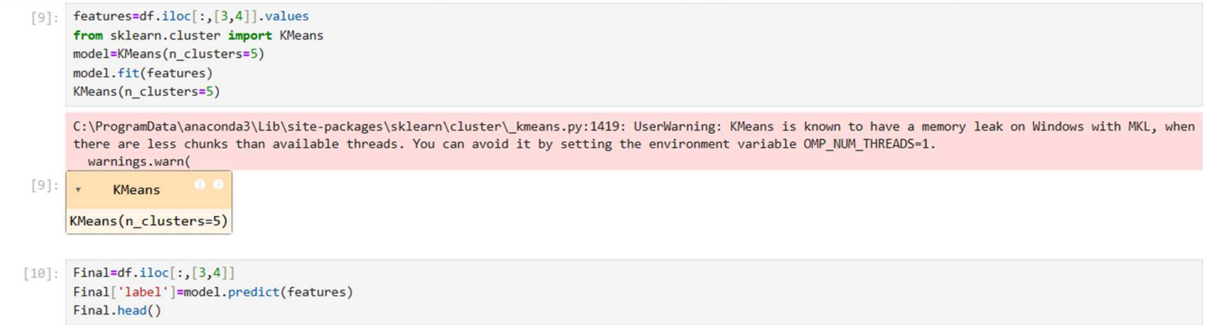
[3]:

	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

[8]:

```
p=sns.pairplot(df)
plt.show()
```





[9]:

```
features=df.iloc[:,[3,4]].values
from sklearn.cluster import KMeans
model=KMeans(n_clusters=5)
model.fit(features)
KMeans(n_clusters=5)
```

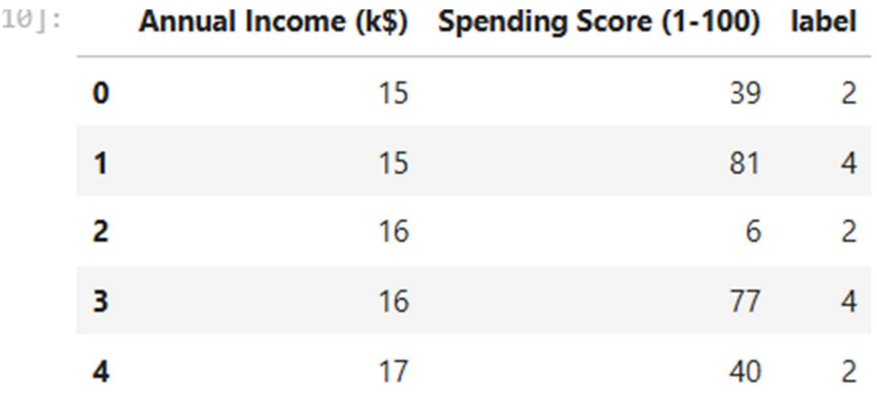
```
C:\ProgramData\anaconda3\Lib\site-packages\sklearn\cluster\_kmeans.py:1419: UserWarning: KMeans is known to have a memory leak on Windows with MKL, when
there are less chunks than available threads. You can avoid it by setting the environment variable OMP_NUM_THREADS=1.
  warnings.warn(
```

[9]:

```
▾ KMeans
KMeans(n_clusters=5)
```

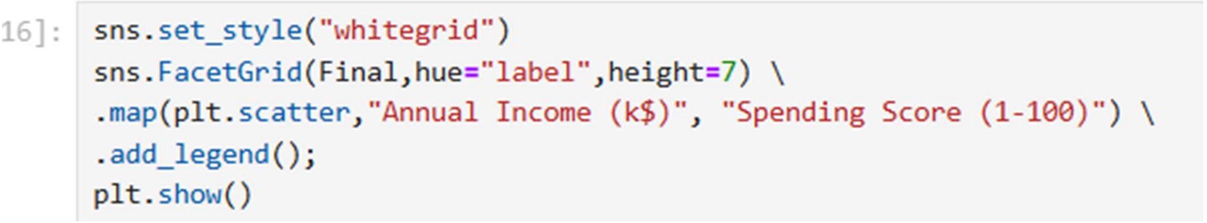
[10]:

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



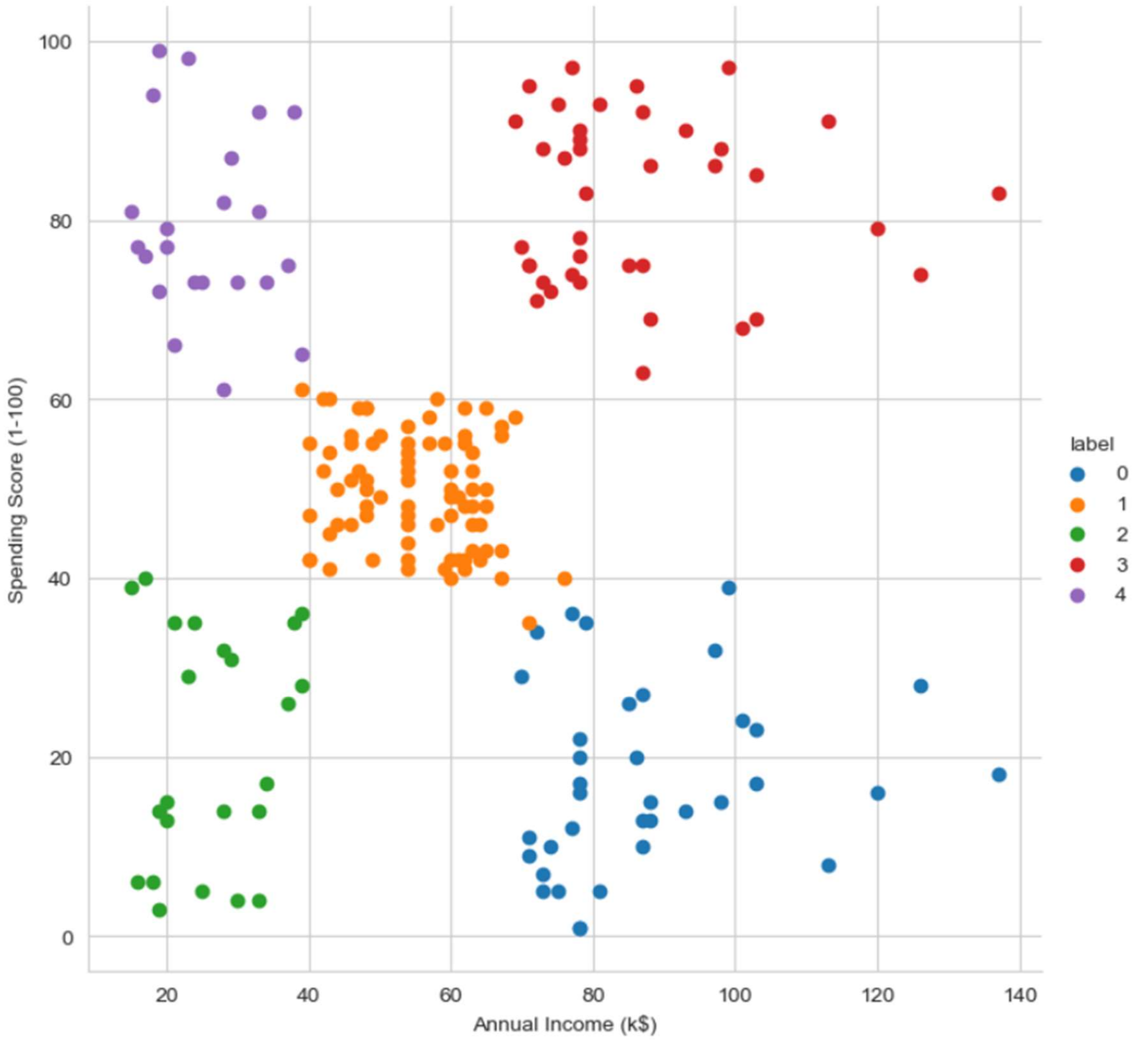
10]:

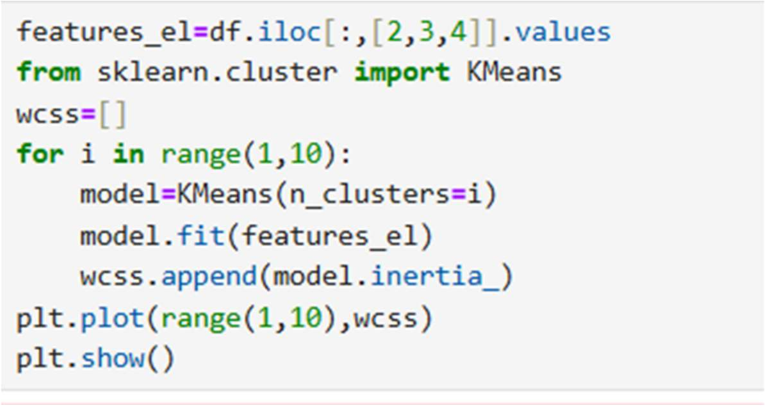
	Annual Income (k$)	Spending Score (1-100)	label
0	15	39	2
1	15	81	4
2	16	6	2
3	16	77	4
4	17	40	2



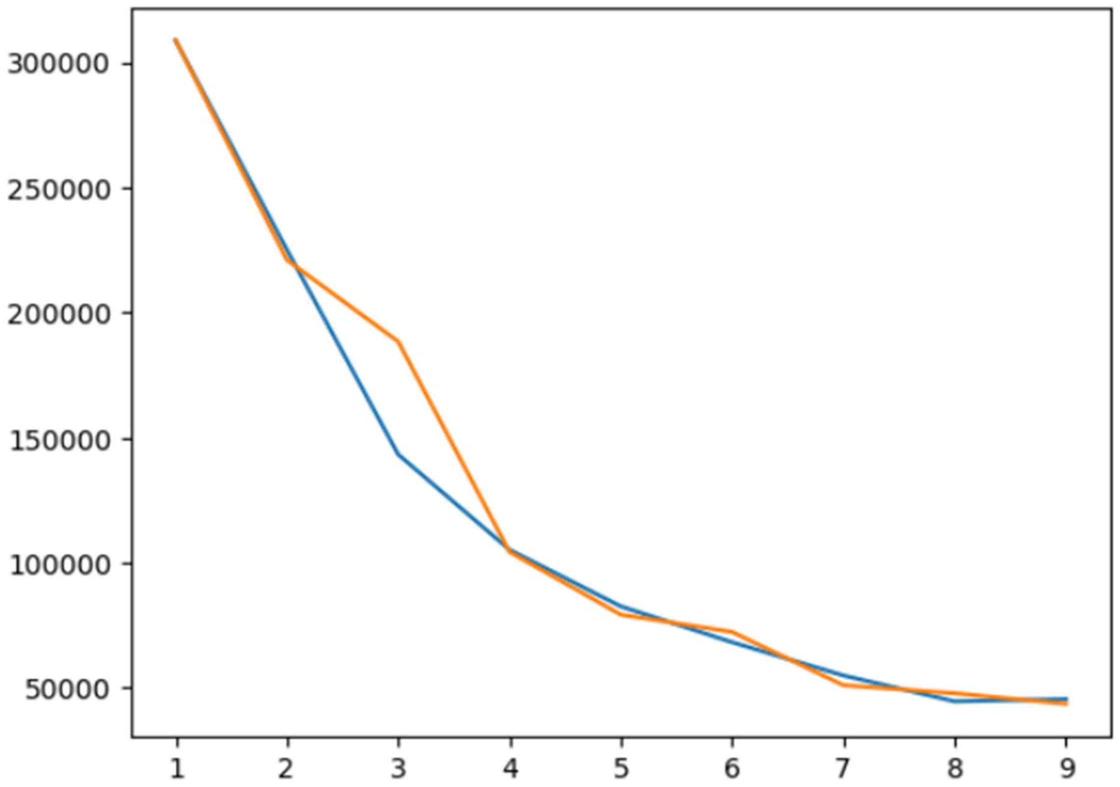
16]:

```
sns.set_style("whitegrid")
sns.FacetGrid(Final,hue="label",height=7) \
.map(plt.scatter,"Annual Income (k$)", "Spending Score (1-100)") \
.add_legend();
plt.show()
```





```
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:

The K-Means algorithm groups customers into 5 clusters based on their spending habits and income. The Elbow Method helps confirm that 5 clusters provide the most balanced segmentation, clearly showing different customer behavior groups.