CS-23334 FUNDAMENTALS OF DATA SCIENCE

ASHWIN C 240701058

EXPERIMENT 10 **Date: 02.10.2025**

10. Experiment to understand K-means clustering algorithm for a given data set.

Aim:

To conduct experiment to understand K-Means Clustering Algorithm for a given data set

Description:

Understand the K-Means Clustering algorithm for the dataset given.

Algorithm:

Step 1: Select Features and Preprocess the Data

Step 2: Choose the Number of Clusters (K)

Step 3: Apply the K-Means Algorithm and Fit the Model

Step 4: Visualize Clusters and Centroids

Step 5: Interpret Cluster Assignments and Evaluate Results

About Dataset:

This dataset contains customer demographic and behavioral data, including Customer ID, Gender, Age, Annual Income (in thousands), and a Spending Score from 1 to 100.

Code With Output:

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

df=pd.read_csv(r'D:\REC 2nd Year\Data Science\Data Sets\Mall
Customers.csv')

print(df.info())
```

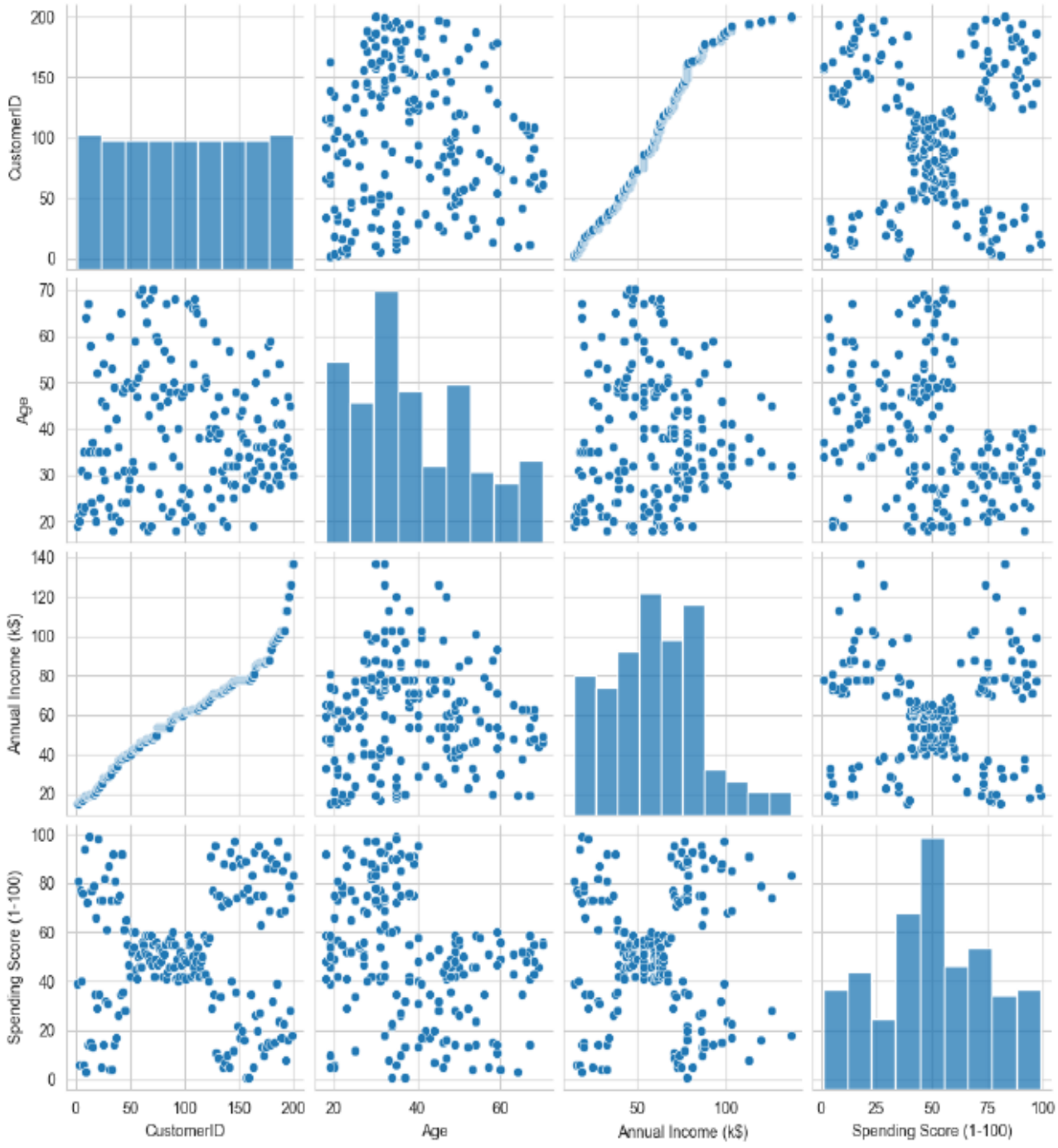
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 200 entries, 0 to 199
Data columns (total 5 columns):
 #   Column                  Non-Null Count  Dtype
---  ------                  --------------  -----
 0   CustomerID              200 non-null    int64
 1   Gender                  200 non-null    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
None
```

```
df.head()
```

	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

```
sns.pairplot(df)
```

```
<seaborn.axisgrid.PairGrid at 0x1dcec06ed50>
```

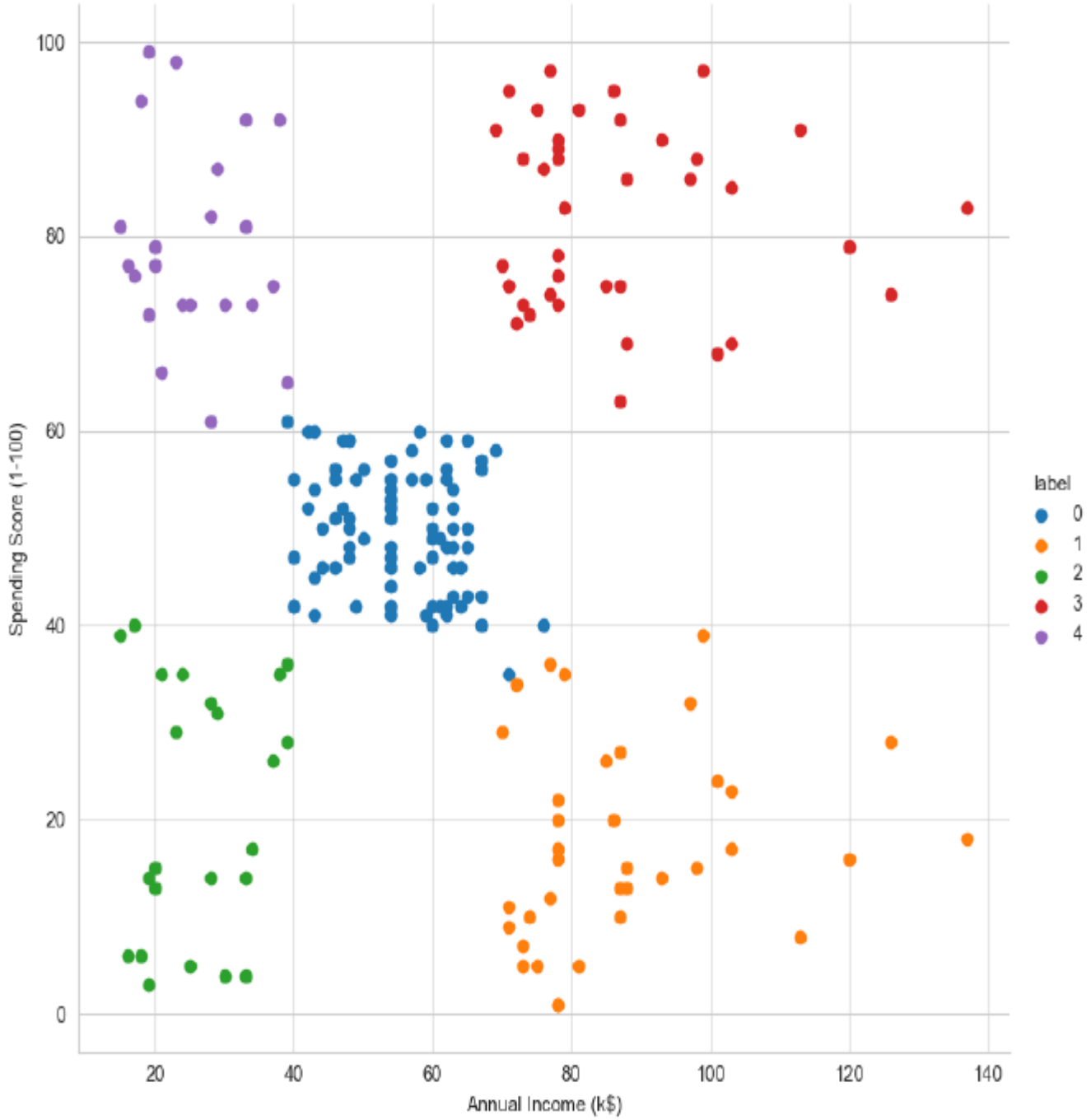

CustomerID
Age
Annual Income (k$)
Spending Score (1-100)
0
50
100
150
200
20
40
60
70
80
120
140
25
75

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

from sklearn.cluster import KMeans
model=KMeans(n clusters=5)
model.fit(features)
KMeans(n_clusters=5)

KMeans(n_clusters=5)

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

```
C:\Users\Abenanthan P\AppData\Local\Temp\
ipykernel 24940\470183701.py:2: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation:
https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#
returning-a-view-versus-a-copy
  Final['label']=model.predict(features)
```

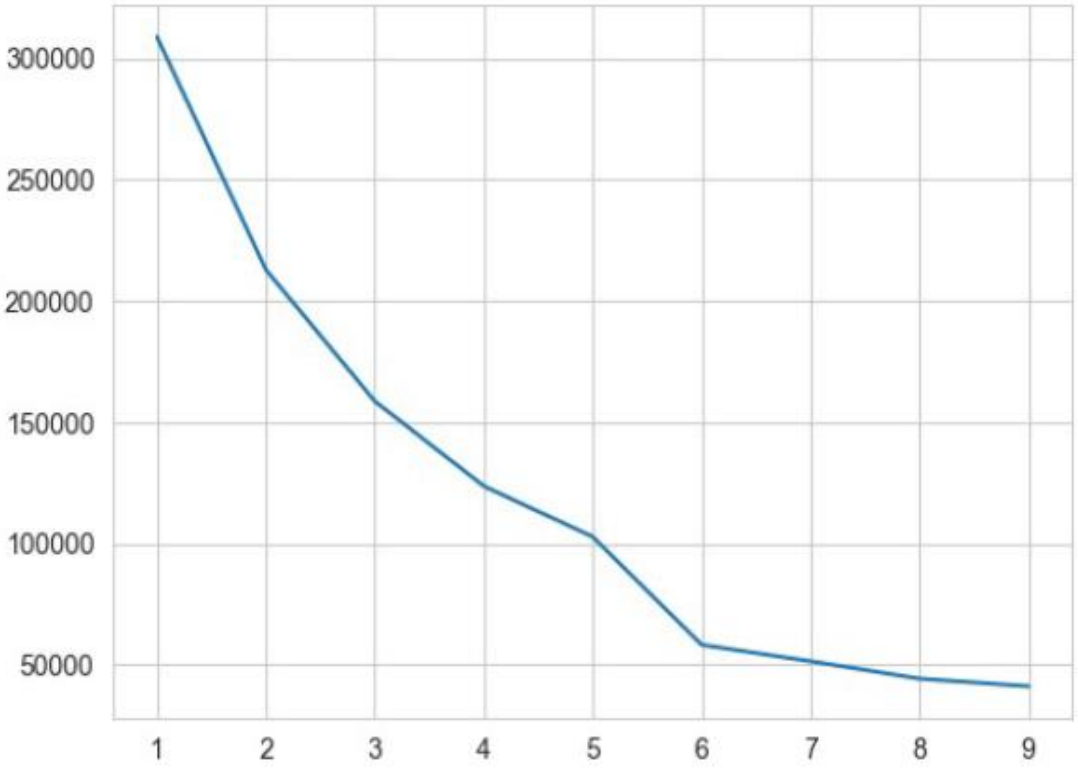
	Annual Income (k$)	Spending Score (1-100)	label
0	15	39	2
1	15	81	4
2	16	6	2
3	16	77	4
4	17	40	2

```
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()
```


Spending Score (1-100)
100
80
60
40
20
0
20
40
60
80
100
120
140
Annual Income (k$)
label
0
1
2
3
4

```
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)
```

```
[<matplotlib.lines.Line2D at 0x1dceef93750>]
```



Result:

Thus python program to understand K-Means Clustering algorithm for dataset is conducted successfully