

Name: Kelvin J Anil

Reg No: 21BCE0002

E-Mail: kelvin.janil2021@vitstudent.ac.in

AI_ML_Assignment 5:




Assignment 5

Market Basket Magic: Extracting Insights for Retail Success

Customer segmentation is a crucial aspect of retail and marketing strategy. Mall Customer Segmentation is a common data analysis project that involves categorizing mall customers into distinct groups or segments based on various characteristics and behaviors. This segmentation is valuable for tailoring marketing efforts, optimizing store layouts, and enhancing customer experiences.

Dataset link: [Here](#)

Task:

-  Understand the data
-  Data Preprocessing
-  Machine Learning approach with clustering algorithm

Colab Link:

<https://colab.research.google.com/drive/1MLw4BEm75Fu1kntyd-Aa58zpM1kYPSO5#scrollTo=Mlj8b7JMzXX6>

```
✓ [4] # import dependencies
2s import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.cluster import KMeans
```

```
✓ #uploading the dataset
0s df = pd.read_csv('/content/Mall_Customers.csv')
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

```
✓ [9] df.tail()
0s
```

	CustomerID	Gender	Age	Annual Income (k\$)	Spending Score (1-100)
195	196	Female	35	120	79
196	197	Female	45	126	28
197	198	Male	32	126	74
198	199	Male	32	137	18
199	200	Male	30	137	83

```
✓ [10] df.shape # finding the number of rows and cols
0s
(200, 5)
```

✓
0s

```
# information about the dataset  
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
```

✓
0s

```
[12] # checking for missing values in the dataset  
df.isnull().sum()
```

```
CustomerID      0  
Gender          0  
Age             0  
Annual Income (k$) 0  
Spending Score (1-100) 0  
dtype: int64
```

▼ there is no missing values in the dataset

✓
0s

```
[13] # we are clustering based on annual income and spending score  
x = df.iloc[:,[3,4]].values  
x
```

```
[ 76,  40],  
[ 76,  87],  
[ 77,  12],  
[ 77,  97],  
[ 77,  36],  
[ 77,  74],  
[ 78,  22],  
[ 78,  90],  
[ 78,  17],  
[ 78,  88],  
[ 78,  20],  
[ 78,  76],  
[ 78,  16],  
[ 78,  89],  
[ 78,  11]
```

```

# finding the number of clusters by using a loop to find the minimum wcss value
wcss=[]
for i in range(1,11):
    kmeans = KMeans(n_clusters=i, init='k-means++',random_state=42)
    kmeans.fit(X)
    wcss.append(kmeans.inertia_)

```

```

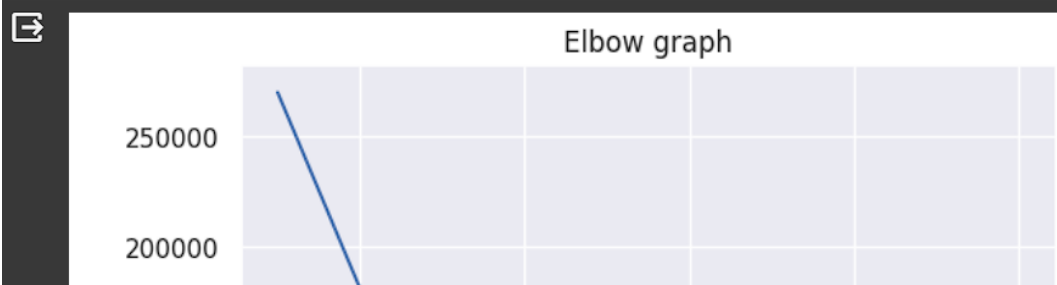
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870: FutureWarning: The default value of `n_init` will change from 10 to 'auto' in 1.4. See https://github.com/scikit-learn/scikit-learn/issues/25763 for more details.
warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870: FutureWarning: The default value of `n_init` will change from 10 to 'auto' in 1.4. See https://github.com/scikit-learn/scikit-learn/issues/25763 for more details.
warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870: FutureWarning: The default value of `n_init` will change from 10 to 'auto' in 1.4. See https://github.com/scikit-learn/scikit-learn/issues/25763 for more details.
warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870: FutureWarning: The default value of `n_init` will change from 10 to 'auto' in 1.4. See https://github.com/scikit-learn/scikit-learn/issues/25763 for more details.
warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870: FutureWarning: The default value of `n_init` will change from 10 to 'auto' in 1.4. See https://github.com/scikit-learn/scikit-learn/issues/25763 for more details.
warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870: FutureWarning: The default value of `n_init` will change from 10 to 'auto' in 1.4. See https://github.com/scikit-learn/scikit-learn/issues/25763 for more details.
warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870: FutureWarning: The default value of `n_init` will change from 10 to 'auto' in 1.4. See https://github.com/scikit-learn/scikit-learn/issues/25763 for more details.
warnings.warn(

```

```

# elbow graph
sns.set()
plt.plot(range(1,11),wcss)
plt.title('Elbow graph')
plt.xlabel('number of clusters')
plt.ylabel('WCSS')
plt.show()

```



Visualizing the clusters (0,1,2,3,4,5)

```
# plotting all the clusters
plt.figure(figsize= (10,5))
plt.scatter(X[Y==0,0],X[Y==0,1],s=50,color='green',label='Cluster1')
plt.scatter(X[Y==1,0],X[Y==1,1],s=50,color='blue',label='Cluster2')
plt.scatter(X[Y==2,0],X[Y==2,1],s=50,color='red',label='Cluster3')
plt.scatter(X[Y==3,0],X[Y==3,1],s=50,color='yellow',label='Cluster4')
plt.scatter(X[Y==4,0],X[Y==4,1],s=50,color='orange',label='Cluster5')

# plotting the centroids
plt.scatter(kmeans.cluster_centers_[0,0],kmeans.cluster_centers_[0,1],s=100,color='black',label='Centroids')

# Labelling
plt.title('Customer Segmentation')
plt.xlabel('Annual Income')
plt.ylabel('Spending Score')
plt.show()
```

Loading...

