Customer Segmentation using K-Means Clustering Algorithm

Problem Statement:- Based on Customer Shoping behaviour create different segments of customers to support designing business plans. Use K-Means Clustering Algorithm.

Solution: - In order to do the segmentation, there has to be some parameters which will play main role in analysis. In this case, Annual Spending and Spending Score is consdiered. These 2 features are used to create different clusters of customers. Clustering is done using K-Means Clustering ALgorithm.

In K-Means CLustering, following steps were taken:-

1. Initialized the number of Clusters to be considered. (For that optimum number of clusters selection must be used first), but in this case it is considered by default.
2. Randomly initialized the centroid points based on the data in 2 columns.
3. Calculated the minimum distance between the data points and the centroids by using Euclidean distance method.
4. Based on minimum distance, assigned the clusters to data points.
5. Re-Calculated the centroids, and applied steps 3 to 4 again on data points.
6. Matched the new clusters with previous clusters. If matched code breaks, if not Repeat Steps 3 to 5 untill the clusters do not change.

In [409…

```
# Importing Required Libraries and dataset
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from IPython.display import display
data1 = pd.read_csv("S:/Nitin/Projects/Project 3 - Customer Segment Analysis/Customers.csv")
data = data1.iloc[:,3:5]
display(data1.head())
display(data.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

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

In [427…

```
# Data Cleaning before proceeding to implementing K-Means Algorithm
data1['Gender']=data1['Gender'].apply(lambda x:x.strip())
Datacheck1 = data1.isnull().sum()
Datacheck2 = data1.isna().sum()
Datacheck3 = data.dtypes
print("Datatype check for the features")
display(Datacheck3)
print(".....................")
print("NULL values stats")
display(Datacheck1)
print(".....................")
print("NA values stats")
display(Datacheck2)
```

```
Datatype check for the features
Annual Income (k$)        int64
Spending Score (1-100)    int64
dtype: object
.....................
NULL values stats
CustomerID                0
Gender                    0
Age                       0
Annual Income (k$)        0
Spending Score (1-100)    0
dtype: int64
.....................
NA values stats
CustomerID                0
Gender                    0
Age                       0
Annual Income (k$)        0
Spending Score (1-100)    0
dtype: int64
```

In [411…

```
# Initialize k value
k = 5
q=min(data['Annual Income (k$)'])
w=max(data['Annual Income (k$)'])
e=min(data['Spending Score (1-100)'])
r=max(data['Spending Score (1-100)'])
C1=np.array(np.random.randint(q,w,(k,1)))
C2=np.array(np.random.randint(e,r,(k,1)))
centroids = np.concatenate((C1,C2),axis=1)
print(centroids)
```

```
[[ 32  81]
 [ 22   4]
 [ 68  76]
 [110  13]
 [ 31  52]]
```

In [414…

```
X = data.shape
row_number = X[0]
X
row_number
check = 0
iteration = 0
All_Clusters = np.full((row_number,40),0)

while check==0:
    #print(".....................")
    print("Iteration = ",iteration)
```

```
# Distance calculation
    distance = list()
    for i in range(0,row_number):
        for j in range(0,k):
            a=data.iloc[i,:].values
            b=centroids[j]
            c=(a-b)**2
            d=np.sqrt(sum(c))
            distance.append(d)
    #print(distance)
    Distance_array = np.array(distance).reshape(row_number,k)
    Distance_DF = pd.DataFrame(Distance_array)
    e=pd.concat([data,Distance_DF],axis=1)
    #print(e)


#Assigning Cluster to the Data Points
    Cluster=list()
    for i in range(0,row_number):
        f=e.iloc[i,-k::1]
        f=list(f)
        f=f.index(min(f))
        Cluster.append(f)
    #print(Cluster)
    Cluster_DF = pd.DataFrame(np.array(Cluster))
    Cluster_DF.columns = ['Clusters']
    #print(Cluster_DF)
    All_Clusters[:,iteration] = Cluster
    #print(All_Clusters)

# Updating Centroids
    g = pd.concat([data,Cluster_DF],axis=1)
    h=g.groupby(by='Clusters').mean()
    print("New Centroids")
    display(h)
    print(".....................")
    p=h.shape

# Check Point to see if need to proceed further
    if p[0]==1:
        check=1
    else:
        if iteration==0:
            pass
        else:
            l=(All_Clusters[:,iteration]==All_Clusters[:,iteration-1]).all()
            if(l == False):
                centroids=h.to_numpy()
            else:
                check=1
    iteration = iteration+1

All_Clusters = pd.DataFrame(All_Clusters)
All_Clusters.drop(All_Clusters.iloc[:,iteration:40],axis=1,inplace=True)
Final_cluster = pd.DataFrame(All_Clusters.iloc[:,-1])
Final_cluster.columns = ['Cluster']
Final_data = pd.concat([data1,Final_cluster],axis=1)
display(Final_data.head())
```

```
Iteration =  0
New Centroids
```

Clusters	Annual Income (k$)	Spending Score (1-100)
0	25.157895	81.789474
1	25.538462	10.846154
2	76.485714	67.857143
3	88.200000	17.114286
4	45.412698	47.555556

```
.....................
Iteration =  1
New Centroids
```

Clusters	Annual Income (k$)	Spending Score (1-100)
0	25.157895	81.789474
1	25.538462	10.846154
2	76.485714	67.857143
3	88.200000	17.114286
4	45.412698	47.555556

```
.....................
```

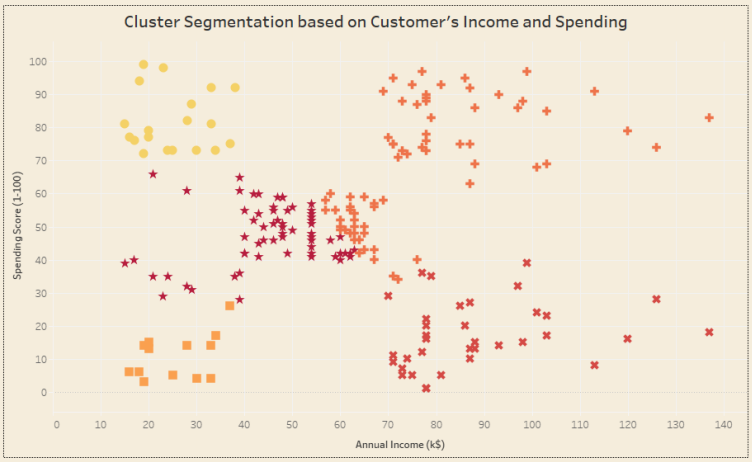
	CustomerID	Gender	Age	Annual Income (k$)	Spending Score (1-100)	Cluster
0	1	Male	19	15	39	4
1	2	Male	21	15	81	0
2	3	Female	20	16	6	1
3	4	Female	23	16	77	0
4	5	Female	31	17	40	4

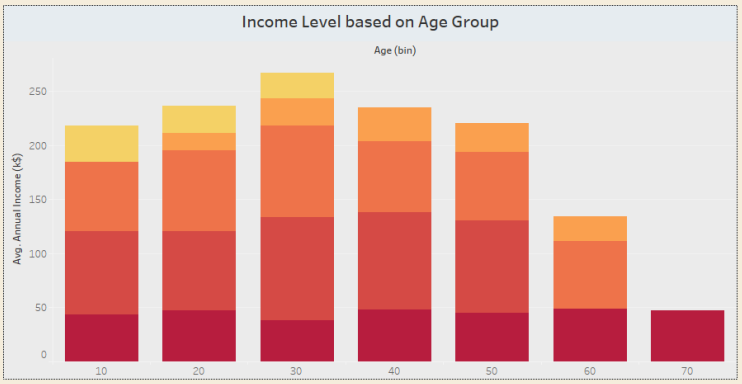
In [428…

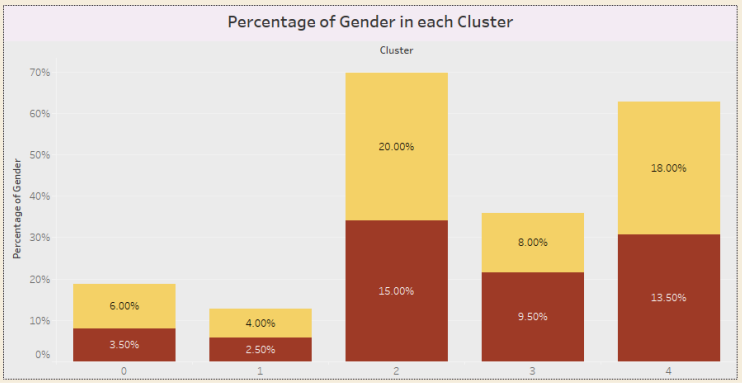
```
# Export data to .csv format to create dashboard in TABLEAU
Final_data.to_csv("S:/Nitin/Projects/Project 3 - Customer Segment Analysis/Output.csv")
```

In [ ]:

Customer Segmentation Analysis







Important Information

Cluster 0 - Customers with ***Low Income - High Spending***
Cluster 1 - Customers with ***Low Income - Low Spending***
Cluster 2 - Customers with ***Medium to High Income - Medium to High Spending***
Cluster 3 - Customers with ***High Income - Low Spending***
Cluster 4 - Customers with ***Low to Medium Income - Medium Spending***

* **Major Contributer in Sales** are the **Females** in all the Cluster, making it **56%** in the database.
* **Cluster 2 is the biggest** with majority of customers. **35% of all the customers** fall in this segment with **20% Females** and **15% Males.**
* Overall **Age Group of 30-40** has the **highest average income.**

Analysis suggests that, company should **focus** on customers who have **Medium to High Income** and are in **age group of 30-50**. **Specific attention** can be given to **Female customers** to increase the sales.