Customer Segmentation using K-means Clustering Algorithm

Md.Miftahul Alam
Department of Computer Science (CS,AIUB)
American International University-Bangladesh (AIUB)
Dhaka, Bangladesh
18-38839-3@student.aiub.edu

Md.Masfikur Rahman Fahim
Department of Computer Science (CS,AIUB)
American International University-Bangladesh (AIUB)
Dhaka, Bangladesh
19-39455-1@student.aiub.edu

Sanzida Tasnim
Department of Computer Science (CS,AIUB)
American International University-Bangladesh (AIUB)
Dhaka, Bangladesh
18-38867-3@student.aiub.edu

Tasnuba Kamal
Department of Computer Science (CS,AIUB)
American International University-Bangladesh (AIUB)
Dhaka, Bangladesh
19-39741-3@student.aiub.edu

***Abstract*—The zeitgeist of the modern era is innovation, where everyone is embroiled into competition to be better than others. This is where machine learning comes into play, various algorithms are applied for unravelling the hidden patterns in the data for better decision making for the future. Any business that wants to make money needs to make wise decisions. Nowadays, there is fierce competition, and every company is pursuing its own strategic plan. We should analyze the evidence and make an informed judgment. Every individual is unique, and we have no way of knowing what he or she purchases or enjoys. However, by applying many algorithms to the dataset, one may filter out the data and discover the target group using machine learning techniques. The process of segmenting the customers with similar behavior into the same segment and with different patterns into different segments is called customer segmentation. In this paper, K-means clustering algorithm have been implemented to segment the customers and finally compare the results of clusters obtained from the algorithm.**

Keywords—Clustering, K-Means Algorithm, Customer Segmentation, Visualization.

I. Introduction

Nowadays, the competition is fierce, and a variety of technologies must be considered in order to achieve successful growth and income development. Data is the most critical component of every business. We can execute certain operations to determine client interests using grouped or ungrouped data.

However, we may not be aware of the true beneficiaries over the entire dataset. Customer segmentation is a technique for dividing a large dataset into various groups based on age, demographics, spending, income, gender, and other factors. Clusters are another name for these groups. We can learn which products have a large number of sales and which age groups are purchasing as a result of this. We can also deliver that good— in a far more cost-effective manner, resulting in increased revenue.

We're going to start with the old data. We know that old is gold, therefore we'll use ancient data to apply the K-means clustering technique, and we'll have to figure out how many clusters there are first. One can easily find the potential group of data while observing that visualization.

The goal of this paper is to identify customer segments using the unsupervised machine learning, using the partitioning algorithm called as K-means clustering algorithm.

II. Proposed System

A. *Existing System*

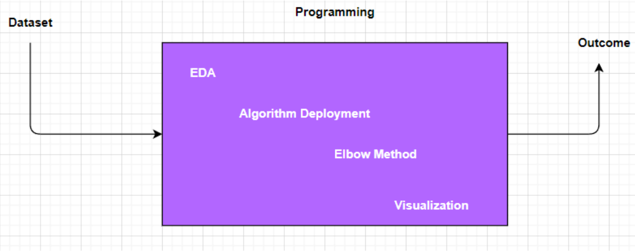
The existing method is storing customer data through paperwork and computer software (digital data) is increasing day by day. At end of the day, they will analyze their data as how many things are sold or actual customer count etc. By analyzing the collected data, they got to know who is beneficial to their business and increase their sales. It requires more time and more paperwork. Also, it is not much effective solution to find the desired customers data.

B. *Proposed Method*

This new way will be critical in overcoming the traditional method of paper work and computerized digital data. Day by day, we collect a large amount of data, which necessitates more paperwork and time. In today's world, new technologies are being developed. Machine Learning is a significant breakthrough that uses a variety of algorithms to predict the ultimate outcome. As a result, for our issue statement, we will use K-Means Clustering, which divides data into groups based on similar qualities. The data will then be visualized.

C. *System Architecture*

We'll look at the dataset first, then perform exploratory data analysis to look for missing data, duplicate values, and null values. Then we'll use our k-means clustering approach, which is unsupervised learning in machine learning. In which distance is calculated using randomly chosen centers and repeated until no change in cluster centers occurs. After that, we'll use data visualization to analyze the data. Finally, we shall have a result.



III. METHODOLOGY

a. First, we'll import all of the required libraries or modules (pandas, NumPy, seaborn).
b. Next, we'll examine the dataset to see if there are any null values, missing values, or duplicate values. So, we'll rectify them by removing or fixing the value with its means, medians, and so on, which is called Data Preprocessing in technical terms.
c. We'll use our model approach, K-Means Clustering, to partition the data into clusters based on shared properties. We'll utilize the elbow approach to figure out how many clusters there are.
d. Finally, we will use Matplotlib to show our data, which will result in customers being separated into groups that are similar to one another.

A. *Overview of K-Mean Clustering Algorithm*

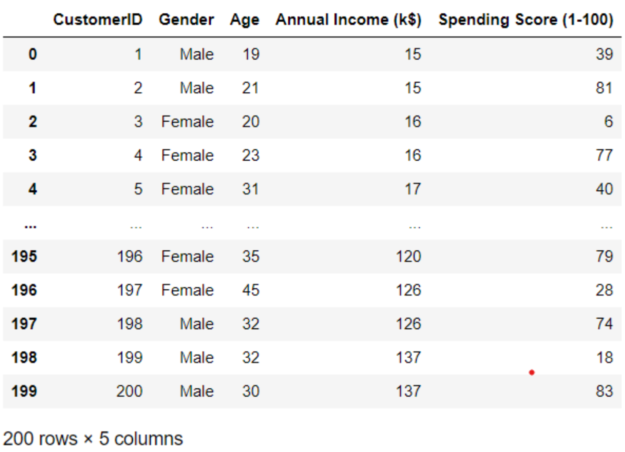
a. K Means algorithm in an iterative algorithm that tries to partition the dataset into K predefined distinct non overlapping sub groups which are called as cluster.
b. Here K is the total no of clusters.
c. Every point belongs to only one cluster.
d. Clusters cannot overlap

B. *Steps of Algorithm*

a. Choose k items at random from D to serve as the first cluster centers.
b. Repeat. Based on the mean value of the objects in the cluster, assign each object to the cluster to which it is most comparable.
c. Until there is no change, update the cluster means, i.e., calculate the mean value of the items for each cluster.

C. *Overview of Dataset*

We are using widely available dataset named “Mall Customers.csv” dataset. This is a mall customer segmentation data which contains 5 columns and 200 rows.

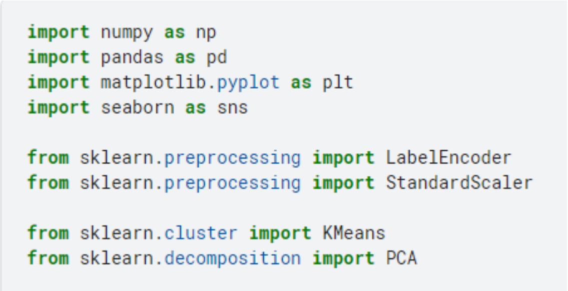


	CustomerID	Gender	Age	Annual Income (k$)	Spending Score (1-100)
0	1	Male	19	15	39
1	2	Male	21	15	81
2	3	Female	20	16	6
3	4	Female	23	16	77
4	5	Female	31	17	40
...	...	...	...	...	...
195	196	Female	35	120	79
196	197	Female	45	126	28
197	198	Male	32	126	74
198	199	Male	32	137	18
199	200	Male	30	137	83

200 rows × 5 columns

D. *Data Preprocessing*

- Importing Libraries



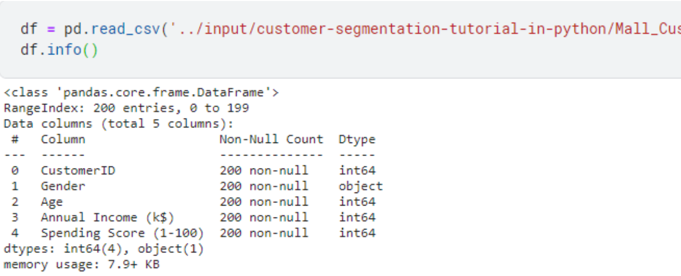
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

from sklearn.preprocessing import LabelEncoder
from sklearn.preprocessing import StandardScaler

from sklearn.cluster import KMeans
from sklearn.decomposition import PCA
```

- Data Information and Dataset loading

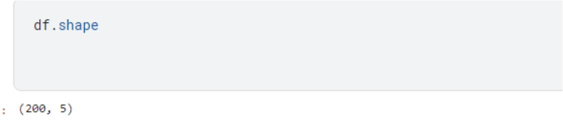
No null values found



```
df = pd.read_csv('../input/customer-segmentation-tutorial-in-python/Mall_Cus
df.info()
```

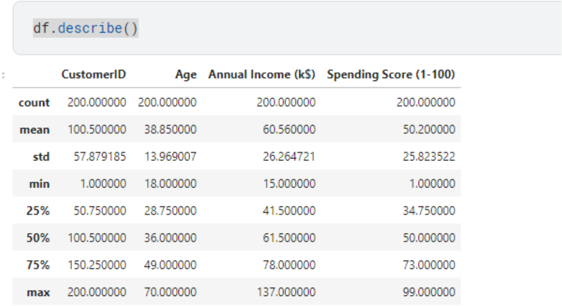
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 200 entries, 0 to 199
Data columns (total 5 columns):
 #   Column                  Non-Null Count  Dtype 
---  ------                  --------------  ----- 
 0   CustomerID              200 non-null    int64 
 1   Gender                  200 non-null    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
```

- Describing and understanding shape of Dataset



```
df.shape
```

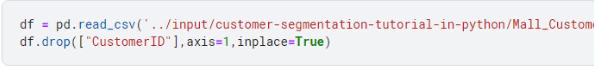
```
: (200, 5)
```



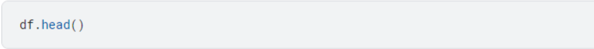
```
df.describe()
```

	CustomerID	Age	Annual Income (k$)	Spending Score (1-100)
count	200.000000	200.000000	200.000000	200.000000
mean	100.500000	38.850000	60.560000	50.200000
std	57.879185	13.969007	26.264721	25.823522
min	1.000000	18.000000	15.000000	1.000000
25%	50.750000	28.750000	41.500000	34.750000
50%	100.500000	36.000000	61.500000	50.000000
75%	150.250000	49.000000	78.000000	73.000000
max	200.000000	70.000000	137.000000	99.000000

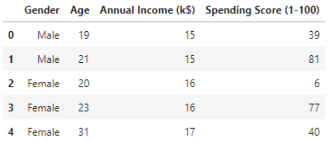
- Dropping unnecessary data like Customer ID.



```
df = pd.read_csv('../input/customer-segmentation-tutorial-in-python/Mall_Custom
df.drop(["CustomerID"],axis=1,inplace=True)
```

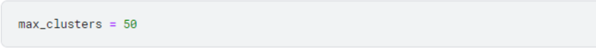


```
df.head()
```

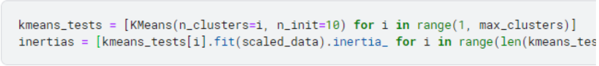


	Gender	Age	Annual Income (k$)	Spending Score (1-100)
0	Male	19	15	39
1	Male	21	15	81
2	Female	20	16	6
3	Female	23	16	77
4	Female	31	17	40

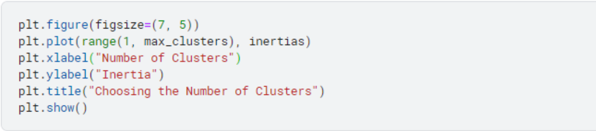
- Cluster selection and choosing number of clusters



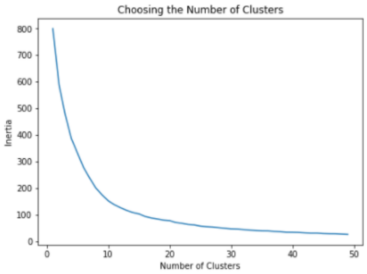
```
max_clusters = 50
```



```
kmeans_tests = [KMeans(n_clusters=i, n_init=10) for i in range(1, max_clusters)]
inertias = [kmeans_tests[i].fit(scaled_data).inertia_ for i in range(len(kmeans_tes
```



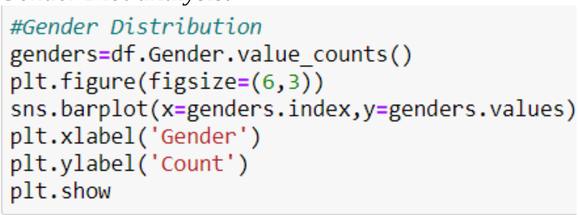
```
plt.figure(figsize=(7, 5))
plt.plot(range(1, max_clusters), inertias)
plt.xlabel("Number of Clusters")
plt.ylabel("Inertia")
plt.title("Choosing the Number of Clusters")
plt.show()
```



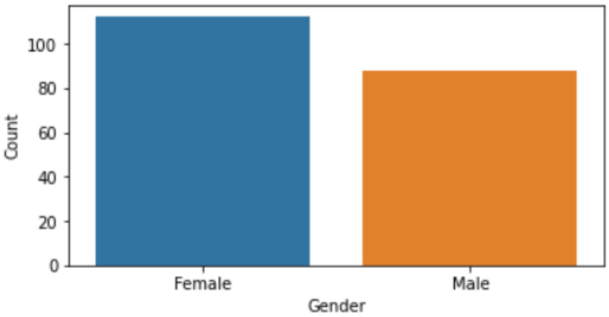
IV. DATA ANALYSIS AND THEIR OUTCOMES

A. *Data analysis*

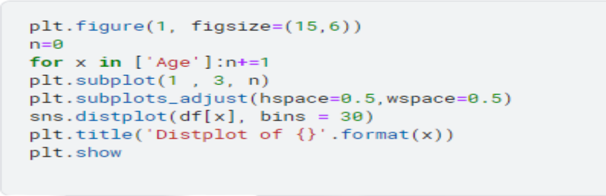
- *Gender Plot analysis:*



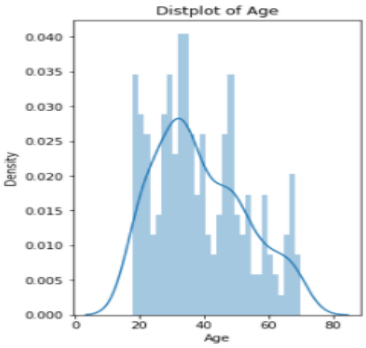
```
#Gender Distribution
genders=df.Gender.value_counts()
plt.figure(figsize=(6,3))
sns.barplot(x=genders.index,y=genders.values)
plt.xlabel('Gender')
plt.ylabel('Count')
plt.show
```



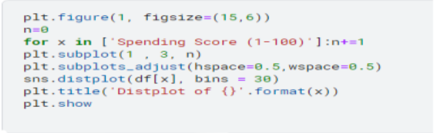
- *Age Plot analysis*



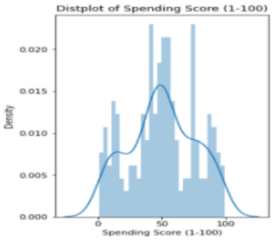
```
plt.figure(1, figsize=(15,6))
n=0
for x in ['Age']:n+=1
plt.subplot(1 , 3, n)
plt.subplots_adjust(hspace=0.5,wspace=0.5)
sns.distplot(df[x], bins = 30)
plt.title('Distplot of {}'.format(x))
plt.show
```



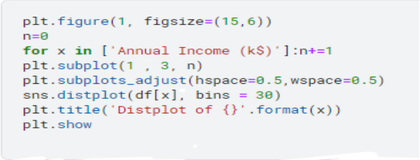
- *Spending Score Plot analysis*



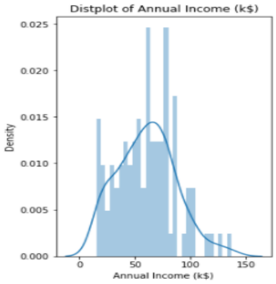
```
plt.figure(1, figsize=(15,6))
n=0
for x in ['Spending Score (1-100)']:n+=1
plt.subplot(1 , 3, n)
plt.subplots_adjust(hspace=0.5,wspace=0.5)
sns.distplot(df[x], bins = 30)
plt.title('Distplot of {}'.format(x))
plt.show
```



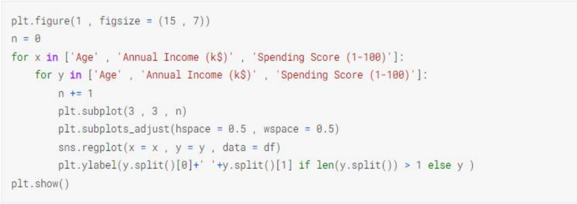
- *Annual Income Plot analysis*



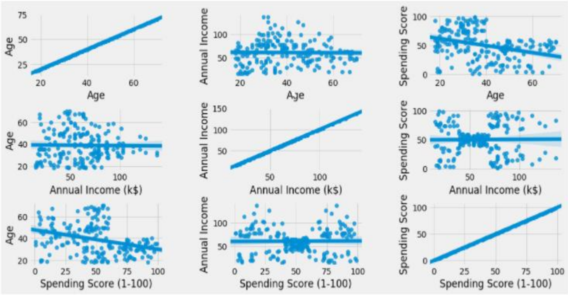
```
plt.figure(1, figsize=(15,6))
n=0
for x in ['Annual Income (k$)']:n+=1
plt.subplot(1 , 3, n)
plt.subplots_adjust(hspace=0.5,wspace=0.5)
sns.distplot(df[x], bins = 30)
plt.title('Distplot of {}'.format(x))
plt.show
```



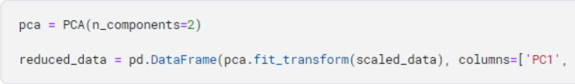
- *Plotting relation between age ,gender, annual income and*



```
plt.figure(1 , figsize = (15 , 7))
n = 0
for x in ['Age' , 'Annual Income (k$)' , 'Spending Score (1-100)']:
    for y in ['Age' , 'Annual Income (k$)' , 'Spending Score (1-100)']:
        n += 1
        plt.subplot(3 , 3 , n)
        plt.subplots_adjust(hspace = 0.5 , wspace = 0.5)
        sns.regplot(x = x , y = y , data = df)
        plt.ylabel(y.split()[0]+' '+y.split()[1] if len(y.split()) > 1 else y )
plt.show()
```

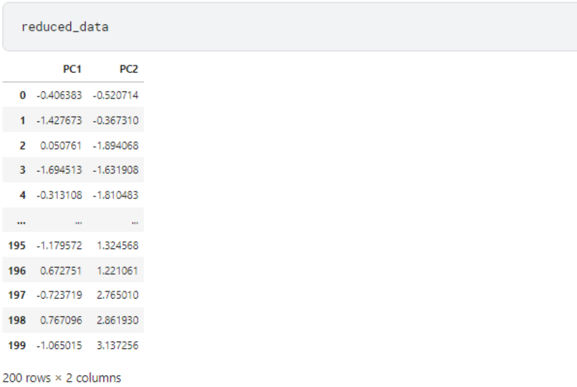


- *Principal Component Analysis*



```
pca = PCA(n_components=2)

reduced_data = pd.DataFrame(pca.fit_transform(scaled_data), columns=['PC1',
```

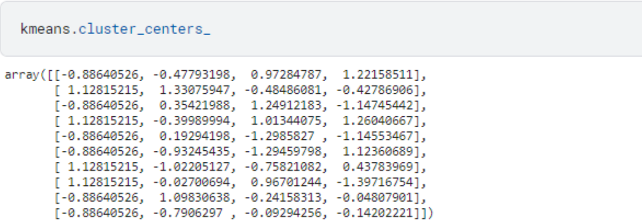


```
reduced_data
```

	PC1	PC2
0	-0.406383	-0.520714
1	-1.427673	-0.367310
2	0.050761	-1.894068
3	-1.694513	-1.631908
4	-0.313108	-1.810483
...	...	...
195	-1.179572	1.324568
196	0.672751	1.221061
197	-0.723719	2.765010
198	0.767096	2.861930
199	-1.065015	3.137256

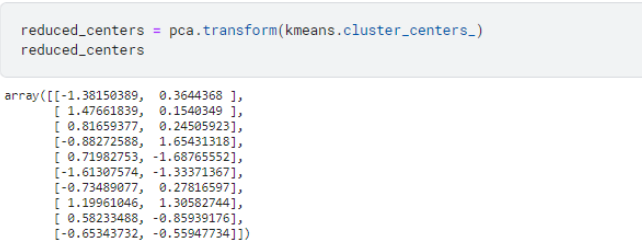
200 rows × 2 columns

- *K-means cluster centers and reduced data centers*



```
kmeans.cluster_centers_
```

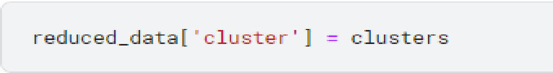
```
array([[-0.88640526, -0.47793198,  0.97284787,  1.22158511],
       [ 1.12815215,  1.33075947, -0.48486081, -0.42786906],
       [-0.88640526,  0.35421988,  1.24912183, -1.14745442],
       [ 1.12815215, -0.39989994,  1.01344075,  1.26040667],
       [-0.88640526,  0.19294198, -1.2985827 , -1.14553467],
       [-0.88640526, -0.93245435, -1.29459798,  1.12360689],
       [ 1.12815215, -1.02205127, -0.75821082,  0.43783969],
       [ 1.12815215, -0.02700694,  0.96701244, -1.39716754],
       [-0.88640526,  1.09830638, -0.24158313, -0.04807901],
       [-0.88640526, -0.7906297 , -0.09294256, -0.14202221]])
```



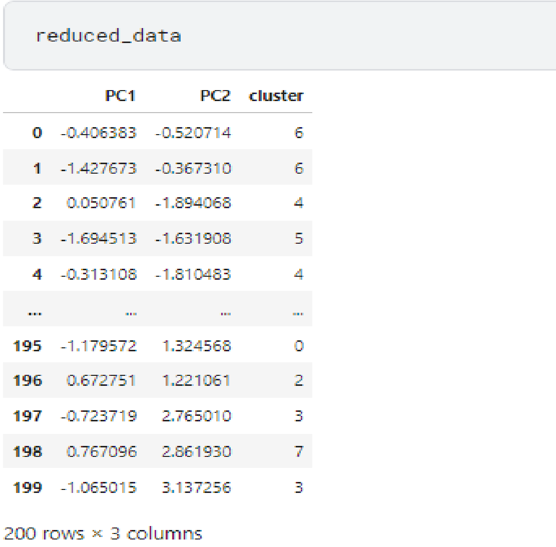
```
reduced_centers = pca.transform(kmeans.cluster_centers_)
reduced_centers
```

```
array([[-1.38150389,  0.3644368 ],
       [ 1.47661839,  0.1540349 ],
       [ 0.81659377,  0.24505923],
       [-0.88272588,  1.65431318],
       [ 0.71982753, -1.68765552],
       [-1.61307574, -1.33371367],
       [-0.73489077,  0.27816597],
       [ 1.19961046,  1.30582744],
       [ 0.58233488, -0.85939176],
       [-0.65343732, -0.55947734]])
```

- *Reduced data and their clusters*



```
reduced_data['cluster'] = clusters
```

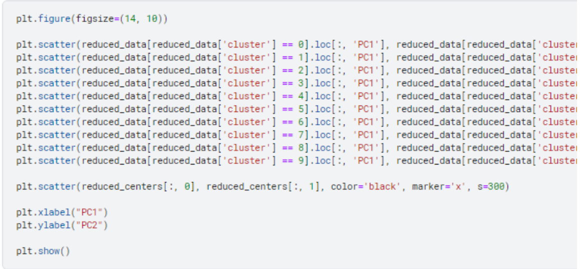


```
reduced_data
```

	PC1	PC2	cluster
0	-0.406383	-0.520714	6
1	-1.427673	-0.367310	6
2	0.050761	-1.894068	4
3	-1.694513	-1.631908	5
4	-0.313108	-1.810483	4
...	...	...	...
195	-1.179572	1.324568	0
196	0.672751	1.221061	2
197	-0.723719	2.765010	3
198	0.767096	2.861930	7
199	-1.065015	3.137256	3

200 rows × 3 columns

- *Final Data Visualization of Clustering*



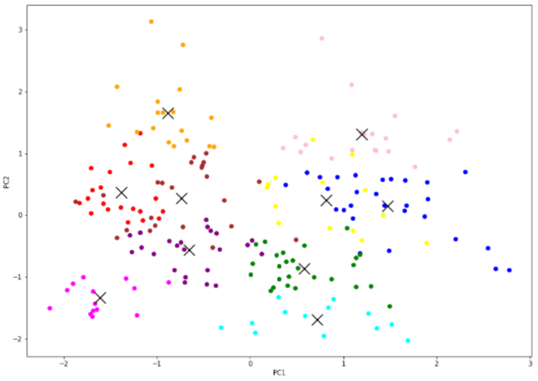
```
plt.figure(figsize=(14, 10))

plt.scatter(reduced_data[reduced_data['cluster'] == 0].loc[:, 'PC1'], reduced_data[reduced_data['cluste
plt.scatter(reduced_data[reduced_data['cluster'] == 1].loc[:, 'PC1'], reduced_data[reduced_data['cluste
plt.scatter(reduced_data[reduced_data['cluster'] == 2].loc[:, 'PC1'], reduced_data[reduced_data['cluste
plt.scatter(reduced_data[reduced_data['cluster'] == 3].loc[:, 'PC1'], reduced_data[reduced_data['cluste
plt.scatter(reduced_data[reduced_data['cluster'] == 4].loc[:, 'PC1'], reduced_data[reduced_data['cluste
plt.scatter(reduced_data[reduced_data['cluster'] == 5].loc[:, 'PC1'], reduced_data[reduced_data['cluste
plt.scatter(reduced_data[reduced_data['cluster'] == 6].loc[:, 'PC1'], reduced_data[reduced_data['cluste
plt.scatter(reduced_data[reduced_data['cluster'] == 7].loc[:, 'PC1'], reduced_data[reduced_data['cluste
plt.scatter(reduced_data[reduced_data['cluster'] == 8].loc[:, 'PC1'], reduced_data[reduced_data['cluste
plt.scatter(reduced_data[reduced_data['cluster'] == 9].loc[:, 'PC1'], reduced_data[reduced_data['cluste

plt.scatter(reduced_centers[:, 0], reduced_centers[:, 1], color='black', marker='x', s=300)

plt.xlabel("PC1")
plt.ylabel("PC2")

plt.show()
```



B. *Outcomes of analysis and their visualization*

- Gender plot analysis gave us understanding of most gender who buys products mostly from mall.
- Age plot analysis gave us understanding of which age group buys product mostly
- Spending score (1-100) plotting gave us brief understanding which score people are maximum and minimum density.
- Annual income plotting gave us the understanding of which income group are of which density.
- Lastly, we did create multiple relations between age, gender, annual income and spending score (1-100).
- We did PCA for dimensionality reduction and efficient feature analysis
- Identification of K-mean cluster centers and data reduction done to make clustering easy
- Finally clustering data visualization done with reduced data and PCA analysis (PC1 vs PC2).

C. *Tables of outcomes*

Name of Group/Cluster	Annual Income(type)	Spending Score	Category for targeting
1(Red Color)	High	High	Ideal
2(Blue Colored)	Low	High	Average
3(Green Color)	High	Low	Could be ideal
4(Yellow color)	Low	Low	Not Ideal
5(Pink Color)	Average	Average	Average

Result and Discussion

From all the analysis following things are found:
a) Female customers buy the most.
b) From age 20-40 maximum customers are found
c) Most buyer annual income is between 50K to 100 K US dollars.
d)Density of customers with spending score around 50 is the maximum.
e) In short from 20-40 age group both low and high annual income buyers are maximum.
f) The spending score of most of the females lies between the 35 and 52.
g) Annual income and spending score of male and female and conclude that those females whose annual income in between 40 to 85k in dollar there spending score is also higher.
h) The highest spending score of a male is 73 and the highest annual income is 137k$.
i) We observed that between the age of 23 and 50 females visits Mall frequently and after the age of 50 males visit mall frequently.

References

[1] Cooil, B., Aksoy, L. & Keiningham, T. L. (2008), 'Approaches to customer segmentation', Journal of Relationship Marketing 6(3-4), 9–39.

[2] T. Kanungo, D. M. Mount, N. S. Netanyahu, C. D. Piatko, R.Silverman, and A. Y. Wu, "An efficient K-means clustering algorithm," IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 24, pp. 881-892, 2002.

[3] Marcus, C. (1998), 'A practical yet meaningful approach to customer segmentation approach to customer segmentation', Journal of Consumer Marketing15, 494–504

[4] Customer Segmentation Using Machine Learning
P Monil, P Darshan, R Jecky, C Vimarsh

[5] Algorithms for Decision Making Through Customer Segmentation
Jesus Vargas, Nelson Alberto & Oswaldo Arevalo