Contents

Abstract

This project report focuses on the application of K-Means Clustering Algorithm to segment customers for targeted marketing campaigns. The aim of the study is to identify distinct customer groups based on their purchasing behavior and demographic information, which would enable businesses to tailor their marketing strategies to specific customer segments. The study uses a dataset of customer transactions and demographic information, which was pre-processed and cleaned before applying K-Means clustering. The results of the clustering analysis are presented through visualizations and statistical measures, such as the silhouette coefficient and within-cluster sum of squares . The study found that the optimal number of clusters was five, which corresponded to distinct customer groups based on their purchasing patterns and demographic characteristics. The report concludes by highlighting the potential benefits of using customer segmentation for marketing campaigns and the limitations of the K-Means clustering algorithm.

Keywards : K-Means Cluster, Customer Segmentation, Unsupervised Learning.

1 Introduction

Customer segmentation analysis is a powerful tool that businesses can use to better understand their customers and tailor their marketing strategies accordingly. One popular method for conducting customer segmentation analysis is through the use of unsupervised machine learning algorithms, such as k-means clustering.

K-means clustering is a technique that involves grouping similar data points into clusters based on their similarities. In the context of customer segmentation analysis, this means grouping customers together based on shared characteristics such as demographics, purchasing behavior, and preferences. By doing so, businesses can identify different customer segments with unique needs and tailor their marketing efforts to each segment in a more targeted and effective manner.

In this analysis, we will explore how k-means clustering can be used for customer segmentation analysis, including the steps involved in the process and how to interpret the results. We will also discuss some common challenges and limitations associated with this approach and explore some potential solutions. Overall, this analysis will provide a comprehensive overview of how businesses can use k-means clustering to better understand their customers and improve their marketing strategies.

As the customer base is increasing day by day it has become challenging for the companies to cater to the needs of each and every customer, this is where Data Science serves a very important role to unravel hidden patterns stored in the data.

Product classifications as well as customer segmentation are most frequent used methods. The Customer Segmentation is focused on getting knowledge about the structure of customers and is used for targeted marketing. This segmentation can directly or indirectly influence the marketing strategy as it opens many new paths to discover like for which segment the product will be good, customizing the marketing plans according to the each segment, providing discounts for a specific segment, and decipher the customer and object relationship which has been previously unknown to the company.

Customer segmentation allows companies to visualize what actually the customers are buying which will prompt the companies to better serve their customers resulting in customer satisfaction, it also allows the companies to find who their target customers are and improvise their marketing tactics to generate more revenues from them.

Clustering comes under unsupervised learning, having ability to find clusters over **unlabelled dataset**. In this seminar, K-Means clustering algorithm is being implemented on a data set with 200 observations and 5 attributes.

2 Background

K-means clustering is a popular unsupervised machine learning algorithm used for clustering data points into groups or clusters based on their similarities.

2.1 Cluster

Clustering is an unsupervised machine learning task. You might also hear this referred to as cluster analysis because of the way this method works.

Using a clustering algorithm means you're going to give the algorithm a lot of input data with no labels and let it find any groupings in the data it can.

Those groupings are called clusters. A cluster is a group of data points that are similar to each other based on their relation to surrounding data points. Clustering is used for things like feature engineering or pattern discovery.

When you're starting with data you know nothing about, clustering might be a good place to get some insight.

In data science, a cluster refers to a group of data points that share similar characteristics or properties. Clustering is the process of grouping these data points together based on their similarities, such as their attributes or behaviors. The goal of clustering is to identify groups of similar data points that can be analyzed or used for further processing.

Example 2.1 (**Retail Marketing**)**.** Retail companies often use clustering to identify groups of households that are similar to each other.

For example, a retail company may collect the following information on households:

- Household income
- Household size
- Head of household Occupation
- Distance from nearest urban area

They can then feed these variables into a clustering algorithm to perhaps identify the following clusters:

- Cluster 1: Small family, high spenders
- Cluster 2: Larger family, high spenders
- Cluster 3: Small family, low spenders
- Cluster 4: Large family, low spenders

The company can then send personalized advertisements or sales letters to each household based on how likely they are to respond to specific types of advertisements.

Example 2.2 (**Health Insurance**)**.** Actuaries at health insurance companies often used cluster analysis to identify "clusters" of consumers that use their health insurance in specific ways.

For example, an actuary may collect the following information about households:

- Total number of doctor visits per year
- Total household size
- Total number of chronic conditions per household

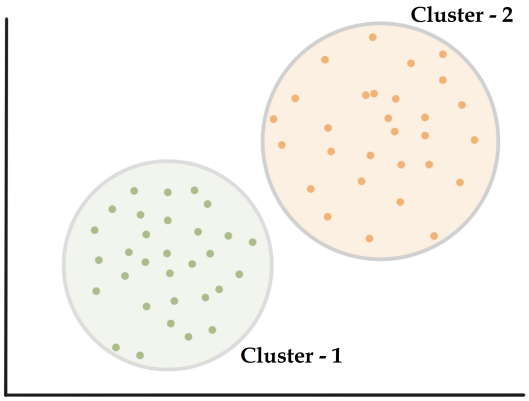

Figure 1: Two Clusters

- Average age of household members

An actuary can then feed these variables into a clustering algorithm to identify households that are similar. The health insurance company can then set monthly premiums based on how often they expect households in specific clusters to use their insurance.

2.2 Properties of Cluster

Property 1 (Cohesion)**.** Clusters are characterized by a high degree of cohesion, which means that the objects within a cluster are more similar to each other than to objects in other clusters. See Figure (1).

Property 2 (Separation)**.** Clusters are also characterized by a high degree of separation, which means that the objects in different clusters are dissimilar from each other

2.3 Applications of Clustering in Real-World Scenarios

Clustering is a widely used technique in the industry. It is actually being used in almost every domain, like -

1. Customer Segmentation
2. Document Clustering
3. Image Segmentation
4. Recommendation Engines

3 Methodology

From the first property of clusters – **it states that the points within a cluster should be similar to each other**. So, our aim here is to minimize the distance between the points within a cluster.

In this seminar, I have used K-Means algorithm for clustering. It is the most efficient clustering algorithms proposed in the literature of data clustering. K-Means algorithm is an iterative algorithm that tries to partition the dataset into K pre-defined distinct non-overlapping subgroups (clusters) where each data point belongs to only one group.

It tries to make the intra-cluster data points as similar as possible while also keeping the clusters as different (far) as possible. It assigns data points to a cluster such that the sum of the squared

distance between the data points and the cluster's centroid (arithmetic mean of all the data points that belong to that cluster) is at the minimum. The less variation we have within clusters, the more homogeneous (similar) the data points are within the same cluster.

3.1 Algorithm : K-Means Cluster

The algorithm for K-means clustering can be summarized in the following steps:

1. First choose the number of clusters K.
2. Randomly select K points from the dataset to serve as the initial centroids for the clusters.
3. Assign each data point to the nearest centroid based on the Euclidean distance between the point and the centroid.
4. Recalculate the centroids for each cluster by taking the mean of all the points assigned to that cluster.
5. Repeat steps 3 and 4 until the centroids no longer move significantly, or a maximum number of iterations is reached.

3.2 Stopping Criteria for K-Means Clustering

There are essentially three stopping criteria that can be adopted to stop the K-means algorithm:

1. Centroids of newly formed clusters do not change
2. Points remain in the same cluster
3. Maximum number of iterations are reached

3.3 Proof of mathematical convergence of K-means clustering:

Given a dataset $D = \{x_1, x_2, ..., x_N\}$ consists of N points, let us denote the clustering obtained after applying K-means clustering by $C = \{C_1, C_2, ..., C_k\}$ The objective is to find clustering that minimizes the **Sum of Square Error (SSE)** score. The SSE for this clustering is defined in the Equation (a) where ck is the centroid of cluster C_k.

The iterative assignment and update steps of the K-means algorithm aim to minimize the SSE score for the given set of centroids. Let us denote C_k as the kth cluster, x_i is a point in C_k, and C_k is the mean of the kth cluster. We can solve for the representative of C_j which minimizes the SSE by differentiating the SSE with respect to C_j and setting it equal to zero.

$$\begin{aligned} SSE(C) &= \sum_{k=1}^{K} \sum_{x_i \in C_k} ||(x_i - C_k)||^2 \\ &= \sum_{k=1}^{K} \sum_{x_i \in C_k} a_{ij} ||(x_i - C_k)||^2 \end{aligned}$$

where $a_{ij} = \begin{cases} 1 & if x_i \text{ assigned to } C_k \\ 0 & else \end{cases}$

(a) Assign x_i to the nearest C_k i.e.

$$a_{ij} = \begin{cases} 1 & \text{if } arg\ min||x_i - C_k||^2 \\ 0 & \text{else} \end{cases}$$

(b)

$$SSE(C) = \sum_{k=1}^{K} \sum_{x_i \in C_k} (x_i - C_k)^2$$

$$\begin{aligned} \frac{\partial\, SSE}{\partial C_i} &= \frac{\partial}{\partial C_i} \sum_{k=1}^{K} \sum_{x_i \in C_k} (x_i - C_k)^2 \\ &= \sum_{k=1}^{K} \sum_{x_i \in C_k} \frac{\partial}{\partial C_i} (x_i - C_k)^2 \\ &= \sum_{x_i \in C_i} 2 * (C_i - x_i) = 0 \end{aligned}$$

$$=> |C_i| \cdot C_i = \sum_{x_i \in C_i} x_i => C_i = \frac{\sum_{x_i \in C_i} x_i}{|G|}$$

From the above equation, we find that in each iteration centroid is updated to mean of the data points in a cluster.

3.4 Data Collection

The data set for this project has been taken from Kaggle, a subsidiary of Google LLC, is an online community of data scientists and machine learning practitioners. This dataset contains customer information such as *CustomerID, Gender, Age, Annual Income (k $) , Spending Score.*

4 Analysis

Here we have implemented K-Means Cluster using Python. The steps are as follows:

Listing 1: Importing the Library

```
import numpy as np # linear algebra
import pandas as pd # data processing, CSV file I/O (e.g. pd.read_csv)

import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline
# for interactive visualizations
import plotly.offline as py
import plotly.graph_objs as go
import plotly.figure_factory as ff
from plotly.offline import init_notebook_mode, iplot
init_notebook_mode(connected = True)
from plotly import tools
```

Listing 2: Reading the CSV data file

```
mall_data = pd.read_csv("Mall_Customers.csv")
data = ff.create_table(mall_data.head())
py.iplot(data)
```

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

Figure 2: First five rows of the data

Listing 3: Data Description

```
Check the Descriptions of the datasets using plotly library.
print(mall_data.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
```

Figure 3: Data Description

Observation:

- Age of the customers ranges from 18-70. This shows that the mall attracts has shops and things which suite all age group people.
- Average age of customers is 39.
- Average income of customers is 60 K$.
- Average spending score of customers is 50.

Listing 4: Caption

```
#Data Visualization
plt.figure(figsize = (16,5))
plt.subplot(1, 3, 1)
sns.distplot(mall_data['Age'])
plt.subplot(1, 3, 2)
sns.distplot(mall_data['Annual Income (k$)'])
plt.subplot(1, 3, 3)
sns.distplot(mall_data['Spending Score (1-100)'])
plt.show()
```

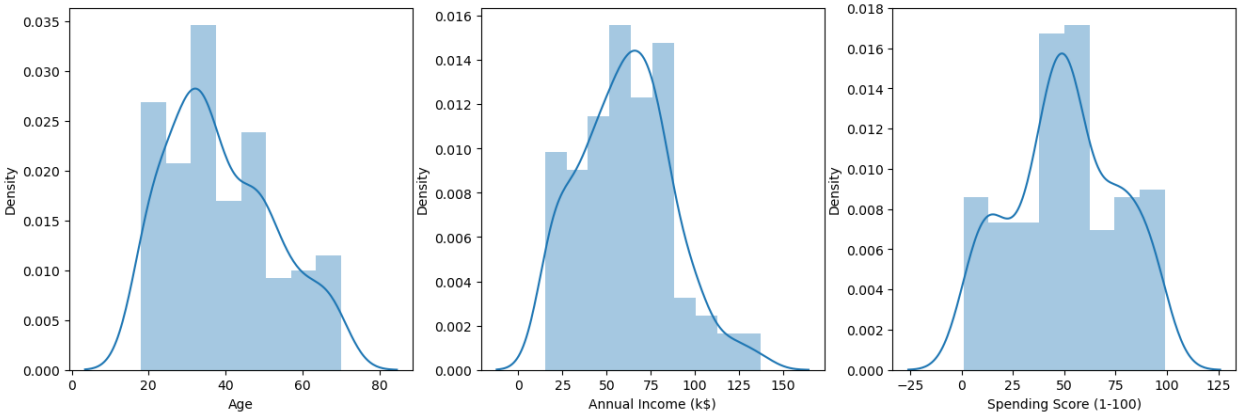

Figure 4: Data Visualization

Listing 5: Preparing Data

```
# Prepare Data
df = mall_data.groupby('Gender').size()
# Make the plot with pandas
df.plot(kind='pie', subplots=True, figsize=(15, 8))
plt.title("Pie Chart of Vehicle Class - Bad")
plt.ylabel("")
plt.show()
```

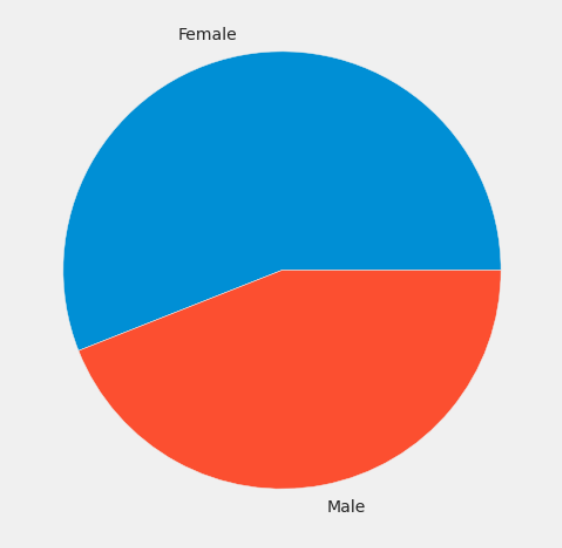

Figure 5: Pie diagram - Male vs Female

Listing 6: Histogram of the data

```
### hist plot
mall_data.hist(figsize = (15, 12))
plt.show()
```

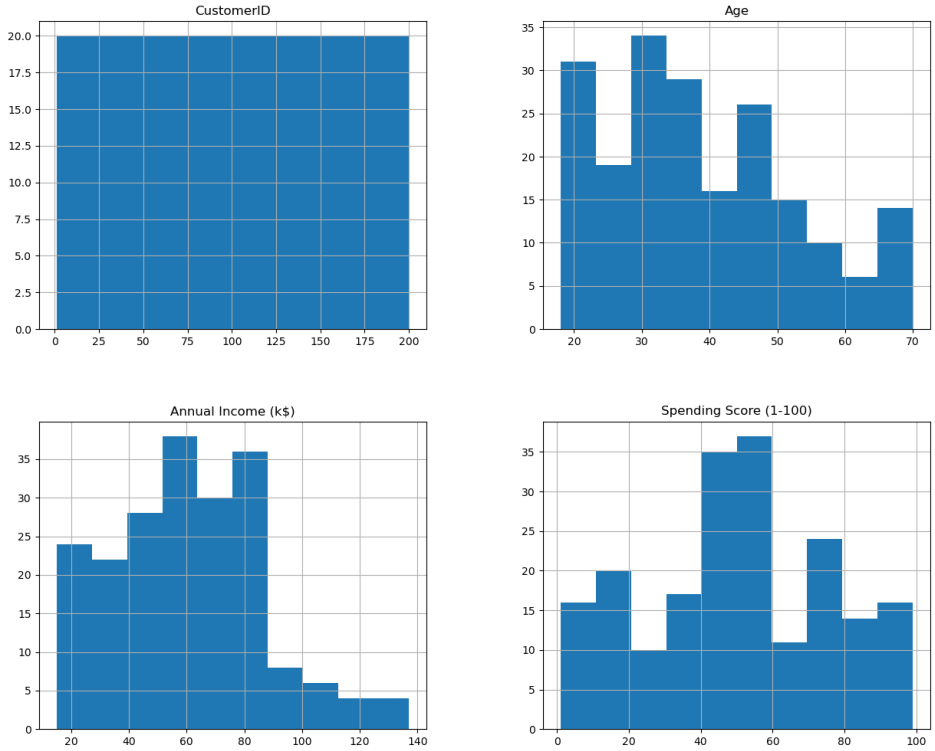

Figure 6: Histogram of the data

5 Estimating the Number of Clusters

The problem of estimating the correct number of clusters (K) is one of the major challenges for the K-means clustering. Several researchers have proposed new methods for addressing this challenge in the literature. We will briefly describe the most prominent method Elbow method to choose number of clusters.

Elbow Method

The Elbow method is one of the most popular ways to find the optimal number of clusters. This method uses the concept of WCSS value. WCSS stands for Within Cluster Sum of Squares, which defines the total variations within a cluster. The formula to calculate the value of WCSS (for k clusters) is given below:

$$WCSS = \sum_{P_i \ in \text{ cluster 1}} distance(P_iC_1)^2 + \sum_{P_i \ in \text{ cluster 2}} distance(P_iC_2)^2 + \cdots + \sum_{P_i \ in \text{ cluster k}} distance(P_iC_k)^2.$$

In the above formula of WCSS,

$\sum_{(P_1)inCluster1} distance(P_iC_1)^2$:

It is the sum of the square of the distances between each data point and its centroid within a cluster1 and the same for the other two terms.

To measure the distance between data points and centroid, we can use any method such as Euclidean distance or Manhattan distance.

To find the optimal value of clusters, the elbow method follows the below steps:

- It executes the K-means clustering on a given dataset for different K values (ranges from 1-10).
- For each value of K, calculates the WCSS value.
- Plots a curve between calculated WCSS values and the number of clusters K.
- The sharp point of bend or a point of the plot looks like an arm, then that point is considered as the best value of K.

Listing 7: Elbow Method

```
#Elbow Method
#Considering only 2 features (Annual income and Spending Score)
X= mall_data.iloc[:, [3,4]].values

#Building the Model
#KMeans Algorithm to decide the optimum cluster number ,
KMeans++ using Elbow Mmethod

from sklearn.cluster import KMeans
k=[]

for i in range(1,11):
    kmeans = KMeans(n_clusters= i, init='k-means++', random_state=0)
    kmeans.fit(X)
    k.append(kmeans.inertia_)
```

```
#Visualizing the ELBOW method to get the optimal value of K

plt.figure(1 , figsize = (15 , 6))
plt.plot(range(1,11), k)
plt.title('The Elbow Method')
plt.xlabel('no of clusters')
plt.ylabel('wcss')
plt.show()
```

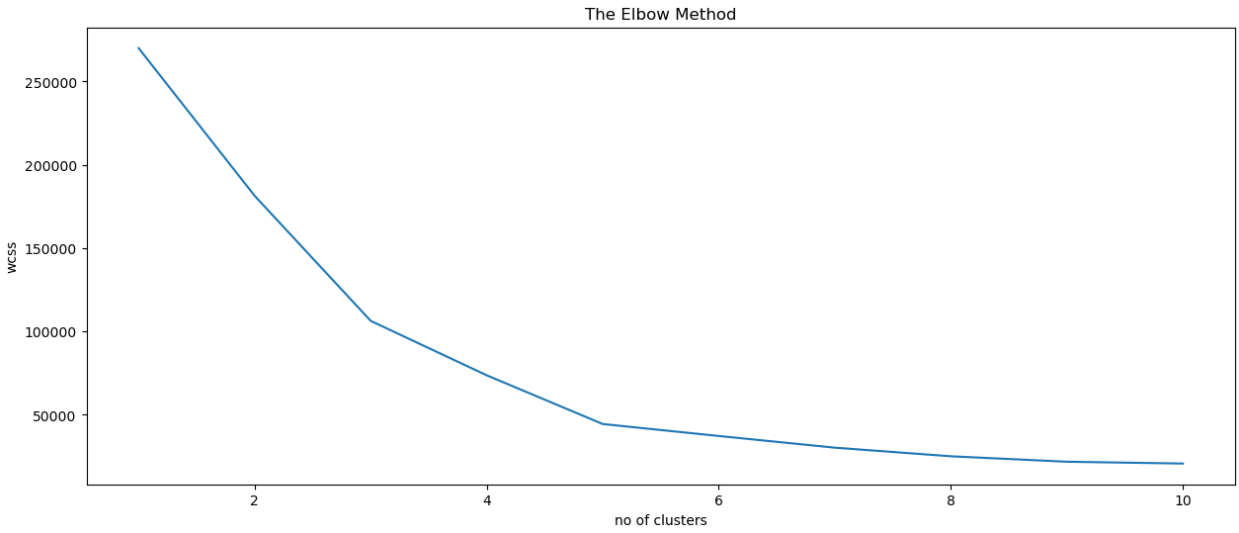

Figure 7: Elbow Curve for number of clusters

Listing 8: Model Building : K-Means Cluster

```
#Model Build
model = KMeans(n_clusters= 5, init='k-means++', random_state=0)
y_kmeans= model.fit_predict(X)
```

Listing 9: Clusters Visualization

```
#Visualizing all the clusters
plt.figure(1 , figsize = (15 , 8))
plt.scatter(X[y_kmeans == 0, 0], X[y_kmeans == 0, 1], s = 100, c = 'magenta',
label = 'Cluster 1')
### Cluster 1
plt.scatter(X[y_kmeans == 1, 0], X[y_kmeans == 1, 1], s = 100, c = 'blue',
label = 'Cluster 2')
## Cluster 2
plt.scatter(X[y_kmeans == 2, 0], X[y_kmeans == 2, 1], s = 100, c = 'cyan',
label = 'Cluster 3')
## Cluster 3
plt.scatter(X[y_kmeans == 3, 0], X[y_kmeans == 3, 1], s = 100, c = 'green',
label = 'Cluster 4')
## Cluster 4
plt.scatter(X[y_kmeans == 4, 0], X[y_kmeans == 4, 1], s = 100, c = 'red',
label = 'Cluster 5')
## Cluster 5
```

```
plt.title('K-Means-Clustering-Algorithm')
plt.xlabel('Annual-Income-(k$)')
plt.ylabel('Spending-Score-(1-100)')
plt.legend()
plt.show()
```

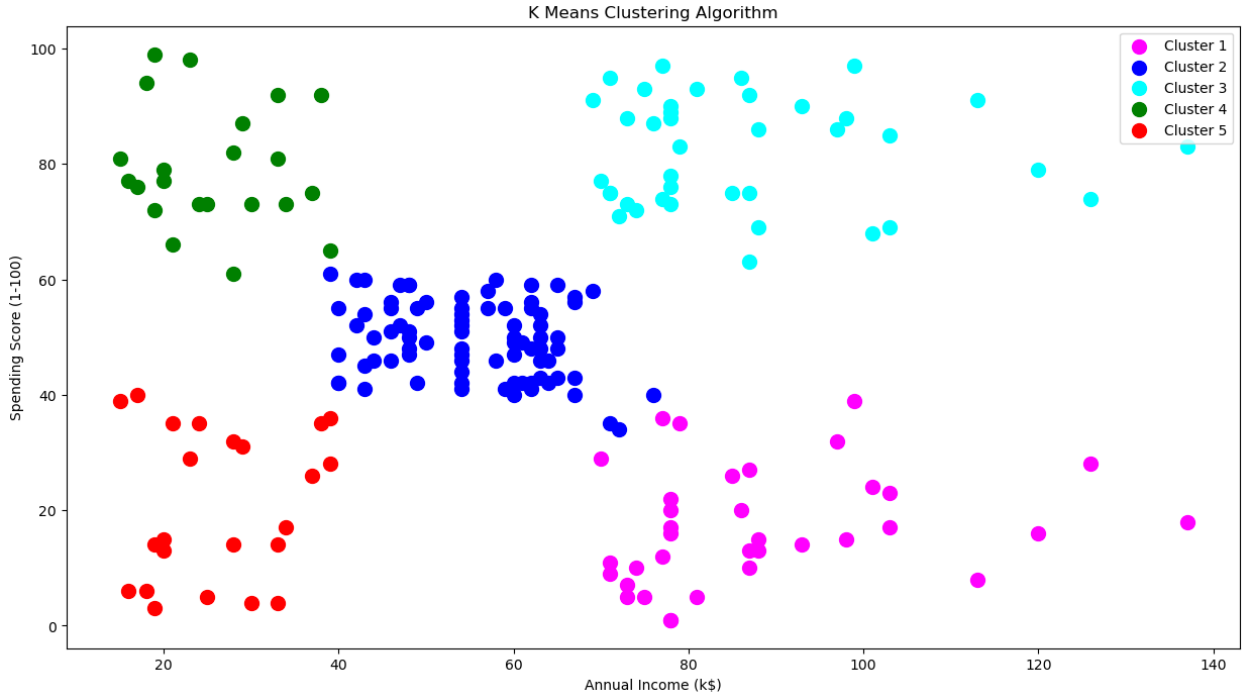

Figure 8: Clusters Visualization

6 Results

Now after analysing the data we get some results. The results are as follows:

1. From the Figure (3) we see that in this mall there are 25% people are of age = 28 years, 50% people are of age = 36 years, 50% people are of age = 50 years.
2. From the Figure (4) we see that :
   - In Age columns most people belong to 18 to 50 Age.
   - Maximum Annual Income 45k to 90k.
   - Maximum Spending Sore is 40 to 60.
3. From the Figure (5) we see that females are going to that mall more than males.
4. From the Figure (7) we see that there are $K = 5$ clusters for this model(Annual Income vs Spending Score). This Clustering Analysis gives us a very clear insight about the different segments of the customers in the Mall. There are clearly Five segments of Customer Clusters namely:
   - Cluster 1 (Magenta Color) : Earning high but spending less.
   - Cluster 2 (Blue Color) : Average in terms of earning and spending.
   - Cluster 3 (Cyan Color) : Earning high and also spending high [**TARGET SET**].

- Cluster 4 (Green Color) : Earning less but spending more.
- Cluster 5 (Red Color) : Earning less , spending less.

7 Conclusion

Targeting both high-earning and high-spending customers in a shopping mall can be a strategic move to increase revenue and profitability. Here are some possible discussions on how to attract and retain these two types of customers:

1. **Understanding the needs and preferences of high-earning customer** : High-earning customers typically prioritize convenience, quality, and exclusivity in their shopping experience. Therefore, shopping malls can cater to their needs by providing valet parking, personalized services, upscale brands, and VIP lounges. Moreover, high-earning customers tend to be time-poor, so offering extended hours, online shopping, and delivery services can increase their loyalty and satisfaction.

2. **Offering value propositions for high-spending customers** : High-spending customers may not necessarily be high-earning, but they are willing to splurge on luxury items or experiences. Therefore, shopping malls can create value propositions for them by offering discounts, loyalty programs, gift cards, free samples, and other incentives that make them feel appreciated and rewarded for their spending. Moreover, organizing events, promotions, and partnerships with premium brands can enhance their shopping experience and entice them to spend more.

3. **Enhancing the ambience and atmosphere of the shopping mall** : Both high-earning and high-spending customers value the ambiance and atmosphere of the shopping mall. A well-designed and maintained environment with pleasant lighting, music, and decor can create a positive mood and stimulate their senses. Moreover, having amenities such as restaurants, cafes, cinemas, spas, and playgrounds can offer a holistic experience that encourages customers to stay longer and spend more.

4. **Leveraging technology to personalize the shopping experience** : Technology can play a crucial role in attracting and retaining high-earning and high-spending customers. For instance, implementing a loyalty app that offers customized recommendations, discounts, and promotions based on their shopping history and preferences can increase their engagement and loyalty. Moreover, using artificial intelligence and data analytics to analyze their behavior and anticipate their needs can offer a seamless and personalized shopping experience that enhances their satisfaction and loyalty.

In conclusion, targeting high-earning and high-spending customers in a shopping mall requires a deep understanding of their needs, preferences, and behavior. By offering tailored value propositions, enhancing the ambiance and atmosphere, leveraging technology, and providing personalized services, shopping malls can attract and retain these valuable customers and boost their revenue and profitability.

References

[1] C. M. Bishop, *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Berlin, Heidelberg: Springer-Verlag, 2006.

[2] J. A. Hartigan and M. A. Wong, "Algorithm as 136: A k-means clustering algorithm," *Journal of the royal statistical society. series c (applied statistics)*, vol. 28, no. 1, pp. 100–108, 1979.

[3] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning*. Springer Series in Statistics, New York, NY, USA: Springer New York Inc., 2001.